

Dynamic Proximal Unrolling Network for Compressive Imaging

Yixiao Yang^a, Ran Tao^{a,*}, Kaixuan Wei^b, Ying Fu^b

^aSchool of Information and Electronics, Beijing Institute of Technology, Beijing, China

^bSchool of Computer Science and Technology, Beijing Institute of Technology, Beijing, China

Abstract

Compressive imaging aims to recover a latent image from under-sampled measurements, suffering from a serious ill-posed inverse problem. Recently, deep neural networks have been applied to this problem with superior results, owing to the learned advanced image priors. These approaches, however, require training separate models for different imaging modalities and sampling ratios, leading to overfitting to specific settings. In this paper, a dynamic proximal unrolling network (dubbed DPUNet) was proposed, which can handle a variety of measurement matrices via one single model without retraining. Specifically, DPUNet can exploit both the embedded observation model via gradient descent and imposed image priors by learned dynamic proximal operators, achieving joint reconstruction. A key component of DPUNet is a dynamic proximal mapping module, whose parameters can be dynamically adjusted at the inference stage and make it adapt to different imaging settings. Moreover, in order to eliminate the image blocking artifacts, an enhanced version DPUNet⁺ is developed, which integrates a dynamic deblocking module and reconstructs jointly with DPUNet to further improve the performance. Experimental results demonstrate that the proposed method can effectively handle multiple compressive imaging modalities under varying sampling ratios and noise levels via only one trained model, and outperform the state-of-the-art approaches. Our code is available at <https://github.com/Yixiao-Yang/DPUNet-PyTorch>.

Keywords: Dynamic neural networks, deep proximal unrolling, compressive imaging, image reconstruction.

1. Introduction

Compressive imaging depicts a novel imaging paradigm for image acquisition and reconstruction that allows the recovery of an underlying image from far fewer measurements than the Nyquist sampling rate [1, 2, 3, 4], and drives a range of practical applications, such as image or video compressive sensing (CS) [2, 5], compressive sensing magnetic resonance imaging (CS-MRI) [6, 7], single-pixel imaging [3, 8], snapshot compressive imaging [9, 10], and compressive phase retrieval (CPR) [11, 12].

Mathematically, given the under-sampled measurements $y \in \mathbb{R}^m$ and observation model $\Phi(\cdot)$, the goal of compressive imaging is to find a solution $\hat{x} \in \mathbb{R}^n$, such that $y \approx \Phi(\hat{x})$ and \hat{x} resides the class of images. Since the sampling ratio, defined as $\frac{m}{n}$, is typically much less than one, reconstructing a unique solution from limited measurements only is difficult or impossible without proper image priors.

To tackle the fundamental ill-posedness of compressive imaging, traditional methods typically exploit the measurement model knowledge and intrinsic image properties [13, 14, 15, 16, 17], and solve a regularized optimization problem in an iterative scheme [18, 19, 20, 21, 22, 23]. They are effective and flexible to handle a wide variety of measurements based on the well-studied forward model and well-understood behavior but limited in unsatisfactory reconstruction quality and high computational complexity.

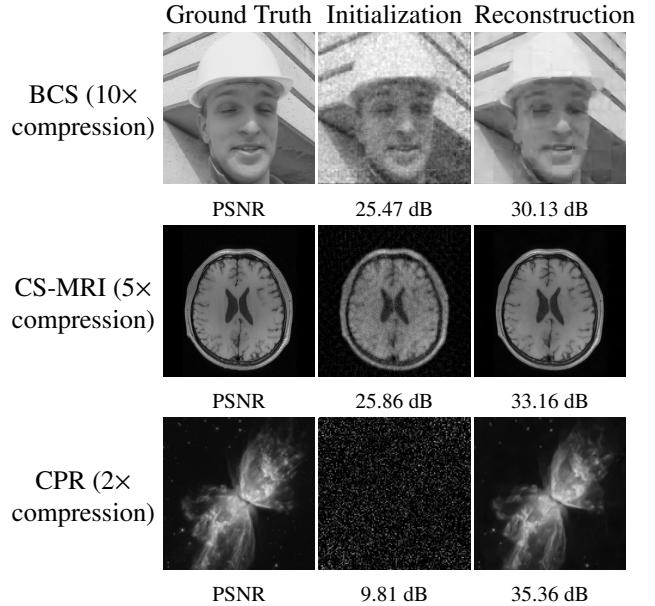


Figure 1: The single trained model of our network is used to handle multiple compressive imaging modalities with different sampling ratios: block-based compressive sensing (BCS) with 10x compression, CS-MRI with 5x compression, and CPR with 2x compression. Note that even though these inverse problems are very different, the proposed network can handle multiple imaging modalities under various imaging conditions via only one trained model without retraining.

In contrast to iterative-based methods, deep-learning-based compressive imaging approaches [24, 25, 26, 27, 28, 29, 30], as

*Corresponding author

Email address: rantao@bit.edu.cn (Ran Tao)

an alternative, employ neural networks to directly learn a mapping from measurements to latent quantities entirely based upon data. Once trained, the inference only requires a single forward of the network, without the need for time-consuming optimization and hyper-parameters selection. Nevertheless, the pure deep learning approach cannot offer the flexibility of variational methods in adapting to different imaging modalities and even sampling ratios, largely due to learning a task-specific mapping. Traditionally one would require a separate deep network for each imaging setting even if there is only a tiny change, which limits its practical applications.

Motivated by the observation that many iterative optimization algorithms can be truncated and unfolded into learnable deep neural networks, researchers have explored the hybrid approach (*i.e.*, deep unrolling) that combines the best of both worlds in compressive imaging [31, 32, 33, 34, 35, 36]. Following the unrolling, regularizers, free hyper-parameters such as the step size or regularization parameters, and even sampling matrices can be learned via end-to-end training, rather than being hand-crafted. Meanwhile, the physical measurement model can be explicitly exploited. In this way, deep unrolling networks take the merits of interpretability and flexibility of optimization-based methods and fast inference of deep-learning-based approaches, while achieving promising reconstruction performance.

Despite these gains, the existing deep unrolling networks still suffer from severe performance degradation when operating on a measurement model significantly different from the training set. An illustrative example of a state-of-the-art CS network, *i.e.*, ISTA-Net⁺, is shown in Figure 2. It can be seen that the model of ISTA-Net⁺ trained on a fixed sampling ratio (“fixed”) produces poor results when testing on unseen sampling ratios. While the performance can be improved when training on all sampling ratios jointly (“all”), there is still a large gap between such model and the task-specific model separately trained on each sampling ratio (“optimal”).

Basically, the ill-posedness of compressive imaging and the difficulty of learning to solve the corresponding inverse problem are very distinct under different imaging conditions. Furthermore, learned parameters of a specific unrolling network remain fixed after training, so the inference cannot adapt dynamically to other imaging parameter configurations. To address this issue, in this paper, a novel deep unrolling architecture was proposed, whose key part is a dynamic proximal mapping module. Specifically, this module consists of a convolutional neural network (CNN) that learns/executes proximal operators and several fully connected networks that perform dynamic modification mechanisms to adjust the parameters of CNN given imaging conditions, which can be jointly trained end-to-end. In this way, fully connected networks will dynamically adjust the learned proximal operators at the inference stage and make the unrolling network adapt to different imaging settings.

The effectiveness of the proposed method is verified on three representative compressive imaging applications, *i.e.*, image compressive sensing, CS-MRI, and CPR under various imaging conditions. Furthermore, the applicability of the proposed framework is explored on multiple imaging modalities with different imaging conditions. Experimental results demonstrate that the

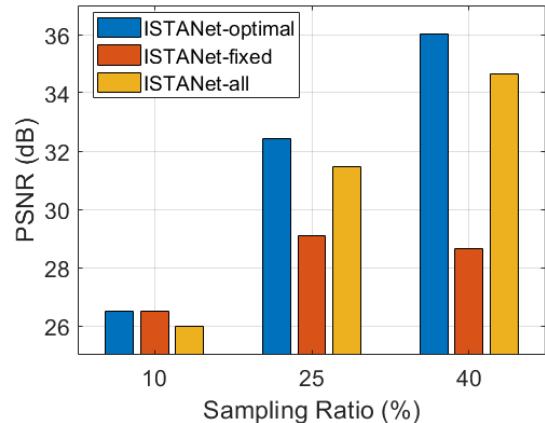


Figure 2: The model generalizability test of a representative deep unrolling method – ISTANet [34] for image compressive sensing under multiple sampling ratios. “ISTANet-optimal” denotes the (three) models trained and tested on consistent sampling ratios. “ISTANet-fixed” indicates the single model trained on the fixed sampling ratio 10% and then tested on all sampling ratios. “ISTANet-all” implies the single model trained and tested on all sampling ratios (*i.e.*, 10%, 25%, and 40%).

proposed method can not only effectively tackle varying imaging conditions for a specific compressive imaging task, but also be able to handle multiple imaging modalities via only *one single model* simultaneously (see Figure 1), and outperform the state-of-the-art approaches.

Our main contributions can be summarized as follows:

- We present a dynamic proximal unrolling network (dubbed DPUNet) that can adaptively handle different imaging conditions, and even multiple compressive imaging modalities via the only one trained model.
- The key part of DPUNet is to develop a dynamic proximal mapping module, which can enable the on-the-fly parameter adjustment at the inference stage and boost the generalizability of deep unrolling networks.
- To remove the blocking artifacts, an enhanced version DPUNet⁺ is developed, which integrates a dynamic de-blocking module and reconstructs jointly with DPUNet to further improve the recovered images.
- Experimental results demonstrate DPUNet can outperform the state-of-the-art on image compressive sensing, CS-MRI, and CPR under various imaging conditions without retraining. In addition, we show the extension of DPUNet can simultaneously handle all these imaging tasks via one single trained model, with promising results.

The remainder of this paper is organized as follows. We review the related works in Section 2. In Section 3, the details of the proposed method including unrolling frameworks and dynamic proximal mapping module are presented. Section 4 provides both experimental settings and qualitative results. In Section 5, we conclude the paper.

2. Related Work

2.1. CS Reconstruction Approaches

Traditionally, the inverse problem of compressive imaging can be attacked by variational optimization methods by minimizing the following cost functional:

$$\underset{x}{\text{minimize}} \quad \frac{1}{2} \|y - \Phi(x)\|_2^2 + \lambda \mathcal{R}(x), \quad (1)$$

by an iterative optimization framework, *e.g.*, the proximal gradient descent (PGD) [18], the alternating direction method of multipliers (ADMM) [22], the iterative shrinkage-thresholding algorithm (ISTA) [20], and the half-quadratic splitting (HQS) [21] algorithm. In the above, $\mathcal{R}(x)$ indicates the regularization term associated with the prior knowledge of images, which alleviates the ill-posedness of compressive imaging. Many image priors have been designed in the imaging community over the past decades. The well-known examples include structured sparsity [37, 38, 39, 40], group sparse representation (GSR) [15] and nonlocal low-rank [14, 41, 42].

