IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 25, 2018, accepted August 22, 2018, date of publication August 28, 2018, date of current version September 21, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2867435

Co-Robust-ADMM-Net: Joint ADMM Framework and DNN for Robust Sparse Composite Regularization

YUNYI LI[1], (Student Member, IEEE), XIEFENG CHENG[1], AND GUAN GUI[2], (Senior Member, IEEE)

[1]College of Electronic and Optical Engineering and College of Microelectronics, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
[2]College of Telecommunication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Corresponding author: Guan Gui (guiguan@njupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61671253, in part by the Jiangsu Specially Appointed Professor under Grant RK002STP16001, in part by the Innovation and Entrepreneurship of Jiangsu High-level Talent under Grant CZ0010617002, in part by NUPTSF under Grant NY2015026, and in part by the 1311 Talent Plan of Nanjing University of Posts and Telecommunications.

ABSTRACT Symmetric α-stable (SαS) noise is a typical form of impulsive noise often generated in signal measurement and transmission systems. The problem of reconstructing an image from a small number of under-sampled data corrupted by impulsive noise is called robust compressive sensing (CS). In this paper, to effectively suppress the outliers and accurately reconstruct the image from compressive measured data in the presence of SαS noise, a novel composite robust alternating direction method of multiplier network-based CS algorithm is proposed. Specifically, we first employ the L_1-norm as the estimator to depress the influence of SαS noise, and then the ADMM framework is employed to address the resulting optimization problem. Moreover, a smoothing strategy is adopted to address the L_1-norm based non-smooth optimization problem. To exploit more prior knowledge and image features, a robust composite regularization model is proposed for training by the deep neural network (DNN). In the training phase, the DNN can be utilized to train the samples for the optimal parameters, the optimal shrinkage function and the optimal transform domain, which can be reserved as the network. In the reconstruction process, the obtained network can be employed for improving the reconstruction performance. Experiments show that our proposed algorithm can obtain higher reconstruction Peak signal-to-noise ratio than the existing state-of-the-art robust CS methods.

INDEX TERMS Symmetric α-stable noise, compressive sensing, composite regularization model, alternating direction method of multipliers, deep neural network.

I. INTRODUCTION

Compressive sensing (CS) [1] is an emerging promising approach that aims for accurate acquisition and reconstruction of the sparse signal from a small amount of sub-Nyquist sampling data. Typical applications include magnetic resonance imaging (MRI) [2], [3], radar imaging [4], hyperspectral imaging [5], and sparse cluster for high-dimensional data [6], [7]. The basic linear observation system can be formulated as follows:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n} \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^{M\times 1}$ denotes the observation data, $\mathbf{A} \in \mathbb{R}^{M\times N}$, $(M \ll N)$ represents the linear operator or random sampling matrix, $\mathbf{x} \in \mathbb{R}^{N\times 1}$ is the desired signal vector that is sparse in some transform domain, and $\mathbf{n} \in \mathbb{R}^{M\times 1}$ is often considered Gaussian with the bounded norm $\|\mathbf{n}\|_2 \leqslant \xi$. The CS theory states that, if the desired unknown signal is inherently, then the ill-posed problem of recovering $\mathbf{x}$ from $\mathbf{y}$ can be accurately addressed by

$$\hat{\mathbf{x}} \leftarrow \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda g(\mathbf{x}) \tag{2}$$

where the L_2-norm term $\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2$ is the observation fidelity term that ensures the concurrence of $\mathbf{y}$ and $\mathbf{x}$, and λ is the regularization parameter. The penalty function $g(\mathbf{x})$ usually provides prior knowledge for the optimization problem via a norm function

$$g(\mathbf{x}) = \|\mathbf{D}\mathbf{x}\|_p^p = \sum_{i=1}^{n} \left|(\mathbf{D}\mathbf{x})_i\right|^p, \quad 0 \leqslant p \leqslant 1 \tag{3}$$

Here, $\mathbf{D}$ denotes the sparsifying transform operator. When $p = 0$, then $g(\mathbf{x}) = \|\mathbf{Dx}\|_0^0$ counts the number of nonzero element of $\mathbf{Dx}$, orthogonal matching pursuit (OMP) [8]–[10] based algorithms are mainly effective method to solve the resulting optimization problem. When $p = 1$, the most successful algorithm is the iterative thresholding algorithm [2].

Symmetric α-stable (SαS) noise is a typical impulsive noise often generated in signal/information measurement and transmission systems that will cause the traditional CS reconstruction algorithms in (2) to degrade severely. To suppress the outliers caused by SαS noise, one popular effective optimization model employs the L_1-norm as the metric for the residual error by

$$\hat{\mathbf{x}} \leftarrow \arg\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{y}\|_1 + \lambda g(\mathbf{x}) \tag{4}$$

The CS formulation in (4), known as robust CS, compared with the quadratic estimator of L_2-norm in (2), is more suited for using the L_1-norm to model large outliers; hence, it has been widely used in designing robust CS algorithms. However, optimization of the objective function is intractable because of the resulting nonsmooth cost function term of the L_1-norm. Among all the existing effective robust recovery algorithms, the alternating direction method of multipliers (ADMM) framework is regarded as one of the more effective and efficient approaches [11]–[14]. Using an operator splitting [14], this framework involves separation of the regularization term by an additional variable.

Although the ADMM framework is generally efficient, it is intractable to tune the relevant parameters, e.g., penalty parameter and regularization parameter. To reduce the influence of these parameters, some effective schemes have proposed, with the regularization path being a popular approach to find an optimal regularization parameter λ [13]. To find the optimal penalty parameter, a choice scheme is to gradually increase its value from a small starting value by iteration until reaching the target value [11], [15], [16]. However, these approaches for searching the optimal value are inefficient, especially for large-scale optimization problem. Moreover, it is also challenging in CS to choose an optimal sparsifying transform domain and the corresponding regularization function $g(\mathbf{x})$. Recently, a deep learning approach based ADMM-net was proposed to train these optimal parameter and was demonstrated to achieve significant improvement for MRI [17]; however, it was considered under a Gaussian-assumption noise environment.

In this paper, we propose a composite robust alternating direction method of multiplier net (Co-Robust-ADMM-Net) to address the reconstruction problem in the presence of SαS noise. We first employ the L_1-norm as an estimator to depress the influence of SαS noise, and then we employ the ADMM framework to address the resulting optimization problem. Furthermore, a smoothing strategy for the loss-function is adopted to solve the nonsmooth optimization problem. The proposed robust-ADMM-net algorithm uses the deep neural network method to train the corrupted image data for optimal parameters. In addition, the ADMM-net can also be utilized for training the shrinkage function and the sparsifying transform dictionary $\mathbf{D}$. Experimental results demonstrate that the proposed robust CS reconstruction approach can outperform state-of-the-art CS robust methods both in reconstruction speed and accuracy.

The main contribution of this work can be summarized as follows. First, this paper proposes a new optimization model that combines the L_1-norm estimator and a composite regularization scheme. The proposed model can exploit more physical mechanism and prior knowledge for training. Then, to solve the nonsmooth optimization problem, a smoothing strategy is adopted to smooth the nonsmooth L_1-norm estimator. Furthermore, the back propagation (BP) approach is proposed to train the penalty parameter, the regularization parameter and the transform domain.

