Quantized Proximal Averaging Networks for Compressed Image Recovery

Nareddy Kartheek Kumar Reddy, Mani Madhoolika Bulusu, Praveen Kumar Pokala,
Chandra Sekhar Seelamantula
Department of Electrical Engineering, Indian Institute of Science, Bengaluru, 560012.

{nareddyreddy, manibulusu}@iisc.ac.in, praveenkumar.pokala@gmail.com, css@iisc.ac.in

Abstract

We solve the analysis sparse coding problem considering a combination of convex and non-convex sparsity promoting penalties. The multi-penalty formulation results in an iterative algorithm involving proximal-averaging. We then unfold the iterative algorithm into a trainable network that facilitates learning the sparsity prior. We also consider quantization of the network weights. Quantization makes neural networks efficient both in terms of memory and computation during inference, and also renders them compatible for low-precision hardware deployment. Our learning algorithm is based on a variant of the ADAM optimizer in which the quantizer is part of the forward pass and the gradients of the loss function are evaluated corresponding to the quantized weights while doing a book-keeping of the high-precision weights. We demonstrate applications to compressed image recovery and magnetic resonance image reconstruction. The proposed approach offers superior reconstruction accuracy and quality than state-of-the-art unfolding techniques and the performance degradation is minimal even when the weights are subjected to extreme quantization.

1. Introduction

The objective in Compressive Sensing (CS) is to reconstruct a sparse signal $x \in \mathbb{R}^n$ from its compressed observation, $y = \Phi x \in \mathbb{R}^m$, where Φ is an $m \times n$ matrix, with $m \ll n$. The standard convex optimization formulation to solve the CS problem is given as follows:

$$\min_{x} \ \frac{1}{2}\|\Phi x - y\|_2^2 + \lambda\|\Psi x\|_1, \qquad (1)$$

where y is the observation, Ψ is the sparsifying transform for x, and sparsity is enforced on Ψx using an appropriate regularizer, which in this case is the ℓ_1-norm, and λ is the regularization parameter.

However it has been shown that ℓ_1-norm regularization results in biased amplitude estimates [1]. Non-convex regularizers are known to reduce the bias in the estimates. One

Q-ISTA-Net$^+$ Q-PAN$^+$ (2R)   Q-PAN$^+$ (3R)

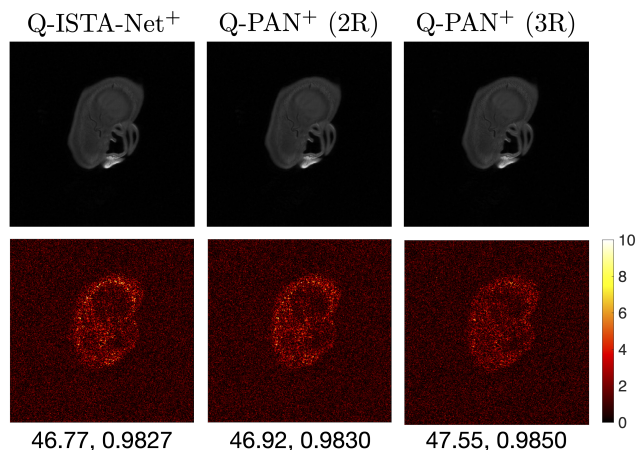46.77, 0.9827   46.92, 0.9830   47.55, 0.9850

Figure 1. Brain MRI image reconstruction for CS ratio 40% by the proposed 1-bit Quantized models. The bottom row visualizes the absolute difference between the ground truth and the reconstructed images. The Q-PAN$^+$ (3R) model captured the brain structure better and hence the difference image is majorly composed of noise.The numbers indicates the PSNR, SSIM values.

can design efficient non-convex regularizers by considering a combination of several sparsity-promoting regularizers. The CS framework has enabled several practical applications particularly in the context of computational imaging [2], ultrasound imaging [3], magnetic resonance imaging [4], cognitive radar [5], cognitive radio [6], etc.

Sparse recovery algorithms are iterative and involve manual parameter tuning and several iterations to achieve accurate reconstruction. Iterative algorithms when unfolded into a feedforward neural network architecture can greatly enhance the reconstruction accuracy since the network has the advantage of *learnability*. Further, the depth of the network is also significantly lower than the number of iterations of the algorithm from which it is inspired. Therefore, the inference time is reduced. However, the price to pay for these advantages is the overhead of training.

A practical bottleneck in deploying such models is that they require more memory to store the high-precision

weights and consequently also more computation. In applications involving low-precision hardware, the reconstruction accuracy reduces drastically. One of the issues that is addressed in this paper is the effect of weight quantization on the performance of unfolded networks. The objective is to obtain comparable accuracy to the full-precision scenario considering weights that are coarsely quantized (which we show in the Figure 1). Such *quantized unfolded networks* are more amenable to practical implementation particularly in resource constrained settings.

Another important aspect that we focus on in this paper is the problem of *analysis prior learning* using a combination of convex/non-convex sparsity promoting regularizers. The analysis prior is more effective than the synthesis prior because most naturally occurring signals and images are sparse only when analyzed using a suitable transformation [7]. Further, employing a combination of penalty functions has been shown to result in superior reconstruction performance than a single penalty function [8] and gives rise to the notion of *proximal averaging*. The penalty functions could be convex or non-convex. *Learning* a convex combination of penalty functions is also likely to improve the sparse recovery performance.

Before proceeding with further developments, we review recent state of the art in these domains.

2. Prior Art

We present the literature in two categories, namely, quantized networks and analysis sparse recovery techniques.

2.1. Quantized Networks

Neural Networks that employ binary weights and activations, also known as binary neural networks (BNNs) [10–12] have been shown to be promising for solving classification problems. Their efficacy has been demonstrated on MNIST [13] and CIFAR10 [14] datasets. The XNOR-Net [15] used a gain term computed from the statistics of the weights to scale the binary weights in order to reduce the quantization error. The XNOR-Net performed relatively better than previous BNN architectures on the ImageNet [16] dataset. DoReFa-Net [17] generalizes the quantization framework with variable bit-precision for the weights, the activations and the gradients required during backpropagation. The learned quantization networks (LQ-Net) [18] was proposed to minimize the quantization error by jointly learning the quantizer and the network. The Bi-Real Network [19] employs residual connections where full-precision activations from the previous layer affect the quantization in the current layer. Bai *et al.* poses quantized network training as a regularized learning problem and optimizes via the proximal gradient method [20]. The problem of outliers affecting the quantization has been addressed by

channel splitting in [21]. On the optimization front Xiangming *et al.* justified the straight-through-estimator (STE) method used for training the BNNs through bayesian learning [22]. The quantization of weights has also been shown to be promising in the context of generative adversarial networks (GANs) [23, 24]. Jhu *et al.* adapts Neural Architecture Search (NAS) to find an optimized architecture for the binarization of Convolutional Neural Networks (CNNs) [25]. Duncan *et al.* showed that quantized neural networks are generally robust relative to their full precision counterpart against adversarial perturbations [26]. While the role of quantization has been considered in classification and generation problems, it has been relatively unexplored in the context of optimizing unfolded networks for sparse recovery.

