

# A Generalized Iterated Shrinkage Algorithm for Non-convex Sparse Coding

Wangmeng Zuo<sup>1,3</sup>, Deyu Meng<sup>2</sup>, Lei Zhang<sup>3</sup>, Xiangchu Feng<sup>4</sup>, David Zhang<sup>3</sup>

<sup>1</sup>Harbin Institute of Technology <sup>2</sup>Xi'an Jiaotong University

<sup>3</sup>Hong Kong Polytechnic University <sup>4</sup>Xidian University

cswmzuo@gmail.com, dymeng@mail.xjtu.edu.cn, {cslzhang, csdzhang}@comp.polyu.edu.hk, xcfeng@mail.xidian.edu.cn

## Abstract

In many sparse coding based image restoration and image classification problems, using non-convex  $\ell_p$ -norm minimization ( $0 \leq p < 1$ ) can often obtain better results than the convex  $\ell_1$ -norm minimization. A number of algorithms, e.g., iteratively reweighted least squares (IRLS), iteratively thresholding method (ITM- $\ell_p$ ), and look-up table (LUT), have been proposed for non-convex  $\ell_p$ -norm sparse coding, while some analytic solutions have been suggested for some specific values of  $p$ . In this paper, by extending the popular soft-thresholding operator, we propose a generalized iterated shrinkage algorithm (GISA) for  $\ell_p$ -norm non-convex sparse coding. Unlike the analytic solutions, the proposed GISA algorithm is easy to implement, and can be adopted for solving non-convex sparse coding problems with arbitrary  $p$  values. Compared with LUT, GISA is more general and does not need to compute and store the look-up tables. Compared with IRLS and ITM- $\ell_p$ , GISA is theoretically more solid and can achieve more accurate solutions. Experiments on image restoration and sparse coding based face recognition are conducted to validate the performance of GISA.

## 1. Introduction

Sparse coding [7, 18, 31] is an effective tool in a myriad of applications such as compressed sensing [11], image restoration [24, 25], face recognition [38], etc. Originally, it aims to solve the following minimization problem:

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_0, \quad (1)$$

where  $\mathbf{y}$  is an  $n \times 1$  vector,  $\mathbf{A}$  is an  $n \times m$  redundant matrix with  $m > n$ , and the  $\ell_0$ -norm  $\|\bullet\|_0$  simply counts the number of non-zero entries in  $\mathbf{x}$ . Unfortunately, solving the minimization in Eq. (1) is NP hard [32] and is computationally infeasible for large scale problems.

Rather than solving the above  $\ell_0$ -minimization problem, one can replace the  $\ell_0$ -norm with the  $\ell_1$ -norm  $\|\bullet\|_1 = \sum_i |x_i|$ , and seek for the desired  $\mathbf{x}$  by solving the following convex

optimization problem

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1. \quad (2)$$

It has been proved that, under certain conditions on  $\mathbf{A}$  [6, 17], the  $\ell_1$ -minimization in (2) is equivalent to the  $\ell_0$ -minimization in (1) with high probability.

However, when the conditions on  $\mathbf{A}$  are not satisfied, the solution by  $\ell_1$ -minimization becomes suboptimal. Actually, both theoretical analysis and numerical experiments [9, 10, 11] have shown that the solution of  $\ell_p$ -norm sparse coding ( $0 \leq p < 1$ )

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_p^p, \quad (3)$$

is close to that of the  $\ell_1$ -minimization and it is sparser. In image restoration, it has been shown that the image gradients of the natural images can be better modeled with hyper-Laplacian distribution with  $0.5 \leq p \leq 0.8$  [25, 28]. In feature selection and compressed sensing,  $\ell_p$  can bridge  $\ell_0$  and  $\ell_1$ , and can achieve better solutions [12, 31].

So far, a number of algorithms have been proposed for solving  $\ell_p$ -norm non-convex sparse coding problems, and they have been applied to various vision and learning tasks, e.g., compressed sensing [10], image restoration [25], face recognition [29], and variable selection [33]. Several typical algorithms include iteratively reweighted least squares (IRLS) [12, 14, 23, 24, 28], iteratively reweighted  $\ell_1$ -minimization (IRL1) [8], iteratively thresholding method (ITM- $\ell_p$ ) [33, 34], and look-up table (LUT) [25]. These algorithms, however, suffer from several limitations. Even for the simplest  $\ell_p$ -minimization problem

$$\min_x \frac{1}{2} (y - x)^2 + \lambda |x|^p, \quad (4)$$

IRLS, IRL1, and ITM- $\ell_p$  would not converge to the global optimal solution. LUT uses look-up tables to store the solutions w.r.t. different values of variable  $x$  and regularization parameter  $\lambda$ . If the values of  $x$  and  $\lambda$  are unconstrained and  $p$  changes dynamically (e.g., multi-stage relaxation), more computational and memory costs are required to construct and store the look-up table. Other algorithms, such as the analytic solutions in [25, 39], can only be used for some specific values of  $p$ .

Inspired by the great success of soft thresholding [16] and iterative shrinkage/thresholding (IST) [15] methods, in this paper, we propose a generalized iterated shrinkage algorithm (GISA) for  $\ell_p$ -norm non-convex sparse coding. The proposed GISA is simple and efficient, and can be adopted for solving  $\ell_p$ -norm sparse coding problems with arbitrary  $p$ ,  $\lambda$  and  $y$  values. Compared with IRLS, IRL1, and ITM- $\ell_p$ , GISA would converge to more accurate solutions. It is easy to implement and can be readily used to solve the many  $\ell_p$ -norm minimization problems in various vision and learning applications.

## 2. Related work

To date, various algorithms have been proposed for  $\ell_p$ -norm non-convex sparse coding. Based on the problems in Eq. (3) and Eq. (4), we provide a brief survey and discussion on IRLS, IRL1, ITM- $\ell_p$ , and LUT.

