QISTA-NET: DNN ARCHITECTURE TO SOLVE $\ell_q$-NORM MINIMIZATION PROBLEM

Gang-Xuan Lin and Chun-Shien Lu

Institute of Information Science
Academia Sinica, Taiwan, ROC.

ABSTRACT

In this paper, we reformulate the non-convex $\ell_q$-norm minimization problem with $q \in (0, 1)$ into a 2-step problem, which consists of one convex and one non-convex subproblems, and propose a novel iterative algorithm called QISTA ($\ell_q$-ISTA) to solve the ($\ell_q$)-problem. By taking advantage of DNN in accelerating optimization algorithms, we also design a DNN architecture associated with QISTA, called QISTA-Net, which is then further speeded up as QISTA-Net+ using the momentum from all previous layers. Extensive experimental comparisons demonstrate that the proposed methods yield better reconstruction qualities than state-of-the-art $\ell_1$-norm optimization (plus learning) algorithms even if the original sparse signal is noisy.

Index Terms— Compressed sensing, $\ell_q$-norm regularization problem, Non-convex optimization, Deep learning

1. INTRODUCTION

1.1. Background and Problem Definition

In sparse signal recovery like compressive sensing (CS) [1, 2], we usually let $x_0 \in \mathbb{R}^n$ denote a $k$-sparse signal to be sensed, let $A \in \mathbb{R}^{m \times n}$ represent a sampling matrix, and let $y \in \mathbb{R}^m$ be the measurement vector defined as

$y = Ax_0,$

where $k < m < n$ and $0 < \frac{m}{n} < 1$ is defined as the measurement rate. At the decoder, $x_0$ can be recovered based on its sparsity by means of solving $\ell_0$-norm regularization problem:

$$\min_{x_0} \frac{1}{2} \| y - Ax_0 \|^2 + \lambda \| x_0 \|_0,$$  \hspace{1cm} (1)

where $\lambda > 0$ is a regularization parameter.

The ($\ell_0$)-problem is NP-hard [3] as it suffers from the non-convexity and discontinuity of objective function so that there is no efficient algorithm to solve its global minima. An effective way to recover the original sparse signal $x_0$ is relaxing the objective function in (1) as the $\ell_1$-norm regularization problem, which is known as “LASSO” [4, 5]:

$$(\text{LASSO}) \min_{x} \frac{1}{2} \| y - Ax \|^2 + \lambda \| x \|_1.$$  \hspace{1cm} (2)

Nevertheless, considering LASSO cannot recover the original sparse signal under low measurement rates (say $m < 3k$) [6], $\ell_q$-norm regularization is suggested [6, 7]. The (non-convex) $\ell_q$-norm regularization problem has the form

$$\min_{x} \frac{1}{2} \| y - Ax \|^2 + \lambda \| x \|_q^q,$$  \hspace{1cm} (3)

where $0 < q < 1$, and $\| x \|_q = \sum_{i=1}^{n} (|x_i|^q)^{1/q}$ is the $\ell_q$-quasi-norm (which is usually called $\ell_q$-norm). In comparison with ($\ell_0$)-problem and LASSO, the authors concluded that decreasing $q$ further decreases the required measurement rate and by less and less as $q$ gets smaller [6, 7].

It is noted that the discussions regarding ($\ell_q$)-problem or effective algorithms in finding its optimal solution are very rare in the literature. Furthermore, ($\ell_q$)-problem is also NP-hard [8], and literature review reveals that solving ($\ell_q$)-problem suffers from non-convexity, leading to a local-non-global optimal solution. Although it is difficult to find the global optimal solution, under good initial iterative point, the limit point gained by iterative algorithms converging to a local-non-global optimal solution is still closer to $x_{\ell_0}^*$ of ($\ell_0$)-problem than $x_{1}^*$ of LASSO [9].

1.2. Related Works

Traditionally, one always approximates the optimal solution to ($\ell_0$)-problem and LASSO, by employing proximal gradient descent method (PGD), which is also known as iterative hard-threshold algorithm (IHT) [10] and iterative soft-threshold algorithm (ISTA) [5]. However, IHT could obtain better reconstruction quality than ISTA only if the original signal is very sparse ($k/n < 5\%$) and/or measurement rate is high ($m/n > 50\%$) [11, 12]. The use of PGD to solve ($\ell_q$)-problem is not popular because there is no closed-form solution to proximal operator associated with its regularization term [11]. Beck and Teboulle speed up ISTA by using
Nesterov’s acceleration method (insert momentum after gradient descent step), which is known as FISTA [13], whereas Donoho et al. consider an efficient algorithm called AMP that incorporates ISTA with Onsager term in measurement residue $y - Ax$ [14]. AMP is a very fast reconstruction process with high performance, but the assumptions therein are impractical.