In contrast to handcrafted priors, deep-learning-based compressive imaging approaches have been proposed and demonstrate promising reconstruction results with fast inference speed. A stacked denoising auto-encode (SDA) is first applied to learn statistical dependencies from data, improving signal recovery performance [43]. Fully connected neural networks are proposed for image and video block-based compressive sensing (BCS) reconstruction [44, 45]. Further several CNN-based approaches are developed, which learn the inverse map from compressively sensed measurements to reconstructed images [46, 27]. Recently, some works jointly optimize a sampling sub-network and a reconstruction sub-network to further enhance the performance. CSNet [28] develops a sampling-recovery framework and BCSNet [30] avoids blocking artifacts by utilizing interblock correlation. SCSNet [29] presents a hierarchical CNN for fine-granular scalable sampling and reconstruction. The basic idea of these works is designing a neural network $f_{NN}(y, \Theta)$ to directly perform the inverse mapping from observed measurements y to desired images x by learning network parameters Θ based upon the training dataset Γ :

$$\hat{\Theta} = \underset{\Theta}{\text{argmin}} \quad \frac{1}{|\Gamma|} \sum_{(y,x) \in \Gamma} \mathcal{L}(x, f_{NN}(y, \Theta)), \quad (2)$$

where \mathcal{L} is a loss function. However, most of the network architectures are predefined and fixed for specific problems, which usually cannot adapt to others. While achieving adaptive sampling and reconstruction with one trained model, SCSNet also requires to update the greedy searching strategy for different sampling ratios. In addition, designing neural networks could be often considered as much an art as a science, without clear theoretical guidelines and domain knowledge [47].

Inspired by the interpretability and flexibility of traditional approaches, an emerging technique called deep unrolling or unfolding has been applied to compressive imaging [31, 32, 33, 34]. By unrolling the iterative optimization framework and introducing learnable parameters Θ , deep unrolling networks can

be learned by minimizing the following empirical risk:

$$\hat{\Theta} = \underset{\Theta}{\text{argmin}} \quad \frac{1}{|\Gamma|} \sum_{(y,x) \in \Gamma} \mathcal{L}(x, f_{NN}(y, \Phi, \Theta)), \quad (3)$$

where Φ is the given physical measurement model. Preliminary attempts focused on learning fast approximation of specialized iterative solvers attached to well-designed priors. The computation schemes of the resulting networks are concise with the original solvers, but with fixed numbers of iterations and some untied/unshared parameters across layers. Well-known examples include learned shrinkage-thresholding algorithms (LISTA) [31] and learned approximate message passing (LAMP) algorithms [32]. A similar idea has also been applied to the alternating direction method of multipliers (ADMM) solver for CS-MRI [48], but with the goal to design powerful networks (ADMM-Net) rather than approximating variational methods. In contrast with encoding the sparsity on linear transform domain to network [48], ISTA-Net [34] goes beyond that to nonlinear transform domain sparsity. Moreover, OPINE-Net [36] considers the joint sampling and reconstruction of image CS that can be further released into purely data-driven prior to boost the performance. Nevertheless, few works consider the performance degradation of deep unrolling networks under mismatched imaging settings during inference, owing to the fixed network parameters.

Another approach that can combine the benefits of both deep learning and optimization methods is called plug-and-play (PnP) priors [49]. Its core idea is to plug a denoiser into the iterative optimization such that the image prior is implicitly defined by the denoiser itself. Many deep-learning-based denoisers have been utilized in the PnP framework to resolve compressive imaging problems, without the need for task-specific training [50, 51, 52, 53, 54, 55]. The main strength of PnP over deep unrolling is the generalizability (only one network is required to handle various compressive imaging problems [50]), but its performance often lags behind the deep unrolling network due to the lack of end-to-end training [56]. Meanwhile, it also faces the limitations of high computational complexity and difficulties of tuning parameters [53, 57]. In this work, we propose a dynamic proximal unrolling approach that improves the generalizability of deep unrolling considerably, while still enjoying the joint optimization of parameters via end-to-end learning.

2.2. Dynamic Neural Networks

Another line of related work is dynamic neural networks, which can dynamically adjust their structures or parameters for different inputs during inference, improving the accuracy, computational efficiency, and adaptiveness of neural networks [58]. Conditionally parameterized convolution (CondConv) [59] and dynamic convolutional neural network (DY-CNN) [60] introduce the soft attention on convolutional weights and improve the representation power on both classification and detection tasks. Dynamic filter networks [61] and HyperNetworks [62] adopt another network to produce dynamic filters and weights for CNNs and RNNs, respectively. Several recent works explore the employ of additional modules to adjust controlling

input for image restoration [63, 64, 65]. Compared with the existing works, we declare three main contributions of our work: first one is that we introduce the insight of dynamic neural network into deep unrolling scheme and present a dynamic proximal unrolling network. Secondly, we investigate and demonstrate that dynamic instance normalization layer is more powerful than dynamic convolution layer. Thirdly, we focus on compressive imaging problems which are more demanding for dynamic neural networks, and apply the proposed DPUNet to handle multiple imaging modalities with varying imaging conditions.

3. Method

In this section, we first introduce the iterative proximal optimization algorithm and the corresponding deep unrolling network. Then, we describe the dynamic proximal mapping module – a core component of the proposed dynamic proximal unrolling network. An overview of the proposed network is shown in Figure 3. What’s more, we present the applicability of the proposed method to other optimization frameworks. To eliminate the blocking artifacts, we further design an enhanced de-blocking version DPUNet⁺.

3.1. Proximal optimization and deep unrolling

In this paper, we adopt the proximal gradient descent (PGD) algorithm [18] for solving Eq.(1), which provides a general and efficient scheme to split the data-fidelity term and regularization term by alternating between the gradient descent step and proximal mapping step. Starting from the initial point x_0 , the whole iterations’ process of PGD can be written as

$$z_k = x_{k-1} - r_{k-1} \nabla \mathcal{D}(x_{k-1}), \quad (4)$$

$$x_k = \text{Prox}_{\lambda \mathcal{R}}(z_k), \quad (5)$$

where k denotes the iteration index, $\mathcal{D}(x) = \frac{1}{2} \|y - \Phi(x)\|_2^2$ is the data-fidelity term, ∇ denotes the vector differential operator that calculates the gradient of a given function, and r_k is the step-size of the k -th iteration. Eq.(4) can be understood as one-step gradient descent for dealing with the data-fidelity term. And $\text{Prox}_{\lambda \mathcal{R}}(z) := \arg\min_x \{\mathcal{R}(x) + \frac{1}{2\lambda} \|x - z\|_2^2\}$ denotes the proximal operator for handling the regularization term [19, 20] in Eq.(5).

Given the modular nature of the proximal optimization framework, the overall iterative procedure can be truncated and unrolled into a trainable reconstruction network by replacing all instances of the proximal operator $\text{Prox}_{\lambda \mathcal{R}}(\cdot)$ with a trainable deep convolutional neural network (CNN). The CNN is expected to learn/perform the proximal operator in Eq.(5), which can project the corrupted image into the clean image manifold. In this paper, we design a simple yet effective CNN consisting of five convolution (Conv) layers separated by an instance normalization (IN) layer [66] and a rectified linear unit (ReLU). IN was first proposed in style transfer and has shown significant improvement by normalizing feature statistics [66]. To

stabilize training [67], an identity skip connection is built between the input and output of the CNN. Mathematically, in the k -th iteration, the CNN maps z_k to x_k through

$$\begin{aligned} f_0 &= z_k, \\ f_j &= \text{ReLU}(\text{IN}_j(W_j * f_{j-1})), \quad j = 1, 2, 3, 4, \\ x_k &= W_5 * f_4 + f_0, \end{aligned}$$

where W_j represents the filters of convolution with a certain kernel and stride size, $*$ denotes the convolution operation, and the IN_j represents the j -th instance normalization operation as

$$\text{IN}_j(x) = \gamma_j \left(\frac{x - u(x)}{\sigma(x)} + \beta_j \right),$$

where γ_j and β_j are affine parameters learned from data; $u(x)$ and $\sigma(x)$ are the mean and standard deviation of the input x , computed across spatial dimensions independently for each feature channel and each sample.

3.2. Dynamic proximal mapping module

Given the training data, the resulting network can be trained end-to-end to learn the proximal mapping. However, the trainable parameters (*i.e.*, W_j, γ_j, β_j) of the CNN are usually fixed once trained, while different imaging conditions such as varying sampling-ratios or noise-levels all affect the performance of the learned proximal operator. To overcome this limitation, we make a step forward by proposing a dynamic proximal mapping module that can dynamically adjust the parameters of CNN according to different imaging conditions, to perform adaptively proximal operators. To this end, we use a set of fully connected networks, whose inputs are imaging parameters θ such as the sampling ratio and noise level, and outputs are the parameters (W_j, γ_j, β_j) of CNN, shown in Figure 4. The fully connected networks aim to generate and update the parameters of CNN, *i.e.*, the weights of convolution filters and affine parameters of IN layers. During the training stage, the proximal CNN and fully connected networks can be jointly trained. In the inference stage, given different imaging parameters, fully connected networks will adaptively adjust the learned proximal operator represented by the CNN, thus improving the representation capability.

Specifically, the imaging parameter is an auxiliary input that feeds into fully connected layers with the final layer outputting the convolution filter weights. The output then is reshaped into a 4D tensor of convolution filter weights and convolved with the input image. Based on empirical results in Section 4, we employ a single fully connected layer to directly learn the weights of convolution filters, which can be written as

$$W_j = A_j \theta + b_j, \quad j = 1, \dots, 5, \quad (6)$$

where $\theta \in \mathbb{R}^l$ is the l imaging parameters related to the imaging settings, $W_j \in \mathbb{R}^{m_j}$ represents the weights (m_j denotes the total number of parameters) of the j -th convolution layer in CNN, and $A_j \in \mathbb{R}^{m_j \times l}, b_j \in \mathbb{R}^{m_j}$ are the weight and bias of the corresponding fully connected layer.

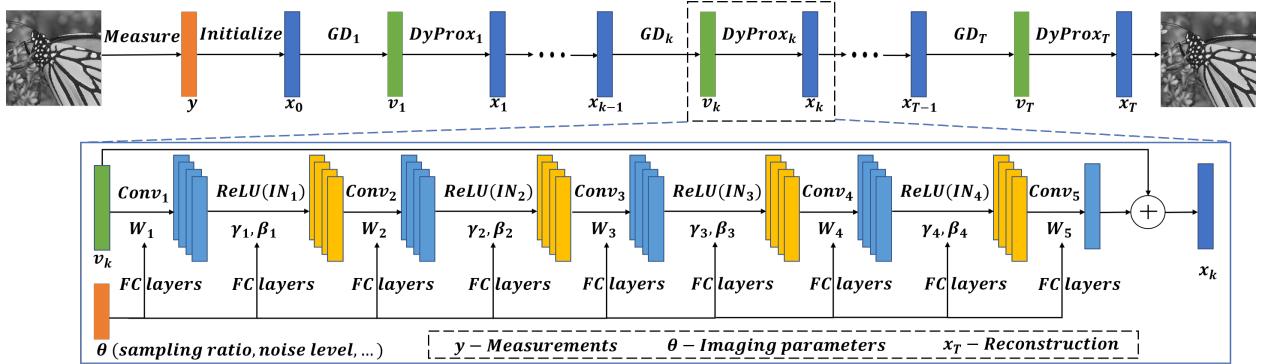


Figure 3: Overview of the proposed dynamic proximal unrolling network. Specifically, our network is unrolled by T iterations (empirically set to 10), and each iteration includes the gradient descent (GD) and the proposed dynamic proximal ($DyProx$) mapping module, which correspond to Eq. (4) and Eq. (5). The $DyProx$ consists of a deep convolutional neural network (CNN) and several fully connected (FC) layers. The CNN is designed as a combination of five convolutions (Conv) layers, each of which is separated by an instance normalization (IN) layer and a ReLU non-linearity. An identity skip connection is added between the input and output. The inputs of FC layers are imaging parameters θ such as sampling ratio and noise level, and the outputs are weights W of convolution (Conv) and affine parameters γ, β of instance normalization (IN). At the inference stage, given the imaging parameters, FC layers will adaptively determine parameters of CNN, which contribute to reconstructions together with gradient descent steps from given measurements.