II. THE PROPOSED CO-ROBUST-ADMM FRAMEWORK

A. ROBUST COMPOSITE REGULARIZATION MODEL FOR DEEP LEARNING

Currently, deep learning is regarded as the representative advancement of artificial intelligence; deep learning approaches are capable of extracting features from images for recognition and restoration. However, deep learning often requires a huge number of samples for training [18]. The main challenge for traditional CS reconstruction algorithms to train the optimal parameters through deep learning is the lack of data. To effectively address this issue, we adopt a robust optimization model to further exploit more prior knowledge for training. It is a fact that a given signal $\mathbf{x}$ can be represented by different dictionaries $\mathbf{D}_l$, $l = 1, 2, \cdots, L$, but with different sparsity; this fact can be utilized to exploit more prior knowledge for CS reconstruction problem. Related algorithms based on the composite regularization have been proposed to improve the reconstruction accuracy, such as Co-L_1 [19] and MUSAI-$L_{1/2}$ [20]. We design the following robust optimization model by

$$\hat{\mathbf{x}} \leftarrow \arg\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{y}\|_1 + \sum_{l=1}^{L} \lambda_l g(\mathbf{D}_l \mathbf{x}) \tag{5}$$

where $g(\cdot)$ denotes the regularization function, such as L_p-norm ($0 < p \leq 1$), $\mathbf{D}_l$ denotes the sparsifying transform dictionary (e.g., sines, wavelet bases), with examples of a typical sparsifying transform including the Discrete Cosine Transform (DCT) and the Discrete Wavelet Transform (DWT); and λ_l denotes the regularization parameter. Unlike the optimization problem (4), the advantage of this model can obtain more physical mechanism and prior knowledge for optimization [20].

B. ADMM FRAMEWORK FOR ROBUST COMPOSITE REGULARIZATION

ADMM is a simple and powerful framework for the high-dimension optimization problem in machine learning and signal processing that adopts a variable-splitting strategy to separate coupled components via auxiliary

variables [21]–[25]. To solve the above optimization problem (5) based on the ADMM framework, we typically use an auxiliary variable $\mathbf{z}_l \in {}^{N\times 1}$, then the optimization problem (5) can be rewritten as

$$\begin{aligned} &\arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_1 + \sum_{l=1}^{L}\lambda_l g(\mathbf{z}_l) \\ &s.t.\ \mathbf{z}_l = \mathbf{D}_l\mathbf{x}, \quad \forall l \in [1,2,\cdots,L] \end{aligned} \tag{6}$$

The augmented Lagrangian of problem (6) is

$$\begin{aligned} L(\mathbf{x},\mathbf{z},\alpha) = \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_1 + \sum_{l=1}^{L}\lambda_l g(\mathbf{z}_l) - \sum_{l=1}^{L}\langle \alpha_l, \mathbf{z}_l - \mathbf{D}_l\mathbf{x}\rangle \\ + \sum_{l=1}^{L}\frac{\rho_l}{2}\|\mathbf{z}_l - \mathbf{D}_l\mathbf{x}\|_2^2 \end{aligned} \tag{7}$$

where $\{\alpha_l\} \in {}^{N\times 1}$ are Lagrangian multipliers, $\{\rho_l\} > 0$ are penalty parameters, and $\{\lambda_l\}$ denotes the regularization parameter; these parameters will play critical roles in the optimization process. According to the ADMM framework, we have the following three steps:

$$\begin{aligned} &\mathbf{x}^{(n+1)} \\ &= \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_1 \\ &\quad - \sum_{l=1}^{L}\left\langle \alpha_l^{(n)}, \mathbf{z}_l^{(n)} - \mathbf{D}_l\mathbf{x}\right\rangle + \sum_{l=1}^{L}\frac{\rho_l}{2}\left\|\mathbf{z}_l^{(n)} - \mathbf{D}_l\mathbf{x}\right\|_2^2 \\ &= \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_1 + \sum_{l=1}^{L}\frac{\rho_l}{2}\left\|\mathbf{z}_l^{(n)} - \mathbf{D}_l\mathbf{x} - \frac{\alpha_l}{\rho_l}\right\|_2^2 \end{aligned} \tag{8}$$

$$\begin{aligned} &\mathbf{z}^{(n+1)} \\ &= \arg\min_{\mathbf{x}} \sum_{l=1}^{L}\lambda_l g(\mathbf{z}_l) \\ &\quad - \sum_{l=1}^{L}\left\langle \alpha_l^{(n)}, \mathbf{z}_l^{(n)} - \mathbf{D}_l\mathbf{x}^{(n+1)}\right\rangle + \sum_{l=1}^{L}\frac{\rho_l}{2}\left\|\mathbf{z}_l - \mathbf{D}_l\mathbf{x}^{(n+1)}\right\|_2^2 \\ &= \arg\min_{\mathbf{x}} \sum_{l=1}^{L}\left[\lambda_l g(\mathbf{z}_l) + \frac{\rho_l}{2}\left\|\mathbf{z}_l - \mathbf{D}_l\mathbf{x}^{(n+1)} - \frac{\alpha_l}{\rho_l}\right\|_2^2\right] \end{aligned} \tag{9}$$

$$\begin{aligned} &\alpha^{(n+1)} \\ &= \arg\min_{\mathbf{x}} \sum_{l=1}^{L}\left\langle \alpha_l, \mathbf{D}_l\mathbf{x}^{(n+1)} - \mathbf{z}_l^{(n+1)}\right\rangle \end{aligned} \tag{10}$$

The $\mathbf{x}$-step in (7) in fact is a penalized least square (LS) problem. To solve this optimization problem, we first smooth the L_1-norm term $\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_1$, specifically, we linearize the term $\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_1$ at the given $\tilde{\mathbf{x}}$ as

$$\begin{aligned} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_1 &= \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_{1,\varepsilon} \\ &= \frac{1}{2}\left\|\mathbf{A}\tilde{\mathbf{x}}-\mathbf{y}\right\|_{1,\varepsilon} + \frac{1}{2}\left\langle \mathbf{A}\mathbf{x}-\mathbf{A}\tilde{\mathbf{x}}, d(\mathbf{A}\tilde{\mathbf{x}}-\mathbf{y})\right\rangle \\ &\quad + \frac{1}{2\tau}\left\|\mathbf{A}\mathbf{x}-\mathbf{A}\tilde{\mathbf{x}}\right\|_2^2 \end{aligned} \tag{11}$$

where $\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_{1,\varepsilon} = \sum_i [(\mathbf{A}\mathbf{x}-\mathbf{y})_i^2 + \varepsilon^2]^{\frac{1}{2}}$, $\varepsilon = 10^{-3}$, $\tau > 0$ is a proximal parameter. Thus, we have

$$\begin{aligned} \mathbf{x}^{(n+1)} = \arg\min_{\mathbf{x}} &\frac{1}{2}\left\|\mathbf{A}\tilde{\mathbf{x}}-\mathbf{y}\right\|_{1,\varepsilon} + \frac{1}{2}\left\langle \mathbf{A}\mathbf{x}, d(\mathbf{A}\tilde{\mathbf{x}}-\mathbf{y})\right\rangle \\ &+ \frac{1}{2\tau}\left\|\mathbf{A}\mathbf{x}-\mathbf{A}\tilde{\mathbf{x}}\right\|_2^2 - \sum_{l=1}^{L}\frac{\rho_l}{2}\left\|\mathbf{z}_l^{(n)} - \mathbf{D}_l\mathbf{x} - \frac{\alpha_l}{\rho_l}\right\|_2^2 \end{aligned} \tag{12}$$

