PAPER

PALADIN: a novel plug-and-play 3D CS-MRI reconstruction method

IOP Publishing Inverse Problems

Inverse Problems **41** (2025) 035014 (32pp) https://doi.org/10.1088/1361-6420/adb8c6

PALADIN: a novel plug-and-play 3D CS-MRI reconstruction method

Jia-Mian Wu[1,3,4], Shi-Bai Yin[1,3,4], Tai-Xiang Jiang[1,3,4,*], Gui-Song Liu[1,3] and Xi-Le Zhao[2]

[1] School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu 611130, People's Republic of China
[2] School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 611731, People's Republic of China
[3] Engineering Research Center of Intelligent Finance, Ministry of Education, Chengdu 611130, People's Republic of China
[4] Kash Institute of Electronics and Information Industry, Kash 844099, People's Republic of China

E-mail: taixiangjiang@gmail.com, 1240701Z1007@smail.swufe.edu.cn, shibaiyin@swufe.edu.cn, gliu@swufe.edu.cn and xlzhao122003@163.com

Received 10 October 2024; revised 4 February 2025
Accepted for publication 19 February 2025
Published 4 March 2025

Abstract

Compressive sensing magnetic resonance imaging (CS-MRI) accelerates data acquisition by reconstructing high-quality images from a limited set of k-space samples. To solve this ill-posed inverse problem, the plug-and-play (PnP) framework integrates image priors using convolutional neural network (CNN) denoisers. However, CNN denoisers often prioritize local details and may neglect broader degradation effects, leading to visually plausible but structurally inaccurate artifacts. Additionally, the theoretical convergence of PnP methods remains a significant challenge. In this work, we propose a novel method, Plug-And-pLAy 3D MRI recoNstruction, to bridge the gap between denoising and MRI reconstruction. Our model employs the tensor tubal nuclear norm (TNN) to capture intrinsic correlations in 3D MRI data. It also incorporates two implicit regularizers. The first leverages CNN denoisers to exploit image priors. The second, introduced here for the first time, is formulated as a CS-MRI reconstruction subproblem and solved using a deep learning-based method to preserve global spatial structure. We solve the proposed model using the alternating direction method of multipliers. We extend existing theoretical results to prove the algorithm's convergence to a fixed point under reasonable assumptions. Experiments on two datasets with three sampling masks show that our method outperforms state-of-the-art MRI reconstruction methods. Ablation

* Author to whom any correspondence should be addressed.

studies confirm that the TNN and the two implicit regularizers work together to improve reconstruction quality.

Keywords: magnetic resonance imaging, compressive sensing, deep neural networks, plug-and-play

1. Introduction

Magnetic resonance imaging (MRI) is an important technique for clinical diagnosis and treatment. However, its effectiveness can be limited by various physiological and technical constraints [33]. For instance, patients must remain still during the scan to avoid motion artifacts, which can distort Fourier measurements and degrade reconstruction quality. These challenges highlight the need for methods that can accelerate MRI acquisition while maintaining high image quality. In recent years, compressive sensing MRI (CS-MRI) has emerged as a highly effective solution to this problem [53].

CS-MRI aims to reconstruct high-quality MRI images from a limited amount of undersampled k-space data, i.e. data sampled at rates significantly below the Nyquist rate [14, 34]. To address this ill-posed inverse problem, traditional model-based methods leverage prior knowledge of MRI data, such as sparsity in the gradient or wavelet domains [4, 8, 10, 35, 54, 58] or nonlocal self-similarity [37, 39]. Meanwhile, deep learning-based methods have gained popularity in MRI reconstruction due to their ability to achieve high-quality results with short testing times, albeit requiring massive amounts of training data [9, 16, 28, 43, 46, 53, 56].

For CS-MRI, traditional methods and deep learning approaches represent complementary strengths and weaknesses. Traditional methods provide strong interpretability and generalization capabilities, but their model representation is often limited by handcrafted regularizers [19]. In contrast, deep (convolutional) neural networks can learn complex patterns from large datasets and exhibit high model capacity. However, they often lack interpretability, and their performance is heavily dependent on the quality and quantity of training data [60].

A promising way to combine the strengths of both paradigms is to unroll iterative optimization algorithms into deep network architectures [15, 44, 50, 55, 61, 69]. This strategy leverages the interpretability of traditional methods while harnessing the representational power of deep learning. Another potential way is the plug-and-play (PnP) framework [29, 47, 62], which offers flexibility by incorporating off-the-shelf denoising algorithms (denoisers) for solving various image inverse problems. Using variable splitting techniques, such as the alternating direction method of multipliers (ADMM) [5], the prior knowledge-related subproblem can be formulated as a standard denoising task. This allows denoisers, such as the well-known BM3D [13] and the deep learning-based FFDNet [64], to be seamlessly integrated as solution mappings. However, convergence analysis for PnP methods remains a significant challenge, often requiring specific assumptions about the denoiser, such as the bounded denoiser assumption [7, 42].

The PnP framework effectively combines traditional model-based approaches with data-driven deep learning techniques, particularly through the use of convolutional neural network (CNN) denoisers, achieving state-of-the-art performance in CS-MRI [17, 21, 45] (see figure 1). Notably, the PnP framework is not limited to denoisers. With appropriate formulations, other restoration algorithms, such as inpainting [57] and super-resolution [65], can also be integrated.

In this work, we focus on 3D MRI reconstruction from limited k-space sampling. Figure 1 shows a 3D magnetic resonance (MR) image reconstructed from fully-sampled measurements, with undersampling performed frontal-slice-wise using various sampling masks. First, we

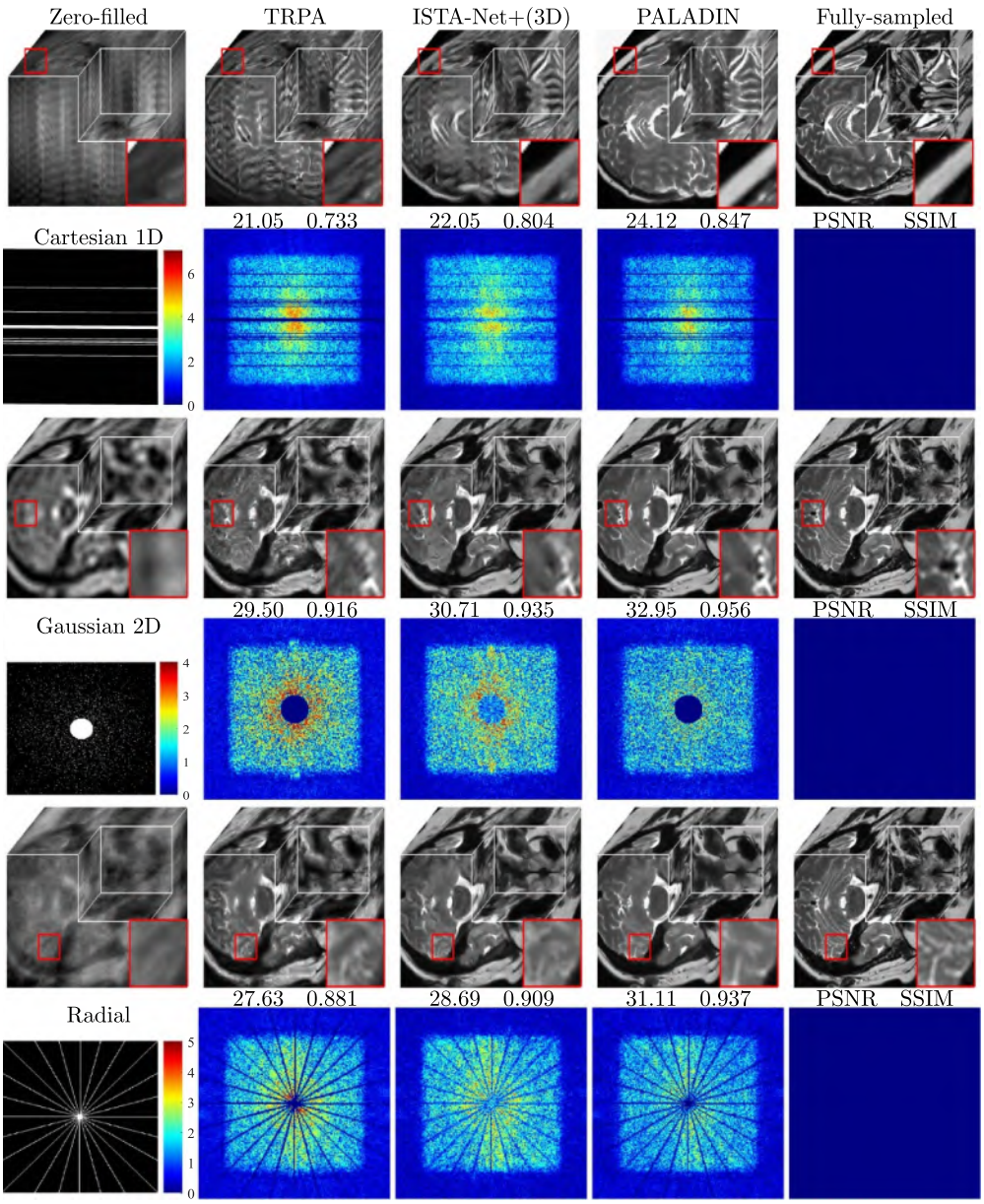

Figure 1. Reconstruction results and corresponding amplitude error images (log-transformed after adding 1) for different methods under various sampling masks. The sampling masks, from top to bottom, are Cartesian 1D (20 ×), Gaussian 2D (20 ×), and radial (20 ×). From left to right, the columns show: the zero-filled data, results from TRPA [21], ISTA-Net+(3D) [61], PALADIN, and the fully-sampled data, respectively.

leverage tensor low-rankness [23, 24, 27, 48, 66] to model frontal-slice coherence in MRI data, employing the tensor tubal nuclear norm (TNN). Then, following previous PnP methods, we introduce an implicit regularizer for plugging in a pretrained deep denoiser. Additionally, we

propose a novel implicit regularizer and formulate its subproblem as a CS-MRI reconstruction subproblem rather than a denoising problem. This allows us to use off-the-shelf CS-MRI reconstruction algorithms (reconstructors) as solution mappings. For example, deep networks like ISTA-Net+ [61] can serve as reconstructors. Our model is optimized using the ADMM algotithm. We extend existing theoretical results to prove the algorithm converges to a fixed point under reasonable assumptions. Like previous PnP methods, our approach combines the interpretability of model-based techniques with the high representation capacity of deep learning. This flexibility offers three key advantages:

- **Superior reconstruction performance:** As shown in figure 1, our method outperforms TRPA [21], which uses only a denoiser. By combining a denoiser and a reconstructor, our approach achieves better results. TRPA struggles to recover low-frequency structural information, while ISTA-Net+(3D) handles structure better but struggles with fine details. Our method restores both global structures and fine details effectively.
- **Theoretical convergence guarantees:** We prove the theoretical convergence of our algorithm, addressing a key challenge in PnP methods. This enhances the reliability of our approach, especially for medical data. As shown in the third row of figure 1 (radial mask), our method produces more convincing results.
- **Flexibility and practicality:** Our method is highly flexible. It can use advanced deep networks as reconstructors or denoisers, improving performance. It also allows reconstructors and denoisers trained on 2D data to be applied to 3D MRI reconstruction. Additionally, our method even allows using the reconstructor trained on natural images, which are much more convenient to access. Our method also adapts to varying noise levels.

The rest of this paper is organized as follows. Section 2 provides the basic mathematical preliminaries. Our method is presented in section 3. Section 4 provides the convergence analysis of Plug-And-pLAy 3D MRI recoNstruction (PALADIN). Section 5 illustrates experimental results. Conclusions and limitations are shown in section 6.

2. Preliminaries

2.1. Notations and tensor basics

In this paper, boldface lowercase letters, e.g. $\mathbf{x}$, are used to denote vectors, boldface uppercase letters, e.g. $\mathbf{X}$, are used to denote matrices, boldface calligraphic letters, e.g. $\mathcal{X}$, are used to denote tensors or operators. The kth frontal slice of a tensor $\boldsymbol{\mathcal{X}} \in \mathcal{C}^{n_1\times n_2\times n_3}$ is denoted as $\mathcal{X}(:,:,k)$ (or briefly $\mathcal{X}[::k]$) for $k = 1,2,\cdots,n_3$. The tensor Frobenius norm of a third-order tensor $\mathcal{X} \in \mathcal{C}^{n_1\times n_2\times n_3}$ is defined as $\|\mathcal{X}\|_F := \sqrt{\sum_{ijk}\mathcal{X}_{ijk}^2}$. For a matrix $\mathbf{X} \in \mathbb{C}^{n_1\times n_2}$, its matrix nuclear norm is denoted as $\|\mathbf{X}\|_* = \sum_{i=1}^{\min\{n_1,n_2\}}\sigma_i(\mathbf{X})$, where $\sigma_i(\mathbf{X})$ is the ith largest singular value of $\mathbf{X}$. The operator $\mathtt{vec}(\cdot)$ rearranges a matrix $\mathbf{X} \in \mathcal{C}^{n_1\times n_2}$ or a tensor $\mathcal{X} \in \mathcal{C}^{n_1\times n_2\times n_3}$ into a vector $\mathbf{x} \in \mathcal{C}^{n_1n_2\times 1}$ or $\mathbf{x} \in \mathcal{C}^{n_1n_2n_3\times 1}$ in the lexicographic order, respectively. Its inverse operation is denoted as $\mathtt{vec}^{-1}(\cdot)$. The apostrophe ($'$s) will be used to denote the plural form of a variable, such as b's referring to the plural of b.

In the following, we introduce very basic definitions for the TNN. Interested readers can refer to [26, 27, 31, 67] for the comprehensive mathematical framework.

Definition 1 (Circular convolution). Let $\mathbf{a}$ and $\mathbf{b}$ are two vectors of the size n_1 and n_2. Define the L-point circular convolution of $\mathbf{a}$ and $\mathbf{b}$ as $\mathbf{c} = \mathbf{a} \circledast \mathbf{b} = [\sum_{m=0}^{L-1}\mathbf{b}(m)\mathbf{a}((n-m))_L]R_L$, where $L \geqslant \max[N_1,N_2]$, L is the circular convolution interval length, R_L is a rectangular sequence,

and $\mathbf{a}((n-m))_L$ can be regarded as the result of periodic continuation after cyclic shift of $\mathbf{a}$ [41].

Definition 2 (Tensor–tensor-product (t-prod) [27]). The tensor–tensor-product (t-prod) $\mathcal{Z} = \mathcal{X} * \mathcal{Y}$ of $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ and $\mathcal{Y} \in \mathbb{C}^{n_2 \times n_4 \times n_3}$ is a tensor of the size $n_1 \times n_4 \times n_3$, where the (i,j)th tube $\mathcal{C}$ is given by $\mathcal{Z}(i,j,:) = \sum_{k=1}^{n_2} \mathcal{X}(i,k,:) \circledast \mathcal{Y}(k,j,:)$, where $\circledast$ denotes the circular convolution between two tubes of the same size.

Definition 3 (Special tensors [27]). $\overline{\mathcal{X}} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ is used to denote the Fourier transformed (along the third mode) tensor of $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$. The conjugate transpose of a tensor $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ is $\mathcal{X}^H \in \mathbb{C}^{n_1 \times n_2 \times n_3}$, which is obtained by conjugate transposing each of the frontal slices of $\mathcal{X}$ and then reversing the order of transposed frontal slices through n_3. The identity tensor $\mathcal{I} \in \mathbb{C}^{n_1 \times n_1 \times n_3}$ is the tensor whose first frontal slice is the $n_1 \times n_1$ identity matrix, and whose other frontal slices are all zeros. A tensor $\mathcal{Q} \in \mathbb{C}^{n_1 \times n_1 \times n_3}$ is an orthogonal tensor if it satisfies $\mathcal{Q}^H * \mathcal{Q} = \mathcal{Q} * \mathcal{Q}^H = \mathcal{I}$. A tensor $\mathcal{X}$ is called f-diagonal if each frontal slice is a diagonal matrix.