To use IRLS for  $\ell_p$ -norm non-convex sparse coding, the problem in Eq. (3) is approximated by [26]

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \sum_i (x_i^2 + \varepsilon)^{p/2-1} x_i^2, \quad (5)$$

where  $\varepsilon \rightarrow 0$  is a small positive number to avoid division by zeros. Given the current estimation  $\mathbf{x}^{(k)}$ , IRLS iteratively solves the following problem

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \sum_i w_i x_i^2, \quad (6)$$

and updates  $\mathbf{x}$  by

$$\mathbf{x}^{(k+1)} = (\mathbf{A}^T \mathbf{A} + \text{diag}(\mathbf{w}))^{-1} \mathbf{A}^T \mathbf{y}, \quad (7)$$

where the  $i^{\text{th}}$  component of weight vector  $\mathbf{w}$  is defined as

$$w_i = p\lambda \left( (x_i^{(k)})^2 + \varepsilon \right)^{1-p/2}. \quad (8)$$

Similarly, to use IRL1 for  $\ell_p$ -norm minimization, the problem in Eq. (3) is approximated by

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \sum_i \lambda p (|x_i| + \varepsilon)^{p-1} |x_i|. \quad (9)$$

Given  $\mathbf{x}^{(k)}$ , IRL1 [8, 21] updates  $\mathbf{x}$  by solving the following problem

$$\mathbf{x}^{(k+1)} = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \sum_i \lambda p (|x_i^{(k)}| + \varepsilon)^{p-1} |x_i| \quad (10)$$

using the existing  $\ell_1$ -minimization algorithms [1, 3, 41]. Based on the theoretical analysis in [20, 26], both IRLS and IRL1 can guarantee to converge, while Chartland and Yin [12] showed that IRLS is theoretically better than IRL1.

However, even for the simplest  $\ell_p$ -minimization problem in Eq. (4), IRLS and IRL1 sometimes cannot converge to the desired solutions. As shown in Fig. 1, given  $p = 0.5$ ,  $\lambda = 1$ , and  $y = 1.3$ , by initializing  $x^{(0)} = y$ , IRLS and IRL1 would converge to the same local minimum. Since the problem in Eq. (4) is for 1D optimization, one can define a proper thresholding function [33] or construct look-up tables (LUTs) [25] in advance. For several special values of

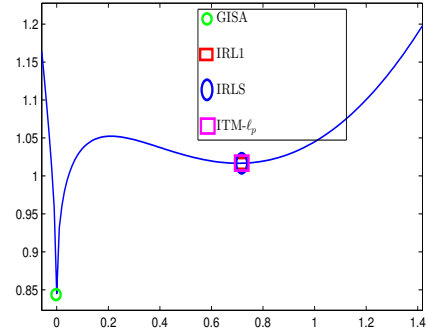


Figure 1. The solutions of GISA, IRL1, IRLS, and ITM- $\ell_p$  for solving the problem in Eq. (4) with  $p = 0.5$ ,  $\lambda = 1$ , and  $y = 1.3$ . IRL1, IRLS, and ITM- $\ell_p$  converge to the same local minimum, but GISA can converge to a better solution.

$p$ , e.g.,  $1/2$  or  $2/3$ , the analytic solutions can be derived [25, 39]. She [33] defined the following  $\ell_p$ -norm thresholding function

$$T_p^{ITM}(y; \lambda) = \begin{cases} 0, & \text{if } |y| \leq \tau_p(\lambda) \\ \text{sgn}(y) S_p^{ITM}(y; \lambda), & \text{if } |y| > \tau_p(\lambda) \end{cases}, \quad (11)$$

where  $\text{sgn}(y)$  denotes the sign of  $y$ ,  $\tau_p(\lambda) = \lambda^{1/(2-p)}(2-p)[p/(1-p)]^{1/(2-p)}$ ,  $g_p(\theta; \lambda) = \theta + \lambda p \theta^{p-1}$ ,  $\theta_0 = [\lambda p(1-p)]^{1/(2-p)}$ , and  $S_p^{ITM}(y; \lambda)$  is the root of the equation  $g_p(\theta; \lambda) = |y|$ . Since  $g_p(\theta; \lambda)$  is monotonically increasing in the range of  $[\theta_0, +\infty)$ , for any  $|y| \in [\theta_0, +\infty)$ ,  $g_p(\theta; \lambda) = |y|$  has one unique root which can be obtained using numerical methods. However, as shown in Fig. 1, the thresholding function in Eq. (11) cannot always guarantee to converge to the global solution. Krishnan and Fergus [25] proposed an LUT method to correctly solve the problem in Eq. (4). In image restoration, the  $p$  value can be fixed and  $|y|$  should fall into the range of  $[0, 1]$ , and thus LUT is very efficient. However, for general  $\ell_p$ -norm non-convex sparse coding problems where the values of  $x$ ,  $\lambda$  and  $p$  are unconstrained, LUT will not be an effective and efficient solution.

In addition, Marjanovic and Solo [30] proposed a very similar method to ours for solving the one-scalar  $\ell_p$ -minimization problem (4). However, our proposed GISA is different from this method across the context. On one hand, we use a direct and very intuitive way to accurately present the global solution of the non-convex problem (4) (see Section 3.2 and Fig. 2 for details), while [30] makes the problem somewhat more complicated through pure mathematical deductions. In particular, our method uses two simple equations (21) and (22) to obtain the two most important numerical values of the problem:  $\tau_p^{GST}(\lambda)$  (the threshold value) and  $x_p^*$  (the minimum at the threshold). The method proposed in [30], however, uses complex mathematics to accomplish the similar task. Our work is thus much easier to understand, and it reveals clearly the physical meaning underlies such kind of non-convex optimization problems,

which are previously believed hard to be solved precisely and understood intuitively. Furthermore, the motivations and the main mechanisms of our method and [30] are significantly different. The main goal of our method is to solve the non-convex sparse coding problems through iterative shrinkage mechanism for computer vision tasks such as image deconvolution and face recognition, while [30] aims mainly at matrix completion by majorization-minimization strategy for DNA microarray analysis.

### 3. Generalized shrinkage / thresholding function

#### 3.1. Soft-thresholding

To solve the  $\ell_1$ -minimization problem:

$$\min_x \frac{1}{2}(y-x)^2 + \lambda|x|, \quad (12)$$

Donoho [16] proposed a soft-thresholding operator:

$$T_1(y; \lambda) = \begin{cases} 0, & \text{if } |y| \leq \lambda \\ \text{sgn}(y)(|y| - \lambda), & \text{if } |y| > \lambda \end{cases}. \quad (13)$$

Generally, if  $|y| \leq \lambda$ , the soft-thresholding operator uses the thresholding rule to assign  $T_1(y; \lambda)$  to 0; otherwise, uses the shrinkage rule to assign  $T_1(y; \lambda)$  to  $\text{sgn}(y)(|y| - \lambda)$ .

#### 3.2. Generalization of soft-thresholding

Inspired by soft-thresholding, we proposed a generalized shrinkage/thresholding operator to solve the  $\ell_p$ -minimization problem in Eq. (4) by modifying the thresholding and the shrinkage rules.