On the other hand, in solving the $(\ell_q)$-problem, Cui et al. [15] propose to utilize the iterative thresholding (IT) algorithm in finding the global optimal solution of surrogate function. Xu et al. [12] design a half-thresholding algorithm by thresholding representation theory to solve the $(\ell_q)$-problem when $q = 1/2$. Cao et al. [16] deduce the thresholding formula in [12] to derive the extension thresholding formula, which can solve the $(\ell_q)$-problem when $q = 2/3$. However, most of the algorithms still suffer from the non-convexity of $\ell_q$-regularized term, leading to a local-non-global optimal solution, though such a solution results in better reconstruct performance than IHT and ISTA, which solve $(\ell_q)$-problem and LASSO, respectively. Moreover, [12] and [16] restrict the choice of $q (= 1/2$ or $= 2/3$) and the other aforementioned methods have to tune an appropriate $q$ to get better results, which violates the fact that $q$ should be small [6].

In this paper, we derive an algorithm solving a non-convex $(\ell_q)$-problem, which obtains reasonable result associated with suggestion of $(\ell_q)$-problem in that the smaller $q$ leads to the better reconstruction performance.

1.3. Contributions

1. We reformulate $(\ell_q)$-norm minimization problem into 2-step problem that transfers the difficulty coming from non-convexity to another non-convex optimization problem that can be trivially solved. Then we design an algorithm called QISTA that approximates the optimal solution of $(\ell_q)$-problem precisely.

2. QISTA-Net is a DNN architecture by unfolding specific parameters in QISTA to accelerate the reconstruction. We also propose to utilize the momentum coming from all previous layers to further speed up QISTA-Net as QISTA-Net$^+$. The use of momentum in this paper has never been found in literature.

3. The performance of QISTA-Net$^+$ is better than state-of-the-art $\ell_1$-norm DNN methods, even in noisy environments.

2. PROPOSED METHOD

We first describe the proposed QISTA algorithm in Sec. 2.1 and its network version in Sec. 2.2.

2.1. Iterative Method for Solving $(\ell_q)$-Problem

In Sec. 2.1.1, we first approximate the $(\ell_q)$-problem and reformulate it into the 2-step problem. We then propose an iterative algorithm for solving the $(\ell_q)$-problem in Sec. 2.1.2.

2.1.1. Reformulate $(\ell_q)$-Problem as 2-Step Problem

To solve $(\ell_q)$-problem, it is first approximated as

$$\min_x F(x) = \frac{1}{2} ||y - Ax||_2^2 + \lambda \sum_{i=1}^{n} \frac{|x_i|}{(|c_i| + \varepsilon_i)^{1-q}}, \tag{4}$$

where $\varepsilon_i > 0$ for all $i \in [1 : n]$. We can see that the objective function in (4) is equivalent to the one in (3) provided $\varepsilon_i = 0$

$$\lim_{\varepsilon_i \to 0^+} \frac{|x_i|}{(|c_i| + \varepsilon_i)^{1-q}} = |x_i|^q.$$

This means the problem (4) approximates to the $(\ell_q)$-problem (3) well if $\varepsilon_i$’s are small enough.

Second, we extend $F(x)$ in the problem (4) into high-dimensional functional $H(x, c)$, then relax the problem (4) (in the sense of feasible set from $\mathbb{R}^n$ to $\mathbb{R}^n \times \mathbb{R}^n$) into

$$\min_{x, c} H(x, c) = \frac{1}{2} ||y - Ax||_2^2 + \lambda \sum_{i=1}^{n} \frac{|x_i|}{(|c_i| + \varepsilon_i)^{1-q}}. \tag{5}$$

We can see that the functional $H(x, c)$ degenerates to $F(x)$ if $c = x$. Thus, we can reformulate the problem (4) as a 2-step problem:

$$\begin{cases}
\min_{\bar{x}} H(\bar{x}, \tilde{c}) = \frac{1}{2} ||y - A\bar{x}||_2^2 + \lambda \sum_{i=1}^{n} \frac{|\bar{x}_i|}{(|\tilde{c}_i| + \varepsilon_i)^{1-q}}, \\
\min_{c} |H(\bar{x}, c) - H(\bar{x}, \tilde{c})|,
\end{cases} \tag{6}$$

where $\bar{x}$ and $\tilde{c}$ are optimal solutions to the first problem (called $x$-subproblem) and the second problem (called $c$-subproblem), respectively.