Theorem 1 (Tensor-singular value decomposition (t-SVD) [27]). *For $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$, the t-SVD of $\mathcal{X}$ is given by*

$$\mathcal{X} = \mathcal{U} * \mathcal{S} * \mathcal{V}^H \tag{1}$$

where $\mathcal{U} \in \mathbb{C}^{n_1 \times n_1 \times n_3}$ and $\mathcal{V} \in \mathbb{C}^{n_2 \times n_2 \times n_3}$ are orthogonal tensors, and $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is an f-diagonal tensor.

Definition 4 (Tensor tubal nuclear norm (TNN) [67]). The TNN of a tensor $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$, denoted as $\|\mathcal{X}\|_{\mathrm{TNN}}$, is defined as

$$\|\mathcal{X}\|_{\mathrm{TNN}} \triangleq \sum_{i=1}^{n_3} \|\overline{\mathcal{X}}(:,:,i)\|_*. \tag{2}$$

2.2. Observation model

In this work, we consider the compressive sensing of 3D MRI and formulate the observation model of 3D CS-MRI. Let $\mathcal{X} \in \mathcal{C}^{n_1 \times n_2 \times n_3}$ denotes the 3D MR image. For the kth frontal slice $\mathcal{X}(:,:,k)$ $(k = 1,2,\cdots,n_3)$, the undersampling in the k-space can be formulated as

$$\mathcal{S} \circ \mathcal{F}(\mathcal{X}(:,:,k)) = \mathbf{b}_k,$$

where $\mathcal{F}$ is the Fourier transform operation, $\mathcal{S}$ is the undersampling operation, $\mathcal{S} \circ \mathcal{F}$ is the composition of the two operators, $\mathbf{b}_k$ is the observed measurements. Using the vectorization of $\mathcal{X}(:,:,k)$, the above observation model is equivalent to

$$\mathbf{S}\mathbf{F}\mathtt{vec}((\mathcal{X}(:,:,k)) = \mathbf{b}_k, \tag{3}$$

where $\mathbf{F} \in \mathbb{C}^{n_1 \times n_2}$ is the 2D discrete Fourier transform (DFT) matrix and the $\mathbf{S}$ is sampled from the rows of an identity matrix according to the sampling strategy. For simplicity, we briefly denote the observation model for the whole $\mathcal{X}$ as

$$\mathcal{A}(\mathcal{X}) = \mathbf{b}, \tag{4}$$

where $\mathcal{A}$ denotes the Fourier transform and undersampling operation and $\mathbf{b}$ is concatenated from $\mathbf{b}_k{}'$s $(k = 1,2,\cdots,n_3)$.

3. Proposed method

Based on the observation model of 3D CS-MRI in (4), we consider that the slices in 3D MRI data are inner-related. We simultaneously introduce the TNN, reconstructor, and denoiser to establish our model. Then, we utilize the ADMM [5] algorithm to optimize the model and theoretically establish convergence of the algorithm, which is important and challenging.

3.1. Proposed algorithm PALADIN

Our model is formulated as:

$$\min_{\mathcal{X}} \lambda\|\mathcal{X}\|_{\mathrm{TNN}} + \Phi(\mathcal{X}) + \Psi(\mathcal{X}) + \frac{\rho}{2}\|\mathcal{A}(\mathcal{X}) - \mathbf{b}\|_F^2 + \frac{\gamma}{2}\|\mathcal{X}\|_F^2, \tag{5}$$

where $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ is the underlying 3D MR image, $\Phi(\cdot)$ and $\Psi(\cdot)$ are implicit regularizers, $\frac{\rho}{2}\|\mathcal{A}(\mathcal{X}) - \mathbf{b}\|_F^2$ is the data fidelity term, $\frac{\gamma}{2}\|\mathcal{X}\|_F^2$ is a Tikhonov regularizer, ρ is a trade-off parameter, and λ, γ are the regularization parameters. As shown in figure 1, the slices of a 3D MR image are inner correlated and we resort to minimizing the $\|\mathcal{X}\|_{\mathrm{TNN}}$ to utilize this. Those two implicit regularizers are formulated to plug in denoisers and reconstructors, respectively.

Then, as it is difficult to directly optimize (5), we resort to ADMM [5]. First, to decouple three terms in the objective function, we introduce three auxiliary variables and then address the following optimization problem,

$$\begin{aligned} \min_{\mathcal{X},\mathcal{Y},\mathcal{Z},\mathcal{W}} \ & \lambda\|\mathcal{Z}\|_{\mathrm{TNN}} + \Phi(\mathcal{X}) + \Psi(\mathcal{W}) + \frac{\rho}{2}\|\mathcal{A}(\mathcal{Y}) - \mathbf{b}\|_F^2 + \frac{\gamma}{2}\|\mathcal{Y}\|_F^2 \\ \text{s.t.} \quad & \mathcal{A}(\mathcal{X}) = \mathcal{A}(\mathcal{Y}), \mathcal{Z} = \mathcal{Y}, \mathcal{W} = \mathcal{Y}. \end{aligned} \tag{6}$$

It is notable that (5) is not mathematically equivalent to (6) as the constraint $\mathcal{A}(\mathcal{X}) = \mathcal{A}(\mathcal{Y})$ only requires the auxiliary variable $\mathcal{X}$ to match $\mathcal{Y}$ after undersampling in the k-space. We can see in the following, that this enables us to formulate the $\mathcal{X}$ subproblem as a typical CS-MRI reconstruction problem.

The augmented Lagrangian function of (5) is:

$$\begin{aligned} L(\mathcal{X},\mathcal{Y},\mathcal{Z},\mathcal{W},\Lambda) =& \lambda\|\mathcal{Z}\|_{\mathrm{TNN}} + \Phi(\mathcal{X}) + \Psi(\mathcal{W}) + \frac{\rho}{2}\|\mathcal{A}(\mathcal{Y}) - \mathbf{b}\|_F^2 + \frac{\gamma}{2}\|\mathcal{Y}\|_F^2 \\ & + \frac{\beta_1}{2}\|\mathcal{A}(\mathcal{X}) - \mathcal{A}(\mathcal{Y}) + \frac{\Lambda_1}{\beta_1}\|_F^2 + \frac{\beta_2}{2}\|\mathcal{Z} - \mathcal{Y} + \frac{\Lambda_2}{\beta_2}\|_F^2 \\ & + \frac{\beta_2}{2}\|\mathcal{W} - \mathcal{Y} + \frac{\Lambda_3}{\beta_2}\|_F^2, \end{aligned} \tag{7}$$

where $\Lambda_i{}'$s $(i = 1,2,3)$ are Lagrangian multipliers and $\beta_i{}'$s $(i = 1,2)$ are non-negative penalty parameters. Then, our ADMM algorithm iteratively updates each variable by solving corresponding subproblems.

3.1.1. $\mathcal{Z}$-subproblem. $\mathcal{Z}$ is updated via solving the following problem:

$$\mathcal{Z}^{t+1} = \arg\min_{\mathcal{Z}} \left(\lambda\|\mathcal{Z}\|_{\mathrm{TNN}} + \frac{\beta_2}{2}\|\mathcal{Z} - \mathcal{Y}^t + \frac{\Lambda_2^t}{\beta_2}\|_F^2 \right). \tag{8}$$

Consider the t-SVD of $\mathcal{Y}^t - \frac{\Lambda_2^t}{\beta}$ as $\mathcal{U}\mathcal{S}\mathcal{V}^H$. We utilize the t-SVT [31, 67], which is implemented by applying thresholding operations on the singular values derived from the SVD of a tensor, solving the closed-form solution of the $\mathcal{Z}$-subproblem, denoted as:

$$\mathcal{Z}^{t+1} = \text{t} - \text{SVT}\left(\mathcal{Y}^t - \frac{\Lambda_2^t}{\beta_2}\right) = \mathcal{U} * \mathcal{D} * \mathcal{V}^H, \tag{9}$$

where $\mathcal{D}$ is an f-diagonal tensor with each frontal slice in the Fourier domain defined as $\overline{\mathcal{D}}(i,i,k) = \max\{\overline{\mathcal{S}}(i,i,k) - \frac{\lambda n_3}{\beta_2}, 0\}$.

3.1.2. $\mathcal{X}$-subproblem. $\mathcal{X}$ is updated by solving

$$\mathcal{X}^{k+1} = \arg\min_{\mathcal{X}} \left(\Phi(\mathcal{X}) + \frac{\beta_1}{2}\|\mathcal{A}(\mathcal{X}) - \hat{\boldsymbol{b}}\|_F^2\right), \tag{10}$$

where $\hat{\boldsymbol{b}} = \mathcal{A}(\mathcal{Y}^t) - \frac{\Lambda_1^t}{\beta_1}$. We can see that (10) is a typical CS-MRI reconstruction problem, where $\Phi(\mathcal{X})$ is the prior term and the second term is the data fidelity term with the degradation $\mathcal{A}$. Thus, we can directly utilize off-the-shelf CS-MRI algorithms to solve (10), and obtain the updating of $\mathcal{X}$ as

$$\mathcal{X}^{t+1} = \text{Reconstructor}\left(\hat{\boldsymbol{b}}\right). \tag{11}$$

Alternatively, as many state-of-the-art CS-MRI methods are originally designed for 2D MR images, we can slice-wisely update $\mathcal{X}$ as

$$\mathcal{X}^{t+1}(:,:,k) = \text{Reconstructor}\left(\hat{\boldsymbol{b}}_k\right) \ (k = 1, 2, \cdots, n_3), \tag{12}$$

where $\hat{\boldsymbol{b}}_k$ is the kth component of $\hat{\boldsymbol{b}}$ related to the kth frontal slice of the 3D MR image.

3.1.3. $\mathcal{W}$-subproblem. $\mathcal{W}$ is updated as:

$$\mathcal{W}^{t+1} = \arg\min_{\mathcal{W}} \left(\Psi(\mathcal{W}) + \frac{\beta_2}{2}\|\mathcal{W} - \left(\mathcal{Y}^t - \frac{\Lambda_3^t}{\beta_2}\right)\|_F^2\right). \tag{13}$$

(13) is a typical denoising problem and we can feed $\mathcal{Y}^t - \frac{\Lambda_3^t}{\beta_2}$ into a denoiser as:

$$\mathcal{W}^{t+1} = \text{Denoiser}_\sigma\left(\mathcal{Y}^t - \frac{\Lambda_3^t}{\beta_2}\right). \tag{14}$$

3.1.4. $\mathcal{Y}$-subproblem. Next, the $\mathcal{Y}$-subproblem is a least squares problem as follows:

$$\begin{aligned}\mathcal{Y}^{t+1} = \arg\min_{\mathcal{Y}} \Bigg(&\frac{\beta_1}{2}\|\mathcal{A}\left(\mathcal{X}^{t+1}\right) - \mathcal{A}(\mathcal{Y}) + \frac{\Lambda_1^t}{\beta_1}\|_F^2 + \frac{\beta_2}{2}\|\mathcal{Z}^{t+1} - \mathcal{Y} + \frac{\Lambda_2^t}{\beta_2}\|_F^2 + \frac{\gamma}{2}\|\mathcal{Y}\|_F^2 \\ &+ \frac{\beta_2}{2}\left\|\mathcal{W}^{t+1} - \mathcal{Y} + \frac{\Lambda_3^t}{\beta_2}\right\|_F^2 + \frac{\rho}{2}\|\mathcal{A}(\mathcal{Y}) - \mathbf{b}\|_F^2\Bigg).\end{aligned} \tag{15}$$

Considering the separability of the Frobenius norm, we can solve (15) slice-wisely. First, we denote $\mathbf{x}_k^{t+1} = \mathtt{vec}\left(\mathcal{X}^{t+1}(:,:,k)\right)$, $\mathbf{z}_k^{t+1} = \mathtt{vec}\left(\mathcal{Z}^{t+1}(:,:,k)\right)$, $\mathbf{y}_k = \mathtt{vec}\left(\mathcal{Y}^{t+1}(:,:,k)\right)$, $\mathbf{w}_k^{t+1} = \mathtt{vec}\left(\mathcal{W}^{t+1}(:,:,k)\right)$, $\boldsymbol{\mu}_k^t = \mathtt{vec}\left(\boldsymbol{\Lambda}_2^t(:,:,k)\right)$, and $\boldsymbol{\nu}_k^t = \mathtt{vec}\left(\boldsymbol{\Lambda}_3^t(:,:,k)\right)$. We compute the derivative of (15) and let it be 0. Recalling (3), we have

$$\begin{aligned} &-\beta_1 \mathbf{F}^H \mathbf{S}^H \left(\mathbf{S}\mathbf{F}\mathbf{x}_i^{t+1} - \mathbf{S}\mathbf{F}\mathbf{y}_k + \frac{\left(\boldsymbol{\Lambda}_1^t\right)_k}{\beta_1} \right) - \beta_2 \left(\mathbf{z}_k^{t+1} - \mathbf{y}_k + \frac{\boldsymbol{\mu}_k^t}{\beta_2} \right) \\ &-\beta_2 \left(\mathbf{w}_k^{t+1} - \mathbf{y}_k + \frac{\boldsymbol{\nu}_k^t}{\beta_2} \right) + \rho \mathbf{F}^H \mathbf{S}^H \left(\mathbf{S}\mathbf{F}\mathbf{y}_k - \mathbf{b}_k \right) + \gamma \mathbf{y}_k = \mathbf{0}, \end{aligned} \tag{16}$$

where $\mathbf{b}_k$ is the same as $\mathbf{b}_k$ in (3), and $\left(\boldsymbol{\Lambda}_2^t\right)_k$ indicates the component of $\boldsymbol{\Lambda}_2^t$ related to the kth frontal slice. Then, we update $\mathbf{y}_k^{t+1}$ as

$$\begin{aligned} \mathbf{y}_k^{k+1} = &\left((\beta_1 + \rho) \mathbf{F}^H \mathbf{S}^H \mathbf{S}\mathbf{F} + (2\beta_2 + \gamma) \mathbf{I} \right)^{-1} \\ &\times \left(\mathbf{F}^H \mathbf{S}^H \left(\beta_1 \mathbf{S}\mathbf{F}\mathbf{x}_k^{t+1} + \left(\boldsymbol{\Lambda}_1^t\right)_k + \rho \mathbf{b}_k \right) + \beta_2 \mathbf{z}_k^{t+1} + \beta_2 \mathbf{w}_k^{t+1} + \boldsymbol{\mu}_k^t + \boldsymbol{\nu}_k^t \right). \end{aligned} \tag{17}$$

In (17), the corresponding computation, $\mathcal{Y}$ is slice-wisely updated as

$$\mathcal{Y}^{t+1}(:,:,i) = \mathtt{vec}^{-1}\left(\mathbf{y}_i^{t+1}\right). \tag{18}$$

3.1.5. Multipliers. The multipliers are updated as:

$$\begin{cases} \Lambda_1^{t+1} = \Lambda_1^t + \beta_1 \left(\mathcal{A}\left(\mathcal{X}^{t+1}\right) - \mathcal{A}\left(\mathcal{Y}^{t+1}\right) \right) \\ \Lambda_2^{t+1} = \Lambda_2^t + \beta_2 \left(\mathcal{Z}^{t+1} - \mathcal{Y}^{t+1} \right) \\ \Lambda_3^{t+1} = \Lambda_3^t + \beta_2 \left(\mathcal{W}^{t+1} - \mathcal{Y}^{t+1} \right). \end{cases} \tag{19}$$

To ensure the reconstructed image maintains symmetry in the Fourier domain, we introduce a post-processing step where the result is transformed back to k-space. In this step, we symmetrically cover the sampled points and then transform the data back to the image domain to obtain the final result. This post-processing step is reasonable because the symmetry in the Fourier domain is crucial for accurate image reconstruction. By addressing any asymmetries in k-space, we can diminish artifacts, leading to more reliable and clinically useful results.