In essence, these fully connected networks introduce a dynamic modulation mechanism for the weights of convolution with imaging parameters. When the fully connected networks contain only linear operations without non-linear activation function, the implicit assumption is that the network parameters for proximal mapping required by different imaging conditions are linear. Considering a special case, it is mathematically equivalent to directly learn the weights of convolution without fully connected layers when the imaging parameters $\theta = 0$, i.e., $W_j = b_j, j = 1, \dots, 5$. When the imaging parameters are non-zero, the fully connected networks will set up a connection between the CNN weight space and imaging parameters.

For instance normalization layers, we employ two fully connected layers separated by a ReLU (non-linearity) for the affine parameters of IN, respectively, which can be written as

$$Q_j = A_{j2} \text{ReLU}(A_{j1}\theta + b_{j1}) + b_{j2}, j = 1, \dots, 4, \quad (7)$$

where $Q_j \in \mathbb{R}^m$ represents the parameters $\gamma_j \in \mathbb{R}^m$ or $\beta_j \in \mathbb{R}^m$ of the j -th instance normalization, $A_{j1} \in \mathbb{R}^{m \times l}$, $A_{j2} \in \mathbb{R}^{m \times m}$ and $b_{j1} \in \mathbb{R}^m$, $b_{j2} \in \mathbb{R}^m$ are the weight and bias of the corresponding two fully connected layers. Note that continually increasing the fully connected layers can still improve the performance, but with more learnable parameters and higher computational complexity (See Section 4).

3.3. Dynamic proximal unrolling network

In essence, the structure of the proposed dynamic proximal unrolling network (dubbed DPUNet) is derived from the truncated proximal gradient descent algorithm, combined with a dynamic proximal mapping module.

When given training data, the inputs of the proposed DPUNet are the measurements y , corresponding physical forward model Φ and imaging parameters θ , which are sent to the reconstruction process and fully connected networks, respectively. So the proposed DPUNet can be trained by minimizing the following

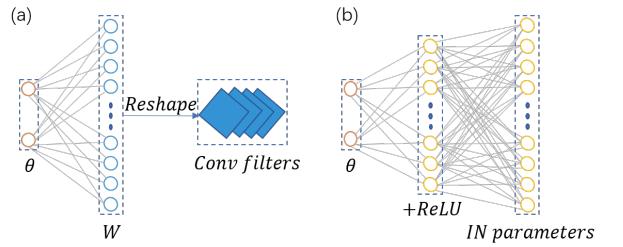


Figure 4: Illustration of the fully connected networks. Subfigure (a) illustrates the fully connected layer for convolution layers. The imaging parameter θ is an auxiliary input that feeds into a single fully-connected layer that outputs the weights W of convolution filters. Subfigure (b) describes two fully connected layers separated by a ReLU for instance normalization (IN) layers. The outputs are affine parameters of IN.

empirical risk:

$$\hat{\Theta} = \underset{\Theta}{\operatorname{argmin}} \quad \frac{1}{|\Gamma|} \sum_{(y, x) \in \Gamma} \mathcal{L}(x, f_{NN}(y, \Phi, g(\theta; \Theta))), \quad (8)$$

where $g(\cdot; \Theta)$ is the operation for generating the dynamic parameters via fully connected layers, and $\mathcal{L}(\cdot, \cdot)$ is exploited by the pixel-wise L_2 loss.

Given imaging parameters, fully connected networks will update the parameters of CNN to execute the proximal operator at each iteration. Finally, the outputs of the proposed network are reconstructed images, which are used to compute loss with the ground-truth. By back-propagation, the trainable parameters of our network, including the weight and bias of fully connected networks and the gradient step sizes, can be jointly optimized.

Once the model is trained, given the imaging parameters and measurements, the fully connected networks can adaptively generate parameters of CNN and the unrolling network performs the reconstructions. In this way, we can dynamically adjust the learned proximal operator based on the designed fully connected networks at the inference stage, thus enabling the

handling of different imaging settings and continuous parameter control.

3.4. Other optimization frameworks

A large variety of first-order proximal algorithms have been developed for solving Eq.(1) efficiently [18, 22, 20, 21]. In this paper, we present other two representative proximal optimization frameworks (*i.e.*, HQS and ADMM) to construct dynamic proximal unrolling networks. It has been observed that both of them perform well with the proposed dynamic proximal mapping module.

Unrolled-HQS. HQS tackles Eq.(1) by introducing an auxiliary variable z , leading to iteratively solving subproblems for z and x as

$$z_k = \operatorname{argmin}_z \left\{ \frac{1}{2} \|y - \Phi(z)\|_2^2 + \frac{\mu_k}{2} \|z - x_{k-1}\|_2^2 \right\}, \quad (9)$$

$$x_k = \operatorname{argmin}_x \left\{ \lambda \mathcal{R}(x) + \frac{\mu_k}{2} \|z_k - x\|_2^2 \right\}, \quad (10)$$

where k denotes the iteration index, μ_k indicates the penalty parameter.

Assuming that $\Phi(\cdot)$ is a linear measurement model, we still adopt the one-step gradient descent and proximal operator to deal with Eq.(9) and Eq.(10), respectively, which can be written as

$$z_k = z_{k-1} - r_{k-1} (\Phi^T (\Phi z_{k-1} - y) + \mu_k (z_{k-1} - x_{k-1})), \quad (11)$$

$$x_k = \operatorname{Prox}_{\frac{\lambda}{\mu_k} \mathcal{R}}(z_k), \quad (12)$$

where Φ^T denotes the transpose of the sampling matrix, Prox denotes the proximal operator, and r_k is step-size at k -th iteration.

Once the proximal optimization is determined, the next step is to unroll the iterative process into a dynamic proximal unrolling network by introducing the designed dynamic proximal mapping module. In this way, the free hyper-parameters (*i.e.*, r_k, μ_k) of HQS and dynamic proximal mapping module can be jointly learned via end-to-end training.

Unrolled-ADMM. Eq.(1) can also be solved by ADMM, whose iterations can be written as

$$x_k = \operatorname{Prox}_{\lambda \mathcal{R}}(z_{k-1} - u_{k-1}), \quad (13)$$

$$z_k = \operatorname{Prox}_{\frac{1}{\mu_k} \mathcal{D}}(x_k + u_{k-1}), \quad (14)$$

$$u_k = u_{k-1} + x_k - z_k, \quad (15)$$

where $\mathcal{D}(x) = \frac{1}{2} \|y - \Phi(x)\|_2^2$, μ_k still indicates the penalty parameter.

Similarly, we use the one-step gradient descent to tackle Eq.(14). Unrolled-ADMM can be derived by unrolling the corresponding iterative optimization and replacing the proximal operator with the designed dynamic proximal mapping module.

Based on empirical results, we demonstrate the proposed dynamic proximal mapping module can be embedded into different proximal optimization frameworks, and boost the generalizability of deep unrolling network. In this paper, we adopt the proximal gradient descent framework as the final choice, owing to its conciseness and effectiveness. More experimental details are presented in Section 4.

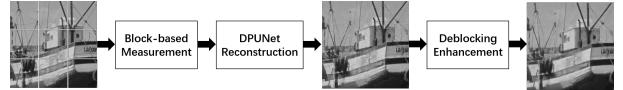


Figure 5: Illustration of the proposed DPUNet⁺ framework, which integrates a deblocking module with DPUNet to eliminate the blocking artifacts and improve the recovered image. (Best viewed with zoom)

3.5. Enhanced Deblocking Version: DPUNet⁺

For block-wise CS sampling, each image block is sampled and reconstructed independently, which will inevitably suffer from blocking artifacts and image quality degradation. To address this issue, we further present an enhanced deblocking framework of DPUNet, termed DPUNet⁺, which is shown in Figure 5. Specifically, we integrate a deblocking module with DPUNet to jointly reconstruct the image. It is worth mentioning that the deblocking module does not perform on each image block independently, but perform on the whole image formed by reassembling all approximated blocks. In this way, the proposed DPUNet⁺ can exploit the inter-block relationship to remove the blocking artifacts.

To this end, we adopt the same U-Net architecture [68] as [69] and add an identity skip connection between the input and output of the deblocking module. Considering that our single network needs to handle different sampling ratios simultaneously, we apply the above dynamic mechanism to all convolutions of U-Net and form a dynamic deblocking module, which can adapt to different blocking artifacts induced by different sampling ratios. Through end-to-end joint training, the single model of DPUNet⁺ can reconstruct the images and eliminate the blocking artifacts under various sampling ratios.

4. Experiments

In this section, we mainly focus on three representative compressive imaging modalities: image compressive sensing, compressive sensing magnetic resonance imaging (CS-MRI), and compressive phase retrieval (CPR), and detail experiments to evaluate the proposed method. We first describe the experimental setting including both physical observation models and implementation details. Then we compare our method against prior art on different tasks under varying imaging conditions and provide an in-depth discussion of the proposed method. Finally, we present an extension of DPUNet to simultaneously handle all these compressive imaging modalities under various imaging conditions via one single model.

4.1. Experimental Setting

4.1.1. Image CS

Image compressive sensing (CS) is a popular linear inverse problem, which enables image or video capturing under a sub-Nyquist sampling rate [2, 70]. In this paper, we focus on the block-based compressive sensing (BCS) task to validate the proposed method. Following common practice and for fair comparison, we use the same training data of 88,912 image blocks (each of size 33×33) provided in [34, 71] for DPUNet. For

training DPUNet⁺, we randomly extract the luminance component of 80,000 image blocks (each of size 99×99) from the public dataset DIV2K [72]. And we use two widely-used benchmark datasets Set11 [27] and BSD68 [73] for testing.

Given the sampling ratio, the measurement y is generated by $y = \Phi x$ with the sampling matrix Φ , where x is the vectorized version of an image block with a size of 33×33 . Specifically, we consider two types of sampling matrices, *i.e.*, fixed random Gaussian matrix (“fixed”) and data-driven adaptively learned matrix (“learned”). And we adopt the data-fidelity term $\mathcal{D}(x) = \frac{1}{2} \|y - \Phi x\|_2^2$, and compute its gradient $\nabla \mathcal{D}(x) = \Phi^T(\Phi x - y)$. Here Φ^T denotes the transpose of Φ . For fixed sampling matrix, we adopt the linear mapping method for initialization, same as ISTA-Net [34]. For learned sampling matrix, we focus on the binarized orthogonal matrix and employ an efficient convolutional implementation of Φ^T , same as OPINE-Net [36].

In order to verify the effectiveness of the proposed method to handle various imaging conditions with one single model, we consider two cases, *i.e.*, noiseless and noisy. For noiseless case, we simulate to generate the BCS measurement of each patch with the sampling ratio η uniformly sampled from $\{1\%, 4\%, 10\%, 25\%, 40\%, 50\%\}$ in the training stage. For noisy case, we further add Gaussian noise with the noise level α uniformly sampled from $[0, 50]$. Then we take $[\eta]$ for noiseless case and $[\eta, \alpha]$ for noisy case as imaging parameters, respectively.