We can obtain the closed-form solution by derivation

$$\begin{aligned} \mathbf{x}^{(n+1)} = &\left[\frac{1}{\tau}\mathbf{A}^T\mathbf{A} + \sum_{l=1}^{L}\rho_l D_l^T D_l\right]^{-1}\left[\frac{1}{\tau}\mathbf{A}^T\mathbf{A}\tilde{\mathbf{x}}\right. \\ &\left. - \frac{1}{2}\mathbf{A}^T d(\mathbf{A}\tilde{\mathbf{x}}-\mathbf{y}) + \sum_{l=1}^{L}\rho_l D_l^T(z_l^{(n)} - \beta_l^{(n)})\right] \end{aligned} \tag{13}$$

where $d(\mathbf{A}\tilde{\mathbf{x}}-\mathbf{y})_i = ((\mathbf{A}\tilde{\mathbf{x}}-\mathbf{y})_i^2 + \varepsilon^2)^{-1/2}$. Specifically, in this paper, we set , where $\mathbf{P} \in \mathbb{R}^{N\times N_1}$ denotes the undersampling matrix, and $\mathbf{F} \in \mathbb{R}^{N_1\times M}$ is a Fourier transform; thus, we have the solution of the $\mathbf{x}$-step,

$$\begin{aligned} \mathbf{x}^{(n+1)} = \mathbf{F}^T&\left[\frac{1}{\tau}\mathbf{P}^T\mathbf{P} + \sum_{l=1}^{L}\rho_l\mathbf{F}\mathbf{D}_l^T\mathbf{D}_l\mathbf{F}^T\right]^{-1}\left[\frac{1}{\tau}\mathbf{P}^T\mathbf{P}\mathbf{F}\tilde{\mathbf{x}}\right. \\ &\left. - \frac{1}{2}\mathbf{P}^T d(\mathbf{P}\mathbf{F}\tilde{\mathbf{x}}-\mathbf{y}) + \sum_{l=1}^{L}\rho_l\mathbf{F}\mathbf{D}_l^T(\mathbf{z}_l^{(n)} - \beta_l^{(n)})\right] \end{aligned} \tag{14}$$

where $d(\mathbf{P}\mathbf{F}\tilde{\mathbf{x}}-\mathbf{y})_i = ((\mathbf{P}\mathbf{F}\tilde{\mathbf{x}}-\mathbf{y})_i^2 + \varepsilon^2)^{-1/2}$, and let $\beta_l = \frac{1}{\rho_l}\cdot\alpha_l$.

To simplify the above solution, we specially set $\tilde{\mathbf{x}} = 0$ to simplify our algorithm, then in the $n+1$ -th iteration, we have

$$\begin{aligned} \mathbf{x}^{(n+1)} = \mathbf{F}^T&\left[\frac{1}{\tau_1}\mathbf{P}^T\mathbf{P} + \sum_{l=1}^{L}\rho_l\mathbf{F}\mathbf{D}_l^T\mathbf{D}_l\mathbf{F}^T\right]^{-1} \\ &\times\left[\frac{1}{2}d(\mathbf{P}^T\mathbf{y}) + \sum_{l=1}^{L}\rho_l\mathbf{F}\mathbf{D}_l^T(\mathbf{z}_l^{(n)} - \beta_l^{(n)})\right] \end{aligned} \tag{15}$$

In the first iteration for initialization, we set $\mathbf{z}_l^{(0)} = 0$, $\beta_l^{(0)} = 0$ then we have

$$\mathbf{x}^{(1)} = \mathbf{F}^T\left[\frac{1}{\tau_1}\mathbf{F}^T\mathbf{P}^T\mathbf{P}\mathbf{F} + \sum_{l=1}^{L}\rho_l\mathbf{F}H_l^T H_l\mathbf{F}^T\right]^{-1}\left[\frac{1}{2}d(\mathbf{P}^T\mathbf{y})\right] \tag{16}$$

where $d(\mathbf{p}^T\mathbf{y})_i = ((\mathbf{p}^T\mathbf{y})_i^2 + \varepsilon^2)^{-1/2}$.

The solution of the $\mathbf{z}$-step (8) can be obtained by the shrinkage function determined by $g(\cdot)$, e.g., the L_1-norm function corresponding with the soft-shrinkage operator [26]. Then, the output of $\mathbf{z}$-step can be described as

$$\mathbf{z}_l^{(n+1)} = S(\mathbf{D}_l\mathbf{x}^{(n)} + \beta_l^{(n)}; {}^{\lambda_l}/_{\rho_l}) \tag{17}$$

where $S(\cdot)$ represents the corresponding nonlinear shrinkage function. For initialization, in the first iteration, we set the

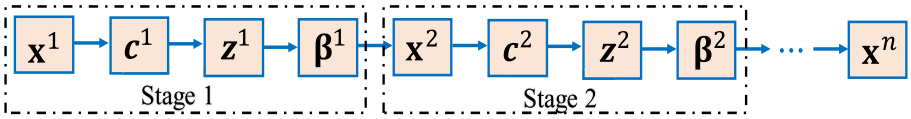

FIGURE 1. The ADMM network structure.

initial $S(\cdot)$ as the soft-shrinkage operator and set $\mathbf{D}_l$ as the DCT basis. For simplicity, let $\beta_l = {}^{\alpha_l}/_{\rho_l}$, then the α-step is converted into the β-step, and the subproblem have the following solution:

$$\beta_l^{(n+1)} = \beta_l^{(n)} + \eta_l \cdot (\mathbf{D}_l\mathbf{x}^{(n+1)} - \mathbf{z}_l^{(n+1)}) \tag{18}$$

where the parameter η_l denotes the update rate.

The Co-Robust-ADMM algorithm can be described as **Algorithm 1** .

Algorithm 1 Co-Robust-ADMM Algorithm

Problem: $\hat{\mathbf{x}} \leftarrow \arg\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_1 + \sum_{l=1}^{L} \lambda_l g(\mathbf{D}_l\mathbf{x})$

1 Input: Measured data $\mathbf{y}$, measurement matrix $\mathbf{A}$, the regularizer number L, the parameter λ_l, η_l and ρ_l;

2 Initialization: $g(\mathbf{D}_l\mathbf{x}) = \|\mathbf{D}_l\mathbf{x}\|_1$, $\mathbf{z}_l^{(0)} = 0$, $\beta_l^{(0)} = 0$.

3 for $t = 1, 2, \cdots, n$

Compute $\mathbf{x}^{(n)}$ using equation (15)

Compute $\mathbf{z}^{(n)}$ using soft-shrinkage operator in equation (17)

Compute $\beta^{(n)}$ using equation (18)

4 end.