3.2. Comparison with other PnP frameworks

In traditional PnP frameworks, the choice of denoiser critically impacts both efficiency and accuracy in solving inverse problems. Commonly used denoisers include BM3D [13], FFDNet [64], and DnCNN [63]. Non-deep learning denoisers like BM3D offer strong interpretability [25, 68], while deep learning-based denoisers (e.g. FFDNet and DnCNN) achieve superior performance by leveraging large-scale training data [2, 49, 62, 68]. However, deep denoisers often lack interpretability, limiting their reliability in medical imaging. Traditional PnP frameworks relying solely on denoisers [11, 21, 40] often yield suboptimal reconstructions. This is because denoising primarily focuses on local detail restoration, frequently neglecting global structural information. As shown in figure 1, such methods may fail to recover critical anatomical structures, leading to diagnostically unreliable results. In contrast, PALADIN introduces a novel combination of the reconstructor and the denoiser, leveraging their complementary strengths: the denoiser restores fine local details, while the reconstructor

Algorithm 1. Proposed algorithm PALADIN.

Input: $\mathcal{Y}, \mathcal{S}, \rho, \beta_1, \beta_2, \lambda, \gamma, k, t = 0$
$\mathcal{Y}^0 = \mathcal{S}\mathcal{F}\mathcal{X}$
Λ_i^0's $= 0\,(i = 1,2,3)$
while $t < k$ do
    Update $\mathcal{Z}^{t+1}$ using (9)
    Update $\mathcal{X}^{t+1}$ using (11)
    Update $\mathcal{W}^{t+1}$ using (14)
    Update $\mathcal{Y}^{t+1}$ using (17) and (18)
    Update Λ_i^{t+1}'s $(i = 1,2,3)$ using (19)
    $t = t + 1$
end while
Output: $\mathcal{Y}$

preserves global structural integrity. This dual-regularization approach addresses the limitations of traditional PnP frameworks, achieving both high reconstruction quality and improved reliability for medical applications.

4. Convergence analysis of PALADIN

In this section, we present the fixed-point analysis of algorithm 1. Since the outline of the convergence analysis is similar to the method in [42], we begin by reformulating our optimization problem into a similar formulation as presented in [42] in section 4.1. Building upon this, we introduce the key assumptions for our proof and discuss their reasonableness in section 4.2. Then, section 4.3 provides the detailed proof of the main theorem. To facilitate the understanding of our proof, section 4.3 consists of (i) the roadmap of our proof and the relationship between the assumption and the key step of the proof, (ii) the necessary lemmas, and (iii) the detailed proof.

4.1. Problem reformulation

Firstly, the original model can be equivalently converted to:

$$\begin{aligned} &\min_{\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{w}} \lambda\|\mathbf{z}\|_{\mathrm{TNN}} + \Psi(\mathbf{w}) + \Phi(\mathbf{x}) + \frac{\rho}{2}\|\mathbf{A}\mathbf{y} - \mathbf{b}\|_F^2 + \frac{\gamma}{2}\|\mathbf{y}\|_F^2 \\ &\text{s.t.} \quad \begin{bmatrix} \mathbf{A}\mathbf{x} \\ \mathbf{z} \\ \mathbf{w} \end{bmatrix} = \begin{bmatrix} \mathbf{A}\mathbf{y} \\ \mathbf{y} \\ \mathbf{y} \end{bmatrix}, \end{aligned} \tag{20}$$

where $\mathbf{A}$ denotes the composition of sub-sampling and Fourier transform, i.e, $\mathbf{A} = \mathbf{SF}$ in (3), and $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$ and $\mathbf{w}$ denote the vectorization of $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$ and $\mathcal{W}$ in (6), respectively.

Denoting $\mathbf{c}=\begin{bmatrix}\mathbf{x}\\ \mathbf{z}\\ \mathbf{w}\end{bmatrix}$, $\mathbf{m}=\begin{bmatrix}\mathbf{y}\\ \mathbf{y}\\ \mathbf{y}\end{bmatrix}$, and $h(\mathbf{c})=\left\|\begin{bmatrix}\Phi(\mathbf{x})\\ \lambda\|\mathbf{z}\|_{\mathrm{TNN}}\\ \Psi(\mathbf{w})\end{bmatrix}\right\|_1$, we convert (20) into the following equation:

$$\begin{aligned}&\min_{\mathbf{c},\mathbf{m}} h(\mathbf{c})+f(\mathbf{m})\\ &\text{s.t.}\quad \mathbf{A}_1\mathbf{c}=\mathbf{A}_1\mathbf{m},\end{aligned}\tag{21}$$

where $\mathbf{A}_1=\begin{bmatrix}\mathbf{A} & & \\ & \mathbf{I} & \\ & & \mathbf{I}\end{bmatrix}$, and $f(\cdot)$ is a scalar-valued function satisfying:

$$f(\mathbf{m})=\frac{\rho}{2}\left\|\mathbf{A}\begin{bmatrix}0 & \mathbf{I} & 0\end{bmatrix}\mathbf{m}-\mathbf{b}\right\|_F^2+\frac{\gamma}{2}\left\|\begin{bmatrix}0 & \mathbf{I} & 0\end{bmatrix}\mathbf{m}\right\|_F^2=\frac{\rho}{2}\|\mathbf{A}\mathbf{y}-\mathbf{b}\|_F^2+\frac{\gamma}{2}\|\mathbf{y}\|_F^2.\tag{22}$$

We denote the $\widehat{\mathbf{\Lambda}}=\begin{bmatrix}\frac{\mathbf{\Lambda}_1}{\beta_1}\\ \frac{\mathbf{\Lambda}_2}{\beta_2}\\ \frac{\mathbf{\Lambda}_3}{\beta_2}\end{bmatrix}$, where $\mathbf{\Lambda}_i(i=1,2,3)$ is the vectorization of $\Lambda_i(i=1,2,3)$ and $\boldsymbol{\beta}=\begin{bmatrix}\boldsymbol{\beta_1}\\ \boldsymbol{\beta_2}\\ \boldsymbol{\beta_2}\end{bmatrix}$ where $\boldsymbol{\beta_1}$ and $\boldsymbol{\beta_2}$ are the vectors containing repetitive $\beta_i(i=1,2)$ and have the same dimension as the $\mathbf{Ay}$ and $\mathbf{y}$, respectively. The augmented Lagrangian function of (21) is:

$$L\left(\mathbf{c},\mathbf{m},\widehat{\mathbf{\Lambda}}\right)=h(\mathbf{c})+\frac{1}{2}\left\|\sqrt{\boldsymbol{\beta}}\odot\left(\mathbf{A}_1\mathbf{c}-\mathbf{A}_1\mathbf{m}+\widehat{\mathbf{\Lambda}}\right)\right\|_F^2+f(\mathbf{m}),\tag{23}$$

where $\odot$ denotes Hadamard product.

The ADMM algorithm can be used to solve the above optimization problem, and we can show the iteration of the variables:

$$\begin{cases}\mathbf{c}^{t+1}=\arg\min\limits_{\mathbf{c}}\left(h(\mathbf{c})+\frac{1}{2}\left\|\sqrt{\boldsymbol{\beta}}\odot\left(\mathbf{A}_1\mathbf{c}-\mathbf{A}_1\mathbf{m}^t+\widehat{\mathbf{\Lambda}}^t\right)\right\|_F^2\right)\\ \mathbf{m}^{t+1}=\arg\min\limits_{\mathbf{m}}\left(f(\mathbf{m})+\frac{1}{2}\left\|\sqrt{\boldsymbol{\beta}}\odot\left(\mathbf{A}_1\mathbf{m}-\mathbf{A}_1\mathbf{c}^t-\widehat{\mathbf{\Lambda}}^t\right)\right\|_F^2\right)\\ \widehat{\mathbf{\Lambda}}^{t+1}=\widehat{\mathbf{\Lambda}}^t+\mathbf{A}_1\mathbf{c}^{t+1}-\mathbf{A}_1\mathbf{m}^{t+1}.\end{cases}\tag{24}$$

We denote the $\mathcal{B}=\begin{bmatrix}\mathcal{H} & & \\ & \mathtt{Prox}_{\lambda g} & \\ & & \mathcal{D}_\sigma\end{bmatrix}$. According to the PNP-ADMM [42], we can equivalently write (24) as

$$\begin{cases}\mathbf{c}^{t+1}=\mathcal{B}\left(\mathbf{A}_1\mathbf{m}^t-\hat{\mathbf{\Lambda}}^t\right)\\ \mathbf{m}^{t+1}=\mathtt{Prox}_{\rho f}\left(\mathbf{A}_1\mathbf{c}^t+\widehat{\mathbf{\Lambda}}^t\right)\\ \widehat{\mathbf{\Lambda}}^{t+1}=\widehat{\mathbf{\Lambda}}^t+\left(\mathbf{A}_1\mathbf{c}^{t+1}-\mathbf{A}_1\mathbf{m}^{t+1}\right),\end{cases}\tag{25}$$

where the $\mathtt{Prox}_{\lambda g}$ denotes the t-SVT operator in (9), $\mathcal{D}_\sigma$ is the denoiser in the denoising subproblem, and $\mathcal{H}$ is the reconstructor in the reconstructing subproblem. If our PNP-ADMM converges to a fixed-point, we denote the $(\mathbf{m}^*,\mathbf{\Lambda}_i{}^*(i=1,2,3))$ as a fixed point of PNP-ADMM.

Then we have:

$$\begin{cases} \mathbf{m}^* = \mathcal{B}\left(\mathbf{A}_1\mathbf{m}^* - \hat{\mathbf{\Lambda}}^*\right) \\ \mathbf{m}^* = \mathtt{Prox}_{\rho f}\left(\mathbf{A}_1\mathbf{m}^* + \hat{\mathbf{\Lambda}}^*\right). \end{cases} \tag{26}$$

If we let $\mathbf{c}^t = \mathbf{m}^t = \mathbf{m}^*$ and $\mathbf{\Lambda}_i^t = \mathbf{\Lambda}_i^* (i = 1,2,3)$ in our PNP-ADMM, we get $\mathbf{c}^{t+1} = \mathbf{m}^{t+1} = \mathbf{m}^*$ and $\mathbf{\Lambda}_i^{t+1} = \mathbf{\Lambda}_i^t = \mathbf{\Lambda}_i^* (i = 1,2,3)$. We let $\mathbf{u}^t = \mathbf{A}_1\mathbf{m}^t + \mathbf{\Lambda}^t$ and convert (25) into the following equation:

$$\begin{cases} \mathbf{m}^{t+1/2} = \mathtt{Prox}_{\rho f}(\mathbf{u}^t) \\ \mathbf{m}^{t+1} = \mathcal{B}\left(2\mathbf{A}_1\mathbf{m}^{t+1/2} - \mathbf{u}^t\right) \\ \mathbf{u}^{t+1} = \mathbf{u}^t + \mathbf{A}_1\mathbf{m}^{t+1} - \mathbf{A}_1\mathbf{m}^{t+1/2}. \end{cases} \tag{27}$$

We call the above method PnP Douglas–Rachford splitting (PNP-DRS). Similarly, if z^* is a fixed point of PNP-DRS, we have:

$$\begin{cases} \mathbf{m}^* = \mathtt{Prox}_{\rho f}(\mathbf{u}^*) \\ \mathbf{m}^* = \mathcal{B}(2\mathbf{A}_1\mathbf{m}^* - \mathbf{u}^*). \end{cases} \tag{28}$$

We can equivalently write (27) as:

$$\begin{cases} \mathbf{m}^{t+1/2} = \mathtt{Prox}_{\rho f}(\mathbf{u}^t) \\ \mathbf{m}^{t+1} = \mathcal{B}\left(2\mathbf{A}_1\mathbf{m}^{t+1/2} - \mathbf{u}^t\right) \\ \mathbf{u}^{t+1} = \mathbf{u}^t + \mathbf{A}_1\mathcal{B}\left(2\mathbf{A}_1\mathbf{m}^{t+1/2} - \mathbf{u}^t\right) - \mathbf{A}_1\mathtt{Prox}_{\rho f}(\mathbf{u}^t). \end{cases} \tag{29}$$

Therefore, the iteration of $\mathbf{u}$ can be written as below:

$$\mathbf{u}^{t+1} = \frac{1}{2}\mathbf{u}^t + \frac{1}{2}\left(2\mathbf{A}_1\mathtt{Prox}_{\rho f} - \mathcal{I}\right)\left(2\mathbf{A}_1\mathcal{B} - \mathcal{I}\right)\mathbf{u}^t. \tag{30}$$

We write the matrix $\mathbf{A}_1$ as the operator $\mathcal{A}_1$, and denote the $\mathcal{A}_1\mathcal{B} = \mathcal{B}_1$. Then we have:

$$\mathbf{u}^{t+1} = \mathcal{T}(\mathbf{u}^t), \tag{31}$$

where $\mathcal{T} = \frac{1}{2}\mathcal{I} + \frac{1}{2}(2\mathcal{A}_1\mathtt{Prox}_{\rho f} - \mathcal{I})(2\mathcal{B}_1 - \mathcal{I})$.