4.1.2. CS-MRI

CS-MRI is an advanced technique for fast MRI through reconstruction of MR images from much fewer under-sampled measurements in k-space (*i.e.*, Fourier domain). Following common practice, we adopt the measurement matrix with the form $\Phi = PF$, where F represents the 2-dimensional Fourier transform and P is an under-sampling matrix taken as a commonly used pseudo radial sampling mask. We use the same training and testing brain medical images as ISTANet [34]. Here we still utilize the data-fidelity term $\mathcal{D}(x) = \frac{1}{2} \|y - \Phi x\|_2^2$, and compute its gradient $\nabla \mathcal{D}(x) = \Phi^T(\Phi x - y)$. For initialization, we project the under-sampled Fourier measurements to the image domain via the inverse Fourier transform. For each image, we simulate to generate subsampled measurements in Fourier domain with the sampling ratio η uniformly sampled from $\{20\%, 30\%, 40\%, 50\%\}$ and add Gaussian noise ϵ with the noise level α uniformly sampled from $[0, 50]$. Similarly, we take $[\eta, \alpha]$ as imaging parameters.

4.1.3. CPR

CPR is a representative non-linear inverse problem, concerned with the recovery of an underlying image from only the subsampled intensity of its complex transform. Mathematically, the measurements of CPR can be written as $y = |\Phi x|^2 + \epsilon$ with Poisson noise $\epsilon \sim \mathcal{N}(0, \alpha^2 \text{Diag}(|\Phi x|^2))$, where the term α controls the noise level in this problem [74, 51]. The data-fidelity term adopts the amplitude loss function $\mathcal{D}(x) = \frac{1}{2} \|\sqrt{y} - |\Phi x|\|_2^2$ in [74]. Notice $\mathcal{D}(x)$ involves complex number operations, we therefore adopt the Wirtinger derivatives [75] to com-

pute its gradient ¹, *i.e.*, $\nabla \mathcal{D}(x) = \frac{1}{2} \Phi^H \left((|\Phi x| - \sqrt{y}) \circ \frac{\Phi x}{|\Phi x|} \right)$, where \circ denotes the Hadamard (element-wise) product and Φ^H denotes the conjugate transpose of Φ . Considering the image is real-valued, we directly take the real part of the gradient $\nabla \mathcal{D}(x)$ to deal with the subtraction in Eq.(4). We test CPR methods under simulated coded Fourier measurements, where the measurement matrix with the form $\Phi = JFM$, where J represents a $m \times n$ matrix made from randomly sampled rows of a $n \times n$ identity matrix and M is a diagonal matrix with nonzero elements drawn uniformly from a unit circle in the complex plane [51]. Here we initialize x_0 with a vector of ones that works sufficiently well. To train the network, we follow the common practice that uses 160000 overlapping patches (with a size of 64×64) cropped from 400 images from the BSD dataset [76]. For each patch, we simulate to generate coded Fourier measurements with the sampling ratio η uniformly sampled from $\{30\%, 40\%, 50\%\}$ and add Poisson shot noise ϵ with the noise level α uniformly sampled from $[0, 30]$, and we take $[\eta, \alpha]$ as imaging parameters.

The inputs of our network are the measurements and corresponding imaging parameters. To keep consistent magnitude of back-propagated gradient, we normalize the maximum value of imaging parameters to one, *i.e.*, $\theta = [(100 \times \eta)/50, \alpha/50]$. We train all models using Adam optimizer with PyTorch on one Nvidia GeForce GTX 3080 Ti GPU. The models of DPUNet on BCS are trained in 200 epochs with the batch size of 64 and learning rate 10^{-4} . The models of DPUNet⁺ on BCS are trained using the pre-trained models of DPUNet as initialization in 20 epochs with the batch size of 64 and learning rate 10^{-4} . The model of DPUNet on CS-MRI is trained in 200 epochs with the batch size of 4 and learning rate 10^{-4} . The model of DPUNet on CPR is trained in 20 epochs with the batch size of 40 and learning rate 10^{-3} .

4.2. Performance Comparison

4.2.1. Validations on BCS

To verify the performance of the proposed method, we mainly compare it against two classic CS approaches, namely TVAL3 [77] and D-AMP [78], three learning-based approaches ReconNet [27], CSNet [79], and SCSNet [29], and two state-of-the-art deep unrolling approaches ISTA-Net⁺ [34] and OPINE-Net [36]. Specifically, CSNet, SCSNet, and OPINE-Net focus on data-driven learned sampling matrix to achieve very high performance. For end-to-end training methods, *i.e.*, ReconNet, CSNet, ISTA-Net, and OPINE-Net, we re-train the corresponding model for each sampling ratio setting. SCSNet also requires to update the greedy searching strategy for different sampling ratios. Table 1 shows the average PSNR and SSIM results of the competing methods for noiseless BCS reconstructions on “fixed” and “learned” sampling matrices with different sampling ratios on Set11 and BSD68, respectively. It can be found that our methods (both DPUNet and DPUNet⁺) can achieve very competitive performance on various sampling ratios with

¹A detailed derivation of the gradient formulation of the data-fidelity term in CPR is provided in Appendix A.

Table 1: Average PSNR/SSIM performance comparisons of various methods for noiseless BCS reconstructions on “fixed” and “learned” sampling matrices with different sampling ratios on Set11 and BSD68. The best results are labeled in **bold** and the second are underlined.

DATASETS	METHOD	TYPE	SAMPLING RATIO (PSNR/SSIM)					
			1%	4%	10%	25%	40%	50%
Set11	TVAL3	“fixed”	11.32/0.2896	18.39/0.4667	22.84/0.6638	27.84/0.8344	29.14/0.9055	33.37/0.9339
	D-AMP		5.21/0.0074	15.52/0.3742	21.18/0.6091	28.24/0.8464	33.47/0.9262	35.79/0.9460
	ReconNet		17.22/0.4032	19.77/0.5168	24.06/0.7223	25.46/0.7687	30.78/0.8932	31.48/0.8993
	ISTA-Net ⁺		17.45/0.4131	21.56/0.6240	26.49/0.8036	32.44/0.9237	36.02/0.9579	38.07/0.9706
	DPUNet		<u>17.48/0.4261</u>	<u>22.21/0.6594</u>	<u>27.31/0.8271</u>	<u>32.99/0.9300</u>	<u>36.48/0.9609</u>	<u>38.35/0.9719</u>
	DPUNet ⁺		18.09/0.4497	22.55/0.6741	27.54/0.8359	33.25/0.9328	36.67/0.9616	38.52/0.9722
	CSNet	“learned”	19.87/0.4977	23.93/0.7338	27.59/0.8575	31.70/0.9274	36.15/0.9625	37.19/0.9700
	SCSNet		21.04/0.5562	25.08/0.7626	28.48/0.8616	33.43/0.9373	36.92/0.9666	39.01/0.9769
	OPINE-Net		20.02/0.5362	25.04/0.7730	29.33/0.8825	34.44/0.9491	38.13/0.9719	<u>39.88/0.9790</u>
	DPUNet		20.08/0.5201	<u>25.22/0.7783</u>	<u>29.40/0.8831</u>	<u>34.55/0.9491</u>	37.89/0.9706	39.71/0.9783
	DPUNet ⁺		<u>20.37/0.5312</u>	25.42/0.7808	29.50/0.8848	34.61/0.9502	<u>38.10/0.9716</u>	39.90/0.9790
BSD68	TVAL3	“fixed”	12.84/0.3177	20.20/0.4715	23.54/0.6165	26.97/0.7734	29.60/0.8603	31.32/0.8998
	D-AMP		6.47/0.0085	17.09/0.3621	21.91/0.5288	25.55/0.6894	27.93/0.7803	29.98/0.8324
	ReconNet		19.00/0.4018	21.23/0.4812	23.88/0.6400	24.99/0.6600	28.96/0.8499	30.13/0.8798
	ISTA-Net ⁺		19.18/0.4201	22.34/0.5573	25.30/0.7001	29.31/0.8507	32.16/0.9158	34.01/0.9421
	DPUNet		<u>19.18/0.4263</u>	<u>22.71/0.5762</u>	<u>25.70/0.7170</u>	<u>29.63/0.8588</u>	<u>32.53/0.9209</u>	<u>34.31/0.9452</u>
	DPUNet ⁺		19.88/0.4483	23.10/0.5921	25.93/0.7270	29.85/0.8644	32.79/0.9246	34.60/0.9483
	CSNet	“learned”	21.91/0.4958	24.63/0.6564	27.02/0.7864	30.22/0.8918	32.53/0.9206	34.82/0.9590
	SCSNet		21.88/0.5250	24.98/0.6843	27.13/0.7785	30.76/0.8973	33.86/0.9348	35.67/0.9640
	OPINE-Net		21.80/0.4972	<u>25.16/0.6841</u>	27.54/0.7966	<u>31.28/0.9034</u>	<u>34.35/0.9482</u>	<u>36.12/0.9646</u>
	DPUNet		<u>21.96/0.5049</u>	25.00/0.6765	<u>27.60/0.7988</u>	31.26/0.9022	34.13/0.9461	35.90/0.9628
	DPUNet ⁺		22.16/0.5135	25.14/0.6819	27.68/0.8024	31.40/0.9058	34.38/0.9493	36.19/0.9653

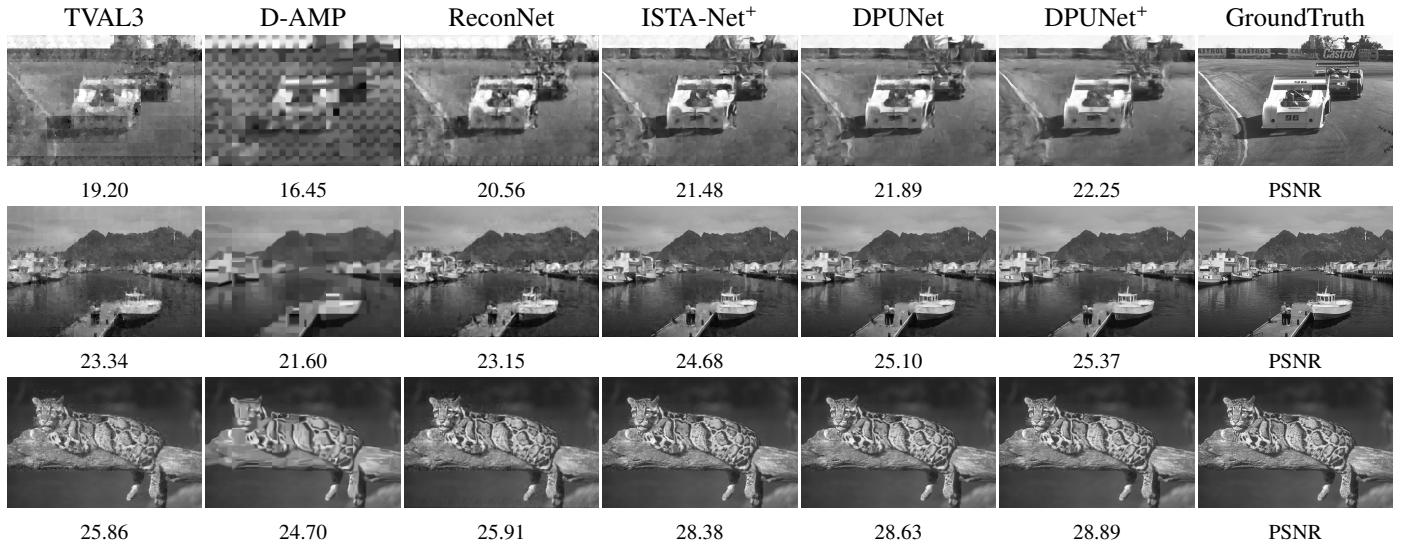


Figure 6: Reconstructed images on BSD68 dataset and corresponding PSNRs (dB) from noiseless BCS measurements on “fixed” sampling matrices with sampling ratios ($\eta = 4\%$, $\eta = 10\%$ and $\eta = 25\%$, from top to bottom) with six image CS algorithms.

one single trained model. Moreover, benefited from data-driven sampling matrix and the proposed dynamic deblocking module, our DPUNet⁺ can produce consistently better reconstruction results. We show visual comparisons of these algorithms on “fixed” and “learned” sampling matrices in Figure 6 and Figure 7, respectively. It can be seen that the proposed methods produce more accurate and clearer reconstructed images than other competing algorithms. In addition, our DPUNet⁺ can eliminate the blocking artifacts and improve the quality of reconstructed images.