2.2. Prior Art on Analysis Sparse Priors

Several works [27–30] have considered the ℓ_1-regularized analysis/synthesis sparse recovery problems owing to its convexity. The analysis-sparse recovery problem can be solved by employing interior-point techniques. However, they become challenging to handle computationally as the dimensionality of the problem grows [31]. An alternative approach for solving the analysis-sparse problem is based on the alternating direction method of multipliers (ADMM) algorithm [32]. Nonetheless, the efficiency of the ADMM method depends highly on the structure of $\mathbf{A}$ [33]. Beck & Teboulle [27] proposed a fast variant of iterative soft-thresholding algorithm (ISTA) [34] based on Nesterov's momentum [35] for solving the synthesis-sparse recovery problem. Smoothing FISTA (SFISTA) [33] was proposed to address the limitation associated with FISTA, and it solved the analysis-sparse recovery problem considering a redundant operator.

Liu *et al.* demonstrated that SFISTA convergence and reconstruction performance is sensitive to the smoothness parameter μ and proposed projected FISTA to overcome the limitation of SFISTA [29]. In order to solve the CS problem in the context of recovering natural images, optimization based techniques assumed that the input images are sparse in a fixed transform domain such as the discrete cosine transform (DCT), wavelet domain [36], gradient-domain [37], etc. Denoising based AMP (D-AMP) [38], etc. ensures good recovery for natural images. Most methods exploit structured sparsity of the data as an image prior and solve the problem iteratively [37–40]. However, natural images are non-stationary and lead to poor reconstruction on a fixed basis. All iterative sparse recovery methods require hundreds of iterations to solve Eq. (1), which inevitably gives rise to high computational overhead thus restricting the application of CS. In addition, the analysis operator or the hyperparameters such as learning rate and regularization parameter are usually hand-crafted.

Table 1. Sparsity-promoting regularizers and corresponding proximal operators employed in this study.

NAME	PENALTY FUNCTION	PROXIMAL OPERATOR
$\ell_1 - \text{norm}\ (\lambda > 0)$	$g_1(x) = \lambda\,|x|$	$\mathcal{P}_1(x) = \text{sgn}(x)\max\left(|x| - \lambda, 0\right)$
MCP [1] $(\lambda > 0, \gamma > 1)$	$g_2(x) = \begin{cases} \lambda\,|x| - \dfrac{|x|^2}{2\lambda\gamma}, & \text{for } |x| \leq \gamma\lambda, \\ \dfrac{\lambda^2\gamma}{2}, & \text{for } |x| \geq \gamma\lambda. \end{cases}$	$\mathcal{P}_2(x) = \begin{cases} 0, & \text{for } |x| \leq \lambda, \\ \text{sgn}\,(x)\,\dfrac{\gamma}{\gamma - 1}\left(|x| - \lambda\right), & \text{for } \lambda < |x| \leq \gamma\lambda, \\ x, & \text{for } |x| > \gamma\lambda. \end{cases}$
SCAD [9] $(\lambda > 0, a > 2)$	$g_3(x) = \begin{cases} \lambda\,|x|, & \text{for } |x| \leq \lambda, \\ \dfrac{|x|^2 - 2a\lambda\,|x| + \lambda^2}{2(1 - a)}, & \text{for } \lambda < |x| \leq a\lambda, \\ \dfrac{(a + 1)\lambda^2}{2}, & \text{for } |x| > a\lambda. \end{cases}$	$\mathcal{P}_3(x) = \begin{cases} \text{sgn}\,(x)\max\left(|x| - \lambda, 0\right), & \text{for } |x| \leq 2\lambda, \\ \dfrac{(a - 1)\,x - \text{sgn}\,(x)\,a\lambda}{a - 2}, & \text{for } 2\lambda < |x| \leq a\lambda, \\ x, & \text{for } |x| > a\lambda. \end{cases}$

To overcome the limitations of optimization based techniques, network-based approaches have been proposed recently. ReconNet is a neural network approach proposed in [41], which takes in CS measurements of an image as input and outputs an intermediate reconstruction. However, it does not utilize the structure offered by iterative algorithms and is not interpretable. Learnable ISTA (LISTA) [42], unrolls the iterations of ISTA [27] into an interpretable neural network architecture. The parameters are learned by end-to-end training instead of being hand-crafted. Several works exploited the idea of deep-unfolding of iterative algorithms and have shown better results than their iterative counterparts [43–45]. In order to solve the CS problem in the context of recovering natural images, it is assumed that the input images are sparse in a fixed transform domain such as the discrete cosine transform (DCT), wavelet domain [36], gradient-domain [37], etc. However, natural images are non-stationary and lead to poor reconstruction on a fixed basis. Networks such as ISTA-Net [45] performed better when the sparsifying transform was learned by imposing the ℓ_1 penalty on the prior. Many CNN based models were proposed to solve the compressed image recovery problem [46, 47].

Two aspects are missing from current literature. First, finer approximation of ℓ_0 pseudonorm in the context of data-driven analysis-prior learning and second, reducing the model size and improved hardware-compatibility of the CS networks. We aim to address these two aspects with application to compressed image reconstruction.

3. Contribution of this Paper

Our contribution lies at the intersection between the two aspects – network quantization and optimization for analysis sparse prior learning while employing several regularizers. The starting point for the developments is a multipenalty/composite regularization formulation of the sparse

recovery problem in an analysis-sparse setting. The result is the proximal-averaged iterative shrinkage algorithm (PAISA), which when unfolded gives rise to the proximal-averaging network (PAN). It is in the context of PAN that we consider the effect of quantization (Q-PAN). On the application front, we consider image recovery from compressed measurements both for natural images and magnetic resonance images.

4. Composite Regularization in the Analysis Setting

Consider the problem of compressed sensing recovery based on multiple penalties $\{g_i\}$ as given below:

$$\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{\Phi x} - \boldsymbol{y}\|_2^2 + \underbrace{\sum_{i=1}^{p} \alpha_i g_i(\mathcal{F}(\boldsymbol{x}))}_{g(\boldsymbol{x})} \tag{2}$$

$$\text{subject to } \sum_{i=1}^{p} \alpha_i = 1, \quad 0 < \alpha_i < 1, \ \forall i,$$

where $\mathcal{F}$ is the data-driven, sparsifying, analysis operator and p is the number of sparsity-promoting regularizers. The convex and non-convex regularizers that are considered in our work are the ℓ_1-norm, minimax-concave penalty (MCP) [1], and smoothly clipped absolute deviation (SCAD) [9]. While the ℓ_1-norm is convex, the other two penalties – MCP and SCAD – are not convex. The MCP and SCAD penalties approximate the ℓ_0 pseudonorm better than the ℓ_1 norm and hence give rise to better recovery performance [1, 48]. The ideal sparse recovery problem requires minimization of the ℓ_0 pseudonorm, but it is NP-hard and hence tractable solvers that approximate ℓ_0 minimization are preferred. The MCP and SCAD penalties perfectly fit the bill and hence we consider them for composite regularization. The combination of these penalties has been shown to outperform the individual penalties for various application [49–51]. The three
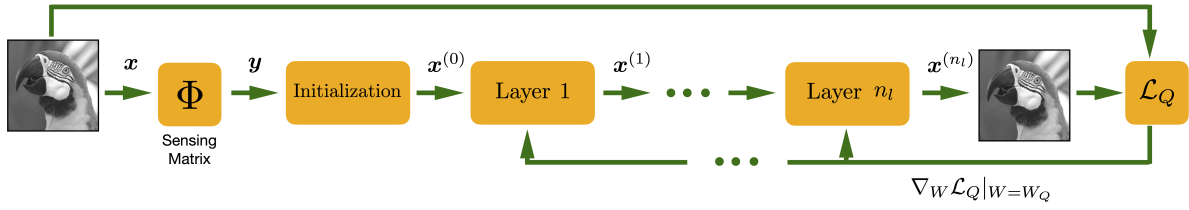
Figure 2. Illustration of the proposed Quantized Proximal Averaging Network (Q-PAN).

penalties under consideration and the corresponding proximal operators are given in Table 1.