4.2. Key assumption and main theorem

If we only use a denoiser, the convergence analysis of PALADIN conforms to theorem 3 in [42] and the road map of our analysis also largely follows the line in [42]. However, the main difference between our method and typical PnP methods makes our analysis more difficult, as we need additional assumptions and verify their reasonability,

We assume $\mathcal{B}_1 = \begin{bmatrix} \mathcal{A}\mathcal{H} & & \\ & \mathtt{Prox}_{\lambda g} & \\ & & \mathcal{D}_\sigma \end{bmatrix} : \mathbb{R}^d \to \mathbb{R}^d$ satisfies

$$\left\|\left(\mathcal{B}_1 - \mathcal{I}\right)(\mathbf{x}) - \left(\mathcal{B}_1 - \mathcal{I}\right)(\mathbf{y})\right\|^2 \leqslant \epsilon^2\|\mathbf{x} - \mathbf{y}\|^2, \quad (A) \tag{32}$$

for all $\mathbf{x},\mathbf{y}\in\mathbb{R}^d$ for some $\epsilon\geqslant 0$. Since $\mathcal{B}_1$ is a block-diagonal operator, if each of the three corresponding operators individually satisfies the assumption, then the entire operator $\mathcal{B}_1$ will also satisfy the assumption. Firstly, for small values of λ, $\texttt{Prox}_{\lambda g}$ will be close to identity operator. Secondly, the operator $\mathcal{D}_\sigma$ will also approximate identity for small σ. Similarly, when a reconstructor is trained well, the $\mathcal{AH}$ is also close to the identity. Next, we specifically analyze the reasonability of $\texttt{Prox}_{\lambda g}$ satisfying the assumption. Because a t-SVT operator is utilized as an effective approach to solve the proximal operator, we can write:

$$\texttt{Prox}_{\lambda g}(\mathcal{X}) = \mathrm{t}-\mathrm{SVT}(\mathcal{X}) = \mathcal{U}*\mathcal{D}*\mathcal{V}^H.$$

The $\mathcal{U}*\mathcal{S}*\mathcal{V}^H$ denotes the t-SVD of $\mathcal{X}$, and $\mathcal{D}$ is an f-diagonal tensor determined by $\mathcal{D}(i,i,k)=\max\{\hat{\mathcal{S}}(i,i,k)-\lambda,0\}$, where $\hat{\mathcal{S}}$ is the Fourier transformed tensor of $\mathcal{S}$ along the third mode. By the lemma 3.19 in [27], we can directly write:

$$\|\mathcal{X}\|_F^2=\|\mathcal{U}*\mathcal{S}*\mathcal{V}^H\|_F^2=\|\mathcal{S}\|_F^2=\left\|\begin{bmatrix}\mathcal{S}_{(1)} & & & \\ & \mathcal{S}_{(2)} & & \\ & & \cdots & \\ & & & \mathcal{S}_{(n_3)}\end{bmatrix}\right\|_F^2, \tag{33}$$

where $\mathcal{S}_{(i)}(i=1,2,3,\ldots,n_3)$ denote the ith frontal slice of $\mathcal{S}$. Similarly, we denote the $\mathcal{F}_n$ as a DFT matrix and write as:

$$\|\mathcal{X}\|_F^2=\left\|\mathcal{F}_{n_3}\otimes\mathcal{I}\begin{bmatrix}\mathcal{X}_{(1)} & & & \\ & \mathcal{X}_{(2)} & & \\ & & \cdots & \\ & & & \mathcal{X}_{(n_3)}\end{bmatrix}\mathcal{F}_{n_3}^*\otimes\mathcal{I}\right\|_F^2. \tag{34}$$

Due to the unitary invariance of Frobenius norm, we have that $\|\hat{\mathcal{X}}\|_F^2=n_3\|\mathcal{X}\|_F^2$ where $\hat{\mathcal{X}}$ denotes the Fourier-transformed tensor of $\mathcal{X}$ along the third mode. Then, we can write:

$$\|\mathcal{X}\|_F^2=\frac{1}{n_3}\left\|\begin{bmatrix}\hat{\mathcal{X}}_{(1)} & & & \\ & \hat{\mathcal{X}}_{(2)} & & \\ & & \cdots & \\ & & & \hat{\mathcal{X}}_{(n_3)}\end{bmatrix}\right\|_F^2=\frac{1}{n_3}\left\|\begin{bmatrix}\hat{\mathcal{S}}_{(1)} & & & \\ & \hat{\mathcal{S}}_{(2)} & & \\ & & \cdots & \\ & & & \hat{\mathcal{S}}_{(n_3)}\end{bmatrix}\right\|_F^2. \tag{35}$$

We write the singular values of all $\hat{\mathcal{X}}_{(i)}$'s as $\hat{\sigma}_1\geqslant\hat{\sigma}_2\geqslant\cdots\geqslant\hat{\sigma}_k>\hat{\sigma}_{k+1}=0$, then we have:

$$\frac{1}{n_3}\left\|\begin{bmatrix}\hat{\mathcal{S}}_{(1)} & & & \\ & \hat{\mathcal{S}}_{(2)} & & \\ & & \cdots & \\ & & & \hat{\mathcal{S}}_{(n_3)}\end{bmatrix}\right\|_F^2=\frac{1}{n_3}\left\|\begin{bmatrix}\hat{\sigma}_1 & & & & \\ & \hat{\sigma}_2 & & & \\ & & \hat{\sigma}_3 & & \\ & & & \cdots & \\ & & & & \hat{\sigma}_k\end{bmatrix}\right\|_F^2. \tag{36}$$

Since $\texttt{Prox}_{\lambda g}(\mathcal{X})=\mathrm{t}-\mathrm{SVT}(\mathcal{X})$, we can directly write:

$$\|\texttt{Prox}_{\lambda g}(\mathcal{X})\|_F^2=\frac{1}{n_3}\left\|\begin{bmatrix}\hat{\sigma}_1-\lambda & & & & & \\ & \cdots & & & & \\ & & \hat{\sigma}_{k_1}-\lambda & & & \\ & & & 0 & & \\ & & & & \cdots & \\ & & & & & 0\end{bmatrix}\right\|_F^2 \tag{37}$$

and

$$\| \left(\texttt{Prox}_{\lambda g} - \mathcal{I}\right)(\mathcal{X}) \|_F^2 = \frac{1}{n_3} \left\| \begin{bmatrix} -\lambda & & & & & & \\ & \cdots & & & & & \\ & & -\lambda & & & & \\ & & & -\hat{\sigma}_{k_1+1} & & & \\ & & & & \cdots & & \\ & & & & & -\hat{\sigma}_k & \end{bmatrix} \right\|_F^2, \tag{38}$$

where k_1 denotes the count of non-zero singular values of $\mathcal{X}$. Based on (38), we reshape the $\mathcal{X}$ and $\mathcal{Y}$ into the vector $\mathbf{x}$ and $\mathbf{y}$, then we can substitute the 2-norm of the vector for the Frobenius norm of the tensor. Therefore, we can write:

$$\begin{aligned} \| \left(\texttt{Prox}_{\lambda g} - \mathcal{I}\right)(\mathbf{x}) \|_2 &= \frac{1}{n_3} \sqrt{k_1\lambda^2 + \sum_{i=k_1+1}^{k} (\hat{\sigma}_i^{\mathbf{x}})^2}, \text{and} \\ \| \left(\texttt{Prox}_{\lambda g} - \mathcal{I}\right)(\mathbf{y}) \|_2 &= \frac{1}{n_3} \sqrt{l_1\lambda^2 + \sum_{i=l_1+1}^{l} \left(\hat{\sigma}_i^{\mathbf{y}}\right)^2}. \end{aligned} \tag{39}$$

Similarly, we can directly write:

$$\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{k} (\sigma_i^{\mathbf{x}})^2}, \quad \|\mathbf{y}\|_2 = \sqrt{\sum_{i=1}^{l} \left(\sigma_i^{\mathbf{y}}\right)^2}. \tag{40}$$

According to the triangle inequality, we can write:

$$\begin{aligned} &\epsilon^2 \|\mathbf{x}-\mathbf{y}\|_2^2 \geqslant \epsilon^2 \left(\|\mathbf{x}\|_2 - \|\mathbf{y}\|_2\right)^2, \\ &\| \left(\texttt{Prox}_{\lambda g} - \mathcal{I}\right)(\mathbf{x}) - \left(\texttt{Prox}_{\lambda g} - \mathcal{I}\right)(\mathbf{y}) \|_2^2 \leqslant \left(\| \left(\texttt{Prox}_{\lambda g} - \mathcal{I}\right)(\mathbf{x}) \|_2 + \| \left(\texttt{Prox}_{\lambda g} - \mathcal{I}\right)(\mathbf{y}) \|_2\right)^2. \end{aligned} \tag{41}$$

Therefore, if the λ , $\{\sigma_i^{\mathbf{x}}\}(i = 1, 2, \cdots, k)$ and $\{\sigma_i^{\mathbf{y}}\}(i = 1, 2, \cdots, l)$ satisfy the below inequality:

$$\begin{aligned} &\left(\sqrt{k_1\lambda^2 + \sum_{i=k_1+1}^{k} (\hat{\sigma}_i^{\mathbf{x}})^2} + \sqrt{l_1\lambda^2 + \sum_{i=l_1+1}^{l} \left(\hat{\sigma}_i^{\mathbf{y}}\right)^2} \right)^2 \\ &\leqslant {n_3}^2\epsilon^2 \left(\sqrt{\sum_{i=1}^{k} (\sigma_i^{\mathbf{x}})^2} - \sqrt{\sum_{i=1}^{l} \left(\sigma_i^{\mathbf{y}}\right)^2} \right)^2, \end{aligned} \tag{42}$$

the Assumption (A) will be true.

Theorem 2 (Convergence of PALADIN). *Assume that $\mathcal{B}_1$ satisfies assumption (A) for some $\epsilon \geqslant 0$. f in (21) is γ-strongly convex and differentiable. Then*

$$\mathcal{T} = \frac{1}{2}\mathcal{I} + \frac{1}{2}\left(2\mathcal{A}_1\texttt{Prox}_{\rho f} - \mathcal{I}\right)\left(2\mathcal{B}_1 - \mathcal{I}\right) \tag{43}$$

satisfies

$$\|\mathcal{T}\mathbf{x}-\mathcal{T}\mathbf{y}\| \leqslant \frac{1+\epsilon+\epsilon\gamma_1+2\epsilon^2\gamma_1}{1+\gamma_1+2\epsilon\gamma_1}\|\mathbf{x}-\mathbf{y}\|$$

for all $\mathbf{x},\mathbf{y}\in\mathbb{R}^d$. *The coefficient is less than 1 if*

$$\frac{\epsilon}{(1+\epsilon-2\epsilon^2)}<\gamma_1,\quad \epsilon<1.$$

4.3. Proof for theorem

4.3.1. The roadmap for proof. Following [42], we validate the nonexpansiveness of the iterative updating, i.e. the operator $\mathcal{T}$ in (43), to obtain the fixed-point convergence. To analyze the property of $\mathcal{T}$, we analyze the the averagedness of its two key components. Thus, we need to prove:

$$\begin{cases}\text{(i) the averagedness of operator } (2\mathcal{A}_1\mathtt{Prox}_{\rho f}-\mathcal{I}),\\ \text{(ii) the averagedness of operator } (2\mathcal{B}_1-\mathcal{I}).\end{cases}$$

The averagedness of operator $(2\mathcal{B}_1-\mathcal{I})$ relies on the aforementioned Assumption (A). Then, after validating the averagedness of those operators, we prove the nonexpansiveness of $\mathcal{T}$, yielding the fixed-point convergence of our algorithm.

4.3.2. Necessary lemmas. Before verifying these conditions, we first give the required lemmas of the proof.

Lemma 1 (proposition 4.35 of [3]). *$\mathcal{T}:\mathbb{R}^d\to\mathbb{R}^d$ is θ-averaged if and only if*

$$\|\mathcal{T}(\mathbf{x})-\mathcal{T}(\mathbf{y})\|^2+(1-2\theta)\|\mathbf{x}-\mathbf{y}\|^2\leqslant 2(1-\theta)\langle\mathcal{T}(\mathbf{x})-\mathcal{T}(\mathbf{y}),\mathbf{x}-\mathbf{y}\rangle \quad (44)$$

for all $\mathbf{x},\mathbf{y}\in\mathbb{R}^d$.

Lemma 2 (proposition 5.4 of [18]). *If f is γ-strongly convex, closed, and proper, then*

$$-(2\mathtt{Prox}_{\rho f}-\mathcal{I})$$

is nonexpansive and $\frac{1}{1+\gamma_1}$-averaged.

Lemma 3 ([42]). *$\mathcal{B}_1:\mathbb{R}^d\to\mathbb{R}^d$ satisfies Assumption (A) if and only if*

$$\frac{1}{1+\epsilon}(\mathcal{B}_1) \quad (45)$$

is nonexpansive and $\frac{\epsilon}{1+\epsilon}$-averaged.

Lemma 4 ([42]). *$\mathcal{B}_1:\mathbb{R}^d\to\mathbb{R}^d$ satisfies Assumption (A) if and only if*

$$\frac{1}{1+2\epsilon}(2\mathcal{B}_1-\mathcal{I}) \quad (46)$$

is nonexpansive and $\frac{2\epsilon}{1+2\epsilon}$-averaged.

Lemma 5 ([42]). *Let $\mathcal{T}:\mathbb{R}^d\to\mathbb{R}^d$. $-\mathcal{T}$ is θ-averaged if and only if $\mathcal{T}\circ(-\mathcal{I})$ is θ-averaged.*

Lemma 6 ([12, 36]). *.Assume $\mathcal{T}_1:\mathbb{R}^d\to\mathbb{R}^d$ and $\mathcal{T}_2:\mathbb{R}^d\to\mathbb{R}^d$ are θ_1 and θ_2-averaged, respectively. Then $\mathcal{T}_1\mathcal{T}_2$ is $\frac{\theta_1+\theta_2-2\theta_1\theta_2}{1-\theta_1\theta_2}$-averaged.*

4.3.3. Proof of theorem 2.

Proof. By lemma 2,

$$-\left(2\texttt{Prox}_{\rho f}-\mathcal{I}\right)$$

is $\frac{1}{1+\rho\gamma}$-averaged, and this implies

$$\left(2\texttt{Prox}_{\rho f}-\mathcal{I}\right)\circ\left(-\mathcal{I}\right)$$

is also $\frac{1}{1+\rho\gamma}$-averaged, by lemma 5. By lemma 2, we have that the $\texttt{Prox}_{\rho f}$ is maximally monotone and strongly monotone. We assume that $\mathcal{A}_1\,\texttt{Prox}_{\rho f}$ is also maximally monotone and γ_1-strongly monotone. Since $\mathcal{A}_1$ is a linear operator that preserves the monotonicity of $\texttt{Prox}_{\rho f}$, the assumption is reasonable. Consequently, the operator

$$\left(2\mathcal{A}_1\texttt{Prox}_{\rho f}-\mathcal{I}\right)\circ\left(-\mathcal{I}\right)$$

is also non-expansive and $\frac{1}{1+\gamma_1}$-averaged.