**Theorem 2.1.** If $(x^*, c^*)$ is an optimal solution pair to 2-step problem (6), then $x^*$ is an optimal solution to problem (4), and vice versa.

**Proof.** Let $(x^*, c^*)$ be an optimal solution pair to (6), since the optimal value of $c$-subproblem is obviously 0, we have $H(x^*, c^*) = H(x^*, x^*)$, which equals to $F(x^*)$ (because $H(x, x) = F(x)$), therefore $x^*$ is an optimal solution to (4).

On the other hand, let $x^*$ be an optimal solution to (4), then $c^* = x^*$ is an optimal solution to $c$-subproblem of (6), whereas $x$-subproblem is exactly equivalent to (4). ■

We can see that the $c$-subproblem has global minimum solution

$$c^* = \bar{x}, \tag{7}$$

whereas the $x$-subproblem is in a weighted-LASSO form

$$\min_{x} \frac{1}{2} ||y - Ax||_2^2 + \lambda \sum_{i=1}^{n} |w_i x_i|,$$
(the weight of $|x_i|$ is $\frac{\lambda}{(|x_i| + \epsilon_i)^q}$), and thus the $x$-subproblem can be solved iteratively by proximal gradient descent algorithm \cite{17} as

$$\begin{cases}
  r^t = x^t + \beta A^T (y - Ax^t) \\
  x^{t+1} = \eta (r^t; \theta),
\end{cases}$$

where $\theta_i = \frac{\beta \lambda}{(|x_i|^q + \epsilon_i)^{1-q}}$, $\forall i$, (when we solve the $x$-subproblem, $c$ is a given constant vector, so $\theta$ is fixed in each iteration) $\eta (\cdot; \cdot)$ is soft-thresholding operator ($\eta (x; \theta) = \text{sign} (x) \cdot \max \{0, |x| - \theta\}$).

2.1.2. QISTA

To solve the 2-step problem (6), we propose an iterative process (see Algorithm 1), which iterates alternately through Eqs. (7) and (8) as

$$\begin{cases}
  c^t = x^t \\
  r^t = x^t + \beta A^T (y - Ax^t) \\
  x^{t+1} = \eta (r^t; \theta),
\end{cases}$$

which can be merged into

$$\begin{cases}
  r^t = x^t + \beta A^T (y - Ax^t) \\
  x^{t+1} = \eta (r^t; \theta),
\end{cases}$$

We call it QISTA ($\ell_q$-ISTA).

Algorithm 1 QISTA

1: Set parameters $\beta, \lambda, \text{TOL}$; 
2: initial $x^0 = x_{-1} \in \mathbb{R}^n$; 
3: repeat 
4: $r^t = x^t + \beta A^T (y - Ax^t)$; 
5: $x^{t+1} = \eta (r^t; \theta), \forall i \in [1:n]$; 
6: until $\|x^t - x^{t-1}\|_2 < \text{TOL}$

2.1.3. Remarks

Basically, in QISTA, we first approximate the ($\ell_q$)-problem by (4), and then reformulate it into the 2-step problem (6). Since the objective function $F(x)$ in (4) is non-convex, it is difficult to attain the global minima; whereas the functional $H(x, c)$ in (6) is convex in $x$ for any given $c$, but non-convex in $c$ for any given $x$. However, since the $x$-subproblem is convex, the proximal gradient descent algorithm (8) to find its optimal solution is global-convergence under mild parameter setting \cite{13}. On the other hand, although the $c$-subproblem is non-convex, the non-convexity can be avoided because there is a trivial global minimum solution, i.e., $c^* = \bar{x}$.

2.2. QISTA-Net

Similar to \cite{18, 19, 20}, we also design a deep neural network (DNN) architecture to accelerate QISTA, which is called QISTA-Net. The feed-forward part of QISTA-Net is shown in Algorithm 2. The learning parameters are $\{A^t, \lambda^t, \mathcal{E}^t\}_{t=1}^T$.