4.1. Proximal-Averaged Iterative Shrinkage Algorithm (PAISA)

The problem in Eq. (2) is optimized based on majorization-minimization [52] and proximal-averaging [53]. The update x at $(k + 1)^{\text{th}}$ iteration is obtained by solving the following optimization problem:

$$\min_{x} \frac{1}{2\rho} \|x - r^{(k+1)}\|_2^2 + \sum_{i=1}^{p} \alpha_i g_i(\mathcal{F}(x)),$$

where $r^{(k+1)} = x^{(k)} - \rho \Phi^{\text{T}}(\Phi x^{(k)} - y)$ and ρ is the step-size. The above problem can be rewritten as

$$\min_{x} \sum_{i=1}^{p} \left(\frac{\alpha_i}{2\rho} \|x - r^{(k+1)}\|_2^2 + \alpha_i g_i(\mathcal{F}(x)) \right). \quad (3)$$

Zhang & Ghanem showed that a data-driven analysis transform improves the reconstruction of natural images from the compressed measurements [45]. Based on this observation, $\mathcal{F}$ is chosen to be a combination of two linear convolutional operators, without bias terms, separated by a rectified linear unit (ReLU) [54]. $\mathcal{F}$ can be formulated in matrix form as $\mathcal{F}(x) = \mathbf{B} \max(\mathbf{A}x, \mathbf{0})$, where $\mathbf{A}$ and $\mathbf{B}$ correspond to the two convolutional operators, and $\max(\cdot, \mathbf{0})$ denotes the ReLU. The convolutional operators $\mathbf{A}$ and $\mathbf{B}$ use n_f filters of size 3×3 and $3 \times 3 \times n_f$, respectively.

Zhang *et al.* showed that, in the context of inverse problems in imaging, it is reasonable to assume that the elements of $(x^{(k+1)} - r^{(k+1)})$ are independent normal distributed random variables with zero mean and variance σ^2 [55]. Let $r^{(k+1)}$ and $\mathcal{F}(r^{(k+1)})$ denote the mean values of x and $\mathcal{F}(x)$, respectively. The following approximation holds [45]:

$$\|\mathcal{F}(x) - \mathcal{F}(r^{(k+1)})\|_2^2 \approx \beta \|x - r^{(k+1)}\|_2^2, \quad (4)$$

where β depends only on the parameters of $\mathcal{F}$. Incorporating the approximation in Eq. (4) into Eq. (3), we obtain

(merging ρ and β into the parameters of the regularizers):

$$x^{(k+1)} = \arg \min_{x} \sum_{i=1}^{p} \alpha_i \|\mathcal{F}(x) - \mathcal{F}(r^{(k+1)})\|_2^2$$
$$+ \sum_{i=1}^{p} \alpha_i g_i(\mathcal{F}(x)). \quad (5)$$

The solution to the above problem relies on proximal-averaging and is given by

$$x^{(k+1)} = \widetilde{\mathcal{F}} \left(\sum_{i=1}^{p} \alpha_i \mathcal{P}_i(\mathcal{F}(r^{(k+1)})) \right), \quad (6)$$

where $\widetilde{\mathcal{F}}$ is the adjoint of $\mathcal{F}$ such that $\widetilde{\mathcal{F}}(\mathcal{F}(x)) = x$. The optimization steps for minimizing the problem in Eq. (2) are listed in Algorithm 1.

4.2. Proximal Averaging Network (PAN)

Unfolding the iterations of PAISA (Algorithm 1) results in a feedforward neural network, which we refer to as the proximal averaging network (PAN). A single PAN layer is represented by

$$x^{(k)} = \widetilde{\mathcal{F}}^{(k)} \left(\sum_{i=1}^{p} \alpha_i \mathcal{P}_i(\mathcal{F}^{(k)}(r^{(k)})) \right). \quad (7)$$

The parameters corresponding to the sparsity promoting regularizers and the analysis transform $\{\lambda_1^{(k)}, \lambda_2^{(k)}, \lambda_3^{(k)}, a^{(k)}, \gamma^{(k)}, \mathcal{F}^{(k)}, \widetilde{\mathcal{F}}^{(k)}\}$ are all learnable subject to the conditions $\lambda_i^{(k)} > 0, \gamma^{(k)} > 1, a^{(k)} > 2$ for each layer k of the network.

4.3. Loss Function

The loss function comprises two terms. The first term captures the mean-squared error between the ground truth x_i and the reconstructed image $x_i^{(n_l)}$, where n_l denotes the number of layers in the network. The second term seeks to enforce invertibility of the sparsifying transform $\mathcal{F}$ across the layers. The loss function $\mathcal{L}$ proposed in [45] is used in
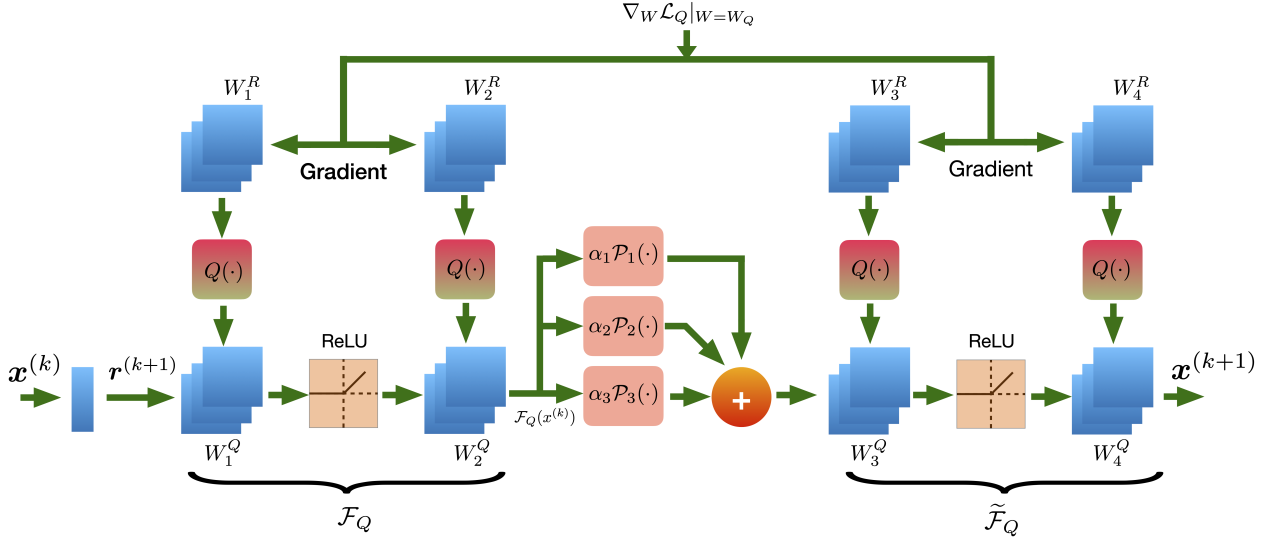
Figure 3. Structure of a layer in the Q-PAN. Quantized weights ($\mathbf{W}_i^Q$) are obtained from their full precision copy ($\mathbf{W}_i^R$) via quantization operator $Q(\cdot)$. The intermediate reconstruction $\boldsymbol{x}^{(k)}$ is processed through convolution filters $\{\mathbf{W}_i^Q\}$, ReLU transform, proximal operators $\{\mathcal{P}_i\}$ to give $\boldsymbol{x}^{(k+1)}$.