By lemma 4, $\frac{1}{1+2\epsilon}\left(2\mathcal{B}_1-\mathcal{I}\right)$ is $\frac{2\epsilon}{1+2\epsilon}$-averaged. Therefore,

$$\frac{1}{1+2\epsilon}\left(2\mathcal{A}_1\texttt{Prox}_{\rho f}-\mathcal{I}\right)\left(2\mathcal{B}_1-\mathcal{I}\right)\circ\left(-\mathcal{I}\right)$$

is $\frac{1+2\epsilon\gamma_1}{1+\gamma_1+2\epsilon\gamma_1}$-averaged by lemma 6, and this implies

$$-\frac{1}{1+2\epsilon}\left(2\mathcal{A}_1\texttt{Prox}_{\rho f}-\mathcal{I}\right)\left(2\mathcal{B}_1-\mathcal{I}\right)$$

is also $\frac{1+2\epsilon\gamma_1}{1+\gamma_1+2\epsilon\gamma_1}$-averaged, by lemma 5. Applying the concept of averagedness, we have:

$$\left(2\mathcal{A}_1\texttt{Prox}_{\rho f}-\mathcal{I}\right)\left(2\mathcal{B}_1-\mathcal{I}\right)=-\left(1+2\epsilon\right)\left(\frac{\gamma_1}{1+\gamma_1+2\epsilon\gamma_1}\mathcal{I}+\frac{1+2\epsilon\gamma_1}{1+\gamma_1+2\epsilon\gamma_1}\mathcal{R}\right), \tag{47}$$

where $\mathcal{R}$ is a nonexpansive operator. Substituting this into our operator $\mathcal{T}$, we obtain:

$$\begin{aligned}\mathcal{T}&=\frac{1}{2}\mathcal{I}-\frac{1}{2}\left(1+2\epsilon\right)\left(\frac{\gamma_1}{1+\gamma_1+2\epsilon\gamma_1}\mathcal{I}+\frac{1+2\epsilon\gamma_1}{1+\gamma_1+2\epsilon\gamma_1}\mathcal{R}\right)\\&=\underbrace{\frac{1}{2\left(1+\gamma_1+2\epsilon\gamma_1\right)}}_{A}\mathcal{I}-\underbrace{\frac{\left(1+2\epsilon\gamma_1\right)\left(1+2\epsilon\right)}{2\left(1+\gamma_1+2\epsilon\gamma_1\right)}}_{B}\mathcal{R},\end{aligned} \tag{48}$$

where define the coefficients A and B for simplicity. Clearly, $A>0$ and $B>0$. Then we can write:

$$\begin{aligned}\left\|\mathcal{T}\mathbf{x}-\mathcal{T}\mathbf{y}\right\|^2=&A^2\left\|\mathbf{x}-\mathbf{y}\right\|^2+B^2\left\|\mathcal{R}\left(\mathbf{x}\right)-\mathcal{R}\left(\mathbf{y}\right)\right\|^2-2\left\langle A\left(\mathbf{x}-\mathbf{y}\right),B\left(\mathcal{R}\left(\mathbf{x}\right)-\mathcal{R}\left(\mathbf{y}\right)\right)\right\rangle\\\leqslant&A^2\left(1+\frac{1}{\delta}\right)\left\|\mathbf{x}-\mathbf{y}\right\|^2+B^2\left(1+\delta\right)\left\|\mathcal{R}\left(\mathbf{x}\right)-\mathcal{R}\left(\mathbf{y}\right)\right\|^2\\\leqslant&\left(A^2\left(1+\frac{1}{\delta}\right)+B^2\left(1+\delta\right)\right)\left\|\mathbf{x}-\mathbf{y}\right\|^2\end{aligned} \tag{49}$$

for any $\delta > 0$. The first line results from substituting (48). The second line is derived by applying Young's inequality to the inner product. The third line is derived from the nonexpansiveness of $\mathcal{R}$.Lastly, using basic calculus, we find that

$$\min_{\delta>0}\left\{A^2\left(1+\frac{1}{\delta}\right)+B^2\left(1+\delta\right)\right\}=\left(A+B\right)^2.$$

According to the above, we get

$$\|\mathcal{T}\mathbf{x}-\mathcal{T}\mathbf{y}\|^2 \leqslant \left(A+B\right)^2\|\mathbf{x}-\mathbf{y}\|^2=\left(\frac{1+\epsilon+\epsilon\gamma_1+2\epsilon^2\gamma_1}{1+\gamma_1+2\epsilon\gamma_1}\right)^2\|\mathbf{x}-\mathbf{y}\|^2. \quad (50)$$

The constraints on γ_1 and ϵ follow from elementary algebraic manipulations. □

5. Numerical experiments

This section presents a comparison of our method with the following CS-MRI reconstruction methods: (1) DCCNN [43], a deep cascade of convolutional neural networks for MRI reconstruction, (2) SSLMRI [22], a self-supervised learning method for MRI reconstruction with a parallel network training framework, (3) LPD-Net [1], a deep neural network designed by unrolling a primal-dual algorithm, (4) ISTA-Net+ [61], a deep neural network designed by unrolling the ISTA algorithm, (5) ISTA-Net+(3D) [61], an 3D version of ISTA-Net+ with adopting the 3D separable convolutions [38] for 3D MRI data modified by us, (6) HQSNet [52], a deep neural network designed by unrolling a half-quadratic splitting algorithm, and (7) TRPA [21], a truncated residual based PnP ADMM algorithm for MRI reconstruction. Our Plug-And-pLAy 3D MRI recoNstruction method is named PALADIN.

5.1. Data sets

The widely used Information eXtraction from Images (IXI) data set is used in this study. The information about IXI data set can be found from the website[5]. We conduct our experiments using T2-weighted MRI images of the brain. All the deep neural network methods are retrained on this data set. For all 2D deep neural network methods, the train set consists of 15 subjects. For ISTA-Net+(3D), the train set consists of 200 subjects. This is accomplished by regarding every continuous 16 images of one subject as a cubic volume. The test set includes 10 subjects, each with a 3D MR image sized at $256 \times 256 \times 128$. Meanwhile, as shown in figure 2, we consider three types of commonly used undersampling masks, i.e. the Gaussian 2D, the Cartesian 1D, and the radial masks. That is, the compressive sensing measurements are obtained by (3) with a selected mask across 128 frontal slices. We select three acceleration rates, i.e. 5 ×, 10 ×, and 20 ×. The NYU fastMRI data set is also used in this study [59]. The train set of 2D deep neural network consists of 100 subjects. The train set of ISTA-Net+(3D) consists of 625 subjects. The test set includes 8 subjects, each with a 3D MR image sized at $320 \times 320 \times 32$. For the fastMRI data set, we used the Cartesian 1D mask and selected three acceleration rates, i.e. 5 ×, 10 ×, and 20 ×. Prior to the undersampling process, all data sets were normalized to the interval of [0,1].

In our experiment, we employed our modified pretrained ISTA-Net+(3D) [61] as the reconstructor for experiments. We adopt the DRUNet [62] as the denoiser and fine tune it on MR

[5] http://brain-development.org/ixi-dataset/.

Figure 2. Examples of undersampling masks used in this study. White pixels indicate sampling points. Left: the Gaussian 2D mask. Middle: the Cartesian 1D mask. Right: the radial mask.

Table 1. Quantitative results by different methods on MR images with the Gaussian 2D mask. The **best** values are highlighted by **bolder fonts**.

Acceleration rate	5×		10×		20×	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Observed	33.95	0.9471	30.55	0.9000	28.19	0.8372
DCCNN	45.09	0.9942	38.33	0.9780	32.79	0.9422
HQSNet	46.06	0.9962	38.30	0.9833	32.67	0.9498
ISTA-Net+	45.31	0.9956	38.38	0.9816	32.87	0.9459
ISTA-Net+ (3D)	45.56	0.9958	38.99	0.9842	33.90	0.9563
LPD-Net	45.45	0.9957	39.29	0.9852	33.06	0.9495
SSLMRI	41.42	0.9885	37.61	0.9783	33.14	0.9480
TRPA	46.31	0.9962	39.47	0.9839	33.34	0.9468
PALADIN	**46.93**	**0.9967**	**40.42**	**0.9884**	**36.04**	**0.9716**

images in the training set. We evaluate the quality of our results by using two quantitative metrics: the peak signal-to-noise ratio (PSNR) and the structural similarity index (SSIM) [51]. The training processes are conducted on the platform of CentOS and Pytorch with an Intel Xeon Gold 6326 CPU and an Nvidia A100 GPU while the testing is conducted on the same platform with a RTX 3090 GPU.

5.2. Results with the Gaussian 2D mask on IXI data set

Table 1 shows the PSNR and SSIM values obtained using different methods with the Gaussian 2D mask and sampling at different acceleration rates. Our method shows superior performance compared to all other methods across all acceleration rates. Meanwhile, as the acceleration rate increases, the advantages of PALADIN become more evident. For example, while the PSNR value of PALADIN is only 0.62dB higher than TRPA at an acceleration rate of 5×, our PALADIN outperforms TRPA by 2.70dB at the acceleration rate of 20×. Figure 3 shows the visual results with the Gaussian 2D mask and acceleration rates of $5\times, 10\times$ and $20\times$, respectively. As shown in the enlarged areas of the results at the acceleration rate of $20\times$, our method generates a excellent result that is very close to the original fully-sampled data. In the enlarged areas of figure 3, we can see that our method generates reliable results while obvious artifacts can be found from results by some deep learning methods, such as LPD-Net.

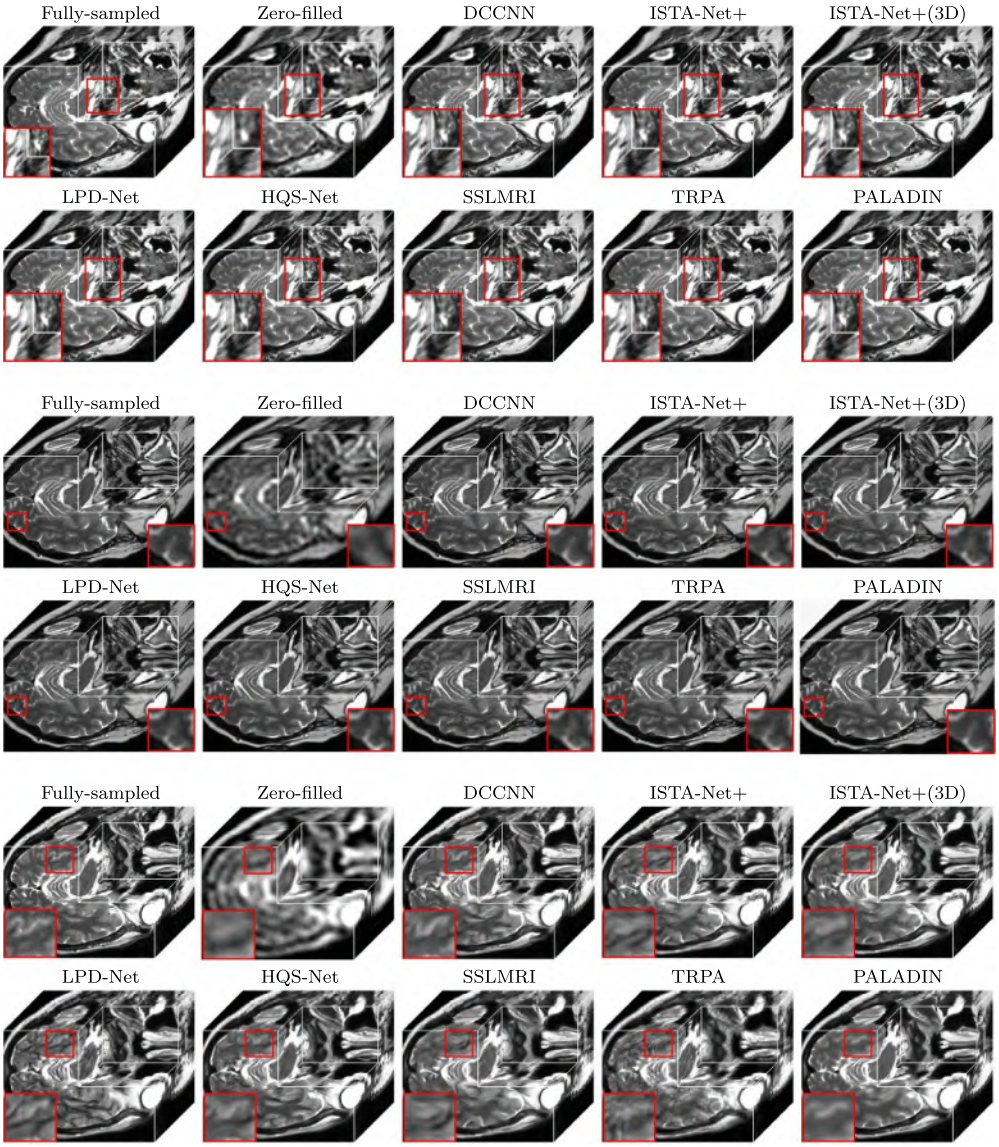

Figure 3. Reconstructed results with the Gaussian 2D mask. The acceleration rate from top to bottom are $5\times, 10\times$ and $20\times$ respectively.

5.2.1. Results with the Cartesian 1D mask on IXI data set. Table 2 presents a comparison of the results obtained through Cartesian 1D sampling. It is evident from the table that obtaining good results and indicators through Cartesian 1D sampling is comparatively difficult. However, our proposed method achieves a significant advantage in Cartesian 1D sampling. Specifically, our method yields a PSNR and SSIM that are 5.55 dB and 0.0608 higher, respectively, than ISTA-Net+(3D) for an accelerated rate of $10\times$. Figure 4 shows the visual results with the Cartesian 1D mask and acceleration rates of $5\times, 10\times$ and $20\times$, respectively. Because the

Table 2. Quantitative results by different methods on **MR images** with Cartesian 1D mask. The **best** values are highlighted by **bolder fonts**.

Acceleration rate	5×		10×		20×	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Observed	28.04	0.8392	24.83	0.7582	22.84	0.6967
DCCNN	37.21	0.9753	28.05	0.8210	22.65	0.7409
HQSNet	36.71	0.9767	28.27	0.8725	23.52	0.7903
ISTA-Net+	35.72	0.9699	28.28	0.8899	23.64	0.7842
ISTA-Net+(3D)	36.45	0.9741	28.70	0.8960	24.58	0.8121
LPDNet	35.73	0.9692	26.94	0.8570	22.11	0.7416
SSLMRI	35.71	0.9691	28.24	0.8904	23.80	0.7860
TRPA	37.78	0.9798	28.21	0.8894	23.76	0.7707
PALADIN	**40.24**	**0.9877**	**34.25**	**0.9568**	**27.16**	**0.8666**

Cartesian 1D mask largely damages the overall structure of the image, the reconstruction results tend to lose the important structure information. Nonetheless, our method effectively preserves the overall structure, as shown in figure 4. In contrast, all other comparing methods suffer from a significant loss of the image's spatial structure.

5.3. Results with the radial mask on IXI data set

Table 3 presents a comparison of the results obtained through radial sampling. Similarly, we can see that our method can obtain the best values. For example, our PALADIN yields an average PSNR value that is 1.95 dB higher than HQSNet at three acceleration rates. Figure 5 exhibits the visual results with the radial mask and acceleration rates of 5 ×, 10 × and 20 ×, respectively. We can see that our results are closer to the original fully-sampled data than the comparing methods at different acceleration rates. Our method successfully recovers detailed textures with minimal artifacts. In the enlarged areas of figure 5, we can see that our method yields excellent results even at an acceleration rate of 20×.

5.4. Results on fastMRI data set

Table 4 presents a comparison of the results on the fastMRI data set with the Cartesian mask at different acceleration rates. In particular, as the acceleration rate increases, our method gains greater advantages over competing methods. Specifically, our method yields a PSNR and SSIM that are 1.52 dB and 0.0249 higher, respectively, than LPDNet for an accelerated rate of 20 ×. Figure 6 shows up the visual quality of results with the Cartesian 1D mask at acceleration rates of 5 ×, 10 × and 20 ×, respectively. We can see that PALADIN can generate most closest results to the original fully-sampled data. The enlarged regions reveal that our method produces clearer details, whereas other methods, such as LPDNet, exhibit significant artifacts. For example, as shown in figure 6, we can observe that our method not only reduces artifacts but also improves image details.

