SIAM J. IMAGING SCIENCES
Vol. 8, No. 3, pp. 1798–1823

A Weighted Difference of Anisotropic and Isotropic Total Variation Model for Image Processing*

Yifei Lou†, Tieyong Zeng‡, Stanley Osher§, and Jack Xin¶

Dedicated to the memory of our good friend and collaborator, Ernie Esser

Abstract. We propose a weighted difference of anisotropic and isotropic total variation (TV) as a regularization for image processing tasks, based on the well-known TV model and natural image statistics. Due to the form of our model, it is natural to compute via a difference of convex algorithm (DCA). We draw its connection to the Bregman iteration for convex problems and prove that the iteration generated from our algorithm converges to a stationary point with the objective function values decreasing monotonically. A stopping strategy based on the stable oscillatory pattern of the iteration error from the ground truth is introduced. In numerical experiments on image denoising, image deblurring, and magnetic resonance imaging (MRI) reconstruction, our method improves on the classical TV model consistently and is on par with representative state-of-the-art methods.

Key words. anisotropic TV, isotropic TV, weighted difference, difference of convex algorithm, convergence to stationary points, stable oscillatory errors, Bregman and split Bregman iterations

AMS subject classifications. 90C90, 65K10, 49N45, 49M20

DOI. 10.1137/14098435X

1. Introduction.

Many image processing tasks can be formulated as an inverse problem, in which the data f is assumed to be obtained approximately by applying a linear operator A on an image u with additive noise. For example, A is the identity matrix for image denoising, a convolution matrix for deblurring, and subsampling of Fourier transform for a magnetic resonance imaging (MRI) reconstruction problem. In most scenarios, solving u from $Au = f$ is ill-posed in the sense that directly inverting A would result in bad and possibly multiple solutions. It is necessary and even desirable to constrain the solutions through regularization, with the help of prior knowledge of images that one wants to reconstruct. A general model for such an inverse problem is

$$\hat{u} := \operatorname{argmin}_u J(u) + \frac{\mu}{2}\|Au - f\|_2^2 , \tag{1.1}$$

*Received by the editors September 2, 2014; accepted for publication (in revised form) July 9, 2015; published electronically September 10, 2015.
http://www.siam.org/journals/siims/8-3/98435.html

†Department of Mathematical Sciences, University of Texas at Dallas, Dallas, TX 75080 (yifei.lou@utdallas.edu). The work of this author was partially supported by NSF grants DMS-0928427 and DMS-1222507.

‡Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong (zeng@hkbu.edu.hk). The work of this author was partially supported by NSFC 11271049, RGC 211911, 12302714, and RFGs of HKBU.

§Department of Mathematics, UCLA, Los Angeles, CA 90095 (sjo@math.ucla.edu). The work of this author was supported by the Keck Foundation, ONR N000141410683, N000141110749, and NSF DMS-1118971.

¶Department of Mathematics, UC Irvine, Irvine, CA 92697 (jxin@math.uci.edu). The work of this author was partially supported by NSF grants DMS-0928427 and DMS-1222507.

where $J(u)$ is the regularization term, μ is a positive parameter to balance $J(u)$, the data fidelity term is $\|Au-f\|_2^2$, and $\hat{u}$ is an optimal solution of the model or a reconstructed result. A classical regularization is the total variation (TV) proposed by Rudin–Osher–Fatemi [37], which is referred to as the ROF model. It is widely used in image processing applications, such as deconvolution [9, 21, 29], inpainting [8], and superresolution [30], just to name a few. The TV model originated in [37] and is isotropic, and later an anisotropic formulation was addressed in the literature (see [12, 15] among others). We give mathematical definitions for both the isotropic and anisotropic TV in the discrete setting. Denoting u as the column vector by a lexicographical ordering of a two-dimensional (2D) image, we have

$$J_{iso}(u) := \|Du\|_{2,1} = \|\sqrt{|D_x u|^2 + |D_y u|^2}\|_1, \tag{1.2}$$

$$J_{ani}(u) := \|Du\|_1 = \|D_x u\|_1 + \|D_y u\|_1, \tag{1.3}$$

where D_x, D_y denote the horizontal and vertical partial derivative operators, respectively, and $D = [D_x; D_y]$ is the gradient operator ∇ in the discrete setting. We shall use $\|\nabla u\|_{2,1}$ and $\|\sqrt{|D_x u|^2 + |D_y u|^2}\|_1$ interchangeably throughout this paper.

Another interpretation of TV can be given from the perspective of compressive sensing (CS) [3, 14], which is reconstructing a signal from an underdetermined system provided that the signal is sufficiently sparse or sparse in a transform domain. For example, a natural image is mostly sparse after taking the gradient. Mathematically, it amounts to minimizing the L_0 norm of the image gradient, i.e., $J(u) = \|\nabla u\|_0$. To bypass the NP-hard L_0 norm, the convex relaxation approach in CS is to replace L_0 by L_1, and L_1 on the gradient is the TV. The restricted isometry property (RIP) condition [3] theoretically guarantees the exact recovery of sparse solutions by L_1. The RIP regime is where the sensing matrix is *incoherent*, such as a random Gaussian matrix. Several nonconvex penalties have been proposed and studied as alternatives to L_1 [23]. A few notable examples are L^p for $p \in (0,1)$ [10, 25, 46], L_1/L_2 (scale invariant L_1), and $L_1 - L_2$ [16, 26, 27, 47, 48]. In particular, the $L_1 - L_2$ penalty is found to be the best among existing methods for recovering sparse solutions when the sensing matrix is highly coherent or significantly violating the RIP condition [27, 48].

TV regularization has been a very active research topic in the past two decades. Though a gradient descent approach in the original paper can be slow to converge, a projection algorithm was later proposed by Chambolle [5] to speed up convergence based on duality. More recently, the Bregman and split Bregman methodology [11, 19, 33] offered another line of fast algorithms equivalent to the role of the alternating direction method of multipliers (ADMM) and the Douglas–Rachford splitting algorithm in the optimization literature dating back to the 1970s. The connection among these optimization algorithms has been observed in different contexts, among which [38, 40, 44] are the first few papers that explicitly address the connection between ROF and Bregman splitting. There are also some approaches to solving the L_0 minimization directly. In [45], a special alternating minimization strategy with half-quadratic splitting is adopted for image smoothing. Image restoration via L_0 is considered in [34], which uses hard shrinkage for L_0 as opposed to soft shrinkage for L_1. In addition, the L_0 on the gradient can be interpreted as the length of the partition boundaries, which leads to the classical Potts model [35] or piecewise constant Mumford–Shah model [32] for image segmentation or

partition. Recently, Storath, Weinmann, and Demaret [39] proposed a hybrid ADMM and dynamic programming method to solve the Potts model.