Algorithm 1: Proximal-Averaged Iterative Shrinkage Algorithm (PAISA)

---

1 **Input:** $\boldsymbol{x}$, K, $\boldsymbol{\lambda}_1$, $\boldsymbol{\lambda}_2$, $\boldsymbol{\lambda}_3$, $\boldsymbol{\gamma}$, $\boldsymbol{a}$
2 Initialize $\boldsymbol{x}_0$, $k = 0$
3 **while** $k \leq K - 1$ **do**
4 $\quad \boldsymbol{x}_k = \widetilde{\mathcal{F}}(\sum_{i=1}^{p} \alpha_i \mathcal{P}_i(\mathcal{F}(\boldsymbol{r}_k)))$
5 $\quad k \leftarrow k + 1$

---

our training:

$$
\begin{aligned}
\mathcal{L} = & \frac{1}{N} \sum_{i=1}^{n_b} \|\boldsymbol{x}_i^{(n_l)} - \boldsymbol{x}_i\|_2^2 \\
& + \gamma \frac{1}{N} \sum_{i=1}^{n_b} \sum_{k=1}^{n_l} \|\widetilde{\mathcal{F}}^{(k)}(\mathcal{F}^{(k)}(\boldsymbol{x}_i)) - \boldsymbol{x}_i\|_2^2,
\end{aligned} \tag{8}
$$

where n_b is the number of training patches extracted from the images in the dataset, n is the size of each training patch, n_l is the number of layers of the network, $N = n_b n$, and γ determines the trade-off between the two terms under consideration. γ is fixed at 0.01 in our experiments.

4.4. PAISA$^+$ Update

The residuals (defined as the difference between the ground truth and its prediction) of natural images and videos are compressible [56, 57]. The residual learning framework named ResNet [58] explicitly formulates the layers of the network to learn residual functions with reference to the layer inputs. Drawing inspiration from ResNet and ISTA-Net$^+$, we propose PAISA$^+$, whose iterations are unfolded to yield PAN$^+$, which learns a residual function for the update $\boldsymbol{x}_{k+1}$, instead of learning it directly as done in PAN.

The update step is: $\boldsymbol{x}^{(k+1)} = \boldsymbol{r}^{(k+1)} + \boldsymbol{w}^{(k+1)} + \boldsymbol{e}^{(k+1)}$, where $\boldsymbol{w}^{(k+1)}$ is the residual and $\boldsymbol{e}^{(k+1)}$ is the error, as recommended in ISTA-Net$^+$ [45]. The residual $\boldsymbol{w}^{(k+1)}$ contains the high-frequency component of $\boldsymbol{x}^{(k+1)}$ missing from $\boldsymbol{r}^{(k+1)}$. It can be extracted from $\boldsymbol{x}^{(k+1)}$ by an affine transformation $\mathcal{R}$ i.e., $\boldsymbol{w}^{(k+1)} = \mathcal{R}(\boldsymbol{x}^{(k+1)}) = \mathcal{G}(\mathcal{D}(\boldsymbol{x}^{(k+1)}))$, where $\mathcal{D}$ has n_f filters (of size 3×3) and $\mathcal{G}$ has one filter (of size $3 \times 3 \times n_f$).

To obtain a closed-form solution, we model $\mathcal{F} = \mathcal{H} \circ \mathcal{D}$, where $\mathcal{H}$ is composed of two convolutional operators $\mathcal{D}$ and $\mathcal{G}$ separated by a ReLU. Substituting $\mathcal{F}$ in Eq. (5) by $\mathcal{H} \circ \mathcal{D}$ gives

$$
\begin{aligned}
\boldsymbol{x}^{(k+1)} = & \arg \min_{\boldsymbol{x}} \ \frac{1}{2} \|\mathcal{H}(\mathcal{D}(\boldsymbol{x})) - \mathcal{H}(\mathcal{D}(\boldsymbol{r}^{(k+1)}))\|_2^2 \\
& + \sum_{i=1}^{p} \alpha_i g_i(\mathcal{H}(\mathcal{D}(\boldsymbol{x}))).
\end{aligned} \tag{9}
$$

Following the same strategy as in PAISA, we define $\widetilde{\mathcal{H}}$, the adjoint of $\mathcal{H}$ to be symmetric. The PAISA$^+$ update is:

$$
\begin{aligned}
\boldsymbol{x}^{(k+1)} = & \ \boldsymbol{r}^{(k+1)} \\
& + \mathcal{G}\left(\widetilde{\mathcal{H}}\left(\sum_{i=1}^{p} \alpha_i \mathcal{P}_i(\mathcal{H}(\mathcal{D}(\boldsymbol{r}^{(k+1)}))) \right) \right).
\end{aligned} \tag{10}
$$

The iterations of PAISA$^+$ are unfolded into the layers of a neural network, henceforth referred to as PAN$^+$. The parameters in all the layers of the model are learnable, i.e., all convolutional operators $\mathcal{H}, \widetilde{\mathcal{H}}, \mathcal{D}, \mathcal{G}$ are learned in each layer of the network, along with parameters involved in the proximal operators $\mathcal{P}_i$. A layer of PAN$^+$ is given by the input-output equation:

$$
\begin{aligned}
\boldsymbol{x}^{(k)} = \boldsymbol{r}^{(k)} \\
+ \mathcal{G}^{(k)}\bigg(\widetilde{\mathcal{H}}^{(k)}\bigg(\sum_{i=1}^{p}\alpha_i\mathcal{P}_i(\mathcal{H}^{(k)}(\mathcal{D}^{(k)}(\boldsymbol{r}^{(k)})))\bigg)\bigg).
\end{aligned}
\tag{11}
$$

The learnable parameters in this case are $\{\lambda_1^{(k)}, \lambda_2^{(k)}, \lambda_3^{(k)}, a^{(k)}, \gamma^{(k)}, \mathcal{H}^{(k)}, \widetilde{\mathcal{H}}^{(k)}, \mathcal{D}^{(k)}, \mathcal{G}^{(k)}\}$ subject to the conditions $\lambda_i^{(k)} > 0, \gamma^{(k)} > 1, a^{(k)} > 2$ for each layer k of the network.