If  $y > 0$ , the solution to Eq. (4) should fall into the range of  $[0, y]$ ; otherwise, into the range of  $[y, 0]$ . Without loss of generality, in the following we only consider the case of  $y > 0$ . Let

$$f(x) = \frac{1}{2}(x-y)^2 + \lambda|x|^p. \quad (14)$$

Note that  $f(x)$  is differentiable in the range of  $(0, +\infty)$ . By setting  $p = 0.5$  and  $\lambda = 1$ , in Fig. 2 we show the plots of  $f(x)$  with five typical  $y$  values. As shown in Fig. 2, given  $p$  and  $\lambda$  there exists a specific threshold  $\tau_p^{GST}(\lambda)$ . If  $y < \tau_p^{GST}(\lambda)$ ,  $x = 0$  is the global minimum; otherwise, the non-zero solution would be optimal. Thus, to generalize soft thresholding for solving the problem in Eq. (4), we could focus on two issues: (1) the calculation of threshold  $\tau_p^{GST}(\lambda)$  and (2) the fast searching of the non-zero solution.

The first- and second-order derivatives of  $f(x)$  are:

$$f'(x) = x - y + \lambda p x^{p-1}, \quad (15)$$

$$f''(x) = 1 + \lambda p(p-1)x^{p-2}. \quad (16)$$

By solving  $f''(x_0^{(\lambda,p)}) = 0$ , we have

$$x_0^{(\lambda,p)} = (\lambda p(1-p))^{\frac{1}{2-p}}. \quad (17)$$

One can easily verify that  $f(x)$  is concave in the range of  $(0, x_0^{(\lambda,p)})$ , and is convex in the range of  $(x_0^{(\lambda,p)}, +\infty)$ . To guarantee that  $f(x)$  has a minimum in  $(x_0^{(\lambda,p)}, +\infty)$ , we should

further require  $f'(x_0^{(\lambda,p)}) \leq 0$ . In [33], She let  $f'(x_0^{(\lambda,p)}) = 0$  and solved the following equation

$$f'(x_0^{(\lambda,p)}) = (\lambda p(1-p))^{\frac{1}{2-p}} - \tau_p^{ITM}(\lambda) + \lambda p(\lambda p(1-p))^{\frac{p-1}{2-p}} = 0. \quad (18)$$

The corresponding threshold on  $y$  is

$$\tau_p^{ITM}(\lambda) = \lambda^{1/(2-p)}(2-p)[p/(1-p)^{1-p}]^{1/(2-p)}. \quad (19)$$

In ITM, She [33] extended the soft-thresholding with the thresholding function in Eq. (11).

However, the thresholding rule in [33] is problematic. Although  $y > \tau_p^{ITM}(\lambda)$  can guarantee that equation

$$x^* - y + \lambda p(x^*)^{p-1} = 0 \quad (20)$$

has a unique solution in  $(x_0^{(\lambda,p)}, +\infty)$ , as shown in Fig. 2(c), this minimum  $f(x^*)$  might be higher than  $f(0)$ . Thus, the thresholding function in Eq. (11) actually is not a good generalization of the soft-thresholding operator for  $\ell_p$ -norm minimization.

From Fig. 2(d), one can see that there exists a specific  $y$ , where  $f(x_p^*)$  is exactly  $f(0)$ . Thus, to generalize soft-thresholding, we should solve the following nonlinear equation system to determine a correct thresholding value  $\tau_p^{GST}(\lambda)$  and its corresponding  $x_p^*$ :

$$\frac{1}{2}(x_p^* - \tau_p^{GST}(\lambda))^2 + \lambda(x_p^*)^p = \frac{1}{2}(\tau_p^{GST}(\lambda))^2 \quad (21)$$

$$x_p^* - \tau_p^{GST}(\lambda) + \lambda p(x_p^*)^{p-1} = 0. \quad (22)$$

Based on Eq. (22), we can substitute  $\tau_p^{GST}(\lambda)$  in Eq. (21) with  $x_p^* + \lambda p(x_p^*)^{p-1}$ , and obtain the following equation

$$(x_p^*)^p \left( 2\lambda(1-p) - (x_p^*)^{2-p} \right) = 0. \quad (23)$$

Thus the only solution of  $x_p^*$  in the range of  $(x_0^{(\lambda,p)}, +\infty)$  can be obtained as

$$x_p^* = (2\lambda(1-p))^{\frac{1}{2-p}}, \quad (24)$$

and the thresholding value  $\tau_p^{GST}(\lambda)$  is

$$\tau_p^{GST}(\lambda) = (2\lambda(1-p))^{\frac{1}{2-p}} + \lambda p(2\lambda(1-p))^{\frac{p-1}{2-p}}. \quad (25)$$

We have the following two theorems, and the proofs of them can be found in the supplementary materials.

**Theorem 1** For any  $y \in (\tau_p^{GST}(\lambda), +\infty)$ ,  $f(x)$  has one unique minimum  $S_p^{GST}(y; \lambda)$  in the range of  $(x_p^*, +\infty)$ , which can be obtained by solving the following equation:

$$S_p^{GST}(y; \lambda) - y + \lambda p(S_p^{GST}(y; \lambda))^{p-1} = 0. \quad (26)$$

**Theorem 2** For any  $y \in (\tau_p^{GST}(\lambda), +\infty)$ , let  $S_p^{GST}(y; \lambda)$  be the unique minimum of  $f(x)$  in the range of  $(x_p^*, +\infty)$ . We have the following inequality:

$$f(0) > f(S_p^{GST}(y; \lambda)). \quad (27)$$

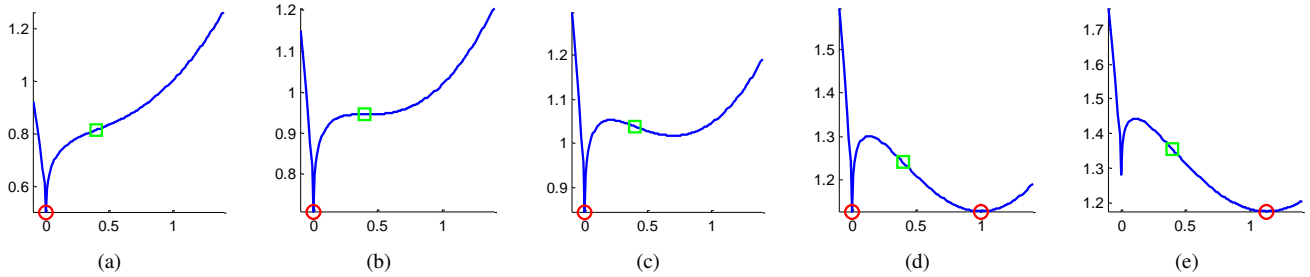


Figure 2. Plots of the function  $f(x)$  in Eq. (14) with different values of  $y$ : (a)  $y = 1$ , (b)  $y = 1.19$ , (c)  $y = 1.3$ , (d)  $y = 1.5$ , and (e)  $y = 1.6$ .