Motivated by $L_1 - L_2$ minimization of coherent CS [27, 48], we propose the following weighted difference of convex regularization:

$$J(u) := J_{ani} - \alpha J_{iso} = \|D_x u\|_1 + \|D_y u\|_1 - \alpha \|\sqrt{|D_x u|^2 + |D_y u|^2}\|_1, \tag{1.4}$$

where $\alpha \in [0, 1]$ is a parameter for a more general model. When $\alpha = 1$, $J(u)$ is to apply $L_1 - L_2$ on the gradient. Two advantages of $L_1 - L_2$ over other nonconvex measures are its Lipschitz regularity and guaranteed convergence via the difference of convex algorithm (DCA) [41, 42], which is analogous to a convex splitting technique [17] for gradient systems. We find that the DCA requires solving the L_1 type of minimization as a subproblem, which can be handled efficiently by utilizing the split Bregman technique. We prove that the DCA approach converges to stationary points, a typical situation for nonconvex problems. In practice, the DCA iterations, when properly stopped, are often close to global minima and produce excellent results. The stopping issue is discussed later based on the oscillatory pattern of the iteration errors.

The rest of the paper is organized as follows. Section 2 describes our model in detail including numerical algorithms and convergence analysis. Section 3 is devoted to numerical experiments, where three image processing applications (denoising, deblurring, and MRI reconstruction) are examined. Finally, discussions and conclusions are given in sections 4 and 5, respectively.

2. Our model. To better understand the novel $L_1 - \alpha L_2$ metric, we plot the level curves corresponding to $L_1 - L_2$ and $L_1 - 0.5L_2$ in comparison with L_0 and L_1 in Figure 1. The level lines of the L_0 norm are 0 at origin, 1 at axes, and 2 elsewhere. The level lines corresponding to $\alpha < 1$ in (1.4) are closer to L_0 than that of $\alpha = 1$ in the sense that the latter yields 0 at both axes. L_1 is the best convex approximation of L_0, which has certain limitations. For example, vast literature shows that the L_1 norm on the gradient, which is the anisotropic TV, will produce "blocky" artifacts, as it prefers a piecewise constant image, where the gradient at every pixel is 1-sparse. For blocky images, it could be true that the gradients are 1-sparse due to the fact that most of the gradient vectors inside the "blocks" are 1-sparse. However, for these images, the gradient vectors at the edges are more important, and they may not be 1-sparse. For this reason, we propose a weighted difference model (1.4) with a constant α taking into account the occurrence of nonsparse gradient vectors.

Let (u_{jx}, u_{jy}) be gradient vector at pixel j. Then (1.4) can be rewritten as

$$J(u) = \sum_j \left(|u_{jx}| + |u_{jy}| - \alpha \sqrt{u_{jx}^2 + u_{jy}^2} \right). \tag{2.1}$$

This pointwise formulation suggests that sparsity is enforced on every gradient vector. More specifically, we encourage the gradient to be 1-sparse at every pixel, which implies that horizontal or vertical edges are more preferable in this model. In order to understand the image gradient and 1-sparsity, we plot the histogram of gradient angles over the range of $[0, 90]$

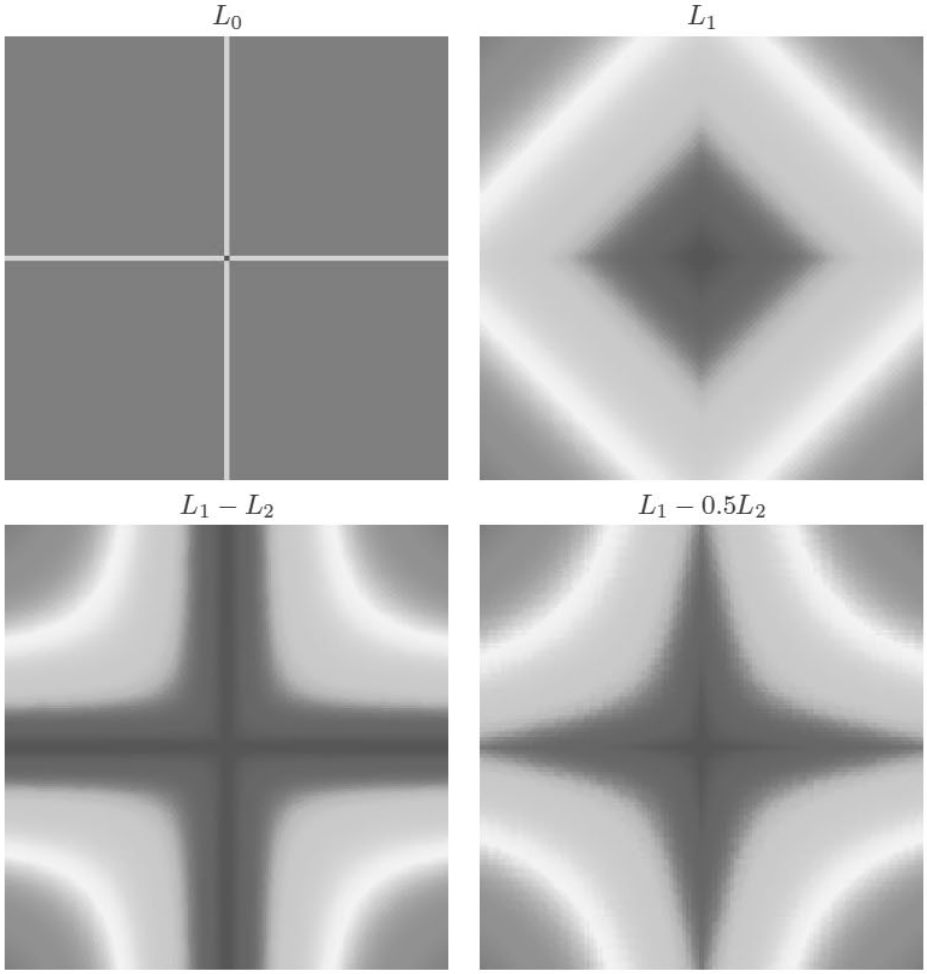

Figure 1. *Level curves of different metrics. The level lines corresponding to* $\alpha < 1$ *in* (1.4) *are closer to* L_0 *than those of* $\alpha = 1$ *in the sense that the latter yield* 0 *at both axes.*

degrees in Figure 2 for a large number of natural images. The angle distribution in other quadrants is similar. As shown in Figure 2, the two largest peaks are at 0 and 90 degrees, which implies that gradient vectors are 1-sparse at a fairly good chance, with nonsparse occurrences also at positive probability. Hence we insert a constant α in (1.4) to reflect such behavior in the histogram.

2.1. Numerical algorithms. We define an objective function in (1.1) with $J(u)$ defined in (1.4):

$$F(u) := \|D_x u\|_1 + \|D_y u\|_1 - \alpha \left\| \sqrt{|D_x u|^2 + |D_y u|^2} \right\|_1 + \frac{\mu}{2} \|Au - f\|_2^2. \tag{2.2}$$