5 Output.

III. PROPOSED CO-ROBUST-ADMM-NET ALGORITHM

A. DEEP NEURAL NETWORK FOR ADMM FRAMEWORK

To connect the ADMM framework and the deep neural network, inspired by the recently work of ADMM-net [17], we first map the iterations in ADMM to the layers of the deep neural network. If every iteration in ADMM is considered as one layer of the deep neural network, in this case, the above three steps can be regarded as three layers: Reconstruction layer $\mathbf{x}^{(n+1)}$ in (15); Nonlinear transform layer $\mathbf{z}_l^{(n+1)}$ in (17); and the Multiplier update layer $\beta_l^{(n+1)}$ in (18). As mentioned before, finding an optimal transform domain is an active research area because a sparser representation often leads to higher reconstruction accuracy. Some popular sparsifying transforms, such as DCT, Fourier and Haar, are often not optimal. In this paper, we also employ a conventional layer $\mathbf{c}_l^{(n+1)}$ to obtain the optimal sparsifying transform domain thus, we have the following four layers:

(1) Reconstruction layer $\mathbf{x}^{(n+1)}$:

$$x^{(n+1)} = \mathbf{F}^T\left[\frac{1}{\tau_1}\mathbf{P}^T\mathbf{P} + \sum_{l=1}^{L}\rho_l\mathbf{F}\mathbf{D}_l^T\mathbf{D}_l\mathbf{F}^T\right]^{-1} \times \left[\frac{1}{2}d(\mathbf{P}^T\mathbf{y}) + \sum_{l=1}^{L}\rho_l\mathbf{F}\mathbf{D}_l^T(\mathbf{z}_l - \beta_l)\right] \tag{19}$$

(2) Convolution layer $c_l^{(n+1)}$:

$$\mathbf{c}_l^{(n+1)} = \mathbf{D}_l^{(n+1)}\mathbf{x}^{(n+1)} \tag{20}$$

(3) Nonlinear transform layer $z_l^{(n+1)}$:

$$\mathbf{z}_l^{(n+1)} = S(\mathbf{D}_l\mathbf{x}^{(n)} + \beta_l^{(n)}; \frac{\lambda_l}{\rho_l}) \tag{21}$$

(4) Multiplier update layer $\beta_l^{(n+1)}$:

$$\beta_l^{(n+1)} = \beta_l^{(n)} + \eta_l \cdot (\mathbf{D}_l\mathbf{x}^{(n+1)} - \mathbf{z}_l^{(n+1)}) \tag{22}$$

B. UPDATING BY BACKPROPAGATION OVER ROBUST-ADMM-NETWORK

In this paper, we choose the popular normalized mean square error (NMSE) as the loss-function for training. The loss-function between the output and the ground-truth is described as

$$Loss = \frac{1}{\Gamma}\sum_{i=1}\frac{\|\hat{\mathbf{x}} - \mathbf{x}\|_2}{\|\mathbf{x}\|_2} \tag{23}$$

where $\hat{\mathbf{x}}$ is the network output (or reconstruction result), $\mathbf{x}$ is the ground-truth and the set Γ represents the number of pairs of under-sampling data and ground-truth images. To obtain the optimal parameters, transform domain and shrinkage function, we employ the backpropagation [17] strategy to compute the gradient w.r.t. the parameters. The procedure of $\mathbf{x}^{(n)} \rightarrow \mathbf{c}^{(n)} \rightarrow \mathbf{z}^{(n)} \rightarrow \beta^{(n)} \rightarrow \mathbf{x}^{(n+1)}$ is the forward pass. In the forward pass, we calculate the NMSE for using the updated parameters, the learned shrinkage function, and the learned transform domain. The procedure of $\mathbf{x}^{(n)} \leftarrow \mathbf{c}^{(n)} \leftarrow \mathbf{z}^{(n)} \leftarrow \beta^{(n)}$ is the backward pass. In the backward pass, we calculate the gradient of NMSE w.r.t. each parameter in every layer, see [17] for more details. Figure 3 details the training and reconstruction process, in the training process, the reconstructed image can be first achieved through 'forward pass'. Then, we caculate the reconstructed loss by (23). In the backward pass, each layer can be effectively updated by the backpropagation strategy. Aftering the two pass of forward and backword, we can obtain the optimal parameters, the optimal transform domian and the optimal sparse regularizers, they are dubbed network parameters; In the reconstruction, the k-space sampling rata can be reconstructed by the 'Stage 1', and then the reconstructed image can be improved stage by stage using the trained network. The proposed Co-Robust-ADMM-net algorithm is described in **Algorithm 2**.

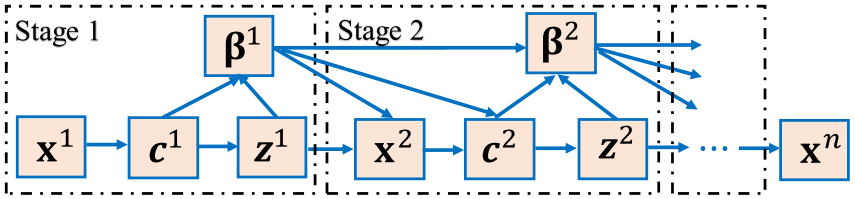

FIGURE 2. An improved robust ADMM network structure.

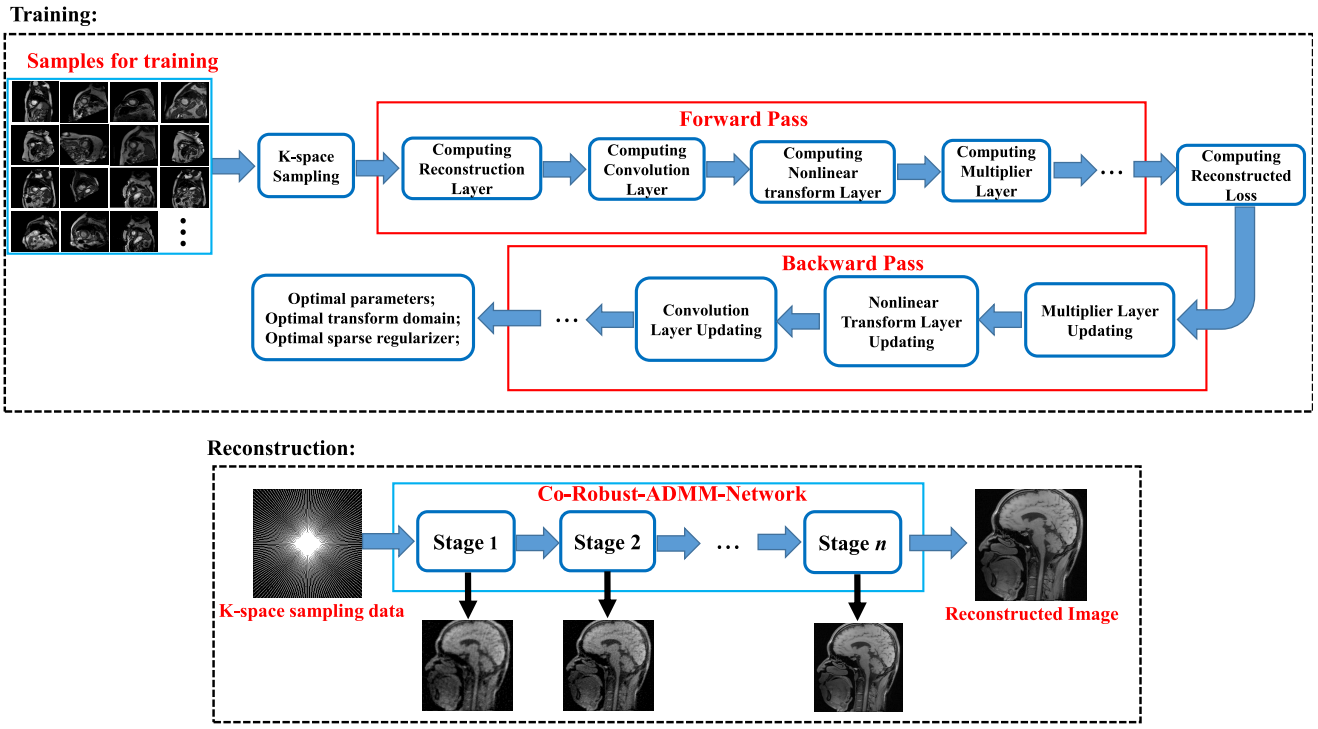

FIGURE 3. Illustrations for training and reconstruction.