---

Algorithm 1 (GST):  $T_p^{GST}(y; \lambda) = GST(y, \lambda, p, J)$

---

**Input:**  $y, \lambda, p, J$

1.  $\tau_p^{GST}(\lambda) = (2\lambda(1-p))^{\frac{1}{2-p}} + \lambda p(2\lambda(1-p))^{\frac{p-1}{2-p}}$
2. **if**  $|y| \leq \tau_p^{GST}(\lambda)$
3.  $T_p^{GST}(y; \lambda) = 0$
4. **else**
5.  $k = 0, x^{(k)} = |y|$
6. **Iterate on**  $k = 0, 1, \dots, J$
7.  $x^{(k+1)} = |y| - \lambda p(x^{(k)})^{p-1}$
8.  $k \leftarrow k + 1$
9.  $T_p^{GST}(y; \lambda) = \text{sgn}(y)x^{(k)}$
10. **end**

**Output:**  $T_p^{GST}(y; \lambda)$

---

To solve Eq. (26), we propose an iterative algorithm  $GST(y, \lambda, p)$ , which is summarized in Algorithm 1.

In Algorithm 1, the output would converge to the correct solution when  $J \rightarrow \infty$ . Empirically we found that satisfactory results can be obtained by choosing  $J = 2$  or 3.

Finally, we propose a generalized soft-thresholding (GST) function for solving the  $\ell_p$ -norm minimization in Eq. (4):

$$T_p^{GST}(y; \lambda) = \begin{cases} 0, & \text{if } |y| \leq \tau_p^{GST}(\lambda) \\ \text{sgn}(y)S_p^{GST}(|y|; \lambda), & \text{if } |y| > \tau_p^{GST}(\lambda) \end{cases} \quad (28)$$

Like the soft-thresholding function, the GST function also involves a thresholding rule  $T_p^{GST}(y; \lambda) = 0$  when  $|y| \leq \tau_p^{GST}(\lambda)$  and a shrinkage rule  $T_p^{GST}(y; \lambda) = \text{sgn}(y)S_p^{GST}(|y|; \lambda)$  when  $|y| > \tau_p^{GST}(\lambda)$ . Compared with the thresholding function in [33], in GST we adopt a different thresholding value  $\tau_p^{GST}(\lambda)$ , and propose an algorithm, i.e., Algorithm 1, to solve the equation in Eq. (26). Based on Theorem 1 and Theorem 2, GST can always find the correct solution to the simple  $\ell_p$ -minimization problem in Eq. (4). Thus, GST can be regarded a better generalization of soft-thresholding for  $\ell_p$ -minimization.

### 3.3. Discussions

Let's further discuss two important cases of GST, i.e., when  $p = 1$  and  $p = 0$ , and their relationships with soft-thresholding [16] and hard-thresholding [2, 19].

When  $p = 1$ , GST will converge after one iteration. S-

ince

$$\lim_{p \rightarrow 1} \tau_p^{GST}(\lambda) = \lambda \lim_{p \rightarrow 1} (1-p)^{p-1} = \lambda, \quad (29)$$

the thresholding value of GST will become  $\lambda$ , and the GST function becomes

$$T_1^{GST}(y; \lambda) = \begin{cases} 0, & \text{if } |y| \leq \lambda \\ \text{sgn}(y)(|y| - \lambda), & \text{if } |y| > \lambda \end{cases} \quad (30)$$

One can see that the soft-thresholding function is a special case of GST with  $p = 1$ .

When  $p = 0$ , GST will also converge after one iteration. The thresholding value of GST will be

$$\tau_0^{GST}(\lambda) = (2\lambda)^{\frac{1}{2}}, \quad (31)$$

and the GST function becomes

$$T_0^{GST}(y; \lambda) = \begin{cases} 0, & \text{if } |y| \leq (2\lambda)^{\frac{1}{2}} \\ y, & \text{if } |y| > (2\lambda)^{\frac{1}{2}} \end{cases}, \quad (32)$$

which is exactly the hard-thresholding function [2, 19] defined for solving the following problem

$$\min_x \frac{1}{2}(y-x)^2 + P(x; \lambda), \quad (33)$$

where the penalty function  $P$  [2, 19, 33] is defined as

$$P(x; \lambda) = \begin{cases} 0, & \text{if } x = 0 \\ \lambda, & \text{if } x \neq 0 \end{cases} \quad (34)$$

Clearly, the hard-thresholding function is a special case of GST with  $p = 0$ .

## 4. Generalized iterated shrinkage algorithm

With the proposed GST in Eq. (28), we can readily have a generalized iterated shrinkage algorithm (GISA) for solving the  $\ell_p$ -norm non-convex sparse coding problem. GST can also be easily applied for image restoration.

### 4.1. GISA

The proposed GISA is an iterative algorithm, and in each iteration it involves a gradient descent step based on  $\mathbf{A}$  or  $\mathbf{y}$ , followed by a generalized shrinkage/thresholding step:

$$\mathbf{x}^{(k+1)} = T_p^{GST}(\mathbf{x}^{(k)} - \|\mathbf{A}\|^{-2} \mathbf{A}^T (\mathbf{A} \mathbf{x} - \mathbf{y}); \|\mathbf{A}\|^{-2} \lambda), \quad (35)$$

where  $\|\mathbf{A}\|$  denotes the spectral norm of the matrix  $\mathbf{A}$ . The proposed GISA algorithm is summarized in Algorithm 2.

---

Algorithm 2 (GISA):  $\mathbf{x} = GISA(\mathbf{y}, \lambda, p, J)$

---

**Input:**  $\mathbf{y}, \lambda, p, J$

1. Initialize  $\mathbf{x}^{(0)}, t = \|\mathbf{A}\|^{-2}$ .
2. **while** not converge **do**
3.      $\mathbf{x}^{(k+0.5)} = \mathbf{x}^{(k)} - t\mathbf{A}^T(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{y})$ .
4.      $\mathbf{x}^{(k+1)} = GST(\mathbf{x}^{(k+0.5)}, t, \lambda, p, J)$ .
5.     **end while**
6.      $\mathbf{x} = \mathbf{x}^{(k)}$ .