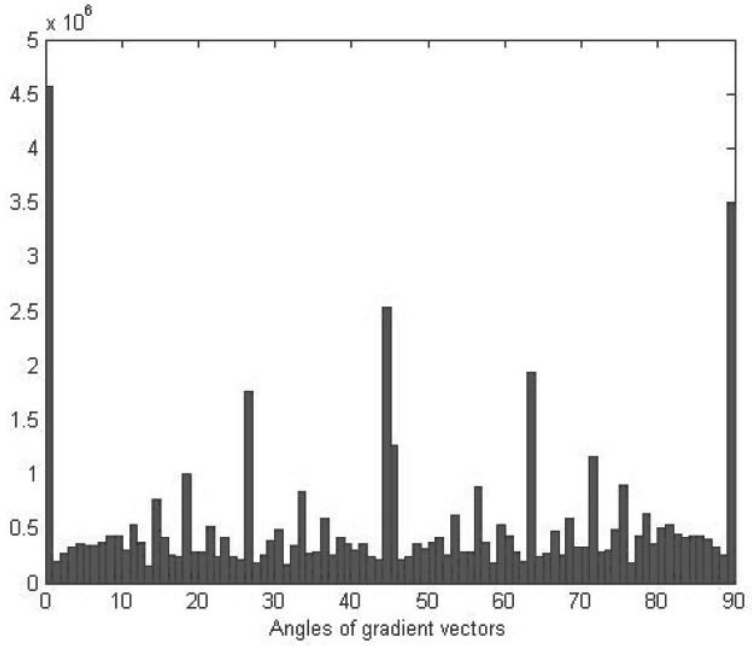

Figure 2. *The histogram of gradient angles over* 300 *images from the Berkeley segmentation dataset* [31]. *Two largest peaks are at* 0 *and* 90 *degrees, indicating that gradient vectors are mostly* 1*-sparse.*

We then decompose $F(u)$ into difference of convex components, i.e., $F(u) = G(u) - H(u)$, where

$$\begin{cases} G(u) = \|D_x u\|_1 + \|D_y u\|_1 + c\|u\|_2^2 + \frac{\mu}{2}\|Au - f\|_2^2, \\ H(u) = \alpha\|\sqrt{|D_x u|^2 + |D_y u|^2}\|_1 + c\|u\|_2^2, \end{cases} \tag{2.3}$$

and c is a positive constant to ensure strong convexity of G and H. After linearizing the H term, we obtain an iterative scheme

$$u^{n+1} = \arg\min_u \|D_x u\|_1 + \|D_y u\|_1 + c\|u\|_2^2 - \alpha\langle Du, q^n\rangle - 2c\langle u, u^n\rangle + \frac{\mu}{2}\|Au - f\|_2^2, \tag{2.4}$$

where $q^n = (q_x^n, q_y^n) = (D_x u^n, D_y u^n)/\sqrt{|D_x u^n|^2 + |D_y u^n|^2}$ at step u^n. Note that q^n is a pointwise calculation, and we adopt the convention that if the denominator is zero at some point, the corresponding q^n value is set to be zero. It means that we select the center of the L_2 norm subgradient (unit ball on the plane) to define q in the algorithm. Each DCA subproblem, (2.4), amounts to solving a TV type of minimization. We employ the split Bregman technique [19] to do the job. Specifically, we introduce two auxiliary variables d_x, d_y as well as two Lagrange multipliers b_x, b_y, while splitting the anisotropic term as

$$\begin{aligned} u^{n+1} = \arg\min_{u,d_x,d_y,b_x,b_y} & \|d_x\|_1 + \|d_y\|_1 + c\|u\|_2^2 - \alpha(d_x^T \cdot q_x^n + d_y^T \cdot q_y^n) \\ & -2c\langle u, u^n\rangle + \frac{\mu}{2}\|Au - f\|_2^2 + \frac{\lambda}{2}\|d_x - D_x u - b_x\|_2^2 + \frac{\lambda}{2}\|d_y - D_y u - b_y\|_2^2. \end{aligned} \tag{2.5}$$

Note that d_x, d_y can be updated via soft shrinkage, defined as

$$\text{shrink}(s,\gamma) = \text{sgn}(s)\max\{|s|-\gamma, 0\}. \tag{2.6}$$

The pseudocode is summarized in Algorithm 1. The algorithm is efficient for many applications where the matrix to be inverted is diagonal or can be diagonalized by Fourier transform, which is true for image denoising, deconvolution, and MRI reconstruction.

Algorithm 1. For solving the unconstrained problem (2.4).

Define $u = q_x = q_y = 0$ and MaxDCA, MaxBregman
for 1 **to** MaxDCA **do**
  $b_x = b_y = 0$
  for 1 **to** MaxBregman **do**
    $u = (\mu A^T A - \lambda D^T D + 2c \cdot I_d)^{-1}(\mu \mathcal{A} f + \lambda D_x^T(d_x - b_x) + \lambda D_y^T(d_y - b_y) + 2cu)$,
    $d_x = \text{shrink}(D_x u + b_x + \alpha q_x/\lambda, 1/\lambda)$,
    $d_y = \text{shrink}(D_y u + b_y + \alpha q_y/\lambda, 1/\lambda)$,
    $b_x = b_x + D_x u - d_x$,
    $b_y = b_y + D_y u - d_y$
  end for
  $(q_x, q_y) = (D_x u, D_y u)/\sqrt{|D_x u|^2 + |D_y u|^2}$
end for

For the corresponding constrained problem,

$$\min \|D_x u\|_1 + \|D_y u\|_1 - \alpha \|Du\|_{2,1} \quad \text{s.t. } Au = f, \tag{2.7}$$

the DCA is expressed as

$$u^{n+1} = \arg\min_u \left\{ \|D_x u\|_1 + \|D_y u\|_1 + c\|u\|_2^2 - \alpha\langle Du, q^n\rangle - 2c\langle u, u^n\rangle \quad \text{s.t. } Au = f \right\}. \tag{2.8}$$

Each DCA subproblem could be reduced to a sequence of unconstrained problems of the form

$$u^{n+1} = \arg\min_u \|D_x u\|_1 + \|D_y u\|_1 + c\|u\|_2^2 - \alpha\langle Du, q^n\rangle - 2c\langle u, u^n\rangle + \frac{\mu}{2}\|Au - z^n\|_2^2, \tag{2.9}$$

$$z^{n+1} = z^n + f - Au^{n+1}. \tag{2.10}$$

The variable z is introduced as a Lagrange multiplier to enforce the constraint $Au = f$ and is updated every step. Again, the first equation can be solved by the split Bregman method. Algorithm 2, for solving the constrained problem (2.8), is almost the same as Algorithm 1, except for an additional update on z.

2.2. Convergence analysis. We want to show that the sequence of $\{u^n\}$ obtained from the DCA iterations, or DCA sequence in short, converges.

We first introduce Lemma 2.1 [41, Theorem 3.7], whose proof is provided to make the paper self-contained.

Lemma 2.1. *If the sequence $\{u^n\}$ is generated by the DCA algorithm (2.4), then $F(u^n) - F(u^{n+1}) \geqslant 2c\|u^n - u^{n+1}\|_2^2$.*

Algorithm 2. For solving the constrained problem (2.8).