IV. EXPERIMENTS

In this section, we will conduct extensive experimental evaluations on a typical MRI problem. Furthermore, some comparison experiments will be conducted to demonstrate the efficiency and priority than other corresponding work. Considering the fact that the noise level is unknown in real-world application, in this paper, we tend to train the network using a relative strong impulsive noise, and then conduct our reconstruction experiment using different levels of noise. All experiments are performed for the real-world magnetic resonance (MR) images (CAF Project: https://masi.vuse.vanderbilt.edu/workshop2013/index.php/Segmentation-Challenge-Details), and are performed on a computer with Intel core i7-6700 CPU.

A. TRAINING

Considering the fact that the desired MR images are unknown in real-world applications, in this paper, we first randomly choose a related real-world MR images for training to obtain the optimal network parameters and then reconstruct the other desired MR images using the learned parameters. In this paper, we typically consider SαS noise corrupted measured data. The characteristic function of a zero-location SαS distribution can be described as

$$\varphi(\omega) = e^{j\theta\omega - \gamma^\alpha |\omega|^\alpha} \tag{24}$$

where $0 < \alpha < 2$ denotes the characteristic exponent; a smaller value of α will often cause a heavier tail of the distribution and more impulsive noise, and the value of α is empirically greater than 1 in applications. The parameter $\gamma > 0$ represents the strength of the SαS noise; the value of γ is empirically set as $10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}$ in applications.

1) LOSS FUNCTION VERSUS ITERATION NUMBER IN THE TRAINING PROCESS

To demonstrate the effects of our proposed network, we first investigate the averaged loss calculated using (23) versus the iteration number in the training process. In this experiment, we empirically adopt a five-stage proposed

Algorithm 2 The Proposed Co-Robust-ADMM-Net Algorithm

Problem: $\hat{\mathbf{x}} \leftarrow \arg\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x}-\mathbf{y}\|_1 + \sum_{l=1}^{L} \lambda_l g(\mathbf{D}_l\mathbf{x})$

1 Input: Measured data $\mathbf{y}$, measurement matrix $\mathbf{A}$, the regularizer number L.

2 Initialization: $g(\mathbf{D}_l\mathbf{x}) = \|\mathbf{D}_l\mathbf{x}\|_1$, $\mathbf{z}_l^{(0)} = 0$, $\beta_l^{(0)} = 0$, Stage number n, the parameter λ_l, η_l and ρ_l;

3 for $t = 1, 2, \cdots, n$

Compute the first stage using the initial setting parameters

Compute $\mathbf{x}^{(1)}$ using equation (16)

Compute $\mathbf{z}^{(1)}$ using equation (20)

Compute $\beta^{(1)}$ using equation (21)

Compute $\mathbf{c}_l^{(1)}$ using equation (22)

Compute the second stage using the learned parameters;

Compute $\mathbf{x}^{(2)}$ using equation (19)

Compute $\mathbf{z}^{(2)}$ using equation (20)

Compute $\beta^{(2)}$ using equation (21)

Compute $\mathbf{c}_l^{(2)}$ using equation (22)

$\vdots$

Compute the $n-$ th stage using the learned parameters;

4 end.

5 Output $\mathbf{x}^{(n+1)}$.

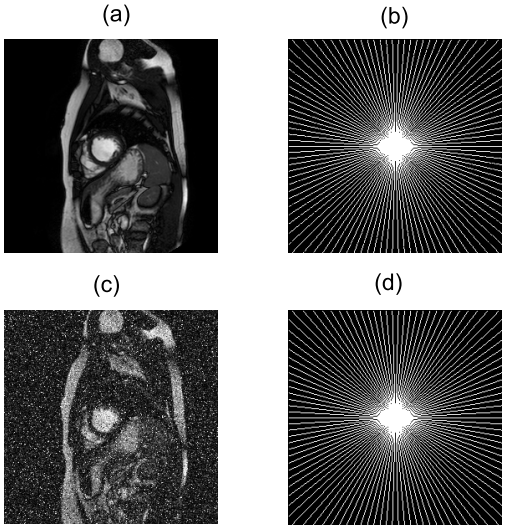

FIGURE 4. (a): A clean Chest magnetic resonance image for training; (b): The clean k-space data; (c): A noisy Chest magnetic resonance image. (d): The noisy k-space data.

Co-Robust-ADMM-net (see figure 2) and employ only a 'Chest' MR image (see figure 4) corrupted by seven levels of SαS noise with 0.3 sampling rate for training, and then evaluate the changes of loss values. Figure 5 describes the averaged loss (averaged NMSE) curve of the proposed Co-Robust-ADMM-net. For every loss curve, the starting

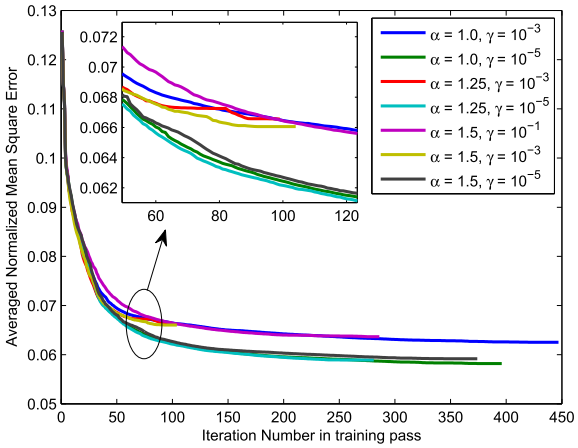

FIGURE 5. Loss function versus iteration number in the training process.

point value denotes the initial result before training, dubbed Initial-Co-Robust-ADMM. The results explicitly show that the reconstruction loss decreases gradually versus the iteration with the training process and then tends to a very small stable value, and these results show that the trained averaged NMSE values are much significantly lower than the initial NMSE values. From the results, we also observe that strong noise need more iterations to achieve the optimal parameters, such as $\alpha = 1.0$, $\gamma = 10^{-3}$, while the image corrupted by weak noise need fewer iterations, like $\alpha = 1.5$, $\gamma = 10^{-3}$. Importantly, unlike common deep learning method, our proposed network can update the corresponding parameters by the backpropagation method only use one figure.