**Output:**  $\mathbf{x}$

---

Actually, GISA is a generalization of the iterative shrinkage/thresholding (IST) method [15], and an example of the iterative thresholding method (ITM) [33]. In [33], She proved that, for any thresholding function  $\Theta(y; \lambda)$  defined for  $-\infty < y < +\infty$  and  $0 \leq \lambda < +\infty$ , if  $\Theta(y; \lambda)$  satisfies the following properties:

- i)  $\Theta(-y; \lambda) = -\Theta(y; \lambda)$ ,
- ii)  $\Theta(y; \lambda) \leq -\Theta(y'; \lambda)$  if  $y \leq y'$ ,
- iii)  $\lim_{y \rightarrow \infty} \Theta(y; \lambda) = \infty$ ,
- iv)  $0 \leq \Theta(y; \lambda) \leq y$  for  $0 \leq y < \infty$ ,

the ITM method would converge to a stationary point. One can easily see that the GST function in Eq. (28) satisfies all these four properties. Thus the convergence of GISA could be guaranteed. From Theorems 1 and 2, one can easily see that, GISA converges to the global optimum when  $\mathbf{A}$  is a positive diagonal matrix. When  $\mathbf{A}$  is unitary, by exploiting the unitary-invariant property of the  $\ell_2$ -norm, GISA can also converge to the optimal solution. Moreover, if  $p = 1$ , GISA would degenerate to IST, and would converge to the global minimum.

Besides, several algorithms, e.g., Two-step IST (TwIST) [5] and accelerated proximal gradient (APG) [4], have been proposed to speedup IST. By substituting the soft-thresholding function with GST, we can also use these algorithms for  $\ell_p$ -norm non-convex sparse coding.

## 4.2. Sparse gradient based deconvolution using GST

One important application of sparse coding is image restoration. As an example, in this subsection we apply the proposed GST to image deconvolution. Let  $\mathbf{x}$  be the original image. In image deconvolution, the degraded image  $\mathbf{y}$  is modeled as first convolving  $\mathbf{x}$  with a blur kernel  $\mathbf{k}$  and then adding additive white Gaussian noise

$$\mathbf{y} = \mathbf{x} \otimes \mathbf{k} + \mathbf{e}, \quad (36)$$

where  $\otimes$  denotes the convolution operator, and  $\mathbf{e}$  is the additive white Gaussian noise with variance  $\sigma^2$ .

A typical image deconvolution model usually includes a fidelity term and a regularization term, where the fidelity term is modeled based on the degradation process, and the regularization term is modeled based on image priors. Recent studies on natural image statistics have shown that the marginal distributions of filtering responses can be modeled

as hyper-Laplacian with  $0 < p < 1$  [25, 28, 35], which had been adopted in many low level vision problems [13, 36]. By using the sparse gradient based image prior, the image deconvolution model can be formulated as

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x} \otimes \mathbf{k} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{D}\mathbf{x}\|_p^p, \quad (37)$$

where  $\lambda$  is the regularization parameter,  $\mathbf{D} = [\mathbf{D}_h, \mathbf{D}_v]$  denotes the gradient operator, and  $\mathbf{D}_h$  and  $\mathbf{D}_v$  are the horizontal and vertical gradient operators, respectively.

Based on [25, 37], we introduce a new variable  $\mathbf{d} = \mathbf{D}\mathbf{x}$ , and reformulate the problem in (37) as

$$\min_{\mathbf{x}, \mathbf{d}} \frac{1}{2} \|\mathbf{x} \otimes \mathbf{k} - \mathbf{y}\|_2^2 + \frac{\eta\lambda}{2} \|\mathbf{D}\mathbf{x} - \mathbf{d}\|_2^2 + \lambda \|\mathbf{d}\|_p^p. \quad (38)$$

When  $\eta \rightarrow \infty$ , the problem in Eq. (38) would have the same solution as the problem in Eq. (37).

We adopt an alternating minimization strategy to solve the problem in Eq. (38). In each iteration, given a fixed  $\mathbf{d}$ ,  $\mathbf{x}$  can be obtained by solving the following subproblem

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x} \otimes \mathbf{k} - \mathbf{y}\|_2^2 + \frac{\eta\lambda}{2} \|\mathbf{D}\mathbf{x} - \mathbf{d}\|_2^2. \quad (39)$$

Actually, the solution to  $\mathbf{x}$  can be written in the closed form [25, 37]

$$\mathbf{x} = \mathcal{F}^{-1} \left( \frac{\mathcal{F}(\mu\lambda\mathbf{D}^T\mathbf{d}) + \mathcal{F}(\mathbf{k})^* \circ \mathcal{F}(\mathbf{y})}{\mu\lambda(\mathcal{F}(\mathbf{D}_h^T\mathbf{D}_h) + \mathcal{F}(\mathbf{D}_v^T\mathbf{D}_v)) + \mathcal{F}(\mathbf{k})^* \circ \mathcal{F}(\mathbf{k})} \right), \quad (40)$$

where  $\mathcal{F}$  denotes the 2D Fourier transform,  $\mathcal{F}^{-1}$  denotes 2D inverse Fourier transform, “\*” denotes complex conjugate, “ $\circ$ ” stands for the component-wise multiplication, and the division is also operated component-wisely.

Given a fixed  $\mathbf{x}$ , let  $\mathbf{d}^{ref} = \mathbf{D}\mathbf{x}$ , and  $\mathbf{d}$  can be obtained by solving the following subproblem:

$$\min_{\mathbf{d}} \frac{\eta}{2} \|\mathbf{d} - \mathbf{d}^{ref}\|_2^2 + \|\mathbf{d}\|_p^p. \quad (41)$$

Using GST, the solution to each  $d_i$  can be written as

$$d_i = T_p^{GST}(d_i^{ref}; 1/\eta). \quad (42)$$

Finally, we summarize the GST based image deconvolution algorithm in Algorithm 3.

Algorithm 3 is similar to the algorithms in [25, 37], but Wang and Yin [37] only studied the Laplacian prior ( $p = 1$ ), and Krishnan and Fergus [25] used look-up table (LUT) to solve the subproblem in Eq. (41). Here we empirically choose  $J = 1$ , making our algorithm very efficient for sparse gradient based image deconvolution.