Moreover, we further conduct noisy BCS experiments on the fixed sampling matrix. Table 2 shows performance of four methods on the public dataset Set11 under various sampling ratios and noise levels. It can be observed that our method outperforms other methods under all imaging conditions via one single trained model. By contrast with ISTA-Net⁺, our method avoids the cumbersome retraining requirement, which is beneficial for real applications. In addition, it is worth mentioning that our method can generalize well to an unseen sampling ratio setting during training, as $\eta = 30\%$.

Table 2: Average PSNR (dB) performance comparisons for noisy BCS reconstructions on Set11 under different imaging conditions (η and α denote the sampling ratio and noise level, respectively). The best results are labeled in **bold** and the second are underlined.

η	α	TVAL3	D-AMP	ISTA-Net ⁺	DPUNet
50%	10	26.83	30.63	<u>31.42</u>	32.00
	30	19.08	25.92	<u>26.87</u>	27.20
	50	14.88	23.56	<u>24.30</u>	24.79
40%	10	26.31	29.29	<u>30.53</u>	31.20
	30	19.33	25.08	<u>26.21</u>	26.59
	50	15.38	22.84	<u>23.78</u>	24.17
30%	10	25.57	27.58	<u>29.45</u>	30.04
	30	19.47	23.93	<u>25.28</u>	25.70
	50	15.81	22.00	<u>22.89</u>	23.29
10%	10	21.91	20.87	<u>24.63</u>	25.40
	30	18.83	19.68	<u>21.85</u>	22.26
	50	16.34	18.53	<u>20.02</u>	20.32
1%	10	11.26	5.20	<u>17.25</u>	17.27
	30	10.90	5.19	<u>16.46</u>	16.58
	50	10.55	5.18	<u>15.44</u>	15.79

4.2.2. Validations on CS-MRI

We mainly compare DPUNet with six competing methods for CS-MRI, including three classic algorithms RecPF [23], FCSA [80] and D-MRI [81], the deep unrolling network ISTA-Net⁺ [34], two plug-and-play (PnP) approaches IRCNN [54] and TFPnP [57]. We train separate models of ISTA-Net⁺ for each sampling ratio in 200 epochs, and retrain TFPnP to adapt to the size of our MR images (*i.e.*, 256×256). Table 3 shows quantitative performance comparisons on 50 brain medical images under different imaging conditions. It can be seen that DPUNet significantly outperforms other competing algorithms under various imaging conditions with only one trained model. The fact that one single DPUNet model performs so well on almost all sampling ratios, particularly in comparison to the ISTA-Net⁺ models, which were trained separately per sampling ratio, again demonstrates that DPUNet generalizes across measurement matrices. What’s more, compared with the PnP approaches IRCNN and TFPnP, DPUNet enjoys higher reconstruction performance owing to end-to-end training and faster inference speed. In Figure 8, we show the reconstructions of three brain MR images and corresponding PSNRs under different sampling ratios ($\eta = 20\%, 30\%, 40\%$) and noise levels ($\alpha = 10, 30, 50$) with seven algorithms. It can be observed that DPUNet can reconstruct more details and sharper edges, especially in case of severe noise.

4.2.3. Validations on CPR

We mainly compare DPUNet with two state-of-the-art approaches (BM3D-prGAMP [55] and prDeep [51]) for CPR. We use their respective authors’ implementations and adopt the same set of twelve images (resized to 128×128) used in [51] to quantitatively evaluate different CPR methods. The results of performance comparisons for CPR are summarized in Table 4. It can be seen that DPUNet can handle these imaging conditions with state-of-the-art results via one single trained network. The visual comparison can be found in Figure 9. It can be found that

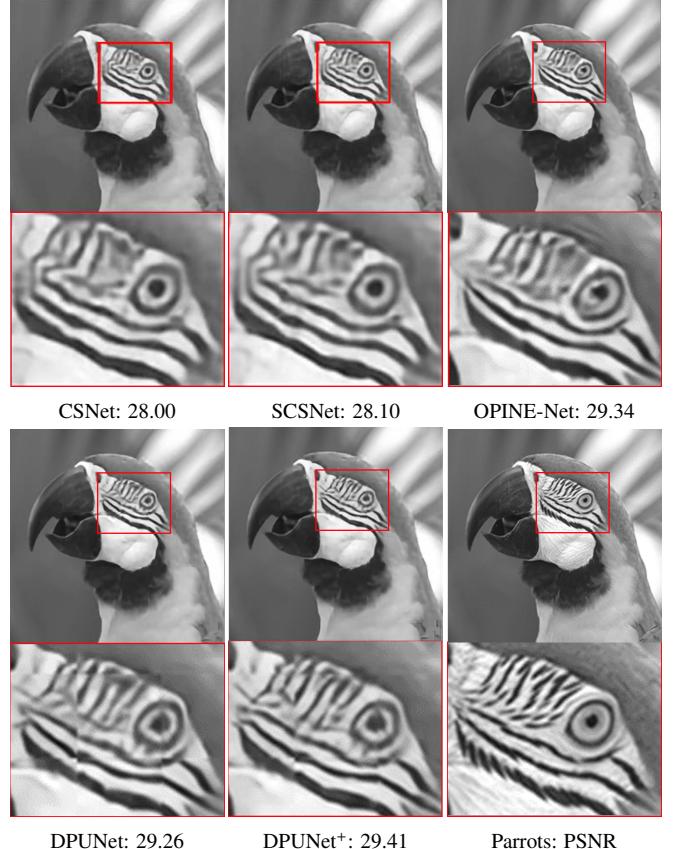


Figure 7: Reconstructed images and corresponding PSNRs (dB) from noiseless BCS measurements on “learned” sampling matrix with the sampling ratio ($\eta = 10\%$) with five image CS algorithms.

our method can still effectively recover desired images, and produce clearer results than other competing methods. In addition, it is worth mentioning that our method trained on a small patch size of 64×64 can perform well on a size of 128×128 , which demonstrates the robustness of our method for different measurement matrices. And it is very important for CPR because its forward measurement matrix is related to the object size.

4.3. Discussion

4.3.1. Effects of different dynamic architectures

To give some insights of the proposed dynamic proximal mapping module, we conduct an ablation study on different dynamic network architectures, including (1) PUNet: the basic proximal unrolling network without fully connected layers; (2) PUNet + DConv1: PUNet with dynamic convolution via one fully connected layer; (3) PUNet + DConv2: PUNet with dynamic convolution via two fully connected layers; (4) PUNet + DIn1: PUNet with dynamic instance normalization via one fully connected layer; (5) PUNet + DIn2: PUNet with dynamic instance normalization via two fully connected layers; (6) PUNet + DIn3: PUNet with dynamic instance normalization via three fully connected layers; (7) PUNet + DConv1In1: PUNet with dynamic convolution and instance normalization via one fully connected layer; (8) PUNet + DConv2In1: PUNet with dynamic convolution via two fully connected layers and

Table 3: Average PSNR (dB) and Run Time (s) performance comparisons for CS-MRI on 50 brain medical images under different imaging conditions (η and α denote the sampling ratio and noise level, respectively). The best results are labeled in **bold** and the second-best results are underlined.

ALGORITHM	$\eta = 20\%$			$\eta = 30\%$			$\eta = 40\%$			$\eta = 50\%$			TIME CPU/GPU
	$\alpha = 10$	30	50										
RecPF	31.41	25.06	21.33	32.47	24.76	20.61	33.05	24.37	19.96	33.46	24.00	19.41	0.39s/-
FCSA	30.71	24.11	19.19	31.31	23.07	17.93	31.28	22.28	17.21	31.29	21.78	16.77	0.64s/-
DMRI	25.77	20.23	22.06	32.15	24.66	20.69	31.80	23.71	19.64	31.35	22.92	18.80	10.28s/-
IRCNN	33.77	30.58	29.09	34.64	31.34	29.70	35.18	31.75	30.02	35.58	32.02	30.18	-/12.16s
ISTA-Net ⁺	32.50	30.78	29.22	34.84	31.72	29.85	35.37	32.04	30.10	35.87	32.25	30.22	-/0.03s
TFPnP	34.44	<u>31.23</u>	<u>29.58</u>	<u>35.30</u>	<u>31.82</u>	<u>30.11</u>	<u>35.83</u>	<u>32.16</u>	<u>30.31</u>	<u>36.19</u>	<u>32.36</u>	<u>30.47</u>	-/0.05s
DPUNet	34.39	31.44	29.74	35.41	32.11	30.32	35.98	32.43	30.59	36.34	32.62	30.70	-/0.04s

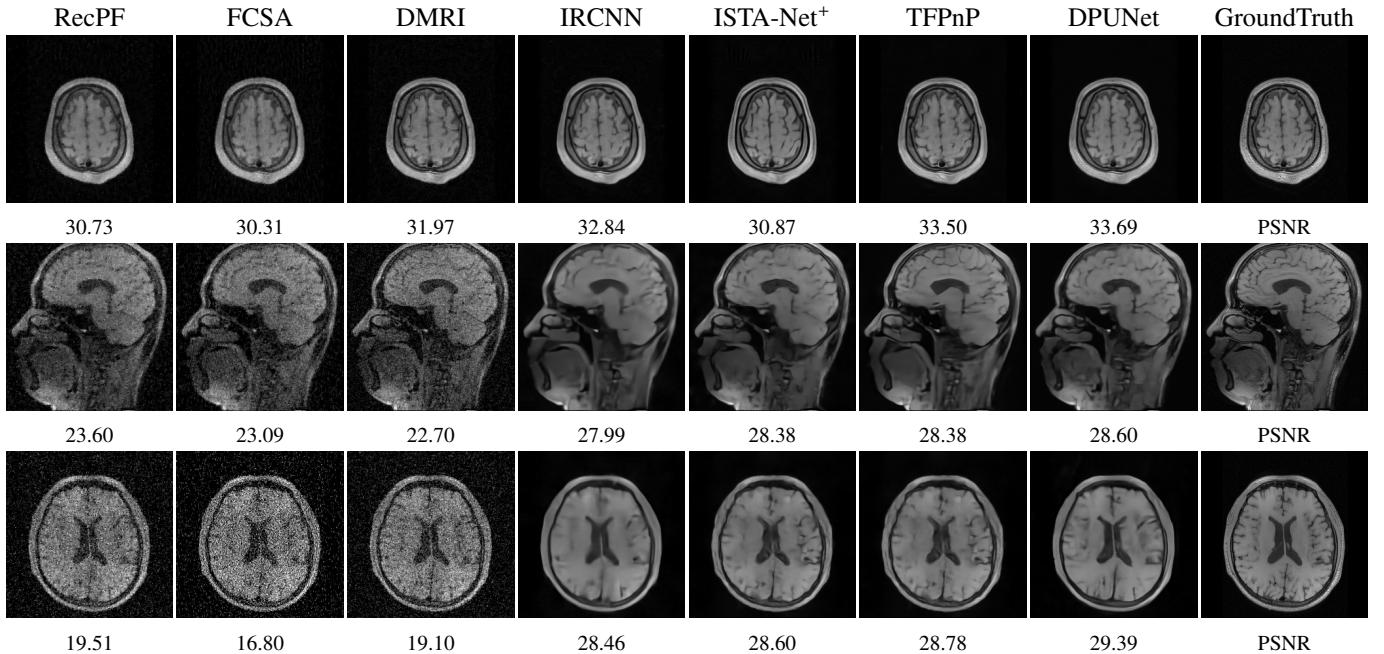


Figure 8: Reconstructed images and corresponding PSNRs (dB) for CS-MRI under various sampling ratios ($\eta = 20\%, 30\%, 40\%$, from top to bottom) and different noise levels ($\alpha = 10, 30, 50$, from top to bottom) with seven algorithms.