Algorithm 2 QISTA-Net

1: for $t = 1$ to $T$ do 
2: $r^t = x^{t-1} + A^t (y - Ax^{t-1})$; 
3: $x^t_i = \eta (r^t_i; \frac{\beta \lambda}{(|x^t_i|^{q-1} + \mathcal{E}^t_i)^{1-q}}) \forall i \in [1:n]$ 
4: end for

Remark that step 2 in Algorithm 2 is corresponding to gradient descent step (as in step 4 of Algorithm 1), which unfold both $A$ and $\beta A^T$ to be learning parameters is commonly adopted in LISTA [19], LAMP [18], and other DNN models. According to Theorem 1 in [20], we only set $\beta A^T$ to be a learning parameter and keep $A$ as the original matrix to reduce the training time without loss of performance.

2.3. QISTA-Net$^+$

Motivated by the acceleration techniques in FISTA \cite{13} and AMP \cite{14} in that the gradient descent step in FISTA follows the previous iterative direction called momentum, and the residue in measurement domain in AMP follows the previous iterative residue called Onsager term, we extend QISTA-Net to QISTA-Net$^+$ (see Algorithm 3) by adding the momentum coming from the descent direction of all previous layers. As shown in Algorithm 3, $D^t$ in Step 2 is the descent direction of the current layer $t$, $\sum_{j=1}^{t-1} D^j$ in Step 3 is the momentum consisting of the descent directions of previous layers, and Step 4 controls the effect of momentum coming from all previous layers appropriately. However, unlike the acceleration of traditional iterative methods such as FISTA [13], the improvement effect in QISTA-Net$^+$ can only draw empirical conclusion without being able to conduct mathematical analysis. Moreover, in Algorithm 3, the learning parameters are $\{A^t, \lambda^t, \mathcal{E}^t\}_{t=1}^T$.

Algorithm 3 QISTA-Net$^+$

1: for $t = 1$ to $T$ do 
2: $D^t = A^t (y - Ax^{t-1})$; 
3: $r^t = x^{t-1} + \sum_{j=1}^{t-1} D^j$; 
4: $D^j = \frac{\alpha}{m} \cdot D^j, \forall j \in [1:t]$ 
5: $x^t_i = \eta (r^t_i; \frac{\beta \lambda^t}{(|x^t_i|^{q-1} + \mathcal{E}^t_i)^{1-q}}) \forall i \in [1:n]$ 
6: end for
3. EXPERIMENTAL RESULTS

Our experiments were conducted on NVIDIA GeForce GTX 1060 GPU, Python 3.6 with Pytorch version 0.4.1, with two kinds of performance comparison: conventional iterative optimization style (Sec. 3.2.1) and deep learning-based style (Sec. 3.2.2).

3.1. Parameter Setting

For a fair comparison, in Sec. 3.2.1, we followed the same setting as in [15] that the problem dimensions were \( n = 1024 \) and \( m = 256 \), and the ground-truth \( x_0 \in \mathbb{R}^n \) was a \( k \)-sparse signal, where the non-zero entries followed i.i.d. Gaussian distribution \( \mathcal{N}(0,1) \). For the sensing matrix \( A \in \mathbb{R}^{m \times n} \), its entries \( A_{i,j}'s \) followed i.i.d. Gaussian distribution \( \mathcal{N}(0,1) \) (without column normalization). In Sec. 3.2.2, we followed the same setting as in [18, 20, 21] that \( n = 500, m = 250 \), and the entries of input \( (k \)-sparse signal) \( x_0 \in \mathbb{R}^n \) followed i.i.d. Gaussian distribution \( \mathcal{N}(0,1) \) with probability 10\% (that is \( x_0 \) is Bernoulli-Gaussian with \( k \approx n \times 10\% = 50 \)). For the sensing matrix \( A \in \mathbb{R}^{m \times n} \), its entries \( A_{i,j}'s \) followed i.i.d. Gaussian distribution with column normalization \( \mathcal{N}(0, \frac{1}{m}) \).

The parameters in QISTA were \( \beta = \frac{1}{\|A\|_2^2} \), where \( \|A\|_2 \) is the spectral norm of \( A \), \( \lambda = 10^{-4}, q = 0.05 \), and \( \varepsilon = I_n \), where \( I_n \) is a vector in \( \mathbb{R}^n \) with each component being equal to 1. The parameters in QISTA-Net and QISTA-Net\(^+ \) were \( \beta = \frac{1}{\|A\|_2^2}, q = 0.05 \), and \( \gamma = 0.1 \). The training parameters of QISTA-Net and QISTA-Net\(^+ \) were initialized as \( \lambda^t = 10^{-4}, A^t = \beta A^T, \) and \( \mathcal{E}_i^t = 0.1 \cdot I_n \). Moreover, since \( \mathcal{E}_i^t \) plays the same role with \( \varepsilon_i \) in (4), and the value of \( \mathcal{E}_i^t \) may be negative after doing back-propagation, we further restrict \( \mathcal{E}_i^t \) after each back-propagation to remain positive by letting \( \mathcal{E}_i^t = \max \{ \mathcal{E}_i^t, 0.1 \} \).