5. Quantized Deep-Unrolled Compressive Sensing Networks

We now consider the effect of quantization of weights in the network. We consider K bit quantization of the weights. We work with two versions of the filter weights – the unquantized version and the quantized version. In the forward pass of Q-PAN, the full precision weights are mapped to the quantization levels, which are determined in a minimum mean-square error sense based on the weights obtained in the backpropagation update. The gradients of the loss function are computed with respect to the quantized weights. The weight updates happen on the full-precision version. The training strategy for Q-PAN models has been captured in Algorithm 2. The non-linear sparsifying transform $\mathcal{F}$ in Eq. (2) is modeled using a convolution network. In Q-PAN, a learnable quantization scheme maps the filter weights to lower-precision weights resulting in a quantized transform $\mathcal{F}_Q$. The sparsifying transform ($\mathcal{F}_Q$) is represented by conjunction of quantized convolution filters with ReLU. The update $\boldsymbol{x}$ at $(k+1)^{\text{th}}$ iteration with the quantized transform is given by

$$
\boldsymbol{x}_{k+1} = \arg\min_{x} J(x),
\tag{12}
$$

where

$$
J(x) = \sum_{i=1}^{p}\frac{\alpha_i}{2\rho}\|\boldsymbol{x} - \boldsymbol{r}^{(k+1)}\|_2^2 + \sum_{i=1}^{p}\alpha_i g_i(\mathcal{F}_Q(\boldsymbol{x})),
$$

ρ is the step-size and $\boldsymbol{r}^{(k+1)} = \boldsymbol{x}^{(k)} - \rho\boldsymbol{\Phi}^T(\boldsymbol{\Phi}\boldsymbol{x}^{(k)} - \boldsymbol{y})$. $\mathcal{F}_Q$ could be expressed in matrix form as $\mathcal{F}_Q(\boldsymbol{x}) = \mathbf{B}_Q\max(\mathbf{A}_Q\boldsymbol{x}, \mathbf{0})$, where $\mathbf{A}_Q$ and $\mathbf{B}_Q$ correspond to the convolutional operators with quantized weights. Following

the procedure outlined in Section 4, the structure of a layer in Q-PAN is modelled as follows:

$$
\boldsymbol{x} = \widetilde{\mathcal{F}}_Q\bigg(\sum_{i=1}^{p}\alpha_i\mathcal{P}_i(\mathcal{F}_Q(\boldsymbol{r}))\bigg),
\tag{13}
$$

where the learnable parameters are $\{\lambda_i, a, \gamma, \mathcal{F}_Q, \widetilde{\mathcal{F}}_Q\}$ subject to the conditions $\lambda_i > 0, \gamma > 1, a > 2$. The Q-PAN architecture is depicted visually in Figure 2. Similarly, each layer of the Q-PAN$^+$ model is given by:

$$
\boldsymbol{x} = \boldsymbol{r} + \mathcal{G}_Q(\widetilde{\mathcal{H}}_Q(\sum_{i=1}^{p}\alpha_i\mathcal{P}_i(\mathcal{H}_Q(\mathcal{D}_Q(\boldsymbol{r}))))).
\tag{14}
$$

The weights of Q-PAN$^+$ are also quantized to K-bits. The loss function with weight quantization in the learning framework is given by

$$
\begin{aligned}
\mathcal{L}_Q = &\frac{1}{N}\sum_{i=1}^{n_b}\|\boldsymbol{x}_i^{n_l} - \boldsymbol{x}_i\|_2^2 \\
&+ \gamma\frac{1}{N}\sum_{i=1}^{n_b}\sum_{k=1}^{n_l}\|\widetilde{\mathcal{F}}_Q^{(k)}(\mathcal{F}_Q^{(k)}(\boldsymbol{x}_i)) - \boldsymbol{x}_i\|_2^2.
\end{aligned}
\tag{15}
$$

Once the training is completed, only the quantized version of the weights is retained for inference. The architecture of one layer of Q-PAN is depicted in Fig. 3.

Algorithm 2: Training strategy for QPAN

---

1 **Input:** $\boldsymbol{x}$, Φ, $\boldsymbol{y}$ **Output:** $\boldsymbol{x}^*$
2 **while** $\boldsymbol{y}, \boldsymbol{x}$ *in training data* **do**
3   **while** $\mathbf{W}_Q, \mathbf{W}$ *in* $\mathcal{F}_Q, \mathcal{F}$ **do**
4          $\mathbf{W}_Q \leftarrow Q(\mathbf{W})$
5   **end**
6   **while** $\widetilde{\mathbf{W}}_Q, \widetilde{\mathbf{W}}$ *in* $\widetilde{\mathcal{F}}_Q, \widetilde{\mathcal{F}}$ **do**
7          $\widetilde{\mathbf{W}}_Q \leftarrow Q(\widetilde{\mathbf{W}})$
8   **end**
9   **while** $k = 1 : n_l$ **do**
10          $\boldsymbol{d}^{(k+1)} = \boldsymbol{x}^{(k)} - \rho\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{\Phi}\boldsymbol{x}^{(k)} - \boldsymbol{y})$
11          $\boldsymbol{x}^{(k)} = \widetilde{\mathcal{F}}_Q^{(k)}\left(\sum_{i=1}^{p}\alpha_i\mathcal{P}_i(\mathcal{F}_Q^{(k)}(\boldsymbol{d}^{(k+1)}))\right)$
12   **end**
13   Compute $\mathcal{L}_Q$ per Eq. (15)
14   **while** $\mathbf{W}, \widetilde{\mathbf{W}}$ *in* $\mathcal{F}, \widetilde{\mathcal{F}}$ **do**
15          $\mathbf{W} \leftarrow ADAM(\mathbf{W}, \nabla_{\mathbf{W}}\mathcal{L}_Q\,|_{\mathbf{W}=\mathbf{W}_Q})$
         $\widetilde{\mathbf{W}} \leftarrow ADAM(\widetilde{\mathbf{W}}, \nabla_{\mathbf{W}}\mathcal{L}_Q\,|_{\mathbf{W}=\widetilde{\mathbf{W}}_Q})$
16   **end**
17 **end**

---

6. Experimental Results and Discussion

We validate the performance of our networks, PAN, PAN$^+$ and their quantized counterparts, namely, Q-PAN

Table 2. Comparison between quantized and unquantized networks over average PSNR [dB] , SSIM values on Set11 and BSD68 datasets for different compression ratios.