Table 4: Average PSNR (dB) performance comparisons for CPR on twelve images under different imaging conditions (η and α denote the sampling ratio and noise level, respectively). The best results are labeled in **bold** and the second-best results are underlined.

η	α	prGAMP	prDeep	DPUNet
50%	10	<u>31.56</u>	30.64	33.18
	30	27.46	<u>27.69</u>	28.63
40%	10	<u>31.26</u>	30.10	32.34
	30	26.77	<u>27.46</u>	28.37
30%	10	29.05	<u>30.00</u>	30.96
	30	26.47	<u>26.99</u>	27.76

instance normalization via one fully connected layer; (9) PUNet + DConv2In2: PUNet with dynamic convolution and instance normalization via two fully connected layers; (10) DPUNet: PUNet with dynamic convolution via one fully connected layer and dynamic instance normalization via two fully connected layers. All models are trained and tested for image CS tasks

on the same experimental setting with noise-free ².

To compare the performance, we show PSNRs (dB) of BCS reconstructions ($\eta = 10\%, 25\%, 40\%$) on Set11 with different structures and the number of model parameters (Million), provided in Table 5. It can be seen that the performance of PUNet could be significantly improved by using dynamic convolution or dynamic instance normalization with two fully connected layers, especially by using them together. Considered by the complexity of the model, our final choice (*i.e.*, DPUNet) achieves performance near to the top with a reasonable number of parameters.

4.3.2. Effects of different optimization frameworks

To provide the insight into the choice of optimization framework, we compare the performance of the proposed three unrolling frameworks on image CS tasks, including their unrolled networks and corresponding dynamic versions. In Table 6, it

²Notice that unless otherwise specified, the following experiments are conducted on the fixed sampling matrix under noiseless case.

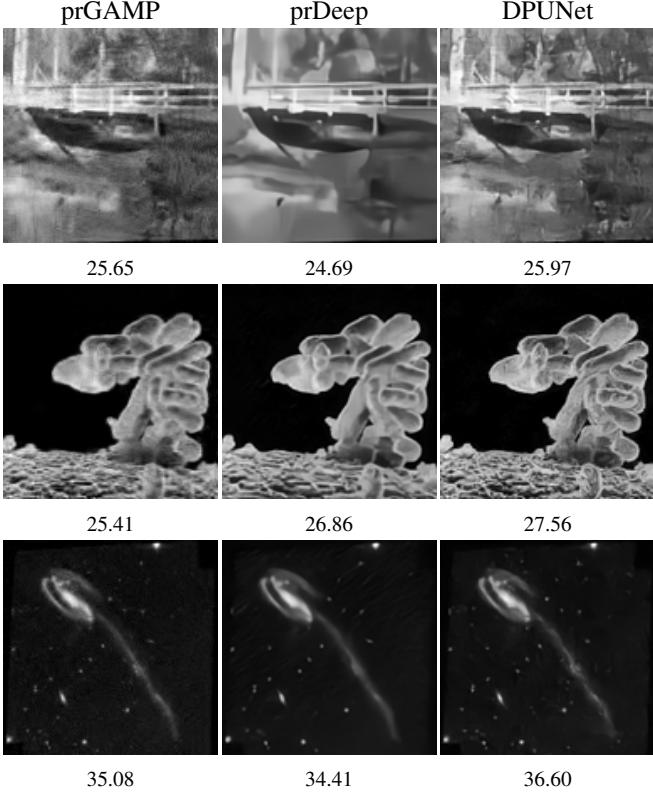


Figure 9: Reconstructed images and corresponding PSNRs (dB) for CPR under different sampling ratios ($\eta = 30\%, 40\%, 50\%$, from top to bottom) and noise level ($\alpha = 10$) with three algorithms.

Table 5: Comparisons of different structures in the proposed dynamic proximal mapping module. We show PSNRs (dB) of BCS reconstructions under multiple sampling ratios on Set11 with different architectures and the number of model parameters (Million). The best results are labeled in **bold** and the second-best results are underlined.

ARCHITECTURES	SAMPLING RATIOS			PARAMS
	10%	25%	40%	
PUNet	26.57	31.80	34.85	1.12
PUNet + DConv1	26.87	32.47	35.50	2.24
PUNet + DConv2	27.05	32.73	36.05	72.65
PUNet + DIn1	26.87	32.39	35.70	1.13
PUNet + DIn2	27.15	32.74	36.12	1.46
PUNet + DIn3	27.25	32.85	36.23	1.79
PUNet + DConv1In1	26.49	32.04	34.84	2.25
PUNet + DConv2In1	26.76	32.51	35.92	72.66
PUNet + DConv2In2	27.33	32.99	36.42	72.98
DPUNet	27.31	<u>32.99</u>	36.48	2.58

can be seen that the proposed dynamic proximal mapping module can consistently boost the performance of unrolling network derived from PGD, HQS, and ADMM, with average performance gain 1.45dB, 1.60dB, and 1.45dB, respectively. Meanwhile, DPUNet-PGD achieves the highest performance under various sampling ratios compared against other networks, chosen as the final choice.

Table 6: Comparisons of unrolling networks based on PGD, HQS, and ADMM. We show PSNRs (dB) of image compressive sensing reconstructions under various sampling ratios on Set11.

EXTENSIONS	SAMPLING RATIOS			
	10%	25%	40%	50%
PUNet-PGD	26.57	31.80	34.85	36.12
DPUNet-PGD	27.31	32.99	36.48	38.35
PUNet-HQS	26.05	31.18	34.16	35.55
DPUNet-HQS	26.74	32.49	36.08	38.04
PUNet-ADMM	26.16	31.30	34.33	35.80
DPUNet-ADMM	26.80	32.51	36.05	38.02

4.3.3. Generalizability of DPUNet

To further investigate the generalizability of DPUNet, we train and test DPUNet on consistent sampling ratios, denoted by DPUNet-optimal. Figure 10 shows the comparison between the "DPUNet-optimal", "DPUNet", and "PUNet" models for image compressive sensing under multiple sampling ratios. Note that "DPUNet-optimal" denotes the (three) models of DPUNet separately trained and tested on each sampling ratio, which is expected to get the optimal results. "DPUNet" and "PUNet" indicate the single model trained and tested on all sampling ratios, and the main difference between them is that "DPUNet" adopts the proposed dynamic module which "PUNet" lacks.

It can be observed that a single trained model of DPUNet can achieve very close performance (the average PSNR difference is about 0.24 dB) with the optimal results, while there is a large gap (the average PSNR difference is about 2.18 dB) between "PUNet" model and the optimal model. Overall, we demonstrate that the proposed dynamic proximal mapping module can significantly boost the generalizability of deep unrolling networks, avoiding time-consuming and storage-consuming re-training.

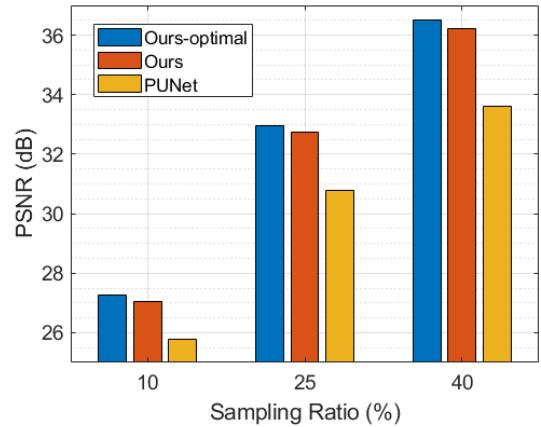


Figure 10: The model generalizability test of our method for image compressive sensing under multiple sampling ratios. "Ours-optimal" denotes the (three) models trained and tested on consistent sampling ratios. "Ours" is the single model trained and tested on all sampling ratios. "PUNet" is the degraded version of "Ours" without a dynamic module, and still the single model trained and tested on all sampling ratios.

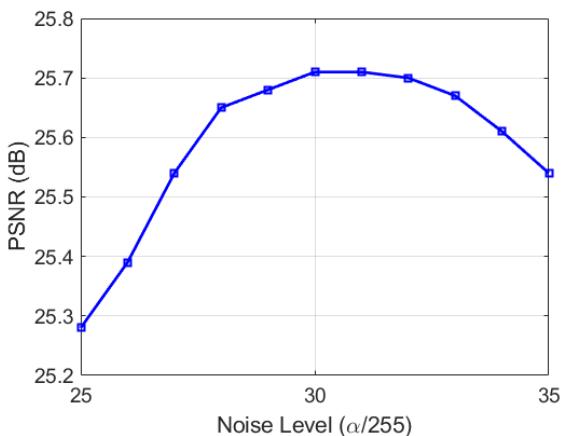


Figure 11: The model robustness test of DPUNet with mismatched imaging parameters on noisy BCS measurements. The real imaging conditions are sampling ratio $\eta = 30\%$, noise level $\alpha = 30$. Here we test DPUNet under a range of parameter values of the noise level as the input and show corresponding reconstruction results (PSNR) on Set11.

4.3.4. Robustness on imaging parameters mismatch

In practical applications, while the sampling ratio can be accurately measured, the noise level or signal-to-noise ratio (SNR) is often estimated by related methods [82, 83, 84] since the ground truth is unknown. Thereby, DPUNet should be robust to mismatched imaging parameters, mainly under noisy conditions. To analyze the robustness of DPUNet for the mismatched imaging parameters, we use a range of parameter values of the noise-level as the input at the inference stage, and show average reconstruction results (PSNRs) on Set11 test set from noisy BCS measurements (sampling ratio $\eta = 30\%$, noise level $\alpha = 30$). Figure 11 illuminates that DPUNet is robust to the inaccuracy of imaging parameters – reaching similar reconstruction results under a range of mismatched imaging parameters. The visual comparisons can see Figure 12, which also shows visually similar results of reconstructed images.

Table 7: Average PSNRs (dB) of reconstruction results with our single trained model (Ours*) for different imaging modalities under varying imaging conditions. The baselines are the models of DPUNet separately trained on a single imaging task (BCS, CS-MRI, and CPR), which represent the performance standard.