5.5. Discussions

5.5.1. Parameter analysis and convergence. As we will discuss in the subsequent section, one limitation of our method is that we need to manually tune parameters in our model. Fortunately, we find that our method is not sensitive to parameters. To validate this, we test our

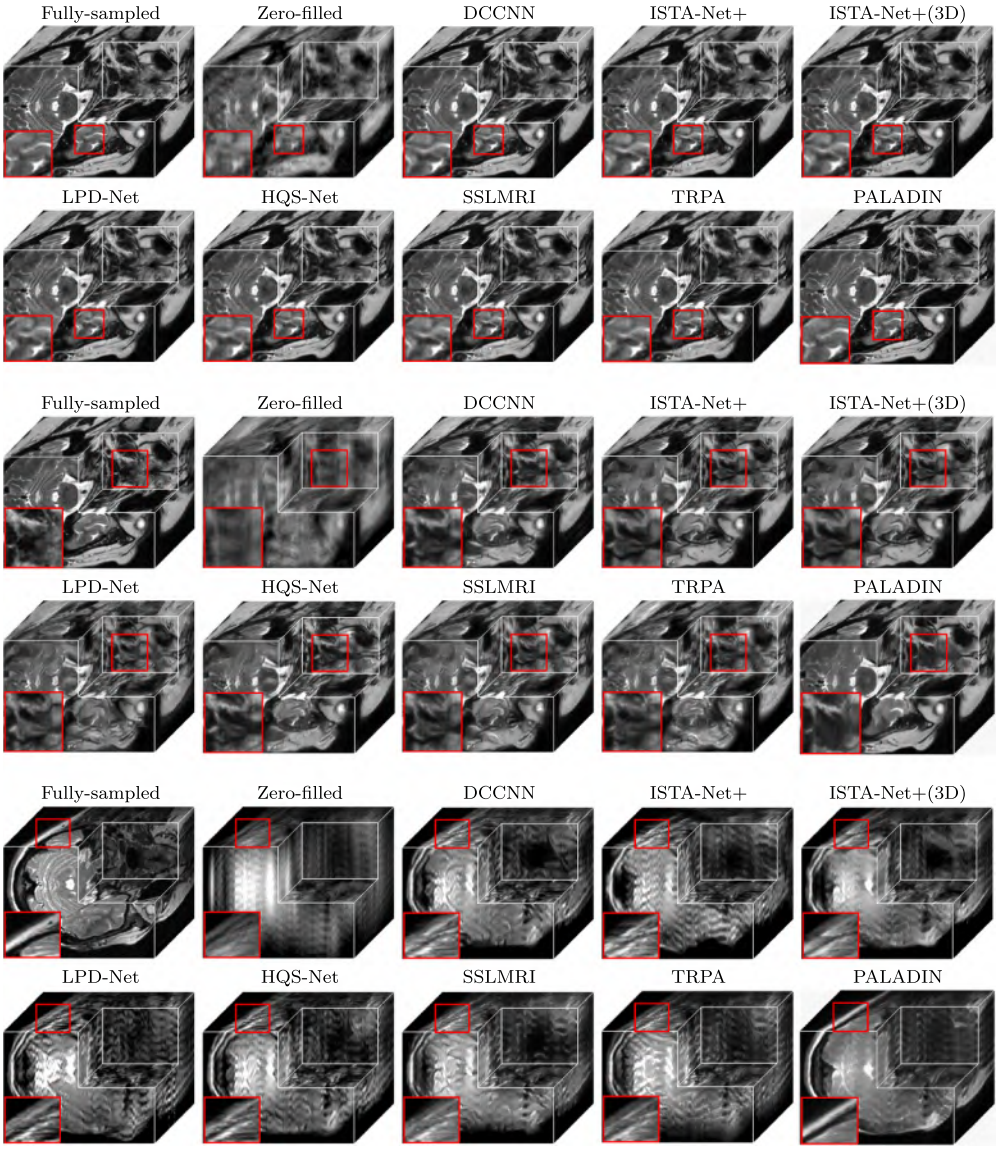

Figure 4. Reconstructed results with the Cartesian 1D mask. The acceleration rate from top to bottom are $5\times, 10\times$ and $20\times$ respectively.

PALADIN with different parameters in our model, i.e. λ, β_1, and β_2. We conducted experiments with the Gaussian 2D mask at the acceleration rate of $5\times$. We first acquire an optimal combination of parameters by manually tuning the parameters. During parameter analysis experiment, one parameter is varying while the other parameters are maintained at optimal values. Figure 7 illustrates the PSNR and SSIM values of results by PALADIN with the different parameters. We observe that the performance of our PALADIN remains stable across different

Table 3. Quantitative results by different methods on MR images with the radial mask. The **best** values are highlighted by **bolder fonts**.

Acceleration rate	5×		10×		20×	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Observed	31.63	0.8810	28.00	0.7962	25.16	0.7143
DCCNN	44.51	0.9934	37.31	0.9696	31.05	0.8982
HQSNet	44.22	0.9950	37.81	0.9812	31.64	0.9304
ISTA-Net+	44.65	0.9950	37.76	0.9799	31.17	0.9279
ISTA-Net+(3D)	45.06	0.9956	38.31	0.9815	32.01	0.9384
LPDNet	45.17	0.9952	37.77	0.9796	31.51	0.9288
SSLMRI	41.58	0.9887	37.31	0.9769	30.83	0.9201
TRPA	45.44	0.9958	38.50	0.9823	31.83	0.9325
PALADIN	**45.88**	**0.9962**	**39.51**	**0.9849**	**34.13**	**0.9560**

parameter settings. Meanwhile, we also illustrate the relative changes of each variable concerning iterations in this experimental setting in figure 7, and the empirical convergence behavior of our method can be found.

5.5.2. Robustness of PALADIN. To test the robustness of different methods against noise, we evaluated all methods on MRI data corrupted by additive white Gaussian noise (with the standard deviation $\sigma = 5$, 10, and 15) and Poisson noise (with the SNR = 20, 15, and 10). The quantitative results are summarized in tables 5 and 6. We observe that deep learning methods trained on noise-free data exhibit significant performance degradation under increasing noise levels. Our PALADIN demonstrates superior robustness, particularly under high noise. For Gaussian noise with $\sigma = 15$, PALADIN achieves a PSNR of 33.41 dB and SSIM of 0.9320, outperforming the second-best method (TRPA: PSNR = 30.73 dB) by 2.68 dB. Similarly, under Poisson noise with SNR = 10, PALADIN maintains high reconstruction quality (PSNR = 34.30 dB, SSIM = 0.9529), surpassing ISTANet+(3D) and SSLMRI by 4.91 dB and 3.97 dB, respectively. This advantage stems from PALADIN's unique design, which integrates a CNN denoiser for local detail restoration and an MRI reconstructor for global structure preservation. In contrast, other methods struggle under high noise. For example, HQSNet's SSIM drops to 0.7418 at $\sigma = 15$, and DCCNN's PSNR falls to 27.44 dB under the same conditions. While ISTANet+(3D) shows better resilience than its 2D counterpart, it still underperforms compared to PALADIN, particularly under extreme noise. These results highlight PALADIN's ability to maintain high reconstruction quality across diverse noise levels, making it a promising solution for real-world MRI applications where noise characteristics are highly variable.

5.5.3. Ablation study

Ablation study on regularizers As there are three regularizers in our model, it is essential to test their distinct effects on the performance. We conduct experiments with the Cartesian 1D mask and acceleration rates of 5×, 10×, and 20×, to test different degenerated versions of our PALADIN. All combinations of three regularizers are considered. Then we also consider the 2D deep neural networks for the reconstructor, i.e. ISTA-Net+ [61] and ReconFormer [20], of which ReconFormer is the latest state-of-the-art method. As shown in table 7, when ISTA-Net+(3D) was plugged into our algorithm, it outperformed the insertion of 2D neural networks, i.e. PALADIN with ISTA-Net+. However, from the last three rows of table 7, we

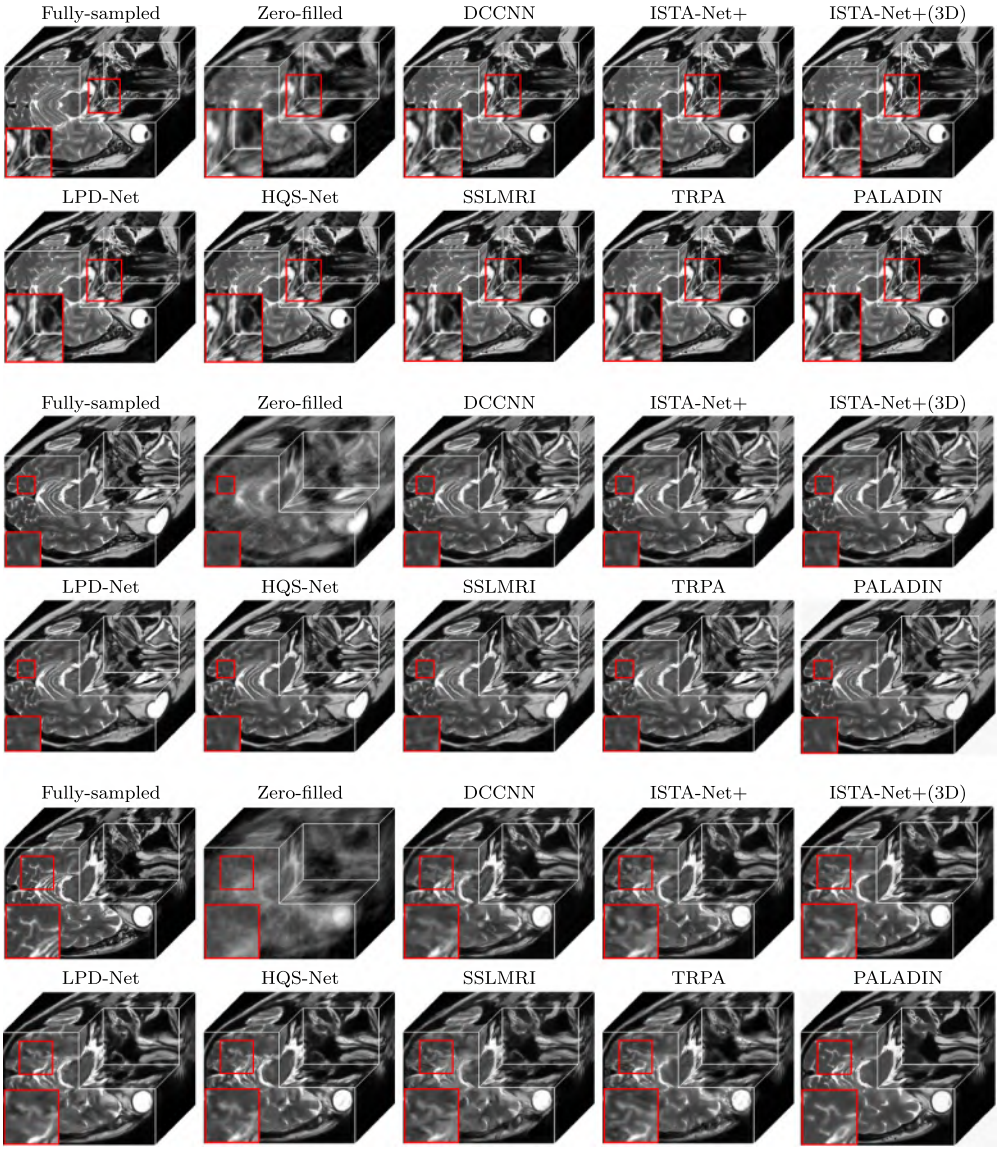

Figure 5. Reconstructed results with the radial mask. The acceleration rate from top to bottom are $5\times, 10\times$ and $20\times$ respectively.

can see that our methods with different reconstructors all achieve better results than compared methods in table 4. If we consider the ReconFormer for the reconstructor, which utilizes the advanced vision transformer structure, our method can also acquire more excellent performance.

From table 7, we can see that the reconstructor plays a dominant role in our method. Meanwhile, the denoiser and the TNN also contribute to the superior performance of our method. Specifically, the priors learned by the denoiser and reconstructor are not the same.

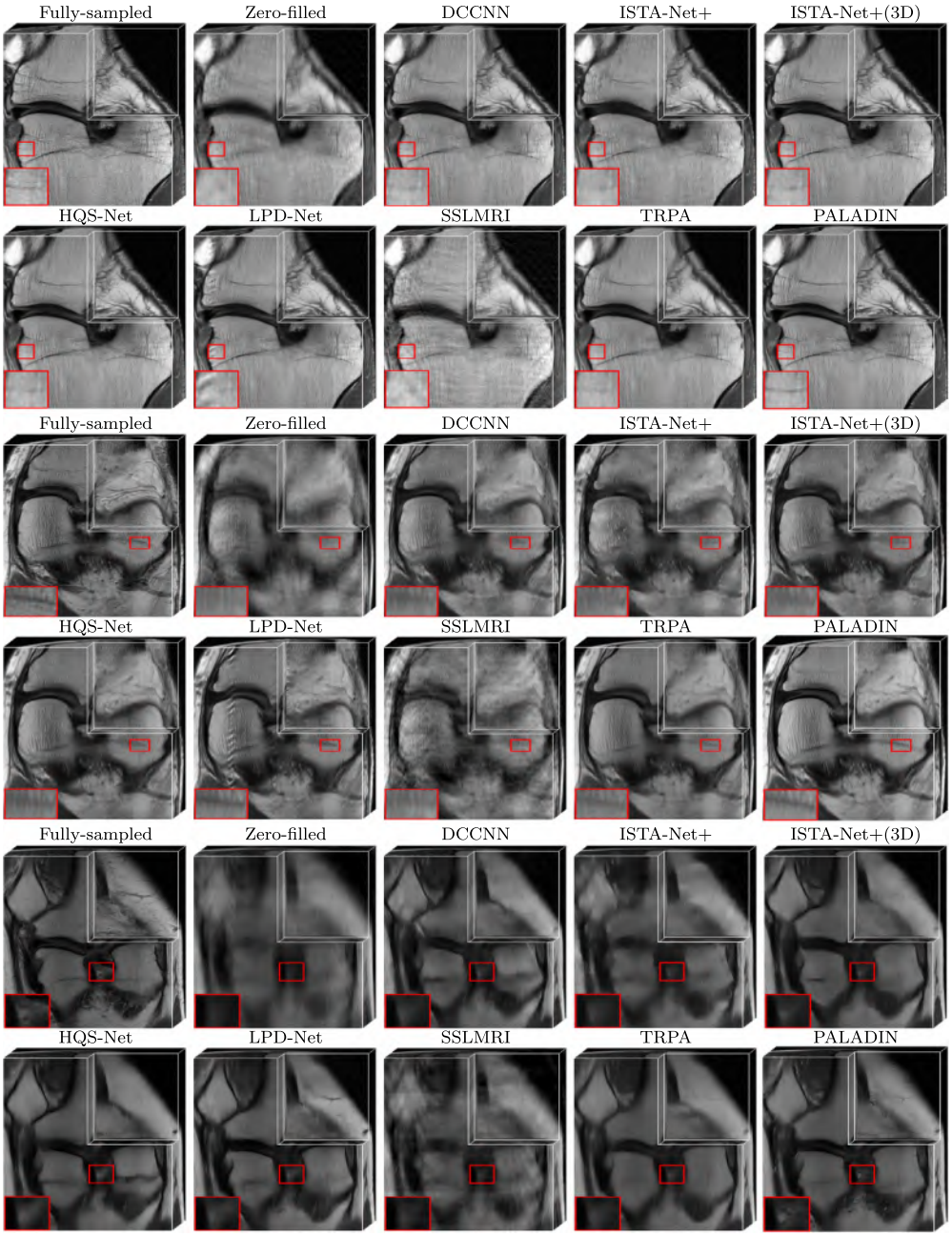

Figure 6. Reconstructed results on fastMRI data set with the Cartesian 1D mask. The acceleration rate from top to bottom are $5\times, 10\times$ and $20\times$ respectively.

Table 4. Quantitative results by different methods on MR images of the fastMRI data set with the Cartesian 1D mask. The **best** values are highlighted by **bolder fonts** .