3.2. Performance Comparison

3.2.1. Traditional Iterative Methods

We compare the proposed method, QISTA, with traditional iterative methods IHT [10], FISTA [13] (ISTA [5] was not included because both ISTA and FISTA have exactly the same reconstruction performance), half thresholding algorithm [12], 2/3 algorithm [16], and \( 1/2 - \varepsilon \) algorithm [15]. The criterion, declaring a successful perfect reconstruction of the ground-truth if the relative error \( RE = \frac{\|x^* - x_{old}\|_2}{\|x^*\|_2} \leq 10^{-4} \) holds [15], was adopted. In Fig. 1, the results were shown in terms of the success rate averaged at 20 tests vs. sparsity \( k \). We can see that QISTA can perfect reconstruct the ground-truth until \( k \) is 94 (in this case, \( 3k = 282 > m \)) but LASSO fails to recover the ground-truth [6]. The \( 1/2 - \varepsilon \) algorithm

3.2.2. Deep Learning Methods

We first show the performance comparison between QISTA-Net (Algorithm 2) and QISTA-Net\(^+ \) (Algorithm 3) in Fig. 2. The results are presented in an average of 100 tests. We can see that, under the same SNR values, QISTA-Net\(^+ \) needs fewer layers than QISTA-Net, implying that QISTA-Net\(^+ \) indeed is speeded up from QISTA-Net.

Second, in Fig. 3, we demonstrate the reconstruction performance of QISTA-Net\(^+ \) with respect to \( q \). We observe that reconstruction quality is increased when \( q \) is decreased. This indicates that QISTA-Net\(^+ \) better approaches the \( \ell_0 \)-norm minimization problems than its \( \ell_1 \)-norm counterpart.

Third, we conducted comparisons of proposed methods with several known DNN methods, including LAMP tied [18] and LAMP untied [18], LISTA-CP [20], LISTA-SS [20], and LISTA-CPSS [20], ALISTA [21] and TiLISTA [21], and

---

\(^1\)Our Python implementation codes can be downloaded from https://github.com/spybeiman/QISTA/
QISTA-Net$^+$ in reconstructing the exactly $k$-sparse ground-truth, with measurement rate 50% ($n = 500, m = 250$), under various $q$’s.

In Fig. 4, we show the performance comparison between QISTA-Net$^+$ and the other state-of-the-art $\ell_1$-based DNN methods, in reconstructing the exactly $k$-sparse ground-truth ($n = 500, m = 250$, and $k \approx 50$). In addition, the performance comparison of reconstructing the exactly $k$-sparse ground-truth under measurement rate 30% ($n = 500, m = 150$, and $k \approx 50$) is shown in Fig. 5. In Fig. 6, we demonstrate the performance comparison in reconstructing the ground-truth with measurement noise at SNR=20dB.

In summary, we can see from Fig. 4, Fig. 5, and Fig. 6 that QISTA-Net$^+$ outperforms all the other existing works in reconstruction quality. Moreover, in Fig. 5, all the $\ell_1$-based DNN methods used for comparison only achieve a maximum of 17dB (at 16th layer). We conjecture that this is because, for
the $\ell_1$-based iterative methods, $m$ must be greater than $3k$ to achieve a good reconstruction performance [6]. Furthermore, Fig. 7 actually indicates that QISTA-Net+ offers state-of-the-art performance in terms of reconstruction quality and speed.

4. CONCLUSION AND FUTURE WORK

In this paper, we first reformulate the $\ell_q$-norm minimization problem into a 2-step problem (6), which excludes the difficulties coming from the non-convexity. We then propose QISTA to solve the $(\ell_q)$-problem via the $\ell_1$-norm based iterative algorithm. Moreover, with the help of DNN, we propose an $\ell_q$-based DNN method, QISTA-Net. Finally, by employing the strategy in Algorithm 3, we further speed up QISTA-Net as QISTA-Net+. The resultant QISTA-Net+ retains better reconstruction performance and faster reconstruction speed than state-of-the-art $\ell_1$-norm DNN methods, even if the original sparse signal is noisy.