Task	η	α	Ours*	Baselines
BCS	30%	10	29.97	30.04
		30	25.57	25.70
	50%	10	31.89	31.99
		30	27.03	27.20
CS-MRI	30%	10	35.08	35.41
		30	31.61	32.11
	50%	10	36.00	36.34
		30	32.07	32.62
CPR	30%	10	30.85	30.96
		30	27.75	27.75
	50%	10	33.11	33.14
		30	28.55	28.63

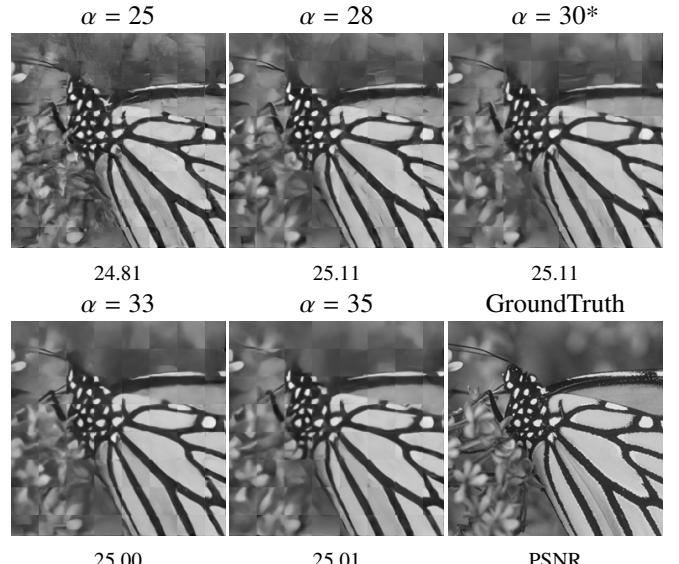


Figure 12: Reconstructed images and corresponding PSNRs (dB) from noisy BCS measurements ($\eta = 30\%$, $\alpha = 30$) under a range of parameter values of the noise-level at the inference stage.

4.4. Extension for multiple compressive imaging tasks

Inspired by the same network structure for different compressive imaging modalities, we explore the potential of our method to handle multiple imaging tasks with one single trained model. To verify this claim, we present an extension of our method and run a mixed experiment, where we consider three imaging modalities mentioned above under different sampling ratios with varying noise levels. To implement this, we simulate to generate the BCS measurements with the sampling ratio η uniformly sampled from $\{1\%, 4\%, 10\%, 25\%, 40\%, 50\%\}$ from CS training data-set, CS-MRI measurements with the sampling ratio η uniformly sampled from $\{20\%, 30\%, 40\%, 50\%\}$ from MRI training data-set, and the under-sampling coded diffraction measurements uniformly sampled from $\{30\%, 40\%, 50\%\}$ from CPR training data-set, all adding noise with α uniformly sampled from $[0, 50]$.

Moreover, we add an extra parameter κ to represent the imaging modality³, and the other parameters are related to the imaging condition $\{\eta, \alpha\}$. During training, the BCS, CS-MRI, and CPR training data pairs that include image patches and corresponding measurements are alternately fed to our network, together with imaging parameters $\{\kappa, \eta, \alpha\}$. The network is trained in 200000 iterations using pixel-wise L_2 loss and Adam optimizer with learning rate 10^{-4} . It takes about 18 hours to train the model.

The reconstruction results with the extension of our method for all imaging modalities and imaging conditions are provided in Table 7. We can see that our method can not only handle various imaging conditions but also totally different imaging modalities via one single model without retraining. Here we

³For simplicity, we input the parameters $\kappa = 1$ for dealing with BCS task, $\kappa = 2$ for CS-MRI, and $\kappa = 3$ for CPR.

also compare with the results of our baselines, which represent the models of DPUNet trained on a single imaging task (BCS, CS-MRI, or CPR), respectively. These baseline models can be regarded as the upper bound. It can be seen that the extension of our method still achieves close results to the baselines, despite small performance degradation. It means with the same network structure, our method can handle different imaging tasks via one single trained model without losing much accuracy. We show reconstructed results of a single trained model for multiple imaging modalities with noise level ($\alpha = 10$) and different sampling ratios ($\eta = 10\%$ for BCS, $\eta = 20\%$ for CS-MRI, $\eta = 50\%$ for CPR) in Figure 1. It can be found that the proposed network is flexible for various imaging conditions and universal for different imaging modalities.

5. Conclusion

In this paper, we propose a dynamic proximal unrolling network (DPUNet) for a variety of compressive imaging problems under varying imaging conditions. The main contribution of the proposed method is developing a dynamic proximal mapping module, which can dynamically update parameters of the proximal network at the inference stage and make it adapt to different imaging settings and even imaging modalities. As a result, the proposed method can handle a wide range of compressive imaging tasks, including image compressive sensing, CS-MRI, and compressive phase retrieval under varying imaging conditions via one single trained model. Moreover, an enhanced version DPUNet⁺ is developed for block-wise CS sampling, which integrates a dynamic deblocking module and reconstructs jointly with DPUNet to further improve the performance. Experimental results demonstrate the effectiveness and competitive performance of the proposed method. Thereby, we envision the proposed method to be applied to embedded mobile devices where storage and computational resources demands become prohibitive, and to handle a variety of imaging tasks via only one trained model.

6. Acknowledgements

We thank the anonymous reviewers for their helpful comments and suggestions to improve this paper. This work was supported by the National Science Foundation of China under Grants no. 62027801 and U1833203.

Appendix A. Formulas for the gradient in CPR

Proof. We start with

$$\begin{aligned} \mathcal{D}(x) &= \frac{1}{2} \|\sqrt{y} - |\Phi x|\|_2^2 \\ &= \frac{1}{2m} \sum_{r=1}^m \left(\sqrt{y_r} - \sqrt{x^T (\phi_r^H \phi_r)^T x^*} \right)^2, \end{aligned} \quad (\text{A.1})$$

where y_r is the r -th element of y , ϕ_r is the r -th row vector of Φ , and x^* denotes the conjugate of x .

Concerning Wirtinger derivatives of real valued functions over complex variables [75], we establish

$$\frac{\partial \mathcal{D}(x)}{\partial x} = \frac{1}{m} \sum_{r=1}^m \left(\sqrt{x^T (\phi_r^H \phi_r)^T x^*} - \sqrt{y_r} \right) \frac{(\phi_r^H \phi_r)^T x^*}{2 \sqrt{x^T (\phi_r^H \phi_r)^T x^*}}. \quad (\text{A.2})$$

This gives

$$\begin{aligned} \nabla \mathcal{D}(x) &= \left(\frac{\partial \mathcal{D}(x)}{\partial x} \right)^* \\ &= \frac{1}{2m} \sum_{r=1}^m \left(\sqrt{x^T (\phi_r^H \phi_r)^T x^*} - \sqrt{y_r} \right) \frac{(\phi_r^H \phi_r)x}{\sqrt{x^T (\phi_r^H \phi_r)^T x^*}} \\ &= \frac{1}{2} \Phi^H \left((|\Phi x| - \sqrt{y}) \circ \frac{\Phi x}{|\Phi x|} \right), \end{aligned} \quad (\text{A.3})$$

where \circ denotes the Hadamard (or elementwise) multiplication. \square

References

- [1] D. L. Donoho, Compressed sensing, IEEE Transactions on information theory 52 (4) (2006) 1289–1306.
- [2] A. Liutkus, D. Martina, S. Popoff, G. Chardon, O. Katz, G. Lerosey, S. Gigan, L. Daudet, I. Carron, Imaging with nature: Compressive imaging using a multiply scattering medium, Scientific reports 4 (1) (2014) 1–7.
- [3] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, R. G. Baraniuk, Single-pixel imaging via compressive sampling, IEEE signal processing magazine 25 (2) (2008) 83–91.
- [4] R. Kerviche, N. Zhu, A. Ashok, Information-optimal scalable compressive imaging system, in: Computational Optical Sensing and Imaging, Optical Society of America, 2014, pp. CM2D–2.
- [5] A. C. Sankaranarayanan, C. Studer, R. G. Baraniuk, Cs-muvi: Video compressive sensing for spatial-multiplexing cameras, in: 2012 IEEE International Conference on Computational Photography (ICCP), IEEE, 2012, pp. 1–10.
- [6] M. Lustig, D. L. Donoho, 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 58 (6) (2007) 1182–1195.
- [7] M. Lustig, D. L. Donoho, J. M. Santos, J. M. Pauly, Compressed sensing mri, IEEE signal processing magazine 25 (2) (2008) 72–82.
- [8] F. Rousset, N. Ducros, A. Farina, G. Valentini, C. D’Andrea, F. Peyrin, Adaptive basis scan by wavelet prediction for single-pixel imaging, IEEE Transactions on Computational Imaging 3 (1) (2016) 36–46.
- [9] L. Gao, J. Liang, C. Li, L. V. Wang, Single-shot compressed ultrafast photography at one hundred billion frames per second, Nature 516 (7529) (2014) 74–77.
- [10] X. Yuan, D. J. Brady, A. K. Katsaggelos, Snapshot compressive imaging: Theory, algorithms, and applications, IEEE Signal Processing Magazine 38 (2) (2021) 65–88.
- [11] M. L. Moravec, J. K. Romberg, R. G. Baraniuk, Compressive phase retrieval, in: Wavelets XII, Vol. 6701, International Society for Optics and Photonics, 2007, p. 670120.
- [12] H. Ohlsson, A. Yang, R. Dong, S. Sastry, Cprl—an extension of compressive sensing to the phase retrieval problem, Advances in Neural Information Processing Systems 25 (2012) 1367–1375.
- [13] S. Ma, W. Yin, Y. Zhang, A. Chakraborty, An efficient algorithm for compressed mr imaging using total variation and wavelets, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2008, pp. 1–8.
- [14] W. Dong, G. Shi, X. Li, Y. Ma, F. Huang, Compressive sensing via non-local low-rank regularization, IEEE Transactions on Image Processing 23 (8) (2014) 3618–3632.
- [15] J. Zhang, D. Zhao, W. Gao, Group-based sparse representation for image restoration, IEEE Transactions on Image Processing 23 (8) (2014) 3336–3351.