## 5. Experimental results

In this section, we evaluate the proposed GISA on two representative vision applications: image deconvolution and face recognition. In image deconvolution experiments, we compare GISA with four state-of-the-art algorithms of  $\ell_p$ -norm non-convex sparse coding: LUT, IRLS, IRL1, and ITM- $\ell_p$ . The results show that GISA is as accurate as LUT but is more efficient, and it is more accurate and

---

Algorithm 3 (GST\_DeConv):  $\mathbf{x} = DeConv(\mathbf{y}, \mathbf{k}, \lambda, p, J)$ 


---

**Input:**  $\mathbf{y}, \mathbf{k}, \lambda, p, J$ 

1. Initialize  $k = 0, \eta^{(0)}, \eta_{\max}, \rho, T, \mathbf{D}$ .
2. Initialize  $\mathbf{x}^{(0)}, \mathbf{d}^{(0)}$ .
3. **while**  $\eta^{(k)} < \eta_{\max}$  **do**
4.      $iter = 0$
5.     **for**  $iter = 1$  to  $T$  **do**
6.         Update  $\mathbf{x}^{(k+1)}$  using Eq. (40)
7.          $\mathbf{d}^{ref} = \mathbf{D}\mathbf{x}^{(k+1)}$
8.          $\mathbf{d}^{(k+1)} = GST(\mathbf{d}^{ref}, 1/\eta, p, J)$
9.     **end for**
10.      $\eta^{(k+1)} = \rho\eta^{(k)}$
11.      $k = k + 1$
12. **end while**
13.  $\mathbf{x} = \mathbf{x}^{(k)}$ .

**Output:**  $\mathbf{x}$ 


---

efficient than the other methods. In face recognition, we use GISA to solve the sparse representation-based classification (SRC) model [38], and show that the performance of SRC can be improved by using GISA with  $p < 1$ . The Matlab source code of GISA can be downloaded at <http://www4.comp.polyu.edu.hk/~cslzhang/code.htm>.

### 5.1. Image deconvolution

In image deconvolution, we followed the experiment setting in [25]. The source code of LUT is from [25] and the source code of IRLS is from [28]. The IRL1 method is implemented by modifying the code in [8]. And we implement the ITM- $\ell_p$  and GISA methods.

We use 8 clean images (see Fig. 3) and five real world camera shake kernels (see Fig. 4) to generate the blurry images, and additive Gaussian white noise with variance of 0.01 is further added to the blurry images. For performance evaluation, apart from the peak signal-noise ratio (PSNR) and running time, we also use the energy function value of the restored image  $\mathbf{x}$ :

$$F(\mathbf{x}) = \frac{1}{2} \|\mathbf{x} \otimes \mathbf{k} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{D}\mathbf{x}\|_p^p \quad (43)$$

as an indicator to evaluate the convergence of algorithms.



Figure 3. The test images.



Figure 4. The five blur kernels.

Fig. 5 shows the deconvolution results of GISA on a test image by using  $p = 1$  and  $p = 0.7$ , respectively. GISA with  $p = 0.7$  is much better than that with  $p = 1$  in terms of suppressing noise and ring effects and preserving edge details, which indicates that non-convex image deconvolution can much improve the deconvolution performance.

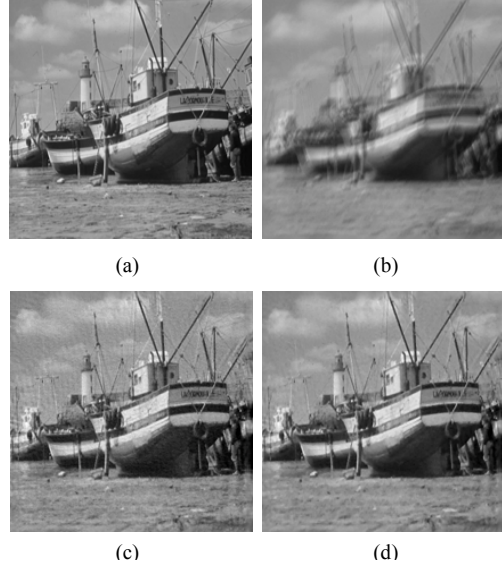


Figure 5. Image deconvolution with GISA: (a) original image, (b) blurry image, (c) deconvolution result (PSNR: 27.17) of GISA with  $p = 1$ , and (d) deconvolution result (PSNR: 28.64) of GISA with  $p = 0.7$ .

Table 1. The PSNR, running time, and energy function values of different algorithms.

Method	IRLS	IRL1	ITM- $\ell_p$	LUT	GISA
PSNR	28.94	28.95	29.27	<b>29.49</b>	<b>29.50</b>
Times(s)	140.32	188.56	0.60	0.88	<b>0.37</b>
EnergyF(x)	21.84	21.79	20.51	<b>20.29</b>	<b>20.32</b>

Table 1 lists the PSNR values, running time (s), and  $F(\mathbf{x})$  values of the competing algorithms on one  $512 \times 512$  image. More results on images with different blur kernels and sizes are provided in the supplementary materials. Here we set  $p = 0.7$ . Compared with IRLS, IRL1 and ITM- $\ell_p$ , GISA can achieve much higher PSNR values, and is computationally more efficient. Note that the  $F(\mathbf{x})$  value of GISA is lower than those of IRLS, IRL1 and ITM- $\ell_p$ . We argue that GISA converges to a better minimum, and this might be the reason of the relative higher PSNR values of GISA. GISA and LUT obtain similar PSNR and  $F(\mathbf{x})$  values, which indicate that these two methods lead to similar solutions. Compared with LUT, however, GISA is more efficient, and does not require the generation and storage of the look-up table.

## 5.2. Face recognition via sparse coding

Given a test sample  $\mathbf{y}$  and the training data matrix  $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K]$ , where  $\mathbf{X}_k, k = 1, 2, \dots, K$ , is the sample matrix of class  $k$ , Wright et al. [38] proposed a sparse representation based classification (SRC) method for face recognition (FR). SRC first seeks the solution of the following sparse coding problem:

$$\hat{\alpha} = \arg \min_{\alpha} \|\mathbf{y} - \mathbf{X}\alpha\|_q^q + \lambda \|\alpha\|_p^p, \quad (44)$$

and then classifies  $\mathbf{y}$  based on the following rule:

$$\text{Label}(\mathbf{y}) = \arg \min_k \{d_k = \|\mathbf{y} - \mathbf{X}_k \hat{\alpha}_k\|_2\}. \quad (45)$$

where  $\hat{\alpha} = [\hat{\alpha}_1; \hat{\alpha}_2; \dots; \hat{\alpha}_K]$ .