2) RECONSTRUCTION PERFORMANCE VERSUS THE NUMBER OF NETWORK STAGES

In the second experiment, we evaluate the performance for different numbers of network stages. The number of stages represents the network depth (see figure 2), in general, a deeper network can often obtain higher reconstruction accuracy but also more computational overhead costs. To make a tradeoff between the reconstruction accuracy and speed, in the following, we will investigate the impact of the stage number on the reconstruction results. In this experiment, we empirically choose 5 MR 'Chest' images with a strong impulsive noise ($\alpha = 1.0, \gamma = 10^{-4}$) for training, and then reconstruct the same training 'Chest' MR image (see figure 4) using our trained network parameters by our proposed Co-Robust-ADMM-net. Figure 6 presents the reconstruction results versus the number of stages. From the results, we can observe that the reconstruction PSNR tends to a higher value with the stage number increase, but then decreases; as shown in the left of figure 6, the reconstruction algorithm can obtain the highest PSNR value at 5, 6 and 7. The right figure describes the CPU time versus the stage number; all results show that the computational overhead increases linearly with the number of stages. To achieve

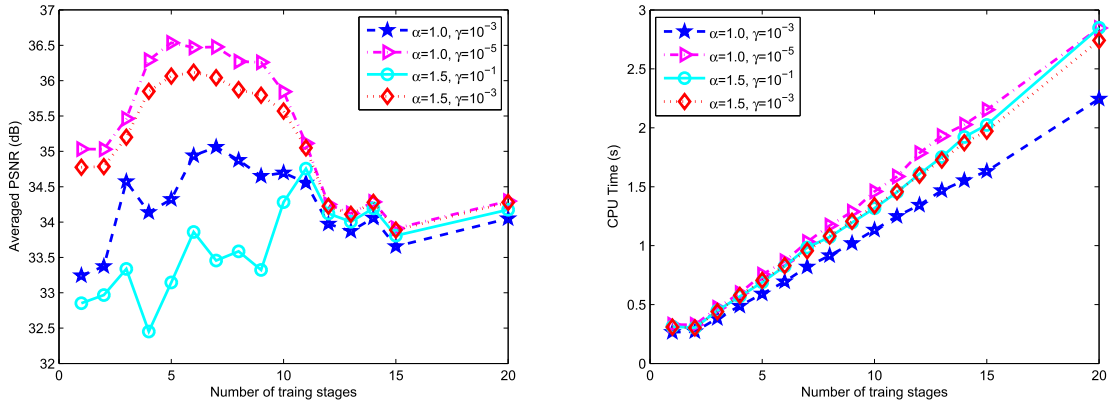

FIGURE 6. Impact of different stage numbers on the reconstruction. Left: The reconstruction averaged PNSRs versus different number of stages; Right: the reconstruction averaged CPU time versus different number of stages.

a good tradeoff between the reconstruction accuracy and computational time, we choose a 5-stage network as the considerable network and repeat the experiment 100 times, and then average the results.

3) RECONSTRUCTION PERFORMANCE VERSUS THE NUMBER OF TRAINING IMAGE

To evaluate the reconstruction results of proposed Co-Robust-ADMM-net using different number of training samples, in this experiment, we employ 1, 5, 10, 20 and 30 'Chest' MR images for training, and then reconstruct the 'Brain' images using the learned network under four different levels of SαS noise shown in figure 7. Figure 8 shows the reconstruction PSNRs and the corresponding CPU time under different cases. From the results, we can explicitly observe that increasing the number of training samples can achieve higher reconstruction accuracy and then gradually become stable. Moreover, the computational overhead does not significantly increase. However, increasing the number of training samples will cause overwhelming computational overhead in the training process; hence it is not unnecessary to increase the sampling number in application.

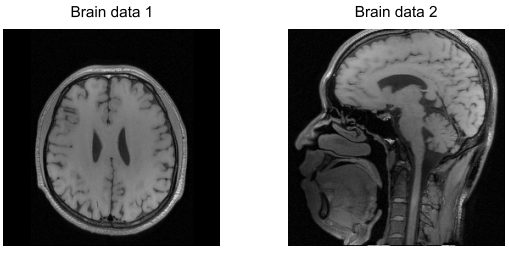

FIGURE 7. Two typical magnetic resonance images.

B. TEST

To further demonstrate the superiority of the proposed Co-Robust-ADMM-net algorithm, in this section, we compare the proposed algorithm with some existing corresponding work. The YALL1 algorithm of [21] is a well-known robust reconstruction algorithm, and the LqLA-ADMM algorithm of [11] is a more recent approach for robust compressive sensing reconstruction. Both of them employ the ADMM framework to address the robust reconstruction problem and have achieved good performances, in this paper, we conduct comparison experiment using the provided matlab code in https://github.com/FWen/LqLA-Sparse-Recovery.git. The ADMM-net algorithm of [17] is a similar work based on a denoise model called BM3D-MRI [27], although the ADMM-net is proposed in Gaussian environment, we find that the algorithm can suppress the outliers effectively, and the algorithm of Co-Robust-ADMM is the proposed algorithm without training. For a fair comparison, the robust reconstruction algorithms of YALL1, LqLA-ADMM and Co-Robust-ADMM employ the discrete cosine transformation (DCT) as the sparsifying transform; the algorithms of ADMM-net and Co-Robust-ADMM-net employ DCT as the initial transform. Following the above analysis, we employ the proposed network with 5 stages for training using the 'Chest' image shown in figure 3.

TEST 1

In this experiment, we train the samplings using one image shown in figure 3 (a) under a very impulsive noise environment ($\alpha = 1.0, \gamma = 10^{-4}$), and then reconstruct two different 'Brain' MR images using the learned parameters. As shown in figure 6, the two 'Brain' magnetic resonance images are widely used to evaluate the CS reconstruction performance. Table 1 and Table 2 detail the reconstruction results of four algorithms for different levels of SαS noise-corrupted 'Brain 1' data and 'Brain 2' data with four sampling rates of 0.2, 0.3, 0.4 and 0.5. In the tables, '–' indicates that the performance of the corresponding algorithm is poor. We can observe that the algorithms of YALL1 and LqLA-ADMM fail to reconstruct the images when the strength of $S\alpha S$ noise is too high (e.g., $\gamma = 10^{-1}$ and 10^{-3}). The algorithms of ADMM-net and Co-Robust-ADMM can obtain considerable reconstruction results, with the proposed Co-Robust-ADMM-net achieving the best performance.

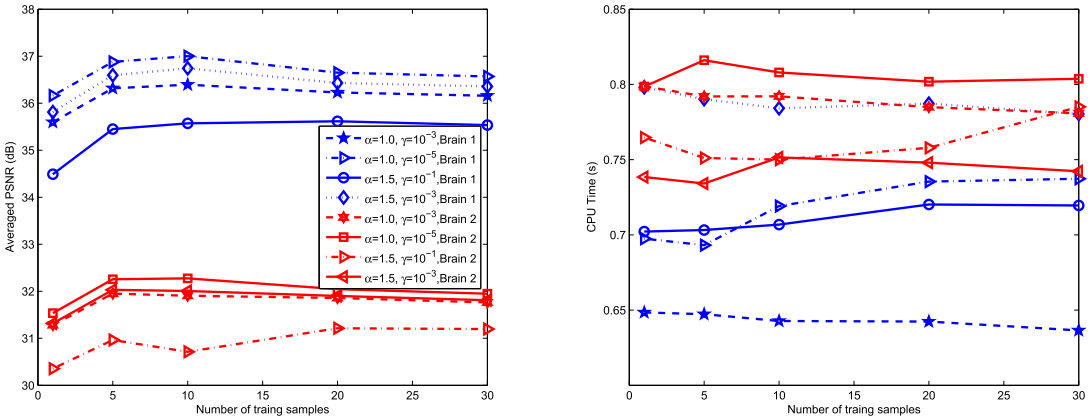

FIGURE 8. Reconstruction Performance versus the training number. Left: the reconstruction averaged PNSRs versus different number of training samples; Right: the reconstruction averaged CPU time versus different number of training samples.