Define $u = q_x = q_y = 0, z = f$ and MaxDCA, MaxBregmanInner, MaxBregmanOuter
for 1 **to** MaxDCA **do**
  $b_x = b_y = 0$
  for 1 **to** MaxBregmanOuter **do**
    for 1 **to** MaxBregmanInner **do**
      $u = (\mu A^T A - \lambda D^T D + 2c \cdot I_d)^{-1}(\mu \mathcal{A} z + \lambda D_x^T(d_x - b_x) + \lambda D_y^T(d_y - b_y) + 2cu)$,
      $d_x = \text{shrink}(D_x u + b_x + \alpha q_x/\lambda, 1/\lambda)$,
      $d_y = \text{shrink}(D_y u + b_y + \alpha q_y/\lambda, 1/\lambda)$,
      $b_x = b_x + D_x u - d_x$,
      $b_y = b_y + D_y u - d_y$
    end for
    $z = z + f - Au$
  end for
  $(q_x, q_y) = (D_x u, D_y u)/\sqrt{|D_x u|^2 + |D_y u|^2}$
end for

Proof. It follows from the first-order optimality condition at u^{n+1} that there exist $p^{n+1} \in \partial \|Du^{n+1}\|_1$ such that

$$p^{n+1} - \alpha D^T q^n + 2c(u^{n+1} - u^n) + \mu A^T(Au^{n+1} - f) = 0. \tag{2.11}$$

A simple calculation shows that

$$\begin{aligned} F(u^n) - F(u^{n+1}) = \frac{\mu}{2}\|A(u^n - u^{n+1})\|_2^2 + \mu\langle A(u^n - u^{n+1}), Au^{n+1} - f\rangle \\ + \|Du^n\|_1 - \|Du^{n+1}\|_1 - \alpha(\|Du^n\|_{2,1} - \|Du^{n+1}\|_{2,1}). \end{aligned} \tag{2.12}$$

Left-multiplying (2.11) by $(u^n - u^{n+1})^T$ and plugging the result into (2.12), we get

$$\begin{aligned} F(u^n) - F(u^{n+1}) = \frac{\mu}{2}\|A(u^n - u^{n+1})\|_2^2 - \langle p^{n+1} - \alpha D^T q^n, u^n - u^{n+1}\rangle + 2c\|u^n - u^{n+1}\|_2^2 \\ + \|Du^n\|_1 - \|Du^{n+1}\|_1 - \alpha(\|Du^n\|_{2,1} - \|Du^{n+1}\|_{2,1}). \end{aligned} \tag{2.13}$$

Due to the convexity of $\|Du\|_1$ and $\|Du\|_{2,1}$, we have the following two inequalities:

$$\|Du^n\|_1 \geq \|Du^{n+1}\|_1 + \langle p^{n+1}, u^n - u^{n+1}\rangle \tag{2.14}$$

and

$$\|Du^{n+1}\|_{2,1} \geq \|Du^n\|_{2,1} + \langle D^T q^n, u^{n+1} - u^n\rangle, \tag{2.15}$$

which conclude the proof. ■

We then prove the coercivity of the objective function.

Lemma 2.2. *Suppose $\mu > 0, 0 < \alpha < 1$, and $ker(A) \bigcap ker(D) = \{0\}$. Then the objective function, defined in (2.2), is coercive.*

Proof. Since $\|Du\|_1 = \|D_x u\|_1 + \|D_y u\|_1 \geqslant \|\sqrt{|D_x u|^2 + |D_y u|^2}\|_1$, we get

$$F(u) \geqslant (1-\alpha)\|Du\|_1 + \frac{\mu}{2}(\|Au\|_2^2/2 - \|f\|_2^2).$$

The functional on the right-hand side is coercive, which is a classical theorem for the original ROF model [6, 7]. Here we provide a simple proof to make the paper self-contained. Suppose there exists a sequence $\{u^n\}$ such that $\|u^n\|_2 \to \infty$, and $F(u^n)$ is bounded. Let $v^n = u^n/\|u^n\|_2$ and $v^n \to v^*$ up to a subsequence with $\|v^*\|_2 = 1$. As $F(u^n)$ is bounded, there exists a constant $C > 0$ such that $\|Du^n\|_1 < C$ and $\|Au^n\|_2 < C$. As a result, we have $\|Dv^n\|_1 < C/\|u^n\|_2 \to 0$ as $n \to \infty$, which implies that $\|Dv^*\|_1 = 0$. Similarly, we have $\|Av^*\|_2 = 0$. By the assumption $ker(A) \bigcap ker(D) = \{\mathbf{0}\}$, we get $v^* = 0$, which contradicts $\|v^*\| = 1$. ∎

The following theorem gives a weak convergence result of the DCA sequence. The weak convergence refers to the fact that

$$\partial(\|Du^*\|_1 - \alpha\|Du^*\|_{2,1}) \subseteq \partial\|Du^*\|_1 - \alpha\partial\|Du^*\|_{2,1}. \tag{2.16}$$

Please refer to the appendix for the proof of this inclusion.

Theorem 2.3. *Under the assumptions in Lemma 2.2, any limit point u^* of $\{u^n\}$ satisfies a weak first-order optimality condition,*

$$\mathbf{0} \in \partial\|Du^*\|_1 - \alpha\partial\|Du^*\|_{2,1} + \mu A^T(Au^* - f). \tag{2.17}$$

Proof. It is easy to check that both difference of convex components defined in (2.3) have modulus of strong convexity of $2c$. It follows from Lemma 2.1 that $F(u^n) - F(u^{n+1}) \geqslant 2c\|u^n - u^{n+1}\|_2^2$. Consequently, the objective function F is monotonically decreasing. As F is bounded from below, $F(u^n)$ converges, which implies that $\|u^n - u^{n+1}\|_2^2 \to 0$. In addition, the sequence $\{u^n\}$ is bounded due to Lemma 2.2. Then it follows from the Bolzano–Weierstrass theorem that there exists a subsequence of $\{u^n\}$, denoted as $\{u^{n_k}\}$, converging to a limit point u^*.

We look at the optimality condition at the $(n_k + 1)$ step of the DCA, i.e.,

$$\mathbf{0} \in \partial\|Du^{n_k+1}\|_1 - \alpha D^T q^{n_k} + 2c(u^{n_k+1} - u^{n_k}) + \mu A^T(Au^{n_k+1} - f). \tag{2.18}$$

As $u^{n_k} \to u^*$ and $u^n - u^{n+1} \to 0$, we have $Du^{n_k} \to Du^*$ and $Du^{n_k+1} \to Du^*$.

Let $v^* = Du^*$ and $v^{n_k+1} = Du^{n_k+1}$. Since $v^{n_k+1} \to v^*$, we have for sufficiently large n_k that $\text{supp}(v^*) \subseteq \text{supp}(v^{n_k+1})$, and if $v_j^* \neq 0$ at some j, then $\text{sign}(v_j^{n_k+1}) = \text{sign}(v_j^*)$. Therefore, $\partial\|v^{n_k+1}\|_1 \subseteq \partial\|v^*\|_1$, which means that $\partial\|Du^{n_k+1}\|_1 \subseteq \partial\|Du^*\|_1$.