Acceleration rate	5×		10×		20×	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Observed	31.16	0.8458	28.11	0.7694	24.88	0.6976
DCCNN	35.11	0.9159	31.65	0.8606	28.31	0.7811
HQSNet	35.20	0.9179	32.11	0.8715	28.58	0.7835
ISTA-Net+	34.10	0.8962	30.18	0.8310	27.62	0.7518
ISTA-Net+(3D)	35.31	0.9180	31.66	0.8601	29.02	0.7978
LPDNet	34.84	0.9154	31.14	0.8529	29.07	0.8029
SSLMRI	31.61	0.8459	28.37	0.7717	26.04	0.7088
TRPA	33.92	0.8660	31.41	0.8143	28.86	0.7519
PALADIN	**35.90**	**0.9269**	**32.60**	**0.8765**	**30.59**	**0.8278**

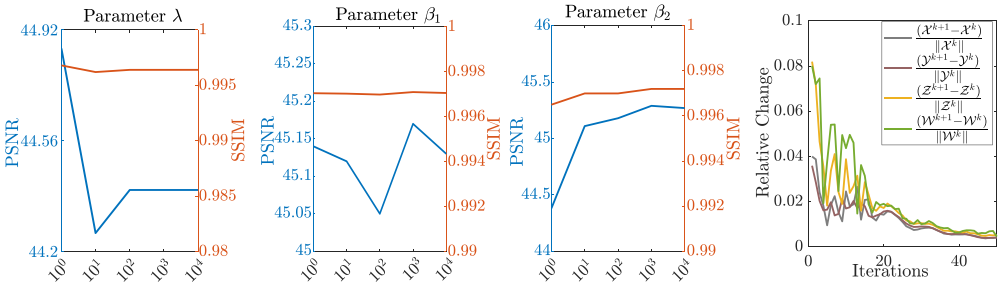

Figure 7. The effect of the parameters β_1, β_2 and λ on the PALADIN and relative changes of variables with respect to iterations.

Generally, denoising CNNs, e.g. the DRUNet utilized in our method, adopt the residual learning strategy of learning the noise from the noisy observation. Therefore, those denoising CNNs concentrate more on the fine details to infer the noise, whose energy is lower than the image signal. In contrast, the MRI reconstruction CNNs need to take the degradation of the undersampling into account. Thus, they might concentrate more on the image structures.

As analyzed above, in our method, the denoiser plays a refining role whereas the reconstructor serves more for structural reconstruction. In figure 8, we can visually observe that all functions appear to be complementary, as previously mentioned. Additionally, we can see that the reconstructor can contribute more to global structural information, which reduces the errors in the low-frequency parts.

Ablation study on training data sets In our method, the reconstructor and denoiser can be trained on natural images. As MR images are relatively fewer than natural images, the training set of natural images might consist of more contents, e.g. geometrical shapes, which would be beneficial for MRI reconstructors to achieve good performance. Table 8 performs the results with different training data sets. As we can see, our method could obtain good performance with different training data sets.

Ablation study on low rank In our method, the low-rank regularizer can be replaced by other alternatives. From table 9, we can see the results of PALADIN with different low-rank terms. Although the effect is relatively minor, different low-rank terms can potentially influence the

Table 5. Quantitative results of different methods on Gaussian noisy MR Images (with the standard deviation $\sigma = 5$, 10, and 15) from the IXI data set with the Gaussian 2D mask at the acceleration factor of $10\times$. The **best** values are highlighted by **bolder fonts**.

	Noise-less		Gaussian					
Noise level	$\sigma = 0$		$\sigma = 5$		$\sigma = 10$		$\sigma = 15$	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Observed	32.43	0.9050	31.44	0.9035	30.74	0.8812	30.05	0.8499
DCCNN	40.07	0.9859	34.16	0.9072	30.33	0.7984	27.44	0.6796
HQSNet	40.06	0.9864	30.07	0.8799	27.96	0.8133	26.12	0.7418
ISTANet+	39.91	0.9854	34.13	0.9499	28.62	0.8628	26.54	0.7795
ISTANet+(3D)	40.61	0.9875	35.12	0.9647	30.59	0.9120	27.99	0.8585
LPDNet	40.97	0.9882	32.09	0.9210	28.94	0.8451	26.66	0.7648
SSLMRI	39.28	0.9823	35.05	0.9594	30.91	0.9037	28.16	0.8376
TRPA	41.01	0.9870	35.72	0.9616	32.88	0.9081	30.73	0.8410
PALADIN	**41.90**	**0.9905**	**37.88**	**0.9755**	**36.05**	**0.9598**	**33.41**	**0.9320**

Table 6. Quantitative results of different methods on Poisson noisy MR Images (with the signal-to-noise ratio (SNR) = 20, 15, and 10) from the IXI data set with the Gaussian 2D mask at the acceleration factor of $10\times$. The **best** values are highlighted by **bolder fonts**.

	Noise-less		Poisson					
Noise level	SNR $= +\infty$		SNR $= 20$		SNR $= 15$		SNR $= 10$	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Observed	32.43	0.9050	31.95	0.9098	31.69	0.9055	31.06	0.8922
DCCNN	40.07	0.9859	35.48	0.9355	33.17	0.9042	29.32	0.8123
HQSNet	40.06	0.9864	30.46	0.9050	29.20	0.8858	26.97	0.8449
ISTANet+	39.91	0.9854	36.95	0.9734	32.49	0.9377	27.82	0.8709
ISTANet+(3D)	40.61	0.9875	37.29	0.9768	33.28	0.9517	29.39	0.9072
LPDNet	40.97	0.9882	32.91	0.9482	30.91	0.9256	28.15	0.8833
SSLMRI	39.28	0.9823	36.92	0.9731	34.20	0.9563	30.33	0.9166
TRPA	41.01	0.9870	37.88	0.9792	34.94	0.9643	30.72	0.9228
PALADIN	**41.90**	**0.9905**	**38.83**	**0.9809**	**36.65**	**0.9707**	**34.30**	**0.9529**

results. In general, the application of SNN on low-rank terms tends to yield relatively better results.

Ablation study on dimensional configuration (2D vs 3D) The 2D adaptation of PALADIN demonstrates dimensional flexibility through three key modifications: (1) Replacement of TNN with NN for intra-slice correlation modeling, (2) Integration of 2D reconstructors/denoisers, and (3) Slice-wise independent processing of 3D MRI data. Quantitative results in table 10 reveal that while 2D reconstruction maintains baseline effectiveness (PSNR = 35.90 dB at $5\times$), 3D reconstruction shows progressive advantages with increased acceleration rates. The performance gap grows from $+0.07$ dB PSNR at $5\times$ to $+0.79$ dB at $10\times$ and $+0.35$ dB at $20\times$, demonstrating TNN's enhanced capability to leverage inter-slice correlations under

Table 7. Quantitative results by different degenerated versions of our PALADIN on MR images of the fastMRI data set with the Cartesian 1D mask. The **best** and the second best values are respectively highlighted by **bolder fonts** and underlines.

Acceleration rate			5×		10×		20×	
TNN	Reconstructor	Denoiser	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
✓			31.21	0.8464	28.10	0.7694	26.68	0.6976
	✓ ISTA-Net+ [61]		34.10	0.8962	30.18	0.8310	27.62	0.7518
		✓	31.31	0.8504	28.34	0.7754	25.05	0.7015
✓		✓	33.47	0.8942	30.97	0.8481	26.83	0.7481
✓	✓ ISTA-Net+ [61]		34.11	0.8963	30.20	0.8312	27.63	0.7529
	✓ ISTA-Net+ [61]	✓	34.19	0.8987	30.38	0.8343	27.94	0.7619
✓	✓ ISTA-Net+ [61]	✓	35.66	0.9220	32.03	0.8686	29.92	0.8068
✓	✓ ISTA-Net+(3D) [61]	✓	35.90	0.9269	32.60	0.8765	30.59	0.8278
✓	✓ ReconFormer [20]	✓	**35.97**	**0.9277**	**33.39**	**0.8847**	**30.94**	**0.8379**

severe undersampling. Figure 9 visualizes this trend across sampling patterns, showing 3D reconstruction's superior structural consistency in low-frequency regions.

6. Conclusions

In this work, we propose a novel PnP method for the reconstruction of 3D MR images from limited sampling measurements in k-space. The innovation of our method is to simultaneously utilize the tensor low-rankness, denoising algorithms, and MRI reconstruction algorithms, in a unified way. Specifically, we formulate a 3D MR image reconstruction model with one TNN and two implicit regularizers. Then, by introducing auxiliary variables, we iteratively solve our model via the ADMM algorithm. The low-rank regularizer exploit the inner correlation of tensor structure by utilizing the TNN. Two implicit regularizers related subproblems are respectively formulated as a denoising problem and an MRI reconstruction problem. Thus, under the PnP framework, we respectively plug in a CNN denoiser and deep learning based MRI reconstructors. The three regularizers complement each other effectively, recovering both the global structure and fine details within the image. In addition, we prove that the proposed algorithm can converge to a fixed point under a certain assumption based on the fixed point theory. Experiments are conducted with different undersampling masks at different acceleration rates. Results illustrate that our method is superior to state-of-the-art methods quantitatively and qualitatively. The ablation studies also demonstrate the high flexibility of our method. Specifically, the ablation study on regularizers shows that our method can be plugged into the most advanced reconstructors to obtain better performance. The ablation study on the training data set demonstrates that our method can utilize readily available 2D natural images to achieve excellent performance. Although our method maintains some merits and yields good performance, there is a limitation of our method that our method requires much more testing time than the deep learning based method, as shown in table 11. The main reason is that we need to run the denoiser and reconstructor in each iteration.

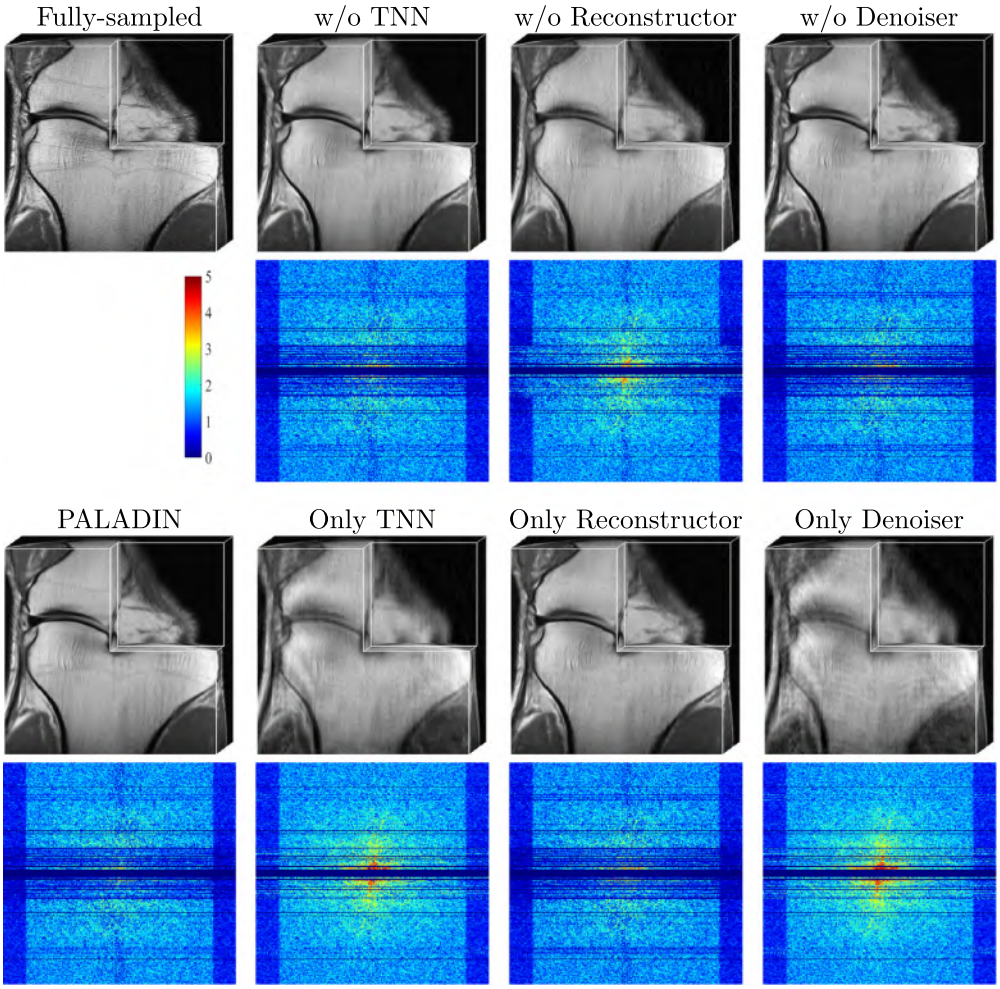

Figure 8. Visual results by different degenerated versions of our PALADIN on MR images, corresponding amplitude error of a middle slice between the results and fully-sampled data with the Cartesian 1D mask (the amplitude error was log-transformed after adding 1).

Table 8. The comparison results of our methods on IXI data set with the Cartesian 1D mask. ISTA-Net+ and DRUNet are trained on different types of dataset.

Acceleration rate		5×		10×		20×	
Training data set							
Reconstructor	Denoiser	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Natural images	Natural images	38.54	0.9832	32.34	0.9401	24.70	0.8162
Natural images	MR images	38.65	0.9822	32.30	0.9472	24.97	0.8062
MR images	Natural images	38.45	0.9822	32.34	0.9410	24.29	0.7943
MR images	MR images	38.36	0.9822	32.56	0.9452	24.75	0.8120

Table 9. Quantitative results on MR images of the IXI data set with the Gaussian 2D mask. The different low rank terms are utilized in our method. The **best** and the second best values are respectively highlighted by **bolder fonts** and underlines.

Acceleration rate	5×		10×		20×	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Observed	33.92	0.9563	30.55	0.9000	28.19	0.8372
TNN [67]	46.34	0.9962	40.20	0.9872	35.51	0.9677
NN [6]	46.27	0.9964	**40.55**	**0.9882**	35.54	0.9665
DCTNN [30]	46.39	0.9964	39.22	0.9870	34.70	0.9628
SNN [32]	**46.56**	**0.9964**	40.44	0.9878	**35.65**	**0.9680**

Table 10. Quantitative results on MR images of the fastMRI data set with the Cartesian 1D mask in 2D versus 3D reconstruction of our PALADIN.

Acceleration rate	5×		10×		20×	
Dimensionality	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
2D reconstruction	35.90	0.9269	32.60	0.8765	30.59	0.8278
3D reconstruction	35.97	0.9277	33.39	0.8847	30.94	0.8379

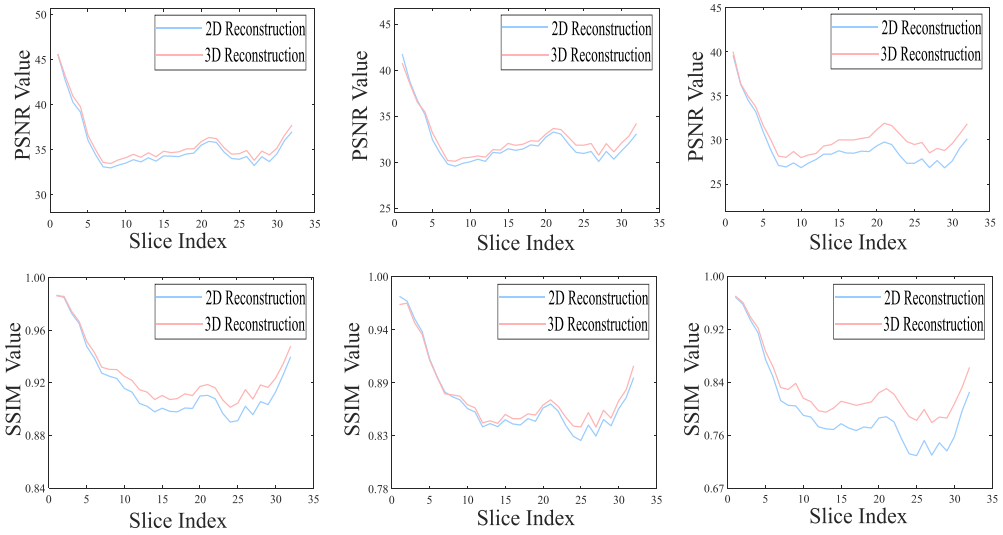

Figure 9. Comparison of PSNR and SSIM for different acceleration rates of Cartesian 1D mask in 2D versus 3D MRI reconstructions. From left to right: the acceleration rate of 5×, 10× and 20×.