CS Ratio	Test Set	ISTA-Net$^+$	PAN$^+$ (3R)	1-bit Q-PAN$^+$ (3R)	3-bit Q-PAN$^+$ (3R)
10%	Set11	26.64, 0.8036	**26.90, 0.8164**	25.15, 0.7481	26.38, 0.7997
10%	BSD68	25.33, 0.6996	**25.48, 0.7073**	24.69, 0.6690	25.27, 0.6957
4%	Set11	21.31, 0.6240	**21.83, 0.6359**	20.95, 0.5715	21.47, 0.6106
4%	BSD68	22.17, 0.5569	**22.49, 0.5639**	21.96, 0.5351	22.28, 0.5537
1%	Set11	17.34, 0.4131	17.43, 0.4126	17.46, 0.4064	**17.49**, 0.4112
1%	BSD68	19.17, **0.4198**	**19.21**, 0.4195	19.15, 0.4172	19.21, 0.4191

and Q-PAN$^+$ on natural image recovery and MR image reconstruction from compressed measurements. The results corresponding to PAN$^+$ and Q-PAN$^+$ are reported here. Due to space constraints, the results corresponding to PAN and Q-PAN are reported in the Supplementary Document. For performance assessment and comparison, we use peak signal-to-noise ratio (PSNR) and structural similarity index metric (SSIM) [59]. Several CS techniques [37, 38, 41, 43, 60] are shown to be under-performing in comparison to ISTA-Net and ISTA-Net$^+$ [45]. Therefore we compare the proposed PAN$^+$ and Q-PAN$^+$ models with ISTA-Net$^+$ and quantized ISTA-Net$^+$ (Q-ISTA-Net$^+$) in the context of compressed image reconstruction.
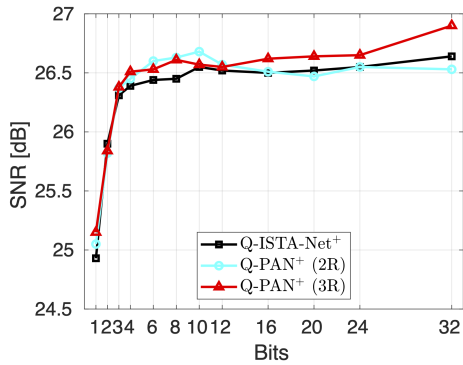
To ensure fair comparison, we use the same set of

Figure 4. Comparative performance of Quantized ISTA-Net$^+$, Q-PAN$^+$ (2R) and Q-PAN$^+$ (3R) on Set11 dataset for CS Ratio = 10 with varying quantization bits

images as recommended in [45] to train our models. The loss function, number of layers in the networks, the size and number of filters learned in the sparsifying function $\mathcal{F}$ are maintained the same for all the networks under consideration. The compressed measurements $\boldsymbol{y}$ are computed as $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}$, where $\boldsymbol{\Phi} \in \mathbb{R}^{m \times n}$ and m varies depending on the compressive sensing (CS) ratio. The measurement matrix $\boldsymbol{\Phi}$ is obtained by orthonormalizing the rows of a random

Gaussian matrix. In training our models, we set the batch size n_b to 64, and the number of layers n_l as 9. In the transform $\mathcal{F}$, $n_f = 32$ filters of size 3×3 are learned in the first convolutional operation. The second convolutional operator learns $n_f = 32$ filters each of size $3 \times 3 \times 32$.

The training data consists of 88,912 cropped image patches each of size 33×33. We trained the network models considering CS ratios of $1\%, 4\%$ and 10% . The training and inference is carried out on a workstation with dual Intel® Xeon® Silver 4110 processors and RTX2080Ti GPU. The models are trained for 100 epochs and take approximately 9 hours. The PAN$^+$ (2R), Q-PAN$^+$ (2R) models employ two regularizers, namely the ℓ_1 penalty and the MC penalty. The convex combination weights are fixed as $\alpha_1 = \alpha_2 = \frac{1}{2}$. The PAN$^+$ (3R), Q-PAN$^+$ (3R) models additionally use the SCAD penalty alongside ℓ_1 and the MCP with $\alpha_1 = \alpha_2 = \alpha_3 = \frac{1}{3}$. The models are tested on the widely used Set11 [41] and BSD68 [61] datasets, which contain 11 and 68 grayscale images respectively. We report the average PSNR and SSIM over the test images in Table 2. To compare the effect of quantization on various models, we trained quantized versions of ISTA-Net$^+$ also. The variation of PSNR with the change in quantization bit-width is captured in Figure 4. We observe that Q-PAN$^+$ (3R) outperforms Q-ISTA-Net$^+$ consistently across different quantizations. The CS reconstructions by the proposed models are shown in Figs. 5 and 6 for visual inspection. From the zoomed-in figure (cf. Fig. 5), one can infer that the PAN$^+$ (3R) model preserves the structure of the ground truth better, in comparison to the benchmark reconstructions. From the reconstruction error images (cf. Fig. 6) from the 1-bit quantized models, one can observe that the Q-PAN$^+$ (3R) model contains fewer artefacts in its reconstruction.

6.1. MR image reconstruction

We evaluate the effectiveness of PAN$^+$ and Q-PAN$^+$ models on the CS MRI problem. The MR image reconstruction methods sample data in the Fourier space and adopt the CS theory to reconstruct images. During the reconstruc-
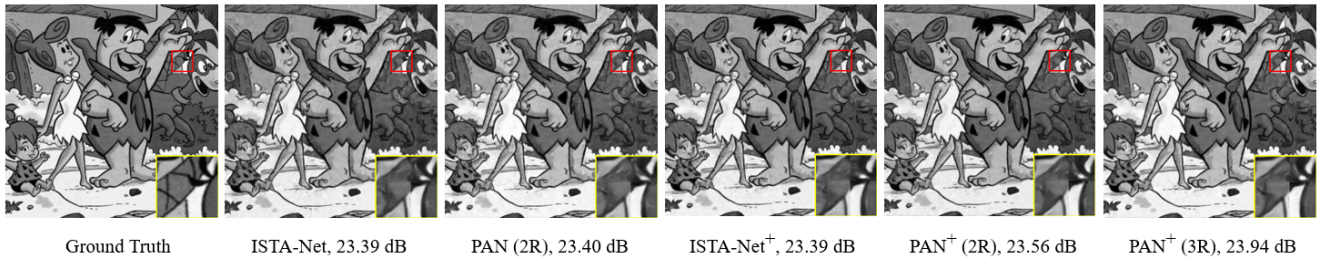
| Ground Truth | ISTA-Net, 23.39 dB | PAN (2R), 23.40 dB | ISTA-Net$^+$, 23.39 dB | PAN$^+$ (2R), 23.56 dB | PAN$^+$ (3R), 23.94 dB |

Figure 5. Reconstruction results for *Flintstones* image based on 10% compressed measurements.