In the original SRC, the typical parameter setting is  $q = 2$  and  $p = 1$  (for FR without corruption) or  $q = 1$  and  $p = 1$  (for robust FR with corruption). In the experiment, we used the efficient augmented Lagrangian method (ALM) [40] to solve the original SRC model. Then, by simply replacing the soft-thresholding operator in ALM by the proposed GST operator, we can embed the proposed GISA algorithm into the ALM method for solving the SRC model with arbitrary values of  $p$  and  $q$ .

By fixing  $q = 2$  and varying  $p$ , in the first FR experiment we use the extended Yale B dataset [22, 27] to test the influence of  $p$  on recognition accuracy. The extended Yale B database contains 2,414 images of 38 subjects. The images are normalized and cropped into size of  $32 \times 32$ . In our experiments, we randomly select 30 images from each subject to construct a training dataset of 1,140 images, and use the remaining images for test. We use the principal component analysis (PCA) to reduce the dimensionality of face images, and test the algorithms in both the original image space (1,024 dimensions) and the PCA subspace with feature dimension 500, 300, and 100, respectively. We set the regularization parameter  $\lambda = 10^{-3}$ .

By setting the feature dimension  $d = 100$ , in Fig. 6 we show the recognition rates of SRC versus different  $p$  values. Hereafter we use SRC- $p$  to denote SRC with  $0 < p < 1$ . One can see that, when  $p = 0.6$ , SRC- $p$  achieves the highest recognition rate (93.25%), much higher than that of original SRC (90.97%). Table 2 lists the recognition rates of SRC and SRC- $p$  ( $p = 0.6$ ) under different dimensions. One can see that, GISA based SRC- $p$  can always achieve higher recognition rates than the original SRC with  $p = 1$ . When the feature dimension is low, the improvement of SRC- $p$  against SRC is more significant. This is because when the face feature dimension is small, the matrix  $\mathbf{X}$  tends to be redundant. Thus, the coding solution should be sparser, and GISA is more probable to obtain the correct solution.

By choosing  $q = p = 1$ , the SRC method would become robust to face corruption/occlusion [38]. It is interesting to know if we set  $0 < q = p < 1$ , will SRC become more robust? We denote by SRC- $p, q$  the SRC method with

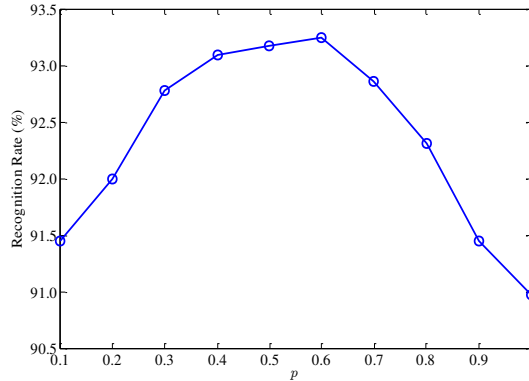


Figure 6. The face recognition rate of SRC by varying the value of  $p$  with GISA.

Table 2. Face recognition rate (%) on the Extended Yale B dataset. In SRC- $p$ , we set  $p = 0.6$ .

Dim.	1024	500	300	100
SRC [38]	96.31	95.91	95.83	90.97
SRC- $p$	<b>96.55</b>	<b>96.31</b>	<b>96.31</b>	<b>93.25</b>

$0 < q = p < 1$ , and embed the proposed GISA into ALM to implement SRC- $p, q$  for robust face recognition. Two types of face image corruption are considered: random pixel corruption and random block occlusion. We set  $p = q = 0.5$  in SRC- $p, q$ .

**Random corruption:** We use the same experiment setting as in the previous experiment, but add random corruption to each test image. Table 3 lists the recognition rates of SRC and SRC- $p, q$  under different ratios of random corruption. One can see that SRC- $p, q$  can always outperform SRC for recognizing face images with random corruption.

**Block occlusion:** In the experiments, we randomly select a square block from each test image, and replace it with an unrelated image. Table 4 lists the recognition rates of SRC and SRC- $p, q$  under different ratios of block occlusion. Again, SRC- $p, q$  can obtain better recognition rates than SRC for face recognition with random block occlusion.

Table 3. Recognition rate (%) on face images with random corruption. In SRC- $p, q$ , we set  $p = q = 0.5$ .

Corruption ratio	10%	20%	40%	60%
SRC [38]	97.48	95.53	90.32	76.64
SRC- $p, q$	<b>97.72</b>	<b>96.64</b>	<b>92.16</b>	<b>78.42</b>

Table 4. Recognition rate (%) on face images with block occlusion. In SRC- $p, q$ , we set  $p = q = 0.5$ .

Occlusion ratio	10%	20%	30%	40%
SRC [38]	89.64	82.58	77.08	69.86
SRC- $p, q$	<b>90.68</b>	<b>83.12</b>	<b>77.73</b>	<b>70.96</b>

## 6. Conclusion

In this paper, we extended the soft-thresholding operator for  $\ell_p$ -minimization. We proposed a generalized shrinkage/thresholding (GST) function and the associated generalized iterated shrinkage algorithm (GISA) for  $\ell_p$ -norm non-convex sparse coding. Compared with the state-of-the-art methods, GISA is theoretically more solid, easier to understand and more efficient to implement, and it can converge to a more accurate solution. Our experimental results on image deconvolution verify the effectiveness and efficiency of GISA, and our experiments on sparse coding based face recognition showed that  $\ell_p$ -norm non-convex sparse coding can improve the recognition rate. With the increasing interests on non-convex  $\ell_p$ -minimization problems, GISA could be potentially used in more and more vision and learning applications.

## 7. Acknowledgements

This work is supported by the Hong Kong RGC / GR-F grant (PolyU 5313/12E), NSFC grant (61271093), Hong Kong Scholar Program, and the program of ministry of education for new century excellent talents (NCET-12-0150).