Table 11. Averaged running time of different methods in seconds (s).

Method	DCCNN	HQSNet	ISTA-Net+	ISTA-Net+(3D)	LPDNet	SSLMRI
Time (s)	4.55	3.22	1.00	1.01	3.02	3.91
Method	TRPA	PALADIN				
Time (s)	837.50	545.23				

Data availability statement

The data that support the findings of this study are openly available at the following URLs: http://brain-development.org/ixi-dataset/ and https://fastmri.med.nyu.edu/.

Acknowledgments

The authors would like to thank the editor and reviewers for giving them many comments and suggestions, which are of great value for improving the quality of this manuscript. The authors wish to express their gratitude to the authors of [1, 21, 22, 43, 52, 59, 61] for providing their codes or data for this study. This work was supported in part by the Sichuan Science and Technology Program under Grants 2024ZYD0147, 2024NSFJQ0038, 2023ZYD0007, and 2024NSFSC0038, in part by the Natural Science Foundation of Xinjiang Uygur Autonomous Region under Grant 2024D01A18, in part by the National Natural Science Foundation of China (NSFC) under Grants 12371456 and 62376228 , in part by Chengdu Science and Technology Program under Grant 2023 JB00-00016-GX, and in part by the Guanghua Talent Project.

ORCID iDs

Jia-Mian Wu https://orcid.org/0009-0008-5789-2229
Shi-Bai Yin https://orcid.org/0009-0005-5811-649X
Tai-Xiang Jiang https://orcid.org/0000-0002-9099-4154
Gui-Song Liu https://orcid.org/0000-0003-2360-0466
Xi-Le Zhao https://orcid.org/0000-0002-6540-946X

References

[1] Adler J and Öktem O 2018 Learned primal-dual reconstruction *IEEE Trans. Med. Imaging* **37** 1322–32
[2] Ahmad R, Bouman C A, Buzzard G T, Chan S, Liu S, Reehorst E T and Schniter P 2020 Plug-and-play methods for magnetic resonance imaging: using denoisers for image recovery *IEEE Signal Process. Mag.* **37** 105–16
[3] Bauschke H H and Combettes P L 2013 *Convex Analysis and Monotone Operator Theory in Hilbert Spaces* (Springer)(https://doi.org/10.1007/978-1-4419-9467-7)
[4] Block K T, Uecker M and Frahm J 2007 Undersampled radial MRI with multiple coils. Iterative image reconstruction using a total variation constraint *Magn. Reson. Med.* **57** 1086–98
[5] Boyd S, Parikh N and Chu E 2011 *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers* (Now Publishers Inc)
[6] Candes E and Recht B 2012 Exact matrix completion via convex optimization *Commun. ACM* **55** 111–9
[7] Chan S H, Wang X and Elgendy O A 2017 Plug-and-play ADMM for image restoration: fixed-point convergence and applications *IEEE Trans. Comput. Imaging* **3** 84–98
[8] Chen C and Huang J 2012 Compressive sensing MRI with wavelet tree sparsity *Advances in Neural Information Processing Systems* vol 25
[9] Chlemper J S, Caballero J, Hajnal J, Price A N and Rueckert D 2017 A deep cascade of convolutional neural networks for dynamic MR image reconstructio *IEEE Trans. Med. Imaging* **37** 491–503
[10] Choi J K, Bao C and Zhang X 2018 PET-MRI joint reconstruction by joint sparsity based tight frame regularization *SIAM J. Imaging Sci.* **11** 1179–204
[11] Cohen R, Elad M and Milanfar P 2021 Regularization by denoising via fixed-point projection (red-pro) *SIAM J. Imaging Sci.* **14** 1374–406

[12] Combettes P L and Yamada I 2015 Compositions and convex combinations of averaged nonexpansive operators *J. Math. Anal. Appl.* **425** 55–70
[13] Dabov K, Foi A, Katkovnik V and Egiazarian K 2007 Image denoising by sparse 3-D transform-domain collaborative filtering *IEEE Trans. Image Process.* **16** 2080–95
[14] Donoho D L 2006 Compressed sensing *IEEE Trans. Inf. Theory* **52** 1289–306
[15] Duan J, Schlemper J, Qin C, Ouyang C, Bai W, Biffi C, Bello G, Statton B, O'regan D P and Rueckert D 2019 VS-Net: variable splitting network for accelerated parallel MRI reconstruction *Int. Conf. on Medical Image Computing and Computer Assisted Intervention* pp 713–22
[16] Eo T, Jun Y, Kim T, Jang J, Lee H-J and Hwang D 2018 KIKI-net: cross-domain convolutional neural networks for reconstructing undersampled magnetic resonance images *Magn. Reson. Med.* **80** 2188–201
[17] Fatania K, Pirkl C M, Menzel M I, Hall P and Golbabaee M 2022 A plug-and-play approach to multiparametric quantitative MRI: Image reconstruction using pre-trained deep denoisers *Int. Symp. on Biomedical Imaging (ISBI)* pp 1–4
[18] Giselsson P 2017 Tight global linear convergence rate bounds for Douglas-Rachford splitting *J. Fixed Point Theory Appl.* **19** 2241–70
[19] Gu H, Yaman B, Moeller S, Ellermann J, Ugurbil K and Akçakaya M 2022 Revisiting ℓ_1-wavelet compressed-sensing MRI in the era of deep learning *Proc. Natl Acad. Sci.* **119** e2201062119
[20] Guo P, Mei Y, Zhou J, Jiang S and Patel V M 2024 Reconformer: accelerated MRI reconstruction using recurrent transformer *IEEE Trans. Med. Imaging* **43** 582–93
[21] Hou R, Li F and Zhang G 2022 Truncated residual based plug-and-play ADMM algorithm for MRI reconstruction *IEEE Trans. Comput. Imaging* **8** 96–108
[22] Hu C, Li C, Wang H, Liu Q, Zheng H and Wang S 2021 Self-supervised learning for MRI reconstruction with a parallel network training framework *Int. Conf. on Medical Image Computing and Computer-Assisted Intervention* pp 382–91
[23] Jiang T-X, Ng M K, Pan J and Song G-J 2023 Nonnegative low rank tensor approximations with multidimensional image applications *Numer. Math.* **153** 141–70
[24] Jiang T-X, Ng M K, Zhao Xi-L and Huang T-Z 2020 Framelet representation of tensor nuclear norm for third-order tensor completion *IEEE Trans. Image Process.* **29** 7233–44
[25] Jiang T-X, Zhuang L, Huang T-Z, Zhao Xi-L and Bioucas-Dias J M 2021 Adaptive hyperspectral mixed noise removal *IEEE Trans. Geosci. Remote Sens.* **60** 1–13
[26] Kilmer M E, Horesh L, Avron H and Newman E 2021 Tensor-tensor algebra for optimal representation and compression of multiway data *Proc. Natl Acad. Sci.* **118** e2015851118
[27] Kilmer M E and Martin C D 2011 Factorization strategies for third-order tensors *Linear Algebr. Appl.* **435** 641–58
[28] Li J, Wang W and Ji H 2023 Self-supervised deep learning for image reconstruction: a langevin Monte Carlo approach *SIAM J. Imaging Sci.* **16** 2247–84
[29] Liu C, Li S, Hu D, Zhong Y, Wang J and Zhang P 2024 Hybrid plug-and-play ct image restoration using nonconvex low-rank group sparsity and deep denoiser priors *Phys. Med. Biol.* **69** 235004
[30] Liu J, Musialski P, Wonka P and Ye J 2012 Tensor completion for estimating missing values in visual data *IEEE Trans. Pattern Anal. Mach. Intell.* **35** 208–20
[31] Lu C, Feng J, Chen Y, Liu W, Lin Z and Yan S 2020 Tensor robust principal component analysis with a new tensor nuclear norm *IEEE Trans. Pattern Anal. Mach. Intell.* **42** 925–38
[32] Lu C, Peng Xi and Wei Y 2019 Low-rank tensor completion with a new tensor nuclear norm induced by invertible linear transforms *IEEEConf. on Computer Vision and Pattern Recognition* pp 5989–97
[33] Lustig M, Donoho D and Pauly J M 2007 Sparse MRI: the application of compressed sensing for rapid MR imaging *Mag. Reson. Med. Off. J. Int. Soc. Mag. Res. Med.* **58** 1182–95
[34] Lustig M, Donoho D L, Santos J M and Pauly J M 2008 Compressed sensing MRI *IEEE Signal Process. Mag.* **25** 72–82
[35] Ma S, Yin W, Zhang Y and Chakraborty A 2008 An efficient algorithm for compressed MR imaging using total variation and wavelets *IEEE Conf. on Computer Vision and Pattern Recognition* pp 1–8
[36] Ogura N 2002 Non-strictly convex minimization over the fixed point set of an asymptotically shrinking nonexpansive mapping *Numer. Funct. Anal. Optim.* **23** 113–37
[37] Pan T, Duan J, Wang J and Liu Y 2022 Iterative self-consistent parallel magnetic resonance imaging reconstruction based on nonlocal low-rank regularization *Magn. Reson. Imaging* **88** 62–75

[38] Qian X, Jiang T-X and Zhao Xi-L 2022 SelfS2: self-supervised transfer learning for sentinel-2 multispectral image super-resolution *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **16** 215–27
[39] Qu X, Hou Y, Lam F, Guo Di, Zhong J and Chen Z 2014 Magnetic resonance image reconstruction from undersampled measurements using a patch-based nonlocal operator *Med. Image Anal.* **18** 843–56
[40] Quan C, Zhou J, Zhu Y, Chen Y, Wang S, Liang D and Liu Q 2021 Homotopic gradients of generative density priors for MR image reconstruction *IEEE Trans. Med. Imaging* **40** 3265–78
[41] Roberts R A and Mullis C T 1987 *Digital Signal Processing* (Addison-Wesley Longman Publishing Co., Inc)
[42] Ryu E, Liu J, Wang S, Chen X, Wang Z and Yin W 2019 Plug-and-play methods provably converge with properly trained denoisers *Int. Conf. on Machine Learning* (PMLR) pp 5546–57
[43] Schlemper J, Caballero J, Hajnal J V, Price A and Rueckert D 2017 A deep cascade of convolutional neural networks for MR image reconstruction *Int. Conf. on Information Processing in Medical Imaging* pp 647–58
[44] Sun J *et al* 2016 Deep ADMM-Net for compressive sensing MRI *Advances in Neural Information Processing Systems* p 29
[45] Tan H Y, Mukherjee S, Tang J and Schönlieb C-B 2024 Provably convergent plug-and-play quasi-newton methods *SIAM J. Imaging Sci.* **17** 785–819
[46] Tewari S, Yousefi S and Webb A 2022 Deep neural-network based optimization for the design of a multi-element surface magnet for MRI applications *Inverse Problems* **38** 035003
[47] Venkatakrishnan S V, Bouman C A and Wohlberg B 2013 Plug-and-play priors for model based reconstruction *IEEE Conf. on Signal and Information Processing* pp 945–8
[48] Wang J-Li, Huang T-Z, Zhao X-L, Luo Y-S and Jiang T-X 2024 Conot: coupled nonlinear transform-based low-rank tensor representation for multidimensional image completion *IEEE Trans. Neural Netw. Learn. Syst.* **35** 8969–83
[49] Wang P, Sun T, Chen Y, Ge L, Wang X and Wang L 2024 Hyperspectral image denoising based on deep and total variation priors *Remote Sens.* **16** 2071
[50] Wang S, Xiao T, Liu Q and Zheng H 2021 Deep learning for fast MR imaging: a review for learning reconstruction from incomplete k-space data *Biomed. Signal Process. Control* **68** 102579
[51] Wang Z, Bovik A C, Sheikh H R and Simoncelli E P 2004 Image quality assessment: from error visibility to structural similarity *IEEE Trans. Image Process.* **13** 600–12
[52] Xin B, Phan T, Axel L and Metaxas D 2022 Learned half-quadratic splitting network for MR image reconstruction *Int. Conf. on Medical Imaging With Deep Learning* pp 1403–12
[53] Yang G *et al* 2018 DAGAN: Deep de-aliasing generative adversarial networks for fast compressed sensing MRI reconstruction *IEEE Trans. Med. Imaging* **37** 1310–21
[54] Yang J, Zhang Y and Yin W 2010 A fast alternating direction method for TVL1-L2 signal reconstruction from partial Fourier data *IEEE J. Sel. Top. Signal Process.* **4** 288–97
[55] Yang Y, Sun J, Li H and Xu Z 2020 ADMM-CSNet: a deep learning approach for image compressive sensing *IEEE Trans. Pattern Anal. Mach. Intell.* **42** 521–38
[56] Yang Y, Wang Y, Wang J, Sun J and Xu Z 2023 An unrolled implicit regularization network for joint image and sensitivity estimation in parallel MR imaging with convergence guarantee *SIAM J. Imaging Sci.* **16** 1791–824
[57] Yao D, Zhuang L, Gao L, Zhang B and Bioucas-Dias J M 2017 Hyperspectral image inpainting based on low-rank representation: a case study on Tiangong-1 data *Int. Geoscience and Remote Sensing Symp.* pp 3409–12
[58] Yao J, Xu Z, Huang X and Huang J 2015 Accelerated dynamic MRI reconstruction with total variation and nuclear norm regularization *Int. Conf. on Medical Image Computing and Computer-Assisted Intervention* pp 635–42
[59] Zbontar J *et al* 2018 fastMRI: an open dataset and benchmarks for accelerated MRI (arXiv:1811.08839)
[60] Zeng G, Guo Y, Zhan J, Wang Z, Lai Z, Du X, Qu X and Guo Di 2021 A review on deep learning MRI reconstruction without fully sampled *k*-space *BMC Med. Imaging* **21** 195
[61] Zhang J and Ghanem B 2018 ISTA-Net: interpretable optimization-inspired deep network for image compressive sensing *IEEE Conf. on Computer Vision and Pattern Recognition* pp 1828–37
[62] Zhang K, Li Y, Zuo W, Zhang L, Van Gool L and Timofte R 2022 Plug-and-play image restoration with deep denoiser prior *IEEE Trans. Pattern Anal. Mach. Intell.* **44** 6360–76

[63] Zhang K, Zuo W, Chen Y, Meng D and Zhang L 2017 Beyond a gaussian denoiser: residual learning of deep CNN for image denoising *IEEE Trans. Image Process.* **26** 3142–55
[64] Zhang K, Zuo W and Zhang L 2018 FFDNet: toward a fast and flexible solution for CNN-based image denoising *IEEE Trans. Image Process.* **27** 4608–22
[65] Zhang K, Zuo W and Zhang L 2019 Deep plug-and-play super-resolution for arbitrary blur kernels *IEEE Conf. on Computer Vision and Pattern Recognition* pp 1671–81
[66] Zhang Y and Hu Y 2022 Dynamic cardiac MRI reconstruction using combined tensor nuclear norm and casorati matrix nuclear norm regularizations *Int. Symp. on Biomedical Imaging* pp 1–4
[67] Zhang Z and Aeron S 2016 Exact tensor completion using t-SVD *IEEE Trans. Signal Process.* **65** 1511–26
[68] Zhao Xi-L, Yang J-H, Ma T-H, Jiang T-X, Ng M K and Huang T-Z 2021 Tensor completion via complementary global, local and nonlocal priors *IEEE Trans. Image Process.* **31** 984–99
[69] Zhuang L, Ng M K, Gao L and Wang Z 2024 Eigen-CNN: eigenimages plus eigennoise level maps guided network for hyperspectral image denoising *IEEE Trans. Geosci. Remote Sens.* pp 1–1