5. REFERENCES

[1] E. J. Candès, J. Romberg, and T. Tao, “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information,” *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[2] D. L. Donoho, “Compressed sensing,” *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[3] B. K. Natarajan, “Sparse approximate solutions to linear systems,” *SIAM J. Comput.*, vol. 24, pp. 227–234, 1995.

[4] J. L. Starck, D. L. Donoho, and E. J. Candès, “Astronomical image representation by the curvelet transform,” *Astron. Astrophys.*, vol. 398, pp. 785–800, 2003.

[5] I. Daubechies, M. Defrise, and C. De Mol, “An iterative thresholding algorithm for linear inverse problems with a sparsity constraint,” *Commun. Pure Appl. Math.*, vol. 57, no. 11, pp. 1413–1457, 2004.

[6] R. Chartrand, “Exact reconstruction of sparse signals via nonconvex minimization,” *IEEE Signal Process. Lett.*, vol. 14, no. 10, pp. 707–710, 2007.

[7] R. Chartrand and W. Yin, “Iteratively reweighted algorithms for compressive sensing,” in *Proc. Int. Conf. Acoust., Speech, Signal Process.* (ICASSP), 2008.

[8] D. Ge, X. Jiang, and Y. Ye, “A note on the complexity of $\ell_p$ minimization,” *Math. Program. Series B*, vol. 129, no. 2, pp. 285–299, 2011.

[9] L. Zheng, A. Maleki, H. Weng, X. Wang, and T. Long, “Does $\ell_q$-minimization outperform $\ell_1$-minimization?,” *IEEE Trans. Inf. Theory*, vol. 63, no. 11, pp. 6896–6935, 2017.

[10] T. Blumensath and M. E. Davies, “Iterative thresholding for sparse approximations,” *J. Fourier Anal. Appl.*, vol. 14, no. 5-6, pp. 629–654, 2008.

[11] F. Wen, L. Chu, P. Liu, and R. C. Qiu, “A survey on non-convex regularization-based sparse and low-rank recovery in signal processing, statistics, and machine learning,” *IEEE Access*, vol. 6, pp. 69883–69906, Nov. 2018.

[12] Z. Xu, X. Chang, F. Xu, and H. Zhang, “$l_{1/2}$ regularization: A thresholding representation theory and a fast solver,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 7, pp. 1013–1027, 2012.

[13] 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.

[14] D. L. Donoho, A. Maleki, and A. Montanari, “Message-passing algorithms for compressed sensing,” *Proc. Nat. Acad. Sci. U.S.A.*, vol. 106, no. 45, pp. 18914–18919, 2009.

[15] A. Cui, J. Peng, H. Li, M. Wen, and J. Jia, “Iterative thresholding algorithm based on non-convex method for modified $l_p$-norm regularization minimization,” *J. Comput. Appl. Math.*, vol. 347, pp. 173–180, Feb. 2019.

[16] W. Cao, J. Sun, and Z. Xu, “Fast image deconvolution using closed-form thresholding formulas of $l_q(q = \frac{1}{2}, \frac{2}{3})$ regularization,” *J. Vis. Commun. Image Representation*, vol. 24, no. 1, pp. 31–41, 2013.

[17] A. Beck, *First-Order Methods in Optimization*, MOS-SIAM Ser. Optim., 2017.

[18] M. Borgerding, P. Schniter, and S. Rangan, “Amplitude-modulated deep networks for sparse linear inverse problems,” *IEEE Trans. Signal Process.*, vol. 65, no. 16, pp. 4293–4308, Aug. 2017.

[19] K. Gregor and Y. LeCun, “Learning fast approximations of sparse coding,” in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2010.

[20] X. Chen, J. Liu, Z. Wang, and W. Yin, “Theoretical linear convergence of unfolded ISTA and its practical weights and thresholds,” in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2018.

[21] J. Liu, X. Chen, Z. Wang, and W. Yin, “ALISTA: Analytic weights are as good as learned weights in LISTA,” in *Proc. Int. Conf. Learn. Representations (ICLR)*, 2019.

[22] D. Ito, S. Takabe, and T. Wadayama, “Trainable ISTA for sparse signal recovery,” *IEEE Trans. Signal Process.*, vol. 67, no. 12, pp. 3113–3125, 2019.