## References

- [1] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. Fast image recovery using variable splitting and constrained optimization. *IEEE T-IP*, 19(9):2345–2356, 2010.
- [2] A. Antoniadis. Wavelets in statistics: a review (with discussion). *Italian Journal of Statistics*, 6(2):97–130, 1997.
- [3] A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE T-IP*, 18(11):2419–2434, 2009.
- [4] A. Beck and M. Teboulle. A fast iterative shrinkage / thresholding algorithm for linear inverse problems. *SIAM J. Imag. Sci.*, 2(1):183–202, 2009.
- [5] J. Bioucas-Das and M. Figueiredo. A new twist: Two step iterative shrinkage / thresholding algorithms for image restoration. *IEEE T-IP*, 16(12):2992C–3004, 2007.
- [6] E. Candes and T. Tao. Near optimal signal recovery from random projections: universal encoding strategies? *IEEE Trans. Inf. Theory*, 52(12):5406–5425, 2006.
- [7] E. J. Candes and J. Romberg. Quantitative robust uncertainty principles and optimally sparse decompositions. *J. Foundations of Computational Mathematics*, 6(2):227–254, 2006.
- [8] E. J. Candes, M. Wakin, and S. Boyd. Enhancing sparsity by reweighted  $\ell_1$  minimization. *Journal of Fourier Analysis and Applications*, 14(5):877C905, 2008.
- [9] M. Cetin and W. C. Karl. Feature-enhanced synthetic aperture radar image formation based on nonquadratic regularization. *IEEE T-IP*, 10(4):623–C631, 2001.
- [10] R. Chartrand. Exact reconstruction of sparse signals via nonconvex minimization. *IEEE Signal Process. Lett.*, 14(10):707C–710, 2007.
- [11] R. Chartrand and V. Staneva. Restricted isometry properties and nonconvex compressive sensing. *Inverse Problems*, 24(035020):1C–14, 2008.
- [12] R. Chartrand and W. Yin. Iteratively reweighted algorithms for compressive sensing. *IEEE ICASSP*, pages 3869C–3872, 2008.
- [13] T. Cho, C. Zitnick, N. Joshi, S. Kang, R. Szeliski, and W. Freeman. Image restoration by matching gradient distributions. *IEEE T-PAMI*, 34(4):683–C694, 2012.
- [14] T. S. Cho, N. Joshi, C. L. Zitnick, S. B. Kang, R. Szeliski, and W. T. Freeman. A content-aware image prior. *Proc. CVPR*, 2010.
- [15] I. Daubechies, M. Defrise, and C. D. Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57:1413–C1457, 2004.
- [16] D. Donoho. De-noising by soft-thresholding. *IEEE Trans. Inf. Theory*, 41(3):613C–627, 1995.
- [17] D. L. Donoho. For most large underdetermined systems of linear equations the minimal  $l_1$ -norm solution is also the sparsest solution. *Communications on Pure and Applied Mathematics*, 59(7):907–C934, 2006.
- [18] M. Elad. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer, 2010.
- [19] J. Fan. Comment on ‘wavelets in statistics: A review’ by A. Antoniadis. *Italian Journal of Statistics*, 6(2):131–C138, 1997.
- [20] S. Foucart and M. Lai. Sparsest solutions of underdetermined linear systems via  $\ell_q$  minimization for  $0 < q < 1$ . *Applied and Computational Harmonic Analysis*, 26(3):395C–407, 2009.
- [21] G. Gasso, A. Rakotomamonjy, and S. Canu. Recovering sparse signals with a certain family of nonconvex penalties and DC programming. *IEEE T-SP*, 57(12):4686C–4698, 2009.
- [22] A. Georgiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE T-PAMI*, 23(6):643C–660, 2001.
- [23] I. F. Gorodnitsky and B. D. Rao. Sparse signal reconstruction from limited data using focuss: A re-weighted minimum norm algorithm. *IEEE T-SP*, 45(3):600–C616, 1997.
- [24] N. Joshi, C. L. Zitnick, R. Szeliski, and D. Kriegman. Image deblurring and denoising using color priors. *Proc. CVPR*, 2009.
- [25] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-laplacian priors. *Proc. NIPS*, 2009.
- [26] M. Lai and J. Wang. An unconstrained  $\ell_q$  minimization with  $0 < q < 1$  for sparse solution of under-determined linear systems. *SIAM Journal on Optimization*, 21(1):82–C101, 2011.
- [27] K. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE T-PAMI*, 27(5):684–C698, 2005.
- [28] A. Levin, R. Fergus, F. Durand, and W. Freeman. Image and depth from a conventional camera with a coded aperture. *ACM TOG*, 26(3):70, 2007.
- [29] Q. Lv, Z. Lin, Y. She, and C. Zhang. A comparison of typical  $\ell_p$  minimization algorithms. *Neurocomputing*, 2013.
- [30] G. Marjanovic and V. Solo. On  $l_q$  optimization and matrix completion. *IEEE Trans. T-SP*, 60(11):5714–5724, 2012.
- [31] R. Mazumder, J. H. Friedman, and T. Hastie. SparseNet: Coordinate descent with nonconvex penalties. *Journal of the American Statistical Association*, 106(495):1125–1138, 2011.
- [32] B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM J. Comput.*, 24(2):227–234, 1995.
- [33] Y. She. Thresholding-based iterative selection procedures for model selection and shrinkage. *Electronic Journal of Statistics*, 3:384–C415, 2009.
- [34] Y. She. An iterative algorithm for fitting nonconvex penalized generalized linear models with grouped predictors. *Computational Statistics & Data Analysis*, 56(10):2976–2990, 2012.
- [35] E. Simoncelli and E. H. Adelson. Noise removal via bayesian wavelet coring. *ICIP*, pages 379C–382, 1996.
- [36] S. Wang, T. Hou, J. Border, H. Qin, and R. Miller. High-quality image deblurring with panchromatic pixels. *ACM TOG*, 31(5):120–111, 2012.
- [37] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM J. Imag. Sci.*, 1(3):248C–272, 2008.
- [38] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE T-PAMI*, 32(2):210–227, 2009.
- [39] Z. Xu, X. Chang, F. Xu, and H. Zhang.  $\ell_{1/2}$  regularization: A thresholding representation theory and a fast solver. *IEEE T-NNLS*, 23(7):1013C–1027, 2012.
- [40] A. Y. Yang, S. S. Sastry, A. Ganesh, and Y. Ma. Fast  $l_1$ -minimization algorithms for robust face recognition, 2012. arXiv:1007.3753v4.
- [41] W. Zuo and Z. Lin. A generalized accelerated proximal gradient approach for total-variation-based image restoration. *IEEE T-IP*, 20(10):2748–2759, 2011.