Define $(u_{jx}, u_{jy}) := Du_j$ as gradient at pixel j, and then $\partial\|Du\|_{2,1} = \prod_j \partial\|Du_j\|_2$. Note that the subgradient of the L_2 norm of the gradient vector (u_x, u_y) has the form

$$\partial\|(u_x, u_y)\|_2 = \begin{cases} (u_x, u_y)/\sqrt{u_x^2 + u_y^2} & \text{if } \sqrt{u_x^2 + u_y^2} \neq 0, \\ u_x^2 + u_y^2 \leqslant 1 & \text{if } (u_x, u_y) = (0,0). \end{cases} \tag{2.19}$$

At pixels j where $(u_{jx}^*, u_{jy}^*) \neq (0,0)$, we have that $q_j^{n_k}$ converges to $(u_{jx}^*, u_{jy}^*)/\sqrt{|u_{jx}^*|^2 + |u_{jy}^*|^2}$ (a unit vector). According to the definition of $q_j^{n_k}$, we know it is either zero if $(u_{jx}^{n_k}, u_{jy}^{nk}) = (0,0)$

or defined on the unit circle otherwise, both of which lie in the unit ball, corresponding to the subgradient of the L_2 norm at discontinuity zero. Therefore, at pixels j where $(u^*_{jx}, u^*_{jy}) = (0,0)$, we have $q_j^{n_k} \in \partial\|(u^*_{jx}, u^*_{jy})\|_2$ (a unit ball). Using the chain rule of subgradient (Corollary 16 in [20]), we have $\partial\|Du^*_j\|_2 = D^T\partial\|(u^*_{jx}, u^*_{jy})\|_2$. To sum up, if $(u^*_{jx}, u^*_{jy}) \neq (0,0)$, then $D^T q_j^{n_k}$ converges to $\partial\|Du^*_j\|_2$; otherwise, $D^T q_j^{n_k}$ belongs to $\partial\|Du^*_j\|_2$.

Putting all of the above together and letting $n_k \to \infty$ in (2.18), we derive that u^* satisfies the weak first-order optimality condition, (2.17). ■

Remark 2.1. The subgradient used in this paper is called a "regular" subgradient (r-sub), while a general subgradient (g-sub) involves a limiting process. Please refer to the book [36] for these two types of subgradients. The relation between the two is r-sub $\subseteq$ g-sub. It seems that r-sub is too restrictive, but (2.16) may not always hold if g-sub is considered.

3. Experiments. We apply the proposed method[1] to three applications: image denoising, deconvolution, and the MRI construction. The matrix A in these examples can be diagonalized by Fourier transform, and hence Algorithm 1 or Algorithm 2 can be efficiently implemented. We compare L_1 and $L_1 - \alpha L_2$ for $\alpha = 0.5$ or 1 (the rationale for $\alpha = 0.5$ is given in section 4.3) with some existing methods, such as L_0 for image smoothing in [45], L_0 in [34], L_p for $p = 2/3$ in [25], and $L_1 + L_2^2$ in [2] for image deblurring. We use the structural similarity index (SSIM) [43] as a quantitative measure for image quality. Let us first define *local* similarity index computed on windows x and y,

$$\text{ssim}(x,y) := \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \tag{3.1}$$

where μ_x, μ_y are the average of x, y; σ_x^2, σ_y^2 are the variance; σ_{xy} is covariance of x, y; and c_1, c_2 are two variables to stabilize the division with weak denominator. The overall SSIM is the mean of local similarity indices, i.e.,

$$\text{SSIM}(X,Y) := \frac{1}{N}\sum_{i=1}^{N} \text{ssim}(x_i, y_i), \tag{3.2}$$

where X is a reference image, Y is a distorted one, x_i, y_i are corresponding windows indexed by i, and N is the number of windows. Here we consider windows of size 8×8.

Image denoising. We examine the problem of image denoising using three piecewise constant images: Shapes, Peppers, and House, in Figures 3–5, as well as a Lena image in Figure 6. We assume zero-mean additive Gaussian noise with standard deviations being 0.2, 0.03, 0.03, and 0.05 for Figures 3–6, respectively. Not only does our method work particularly well on horizontal or vertical edges by design; it can also deal with natural images as well. To verify convergence analysis, the difference of u^n and u^{n-1} versus (outer/DCA) iterations is plotted in logarithm scale for denoising Shape and Lena images in Figure 7, which shows that $L_1 - 0.5L_2$ converges faster than $L_1 - L_2$. Furthermore, we observe numerically that the algorithm still converges without a strong convexity requirement, i.e., $c = 0$ in (2.8). As the ground truth is available, we plot the relative errors versus CPU runtime for $L_1, L_1 - L_2, L_1 - 0.5L_2$ in

[1]Source codes can be downloaded from https://sites.google.com/site/louyifei/Software.

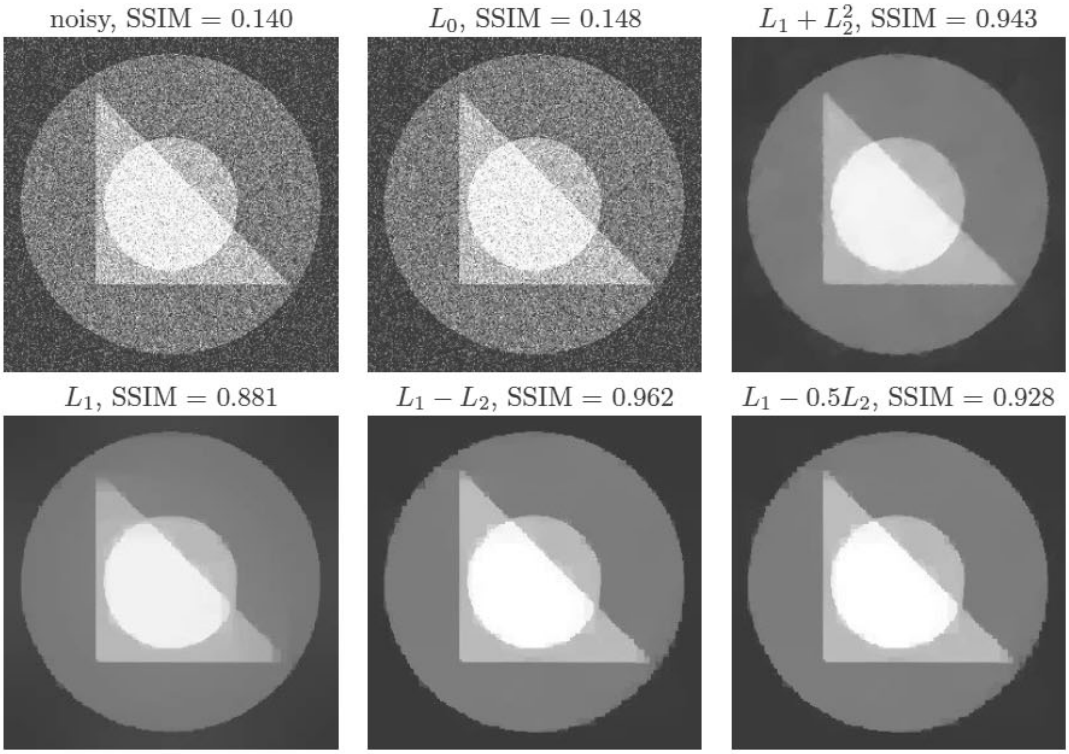

Figure 3. *Denoising results with comparison to L_0 in [45] and $L_1 + L_2^2$ in [2].*

Table 1
Comparisons of different denoising methods in terms of SSIM and computational time (sec).