TABLE 1. Reconstruction PSNR (dB) for 'Brain 1' data with different sampling rates and noises.

Method	$\alpha = 1.0, \gamma = 10^{-3}$				$\alpha = 1.0, \gamma = 10^{-5}$				$\alpha = 1.5, \gamma = 10^{-1}$			
	20%	30%	40%	50%	20%	30%	40%	50%	20%	30%	40%	50%
YALL1	11.18	11.44	12.60	12.79	24.88	27.23	28.89	32.21	–	–	–	–
LqLA-ADMM	–	–	–	–	26.23	28.5	31.19	33.70	–	–	–	–
ADMM-net	31.95	34.20	34.63	35.68	**32.78**	**36.12**	**38.19**	40.04	30.58	33.02	29.59	29.58
Initial-Co-Robust-ADMM	27.29	29.88	30.05	30.60	27.31	29.03	30.10	30.82	27.26	29.01	30.06	30.80
Co-Robust-ADMM-net	**32.47**	**34.70**	**35.58**	**37.10**	32.62	35.89	38.07	**40.21**	**32.35**	**34.77**	**33.26**	**34.01**

TABLE 2. Reconstruction PSNR (dB) for 'Brain 2' data with different sampling rates and noises.

Method	$\alpha = 1.0, \gamma = 10^{-3}$				$\alpha = 1.0, \gamma = 10^{-5}$				$\alpha = 1.5, \gamma = 10^{-1}$			
	20%	30%	40%	50%	20%	30%	40%	50%	20%	30%	40%	50%
YALL1	–	–	–	–	21.06	22.83	25.39	26.31	–	–	–	–
LqLA-ADMM	–	–	–	–	22.18	24.53	26.74	28.37	–	–	–	–
ADMM-net	27.53	30.38	31.48	32.04	28.97	31.45	**33.51**	35.21	27.21	29.11	28.10	28.74
Initial-Co-Robust-ADMM-net	25.53	26.96	28.12	28.83	25.58	27.07	28.20	28.96	25.55	27.05	28.18	28.93
Co-Robust-ADMM-net	**28.27**	**31.17**	**32.17**	**33.43**	**29.05**	**31.68**	33.42	**35.50**	**28.55**	**30.67**	**30.68**	**31.91**

TABLE 3. Reconstruction PSNRs (dB) for 'Brain 1' data with different sampling rates and noises.

Method	$\alpha = 0.75, \gamma = 10^{-5}$		$\alpha = 1.0, \gamma = 10^{-3}$		$\alpha = 1.0, \gamma = 10^{-5}$		$\alpha = 1.5, \gamma = 10^{-3}$	
	20%	40%	20%	40%	20%	40%	20%	40%
YALL1	–	–	–	–	24.88	28.89	13.56	16.76
LqLA-ADMM	–	–	–	–	26.23	31.19	22.93	22.93
ADMM-net	31.80	36.02	32.73	36.92	33.04	37.83	33.04	37.89
Initial-Co-Robust-ADMM-net	27.27	29.69	27.23	29.96	27.31	30.10	27.31	30.10
Co-Robust-ADMM-net	**31.89**	**36.13**	**32.82**	**36.97**	**33.13**	**37.96**	**33.11**	**37.96**

Moreover, the proposed Co-Robust-ADMM-net can improve the reconstruction results significantly after training via the network.

TEST 2

To further demonstrate the robustness and universality of the proposed algorithm, in this experiment, we employ the same sample but corrupted different levels of SαS noise ($\alpha = 1.5, \gamma = 10^{-1}$). Next, we use the learned parameters to reconstruct the same MR images under four different levels of SαS noise. Table 5 details the reconstruction results of four algorithms for 'Brain 1' data and 'Brain 2' data with two sampling rates of 0.2 and 0.4, respectively. The proposed algorithm of Co-ADMM-Robust-net outperform the other four algorithms, and the results are consistent with Tables 3 and 4,

C. FURTHER DISCUSSION ABOUT TRAINING AND TEST

In this section, we will give further discussion and analysis about the achieved results of training and test. From Figure 6, we observe that the reconstruction performance will be improved significantly with the increment of the training stage, and achieve the best results when using about 5, 6, 7 and 8 training stages, however, the performance will declined and then tend to be stable. In general, more stages

TABLE 4. Reconstruction PSNRs (dB) for 'Brain 2' data with different sampling rates and noises.

Method	$\alpha = 0.75, \gamma = 10^{-5}$		$\alpha = 1.0, \gamma = 10^{-3}$		$\alpha = 1.0, \gamma = 10^{-5}$		$\alpha = 1.5, \gamma = 10^{-3}$	
	20%	40%	20%	40%	20%	40%	20%	40%
YALL1	–	–	–	–	21.06	25.39	11.55	13.26
LqLA-ADMM	–	–	–	–	22.18	26.74	22.93	17.20
ADMM-net	28.32	32.83	28.36	33.05	28.98	33.43	28.00	33.44
Initial-Co-Robust-ADMM-net	25.50	28.02	25.57	28.11	25.58	28.15	25.58	28.20
Co-Robust-ADMM-net	**28.46**	**33.03**	**28.45**	**33.19**	**29.08**	**33.65**	**29.05**	**33.60**

means deeeper network, and will provide better training to obtain optimal parameters. We also find that the performance of four cases will close when the stage is over 11, and then rise slowly, considering the CPU cost time increases linearly with the stages number, we choose 5-stage network as the consideration network.

From Figure 8, we can find that the achieved averaged PSNRs increase significantly first, and then decrease slightly till tend to be stable. In general, more samples can provide more details for training the network, and can provide the optimal parameters for optimization, however, these details is not always suit for reconstruction of the unknown desired images, because the image exist difference of structure and details indeed, thus, it is reasonable for our obtained results and we tend to choose 5 MR images for training in this paper.

V. CONCLUSIONS

In this paper, we proposed a robust compressive sensing reconstruction algorithm called the Co-robust-ADMM-net by joining the traditional ADMM framework and the deep neural network. A robust composite regularization model was employed to further exploit more of the physical properties and prior knowledge of the trained image. The strategy of the deep neural network can effectively train the parameters using the back propagation methods. Compared with the no-training algorithm of the Initial-Co-Robust-ADMM, the proposed Co-Robust-ADMM-net algorithm can improve the reconstruction accuracy significantly. Simulation results indicated that the strategy of combining the traditional CS reconstruction algorithms with the deep neural network can improve the recovery performance and should be pursued in our future work.

ACKNOWLEDGMENT

The authors would like to appreciate the handling editor and anonymous reviewers for their constructive comments.

REFERENCES

[1] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[2] Y. Li *et al.*, "Sparse adaptive iteratively-weighted thresholding algorithm (SAITA) for L_p-regularization using the multiple sub-dictionary representation," *Sensors*, vol. 17, no. 12, pp. 2920–2936, 2017.