- [16] J. Xu, Z. Fu, X. Zhu, Image compressive sensing recovery via group residual based nonlocal low-rank regularization, Neurocomputing 449 (2021) 315–329.
- [17] Z. Zha, X. Zhang, Q. Wang, L. Tang, X. Liu, Group-based sparse representation for image compressive sensing reconstruction with non-convex regularization, Neurocomputing 296 (2018) 55–63.
- [18] P. L. Combettes, 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.
- [19] M. V. Afonso, J. M. Bioucas-Dias, M. A. Figueiredo, Fast image recovery using variable splitting and constrained optimization, IEEE transactions on image processing 19 (9) (2010) 2345–2356.
- [20] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM Journal on Imaging Sciences 2 (1) (2009) 183–202.
- [21] D. Geman, C. Yang, Nonlinear image recovery with half-quadratic regularization, IEEE transactions on Image Processing 4 (7) (1995) 932–946.
- [22] 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 3 (1) (2011) 1–122.
- [23] J. Yang, Y. Zhang, W. Yin, A fast alternating direction method for tvl1-l2 signal reconstruction from partial fourier data, IEEE Journal of Selected Topics in Signal Processing 4 (2) (2010) 288–297.
- [24] H. Yao, F. Dai, S. Zhang, Y. Zhang, Q. Tian, C. Xu, Dr2-net: Deep residual reconstruction network for image compressive sensing, Neurocomputing 359 (2019) 483–493.
- [25] J. Du, X. Xie, C. Wang, G. Shi, X. Xu, Y. Wang, Fully convolutional measurement network for compressive sensing image reconstruction, Neurocomputing 328 (2019) 105–112.
- [26] L. Sun, Y. Wu, B. Shu, X. Ding, C. Cai, Y. Huang, J. Paisley, A dual-domain deep lattice network for rapid mri reconstruction, Neurocomputing 397 (2020) 94–107.
- [27] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, 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.
- [28] W. Shi, F. Jiang, S. Liu, D. Zhao, Image compressed sensing using convolutional neural network, IEEE Transactions on Image Processing 29 (2019) 375–388.
- [29] W. Shi, F. Jiang, S. Liu, D. Zhao, Scalable convolutional neural network for image compressed sensing, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 12290–12299.
- [30] S. Zhou, Y. He, Y. Liu, C. Li, J. Zhang, Multi-channel deep networks for block-based image compressive sensing, IEEE Transactions on Multimedia 23 (2020) 2627–2640.
- [31] K. Gregor, Y. LeCun, Learning fast approximations of sparse coding, in: Proceedings of the 27th international conference on international conference on machine learning, 2010, pp. 399–406.
- [32] M. Borgerding, P. Schniter, S. Rangan, Amp-inspired deep networks for sparse linear inverse problems, IEEE Transactions on Signal Processing 65 (16) (2017) 4293–4308.
- [33] Y. Yang, J. Sun, H. Li, Z. Xu, Deep admm-net for compressive sensing mri, in: Proceedings of the 30th international conference on neural information processing systems, 2016, pp. 10–18.
- [34] J. Zhang, B. Ghanem, Ista-net: Interpretable optimization-inspired deep network for image compressive sensing, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1828–1837.
- [35] Z. Zhang, Y. Liu, J. Liu, F. Wen, C. Zhu, Amp-net: Denoising-based deep unfolding for compressive image sensing, IEEE Transactions on Image Processing 30 (2020) 1487–1500.
- [36] J. Zhang, C. Zhao, W. Gao, Optimization-inspired compact deep compressive sensing, IEEE Journal of Selected Topics in Signal Processing 14 (4) (2020) 765–774.
- [37] S. Mun, J. E. Fowler, Block compressed sensing of images using directional transforms, in: 2009 16th IEEE international conference on image processing (ICIP), IEEE, 2009, pp. 3021–3024.
- [38] H. Y. Liao, G. Sapiro, Sparse representations for limited data tomography, in: IEEE International Symposium on Biomedical Imaging: From Nano to Macro, IEEE, 2008, pp. 1375–1378.
- [39] S. Ravishankar, Y. Bresler, Mr image reconstruction from highly undersampled k-space data by dictionary learning, IEEE Transactions on Medical Imaging 30 (5) (2010) 1028–1041.
- [40] Z. Zha, B. Wen, X. Yuan, J. Zhou, C. Zhu, A. C. Kot, A hybrid structural sparsification error model for image restoration, IEEE Transactions on Neural Networks and Learning Systems (2021) 1–15doi:10.1109/TNNLS.2021.3057439.
- [41] J. Mairal, F. R. Bach, J. Ponce, G. Sapiro, A. Zisserman, Non-local sparse models for image restoration, in: The IEEE International Conference on Computer Vision (ICCV), Vol. 29, 2009, pp. 54–62.
- [42] X. Qu, Y. Hou, F. Lam, D. Guo, J. Zhong, Z. Chen, Magnetic resonance image reconstruction from undersampled measurements using a patch-based nonlocal operator, Medical Image Analysis 18 (6) (2014) 843–856.
- [43] A. Mousavi, A. B. Patel, 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.
- [44] M. Iliadis, L. Spinoulas, A. K. Katsaggelos, Deep fully-connected networks for video compressive sensing, Digital Signal Processing 72 (2018) 9–18.
- [45] A. Adler, D. Boublil, M. Elad, M. Zibulevsky, A deep learning approach to block-based compressed sensing of images, arXiv preprint arXiv:1606.01519 (2016).
- [46] A. Mousavi, 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.
- [47] J. R. Hershey, J. L. Roux, F. Weninger, Deep unfolding: Model-based inspiration of novel deep architectures, arXiv preprint arXiv:1409.2574 (2014).
- [48] J. Sun, H. Li, Z. Xu, et al., Deep admm-net for compressive sensing mri, in: Advances in neural information processing systems, 2016, pp. 10–18.
- [49] S. V. Venkatakrishnan, C. A. Bouman, B. Wohlberg, Plug-and-play priors for model based reconstruction, in: IEEE Global Conference on Signal and Information Processing, IEEE, 2013, pp. 945–948.
- [50] J. Rick Chang, C.-L. Li, B. Poczos, B. Vijaya Kumar, 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.
- [51] C. A. Metzler, P. Schniter, A. Veeraraghavan, R. G. Baraniuk, prdeep: Robust phase retrieval with a flexible deep network, in: international conference on machine learning, 2018, pp. 3498–3507.
- [52] T. Meinhardt, M. Moller, C. Hazirbas, D. Cremers, Learning proximal operators: Using denoising networks for regularizing inverse imaging problems, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1781–1790.
- [53] K. Wei, A. Aviles-Rivero, J. Liang, Y. Fu, C.-B. Schönlieb, H. Huang, Tuning-free plug-and-play proximal algorithm for inverse imaging problems, in: International Conference on Machine Learning, PMLR, 2020, pp. 10158–10169.
- [54] K. Zhang, W. Zuo, S. Gu, L. Zhang, Learning deep cnn denoiser prior for image restoration, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 3929–3938.
- [55] C. A. Metzler, A. Maleki, R. G. Baraniuk, Bm3d-prgamp: Compressive phase retrieval based on bm3d denoising, in: 2016 IEEE International Conference on Image Processing (ICIP), IEEE, 2016, pp. 2504–2508.
- [56] K. Zhang, L. V. Gool, R. Timofte, Deep unfolding network for image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3217–3226.
- [57] K. Wei, A. Aviles-Rivero, J. Liang, Y. Fu, H. Huang, C.-B. Schönlieb, Tf-pnp: Tuning-free plug-and-play proximal algorithms with applications to inverse imaging problems, Journal of Machine Learning Research 23 (16) (2022) 1–48.
- [58] Y. Han, G. Huang, S. Song, L. Yang, H. Wang, Y. Wang, Dynamic neural networks: A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence (2021).
- [59] B. Yang, G. Bender, Q. V. Le, J. Ngiam, Condconv: Conditionally parameterized convolutions for efficient inference, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 32, Curran Associates, Inc., 2019.

URL <https://proceedings.neurips.cc/paper/2019/file/>

- f2201f5191c4e92cc5af043eebf0946-Paper.pdf
- [60] Y. Chen, X. Dai, M. Liu, D. Chen, L. Yuan, Z. Liu, Dynamic convolution: Attention over convolution kernels, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11030–11039.
- [61] X. Jia, B. De Brabandere, T. Tuytelaars, L. Van Gool, Dynamic filter networks, in: NIPS, 2016.
- [62] D. Ha, A. Dai, Q. V. Le, Hypernetworks, arXiv preprint arXiv:1609.09106 (2016).
- [63] J. He, C. Dong, Y. Qiao, Interactive multi-dimension modulation with dynamic controllable residual learning for image restoration, in: European Conference on Computer Vision, Springer, 2020, pp. 53–68.
- [64] Q. Fan, D. Chen, L. Yuan, G. Hua, N. Yu, B. Chen, A general decoupled learning framework for parameterized image operators, IEEE transactions on pattern analysis and machine intelligence (2019).
- [65] X. Wang, K. Yu, C. Dong, C. C. Loy, Recovering realistic texture in image super-resolution by deep spatial feature transform, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 606–615.
- [66] D. Ulyanov, A. Vedaldi, V. Lempitsky, Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6924–6932.
- [67] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: European conference on computer vision, Springer, 2016, pp. 630–645.
- [68] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer, 2015, pp. 234–241.
- [69] K. Wei, Y. Fu, Y. Zheng, J. Yang, Physics-based noise modeling for extreme low-light photography, IEEE Transactions on Pattern Analysis and Machine Intelligence (2021).
- [70] A. C. Sankaranarayanan, L. Xu, C. Studer, Y. Li, K. F. Kelly, R. G. Baraniuk, Video compressive sensing for spatial multiplexing cameras using motion-flow models, SIAM Journal on Imaging Sciences 8 (3) (2015) 1489–1518.
- [71] C. Dong, C. C. Loy, K. He, X. Tang, Learning a deep convolutional network for image super-resolution, in: European conference on computer vision, Springer, 2014, pp. 184–199.
- [72] E. Agustsson, R. Timofte, Ntire 2017 challenge on single image super-resolution: Dataset and study, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2017.
- [73] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, Vol. 2, IEEE, 2001, pp. 416–423.
- [74] L. Yeh, J. Dong, J. Zhong, L. Tian, M. Chen, G. Tang, M. Soltanolkotabi, L. Waller, Experimental robustness of fourier ptychography phase retrieval algorithms, Optics Express 23 (26) (2015) 33214–33240.
- [75] E. J. Candes, X. Li, M. Soltanolkotabi, Phase retrieval via wirtinger flow: Theory and algorithms, IEEE Transactions on Information Theory 61 (4) (2015) 1985–2007.
- [76] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, Vol. 2, IEEE, 2001, pp. 416–423.
- [77] C. Li, W. Yin, H. Jiang, Y. Zhang, An efficient augmented lagrangian method with applications to total variation minimization, Computational Optimization and Applications 56 (3) (2013) 507–530.
- [78] C. A. Metzler, A. Maleki, R. G. Baraniuk, From denoising to compressed sensing, IEEE Transactions on Information Theory 62 (9) (2016) 5117–5144.
- [79] W. Shi, F. Jiang, S. Zhang, D. Zhao, Deep networks for compressed image sensing, in: 2017 IEEE International Conference on Multimedia and Expo (ICME), IEEE, 2017, pp. 877–882.
- [80] J. Huang, S. Zhang, D. Metaxas, Efficient mr image reconstruction for compressed mr imaging, Medical Image Analysis 15 (5) (2011) 670–679.
- [81] E. M. Eksioglu, Decoupled algorithm for mri reconstruction using nonlocal block matching model: Bm3d-mri, Journal of Mathematical Imaging and Vision 56 (3) (2016) 430–440.
- [82] D. R. Pauluzzi, N. C. Beaulieu, A comparison of snr estimation techniques for the awgn channel, IEEE Transactions on communications 48 (10) (2000) 1681–1691.
- [83] M. Riffe, M. Blaimer, K. Barkauskas, J. Duerk, M. Griswold, Snr estimation in fast dynamic imaging using bootstrapped statistics, in: Proc Intl Soc Mag Reson Med, Vol. 1879, 2007.
- [84] O. Dietrich, J. G. Raya, S. B. Reeder, M. F. Reiser, S. O. Schoenberg, Measurement of signal-to-noise ratios in mr images: influence of multi-channel coils, parallel imaging, and reconstruction filters, Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine 26 (2) (2007) 375–385.