Denoising	House		Peppers	
	SSIM	Time	SSIM	Time
L_0 [45]	0.9046	0.07	0.8702	0.08
$L_1 + L_2^2$ [2]	0.9214	0.33	0.9452	0.41
L_1	0.9195	0.67	0.9387	0.77
$L_1 - 0.5L_2$	**0.9347**	1.80	**0.9564**	1.82

Figure 8. This figure implies that our solutions oscillate around the ground truth due to the nonconvex nature of our model. Additionally we observe that the larger α is (say, approaching 1), the less well-behaved DCA becomes due to more weight on the nonconvex term. On the other hand, $L_1 - L_2$ yields better results than $L_1 - 0.5L_2$ for the first few DCA iterates. The denoising results presented in Figures 3 and 6 are from stopping the DCA after two iterations. The computational time[2] of denoising two images (House and Peppers) is recorded in Table 1, which shows that $L_1 - 0.5L_2$ gives the best results with extra computational time.

Image deblurring. In Figure 9, a binary image is vertically blurred by motion blur of

[2]All experiments are performed using MATLAB 2014a on a desktop (Windows 7, 3.6GHz CPU, 24GB RAM).

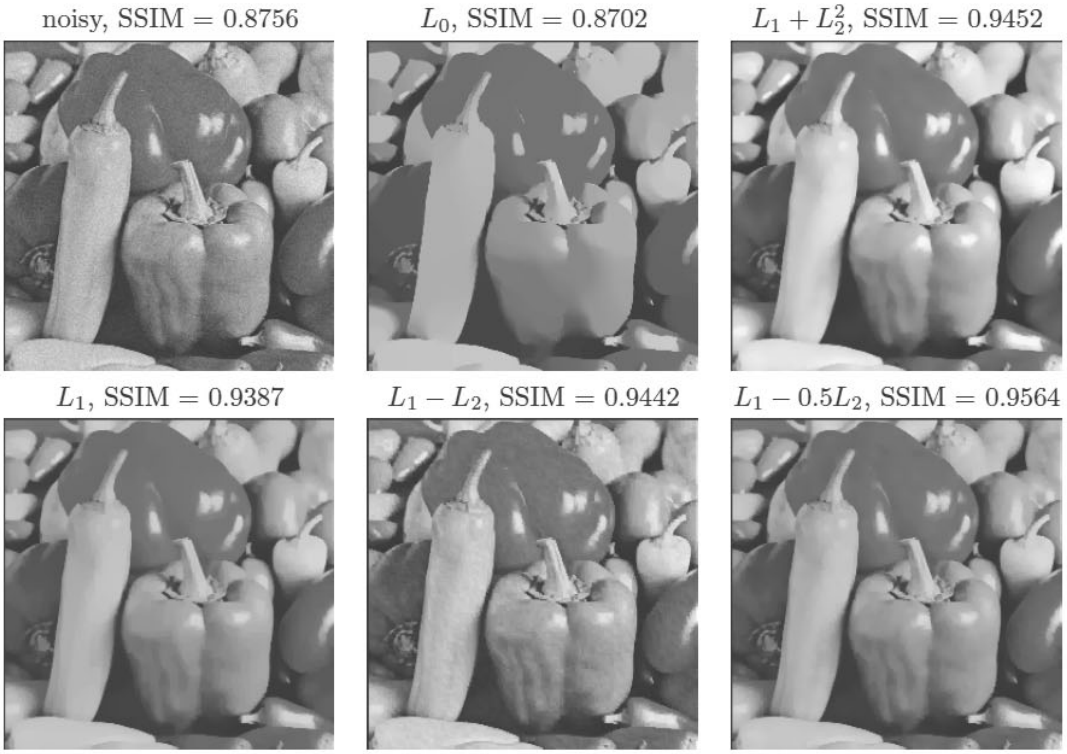

Figure 4. *Denoising results with comparison to* L_0 *in* [45] *and* $L_1 + L_2^2$ *in* [2].

15 pixels plus Gaussian additive noise with zero mean and standard deviation 0.1. Our method outperforms L_0 in [34], L_p for $p = 2/3$ in [25], $L_1 + L_2$ in [2], and the state-of-the-art deblurring method BM3D [13]. In Figures 10–11, we compare all the methods on two piecewise constant images, House and Peppers, where the original images are blurred by 9×9 Gaussian blur whose standard deviation is 1.5 plus Gaussian additive noise with zero mean and standard deviation 0.05. In Figure 13, we present deblurring results for a natural image: Cameraman. The original image is blurred by 15×15 Gaussian blur whose standard deviation is 1.5 plus Gaussian additive noise with zero mean and standard deviation 0.05. In all deblurring examples, our method is better than the classical L_1 approach. We find that our method looks sharper and produces fewer ringing artifacts, compared to $L_0, L_{2/3}$, and BM3D, though these three methods have better SSIM values. The relative errors versus computational time is plotted in Figure 12 for deblurring binary and Cameraman images. It shows behavior similar to that of the denoising problem in that $L_1 - L_2$ tends to worsen beyond certain iterations while $L_1 - 0.5L_2$ is more stable. The deblurring results presented in Figures 9 and 13 are from stopping the DCA after 2 and 10 iterations for $L_1 - 0.5L_2$ and $L_1 - L_2$, respectively. The computational time is listed in Table 2, which suggests that one future direction is accelerating our algorithm.

MRI reconstruction. In Figure 14, we investigate the MRI reconstruction problem using

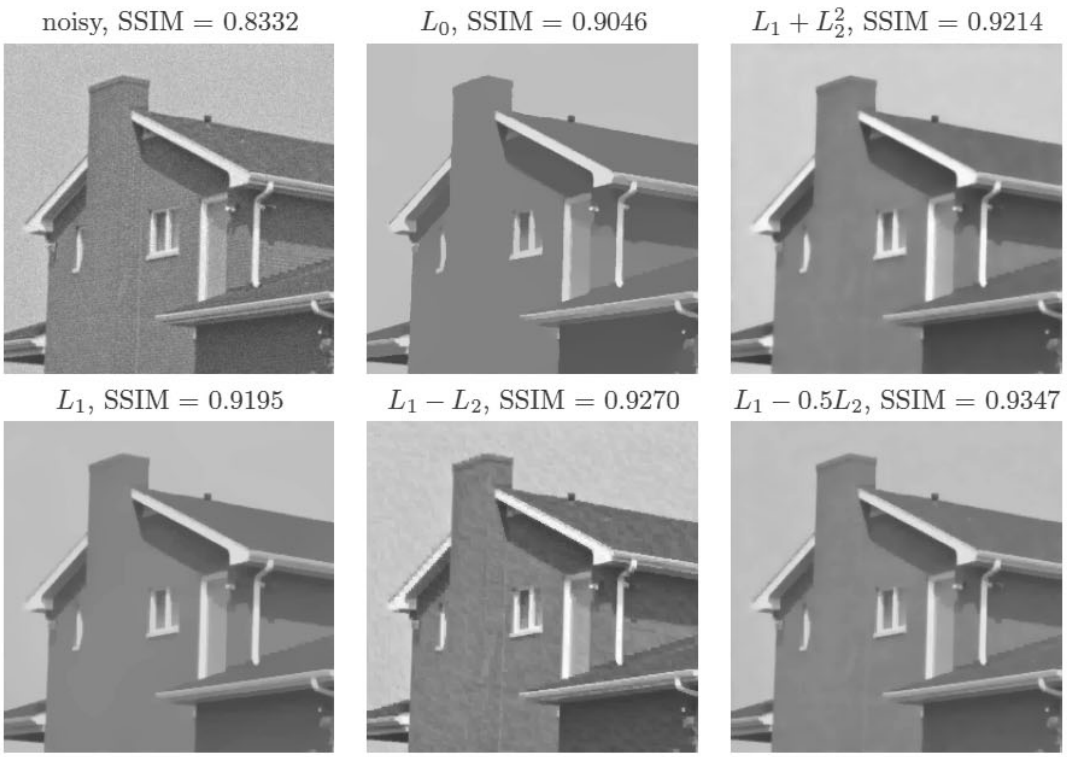

Figure 5. *Denoising results with comparison to L_0 in [45] and $L_1 + L_2^2$ in [2].*

Table 2

Comparisons of different deblurring methods in terms of SSIM and computational time (sec). We find that our method looks sharper and produces fewer ringing artifacts, compared to L_0, $L_{2/3}$, and BM3D, though these three methods have better SSIM values.