Q-ISTA-Net$^+$ Q-PAN$^+$ (2R)   Q-PAN$^+$ (3R)

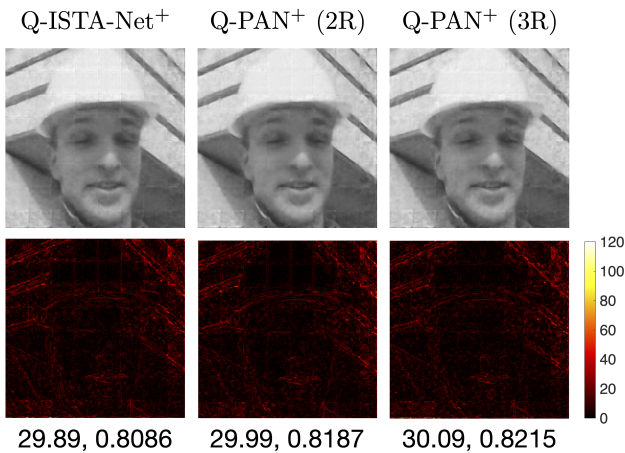29.89, 0.8086   29.99, 0.8187   30.09, 0.8215

Figure 6. Foreman (Set11) image reconstruction for CS ratio 10% by the proposed 1-bit Quantized models. The bottom row visualizes the absolute difference between the ground truth and the reconstructed images. The numbers indicate the PSNR, SSIM values.

Table 3. Comparison of average PSNR [dB] / SSIM on MRI images for different compression ratios.

CS Ratio	ISTA-Net$^+$	PAN$^+$ (3R)	1-bit Q-PAN$^+$ (3R)
20%	38.73, 0.9490	**38.90, 0.9505**	38.17, 0.9400
30%	40.89, 0.9637	**41.02, 0.9640**	40.36, 0.9579
40%	42.52, 0.9729	**42.71, 0.9729**	42.41, 0.9699
50%	44.09, 0.9797	**44.29, 0.9800**	43.84, 0.9769

tion, the sensing matrix is obtained through a combination of under-sampling matrix $\mathbf{P}$ and a discrete Fourier Transform matrix $\mathbf{F}$ as in $\mathbf{\Phi} = \mathbf{PF}$. We use the same training and testing brain medical images as in [45], and train models with $n_l = 11$ layers for CS ratios of $20\%, 30\%, 40\%, 50\%$. The comparison between ISTA-Net$^+$, PAN$^+$ (3R), and 1-bit Q-PAN$^+$ (3R) techniques is summarized in Table 3. The experimental results shows that PAN$^+$ outperforms ISTA-Net$^+$ for all the compression ratios under consideration. Even when extremely quantized 1-bit representations are used, one can observe that the reconstruction performance degradation is at most 0.56 dB.

7. Conclusion

We proposed a multi-penalty formulation for the problem of compressed image reconstruction to learn data-driven analysis prior and developed proximal-averaged iterative shrinkage algorithm for solving it. We then unfolded the iterations to otain a neural network architecture. Making use of the knowledge that natural images are compressible in the residual domain, the enhanced network PAN$^+$ is proposed. We then incorporated a novel learnable quantization strategy into the unfolded networks and showed that the performance degradation even considering the extreme case of 1-bit quantization is less than 1 dB compared with the full precision case. The performance is also superior to the quantized version of ISTA-NET$^+$. The results present strong evidence that unfolded proximal-averaging networks with a quantizer incorporated into the loop in the learning stage offers competitive performance compared with the full-precision case. This makes a strong case for deployment of such networks on low-precision hardware.

References

[1] C. H Zhang, "Nearly unbiased variable selection under minimax concave penalty," *The Annals of Statistics*, vol. 38, no. 2, pp. 894–942, 2010. 1, 3

[2] R. M. Willett, R. F. Marcia, and J. Nichols, "Compressed sensing for practical optical imaging systems: A tutorial," in *IEEE Photonics Conference 2012*, 2012, pp. 586–586. 1

[3] C. Quinsac, A. Basarab, D. Kouamé, and J. Grégoire, "3D Compressed sensing ultrasound imaging," in *2010 IEEE International Ultrasonics Symposium*, 2010, pp. 363–366. 1

[4] M. Jacob, J. C. Ye, L. Ying, and M. Doneva, "Computational MRI: Compressive Sensing and Beyond," *IEEE Signal Processing Magazine*, vol. 37, no. 1, pp. 21–23, 2020. 1

[5] P. Stinco, M. Greco, F. Gini, and M. L. Manna, "Compressed spectrum sensing in cognitive radar systems," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 81–85. 1

[6] Z. Tian and G. B. Giannakis, "Compressed sensing for wideband cognitive radios," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, 2007, vol. 4, pp. IV–1357–IV–1360. 1

[7] R. Rubinstein, T. Peleg, and M. Elad, "Analysis K-SVD: A Dictionary-Learning Algorithm for the Analysis Sparse Model," *IEEE Transactions on Signal Processing*, vol. 61, no. 3, pp. 661–677, 2013. 2

[8] U. S. Kamilov, "A parallel proximal algorithm for anisotropic total variation minimization," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 539–548, 2017. 2

[9] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *Journal of the American Statistical Association*, vol. 96, pp. 1348–1360, 02 2001. 3

[10] M. Courbariaux, Y. Bengio, and J. P. David, "Binaryconnect: Training deep neural networks with binary weights during propagations," in *Advances in neural information processing systems*, 2015, pp. 3123–3131. 2

[11] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks," in *Advances in Neural Information Processing Systems*.

2016, vol. 29, pp. 4107–4115, Curran Associates, Inc. 2

[12] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Quantized neural networks: Training neural networks with low precision weights and activations," *Journal of Machine Learning Research*, vol. 18, no. 187, pp. 1–30, 2018. 2

[13] y. LeCun and c. Cortes, "MNIST handwritten digit database," 2010. 2

[14] Alex Krizhevsky, "Learning multiple layers of features from tiny images," pp. 32–33, 2009. 2

[15] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks," 2016, cite arxiv:1603.05279. 2

[16] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009. 2

[17] S. Zhou, Y. Wu, Z. Ni, X. Zhou, H. Wen, and Y. Zou, "Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients," *arXiv preprint arXiv:1606.06160*, 2016. 2

[18] Z. Dongqing, Y. Jiaolong, Y. Dongqiangzi, and H. Gang, "Lq-nets: Learned quantization for highly accurate and compact deep neural networks," *CoRR*, vol. abs/1807.10029, 2018. 2

[19] L. Zechun, W. Baoyuan, L. Wenhan, Y. Xin, L. Wei, and C. Kwang-Ting, "Bi-Real Net: Enhancing the Performance of 1-bit CNNs with Improved Representational Capability and Advanced Training Algorithm," *CoRR*, vol. abs/1808.00278, 2018. 2

[20] Y. Bai, Y. Wang, and E. Liberty, "Proxquant: Quantized neural networks via proximal operators," *CoRR*, vol. abs/1810.00861, 2018. 2

[21] R. Zhao, Y. Hu, J. Dotzel, C. De Sa, and Z. Zhang, "Improving neural network quantization without retraining using outlier channel splitting," in *Proceedings of the 36th International Conference on Machine Learning*. 09–15 Jun 2019, vol. 97 of *Proceedings of Machine Learning Research*, pp. 7543–7552, PMLR. 2

[22] X. Meng, R. Bachmann, and M. E. Khan, "Training binary neural networks using the Bayesian learning rule," in *Proceedings of the 37th International Conference on Machine Learning*. 13–18 Jul 2020, vol. 119 of *Proceedings of Machine Learning Research*, pp. 6852–6861, PMLR. 2