[3] Z. Lai *et al.*, "Image reconstruction of compressed sensing MRI using graph-based redundant wavelet transform," *Med. Image Anal.*, vol. 27, pp. 93–104, Jan. 2016.

[4] X. Cong *et al.*, "A novel adaptive wide-angle SAR imaging algorithm based on Boltzmann machine model," *Multidimensional Syst. Signal Process.*, vol. 29, no. 1, pp. 119–135, 2016.

[5] W. L. Chan, M. L. Moravec, R. G. Baraniuk, and D. M. Mittleman, "Terahertz imaging with compressed sensing and phase retrieval," *Opt. Lett.*, vol. 33, no. 9, pp. 974–976, 2008.

[6] L. He and H. Zhang, "Iterative ensemble normalized cuts," *Pattern Recognit.*, vol. 52, pp. 274–286, Apr. 2016.

[7] L. He and H. Zhang, "Kernel K-means sampling for Nyström approximation," *IEEE Trans. Image Process.*, vol. 27, no. 5, pp. 2108–2120, May 2018.

[8] J. Wen, Z. Zhou, Z. Liu, M. Lai, and X. Tang, "Sharp sufficient conditions for stable recovery of block sparse signals by block orthogonal matching pursuit," *Appl. Comput. Harmon. Anal.*, to be published, doi: 10.1016/j.acha.2018.02.002.

[9] J. Wen, J. Wang, and Q. Zhang, "Nearly optimal bounds for orthogonal least squares," *IEEE Trans. Signal Process.*, vol. 65, no. 20, pp. 5347–5356, Oct. 2017.

[10] J. Wen, Z. Zhou, D. Li, and X. Tang, "A novel sufficient condition for generalized orthogonal matching pursuit," *IEEE Commun. Lett.*, vol. 21, no. 4, pp. 805–808, Apr. 2017.

[11] F. Wen, L. Pei, Y. Yang, W. Yu, and P. Liu, "Efficient and robust recovery of sparse signal and image using generalized nonconvex regularization," *IEEE Trans. Comput. Imag.*, vol. 3, no. 3, pp. 566–579, Dec. 2017.

[12] Y. H. Xiao, H. Zhu, and S. Y. Wu, "Primal and dual alternating direction algorithms for ℓ_1-ℓ_1-norm minimization problems in compressive sensing," *Comput. Optim. Appl.*, vol. 54, no. 2, pp. 441–459, 2013.

[13] D.-S. Pham and S. Venkatesh, "Efficient algorithms for robust recovery of images from compressed data," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 4724–4737, Dec. 2013.

[14] M. V. Afonso, J.-M. Bioucas-Dias, and M. A. T. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2345–2356, Sep. 2010.

[15] F. Wen, P. Liu, Y. Liu, R. C. Qiu, and W. Yu, "Robust sparse recovery in impulsive noise via ℓ_1-ℓ_1 optimization," *IEEE Trans. Signal Process.*, vol. 65, no. 1, pp. 105–118, Jan. 2017.

[16] Y. Li, Y. Lin, X. Cheng, Z. Xiao, F. Shu, and G. Gui, "Nonconvex penalized regularization for robust sparse recovery in the presence of SαS noise," *IEEE Access*, vol. 6, pp. 25474–25485, 2018.

[17] Y. Yang, J. Sun, H. Li, and Z. Xu, "Deep ADMM-net for compressive sensing MRI," in *proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2016, pp. 10–18.

[18] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[19] R. Ahmad and P. Schniter, "Iteratively reweighted ℓ_1 approaches to sparse composite regularization," *IEEE Trans. Comput. Imag.*, vol. 1, no. 4, pp. 220–235, Dec. 2015.

[20] Y. Li *et al.*, "MUSAI-ℓ_1: MUltiple sub-wavelet-dictionaries-based adaptively-weighted iterative half thresholding algorithm for compressive imaging," *IEEE Access*, vol. 6, pp. 16795–16805, 2018.

[21] J. Yang and Y. Zhang, "Alternating direction algorithms for ℓ_1 problems in compressive sensing," *SIAM J. Sci. Comput.*, vol. 33, no. 1, pp. 250–278, 2011.

[22] R. Cabral, F. De La Torre, J. P. Costeira, and A. Bernardino, "Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2488–2495.

[23] E. Kim, M. Lee, C.-H. Choi, N. Kwak, and S. Oh, "Efficient ℓ_1-norm-based low-rank matrix approximations for large-scale problems using alternating rectified gradient method," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 2, pp. 237–251, Feb. 2015.

[24] E. Kim, M. Lee, and S. Oh, "Elastic-net regularization of singular values for robust subspace learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 915–923.

[25] K. Guo, L. Liu, X. Xu, D. Xu, D. Tao, "GoDec+: Fast and robust low-rank matrix decomposition based on maximum correntropy," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2323–2336, Jun. 2018.
[26] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.
[27] E. M. Eksioglu, "Decoupled algorithm for MRI reconstruction using nonlocal block matching model: BM_3D-MRI," *J. Math. Imag. Vis.*, vol. 56, no. 3, pp. 430–440, 2016.

YUNYI LI (S'18) received the B.Sc. degree in communication engineering from the Hunan University of Science and Technology, Xiangtan, China, in 2014. He is currently pursuing the Ph.D. degree in circuits and systems with the Nanjing University of Posts and Telecommunications, Nanjing, China. His current research interests include intelligent information processing, compressed sensing and sparse recovery and signal processing on wireless communication. He has currently published several journal papers and international conference papers.

XIEFENG CHENG received the M.Sc. degree in communication engineering from Shandong University, Jinan, China, in 2004. He is currently a Professor with the Nanjing University of Posts and Telecommunications, Nanjing, China. His current research interests include intelligent information processing, heart sound signal processing, biometric extraction, and pattern recognition and machine learning.

GUAN GUI (M'11–SM'17) received the Dr. Eng. degree in information and communication engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2012. From 2009 to 2012, with the financial supported from the China Scholarship Council and the Global Center of Education of Tohoku University, he joined the Wireless Signal Processing and Network Laboratory (Prof. Fumiyuki Adachi Laboratory), Department of Communications Engineering, Graduate School of Engineering, Tohoku University, as a Research Assistant and also as a Post-Doctoral Research Fellow, respectively. From 2012 to 2014, he was supported by the Japan Society for the Promotion of Science Fellowship as a Post-Doctoral Research Fellow at the Wireless Signal Processing and Network Laboratory (Prof. Fumiyuki Adachi Laboratory). From 2014 to 2015, he was an Assistant Professor with the Department of Electronics and Information System, Akita Prefectural University. Since 2015, he has been a Professor with the Nanjing University of Posts and Telecommunications, Nanjing, China. He is currently involved in the research of deep learning, compressive sensing, and advanced wireless techniques. He received several Best Paper Awards, such as the CSPS2018, ICNC2018, ICC2017, ICC2014, and VTC2014-Spring. He was selected for the Jiangsu Special Appointed Professor, the Jiangsu High-level Innovation and Entrepreneurial Talent, and the Nanjing Youth Award. He was an Editor of the *Security and Communication Networks* (2012–2016), the IEEE TRANSACTIONS ON VEHICULAR T biosc echnology, and the *KSII Transactions on Internet and Information System*.