Deblurring	Binary		Peppers	
	SSIM	Time	SSIM	Time
BM3D [13]	0.917	1.19	0.873	1.05
L_0 [34]	0.879	7.11	0.875	5.26
$L_{2/3}$ [25]	0.887	0.09	**0.884**	0.45
$L_1 + L_2^2$ [2]	0.934	0.33	0.859	0.54
L_1	0.945	5.88	0.841	6.06
$L_1 - 0.5L_2$	**0.967**	8.69	0.848	8.82

a Shepp–Logan phantom from seven and eight radial projections. There is no noise when we synthesize the data. Consequently we adopt the constrained formulation, i.e., Algorithm 2 for solving (2.8). Due to the presence of complex values in the MRI reconstruction problem, SSIM is no longer applicable; instead we use root-mean-square (RMS) error to measure the performance quantitatively. The RMS between reference and distorted images X, Y is defined as $\mathrm{RMS}(X,Y) = \frac{1}{\sqrt{M}}\|X - Y\|_2$, where M is the number of pixels in images X, Y. Figure 14 shows that our method can get a perfect reconstruction using only eight projections, while a

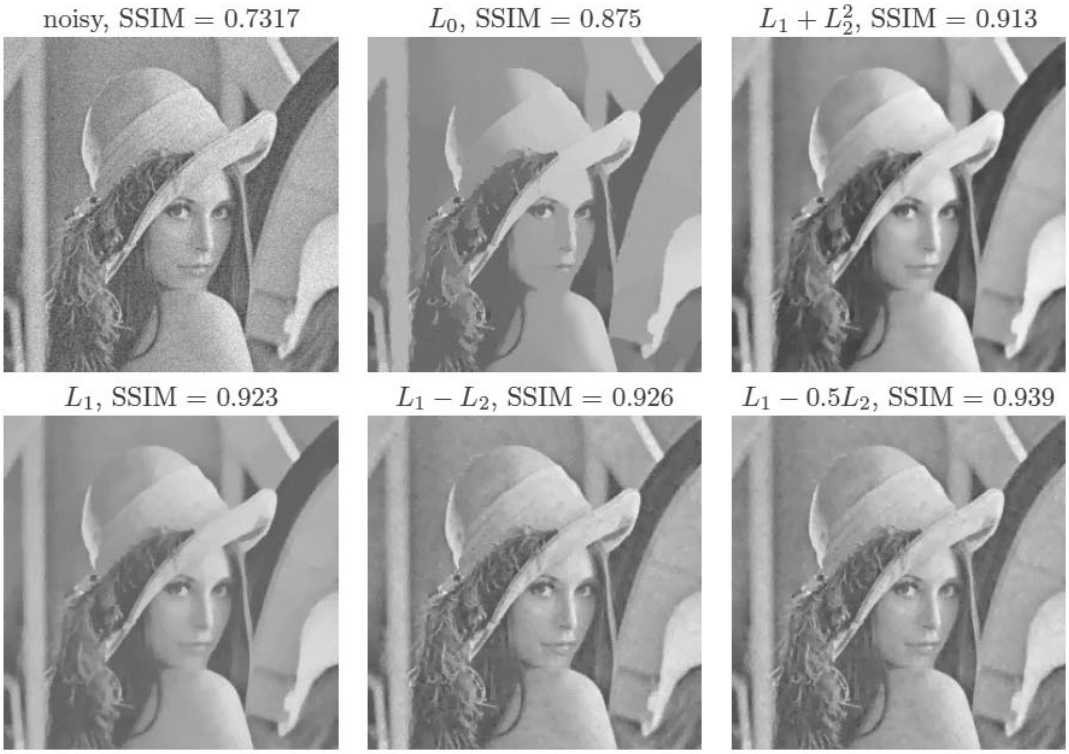

Figure 6. *Denoising results with comparison to L_0 in [45] and $L_1 + L_2^2$ in [2].*

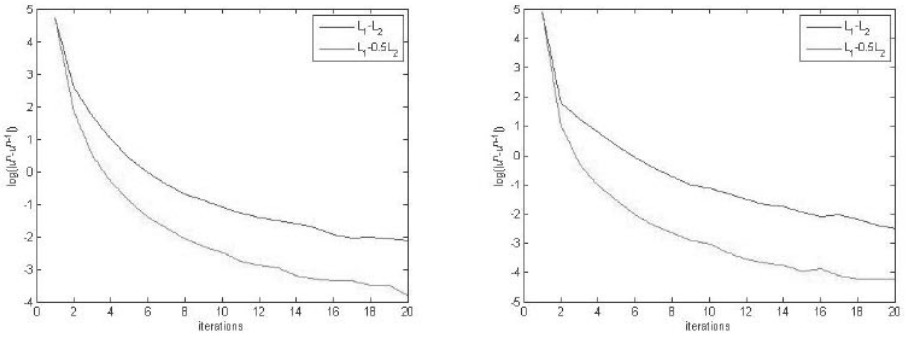

Figure 7. *The difference of u^n and u^{n-1} versus (outer/DCA) iterations is plotted in logarithm scale for denoising examples in Figure 3 (left) and Figure 6 (right). $L_1 - 0.5L_2$ converges faster than $L_1 - L_2$.*

similar work [10] reports that 10 projections are required. When the number of projections is down to seven, $L_1 - 0.5L_2$ is much better than L_1 and $L_1 - L_2$ visually as well as in terms of RMS. The relative errors versus CPU time is plotted in Figure 15. The relative errors of $L_1 - L_2$ iterations in the constrained formulation appear as stable oscillations in contrast to

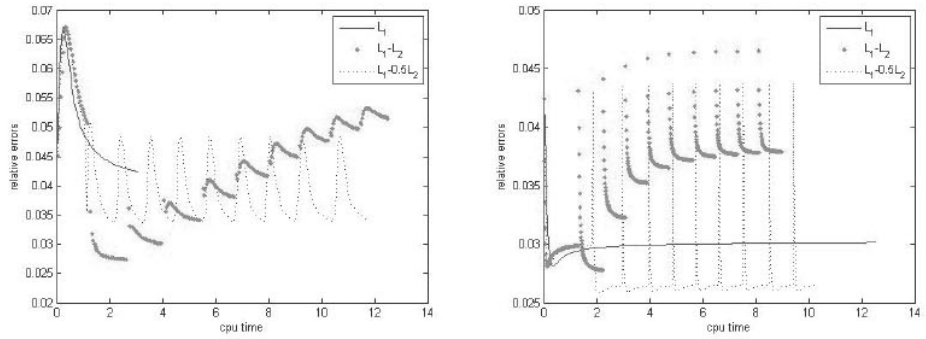

Figure 8. *The relative errors versus runtime for methods $L_1, L_1 - L_2, L_1 - 0.5L_2$ for denoising examples in Figure 3 (left) and Figure 6 (right). Our model solutions are seen to oscillate around the ground truth due to nonconvexity.*

the unstable oscillations in the unconstrained problems.