[23] W. Peiqi, W. Dongsheng, J. Yu, X. Xinfeng, S. Haox-uan, L. XuXin, L. Yongqiang, and X. Yuan, "QGAN: quantized generative adversarial networks," *CoRR*, vol. abs/1901.08263, 2019. 2

[24] D. Wan, F. Shen, L. Liu, F. Zhu, L. Huang, M. Yu, H. T. Shen, and L. Shao, "Deep quantization gener-ative networks," *Pattern Recognition*, vol. 105, pp. 107338, 2020. 2

[25] B. Zhu, Z. Al-Ars, and H. P. Hofstee, "Nasb: Neu-ral architecture search for binary convolutional neural networks," in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–8. 2

[26] K. Duncan, E. Komendantskaya, R. Stewart, and M. Lones, "Relative robustness of quantized neural networks against adversarial attacks," in *2020 Interna-tional Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–8. 2

[27] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009. 2, 3

[28] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse Problems*, vol. 23, no. 3, pp. 947, 2007. 2

[29] Y. Liu, Z. Zhan, J. F. Cai, D. Guo, Z. Chen, and X. Qu, "Projected iterative soft-thresholding algo-rithm for tight frames in compressed sensing magnetic resonance imaging," *IEEE Transactions on Medical Imaging*, vol. 35, no. 9, pp. 2130–2140, 2016. 2

[30] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sens-ing," *in Proceedings of the National Academy of Sciences, U.S.A.*, vol. 106, no. 45, pp. 18914–18919, 2009. 2

[31] T. Plantenga, "A trust region method for nonlin-ear programming based on primal interior-point tech-niques," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 282–305, 1998. 2

[32] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eck-stein, "Distributed optimization and statistical learn-ing via the alternating direction method of multipli-ers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011. 2

[33] Z. Tan, Y. C. Eldar, A. Beck, and A. Nehorai, "Smoothing and decomposition for analysis sparse re-covery," *IEEE Transactions on Signal Processing*, vol. 62, no. 7, pp. 1762–1774, 2014. 2

[34] I. Daubechies, M. Defrise, and C. De Mol, "An itera-tive thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004. 2

[35] Y. Nesterov, "A method for solving the convex pro-gramming problem with convergence rate o$(1/k^2)$," *Proceedings of the USSR Academy of Sciences*, vol. 269, pp. 543–547, 1983. 2

[36] S. Mun and J. E. Fowler, "Block compressed sensing of images using directional transforms," in *2009 16th IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 3021–3024. 2, 3

[37] C. Li, W. Yin, H. Jiang, and Y. Zhang, "An efficient augmented lagrangian method with applications to to-tal variation minimization," *Computational Optimiza-tion and Applications*, vol. 56, pp. 507–530, 12 2013. 2, 3, 7

[38] C. A. Metzler, A. Maleki, and R. G. Baraniuk, "From denoising to compressed sensing," *IEEE Transactions on Information Theory*, vol. 62, no. 9, pp. 5117–5144, 2016. 2, 7

[39] Y. Kim, M. S. Nadar, and A. Bilgin, "Compressed sensing using a gaussian scale mixtures model in wavelet domain," in *2010 IEEE International Con-ference on Image Processing*, 2010, pp. 3365–3368. 2

[40] J. Zhang, C. Zhao, D. Zhao, and W. Gao, "Im-age compressive sensing recovery using adaptively learned sparsifying basis via ℓ_0 minimization," *Sig-nal Processing*, vol. 103, pp. 114–126, 10 2014. 2

[41] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok, "ReconNet: Non-iterative Reconstruc-tion of Images from Compressively Sensed Measure-ments," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 449–458. 3, 7

[42] K. Gregor and Y. LeCun, "Learning fast approxima-tions of sparse coding," *in Proceedings of the 27th International Conference on International Conference on Machine Learning*, 2010. 3

[43] A. Mousavi, A. B. Patel, and R. G. Baraniuk, "A deep learning approach to structured signal recovery," *53rd Annual Allerton Conference on Communication, Con-trol, and Computing*, pp. 1336–1343, 2015. 3, 7

[44] M. Iliadis, L. Spinoulas, and A. K. Katsaggelos, "Deep fully-connected networks for video compressive sensing," *Digital Signal Processing*, vol. 72, pp. 9–18, 2018. 3

[45] J. Zhang and B. Ghanem, "ISTA-Net: Interpretable optimization-inspired deep network for image compressive sensing," *in Proceedings of International Conference on Computer Vision and Pattern Recognition*, pp. 1828–1837, 2018. 3, 4, 5, 7, 8

[46] W. Shi, F. Jiang, S. Liu, and D. Zhao, "Image compressed sensing using convolutional neural network," *IEEE Transactions on Image Processing*, vol. 29, pp. 375–388, 2020. 3

[47] Wuzhen Shi, Feng Jiang, Shaohui Liu, and Debin Zhao, "Scalable convolutional neural network for image compressed sensing," pp. 12282–12291, 2019. 3

[48] P. K. Pokala, Amol G Mahurkar, and Chandra Sekhar Seelamantula, "Firmnet: A sparsity amplified deep network for solving linear inverse problems," pp. 2982–2986, 2019. 3

[49] K. K. R. Nareddy, S. Mache, P. K. Pokala, and C. S. Seelamantula, "An ensemble of proximal networks for sparse coding," pp. 1251–1255, 2022. 3

[50] S. Mache, P. K. Pokala, K. Rajendran, and C. S. Seelamantula, "DuRIN: A deep-unfolded sparse seismic reflectivity inversion network," *CoRR*, vol. abs/2104.04704v2, 2021. 3

[51] S. Mache, P. K. Pokala, K. Rajendran, and C. S. Seelamantula, "*Nu*SPAN: A proximal average network for nonuniform sparse model — application to seismic reflectivity inversion," *CoRR*, vol. abs/2105.00003v2, 2021. 3

[52] M. A. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak, "Majorization-minimization algorithms for wavelet-based image restoration," *IEEE Transactions on Image Processing*, vol. 16, no. 12, pp. 2980–2991, 2007. 4

[53] L. W. Zhong and J. T. Kwok, "Gradient descent with proximal average for nonconvex and composite regularization," *Twenty Eighth AAAI Conference on Artificial Intelligence*, vol. 3, pp. 2206–2212, 01 2014. 4

[54] V. Nair and G. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," *in Proceedings of the International Conference on Machine Learning*, vol. 27, pp. 807–814, 06 2010. 4

[55] J. Zhang, D. Zhao, and W. Gao, "Group-based sparse representation for image restoration," *IEEE Transactions on Image Processing*, vol. 23, no. 8, pp. 3336–3351, 2014. 4

[56] G. K. Wallace, "The JPEG Still Picture Compression Standard," *Communications of the ACM*, vol. 34, pp. 31–44, April 1991. 5

[57] G. J. Sullivan, J. R. Ohm, W. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard.," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012. 5

[58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. 5

[59] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004. 7

[60] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *in Proceedings of International Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2808–2817. 7

[61] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *ICCV*, 2001, vol. 2, pp. 416–423 vol.2. 7