4. Discussions. Let us draw some connections of this work to two existing methods, the Lysaker–Osher–Tai (LOT) model [28] and Bregman iterations [33]. Additionally, we will comment on the stopping criterion and discuss the parameter setting. As we claim to promote 1-sparse gradient vectors via $L_1 - \alpha L_2$, we evaluate the sparsity of the results Du and compare them with those obtained with other sparsity promoting metrics.

4.1. Relation to existing methods. At first, the iterative scheme (2.5) for $\alpha = 1$ resembles the work of denoising the normals, proposed by Lysaker, Osher, and Tai [28],

$$u^{n+1} = \operatorname{argmin}_u \|Du\|_{2,1} - q^n \cdot Du + \frac{\mu}{2}\|Au - f\|_2^2, \tag{4.1}$$

where $q^n = (q_x^n, q_y^n) = (D_x u^n, D_y u^n)/\sqrt{|D_x u^n|^2 + |D_y u^n|^2}$ is the surface normal. Notice that the TV norm in (4.1) is isotropic, while the first term in our model is the anisotropic TV; and hence $L_1 - L_2$ applied to the gradient with linearized L_2 term is different from the LOT model.

On the other hand, the LOT model leads to the discovery of Bregman iterations [33], which relates to the DCA as well. Specifically, the Bregman distance [1] based on a convex functional $J(\cdot)$ between two points u and v is defined as

$$D_J^p(u, v) := J(u) - J(v) - \langle p, u - v\rangle, \tag{4.2}$$

where $p \in \partial J(v)$ is the subgradient of J at the point v. Osher et. al. [33] suggest an iterative refinement procedure to update u as follows:

$$u^{n+1} = \operatorname{argmin}\ D_J^{p^n}(u, u^n) + \frac{\mu}{2}\|Au - f\|_2^2 \tag{4.3}$$

$$= \operatorname{argmin}\ J(u) - \langle p^n, u\rangle + \frac{\mu}{2}\|Au - f\|_2^2, \tag{4.4}$$

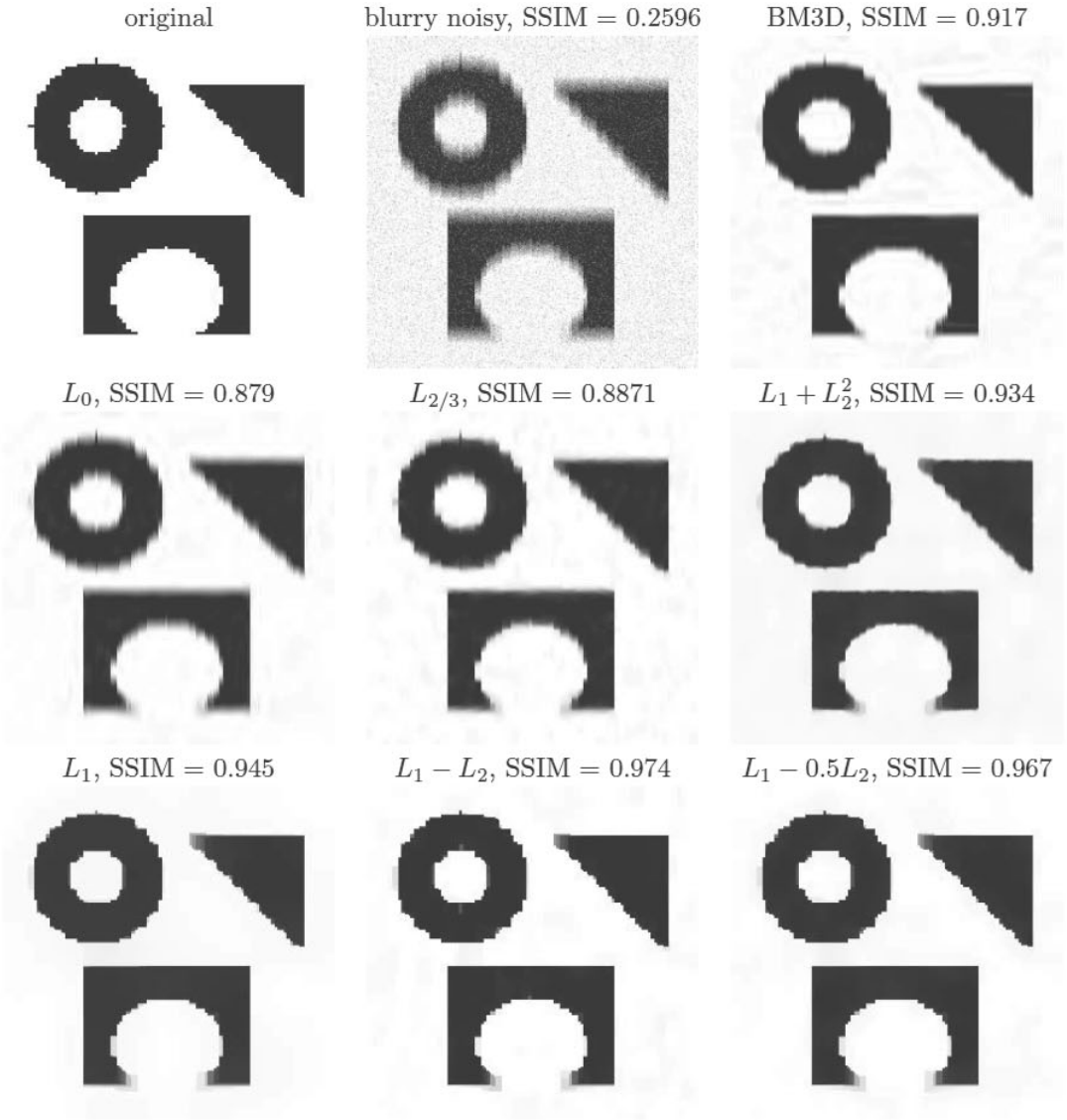

Figure 9. *Deblurring results with comparison to L_0 in [34], L_p for $p = 2/3$ in [25], $L_1 + L_2^2$ in [2], and the state-of-the-art deblurring method BM3D [13].*

which is referred to as the Bregman iterations. Let $J(u) = \|Du\|_2$ be the isotropic TV as in the LOT model, and its subgradient has the form $-\nabla \cdot \frac{Du}{|Du|}$. Consequently, we rewrite the second term in (4.4) as

$$\langle p^n, u\rangle = \left\langle -D \cdot \frac{Du^n}{|Du^n|}, u \right\rangle = \left\langle \frac{Du^n}{|Du^n|}, Du \right\rangle, \tag{4.5}$$

which coincides with the second term in the LOT model (4.1).

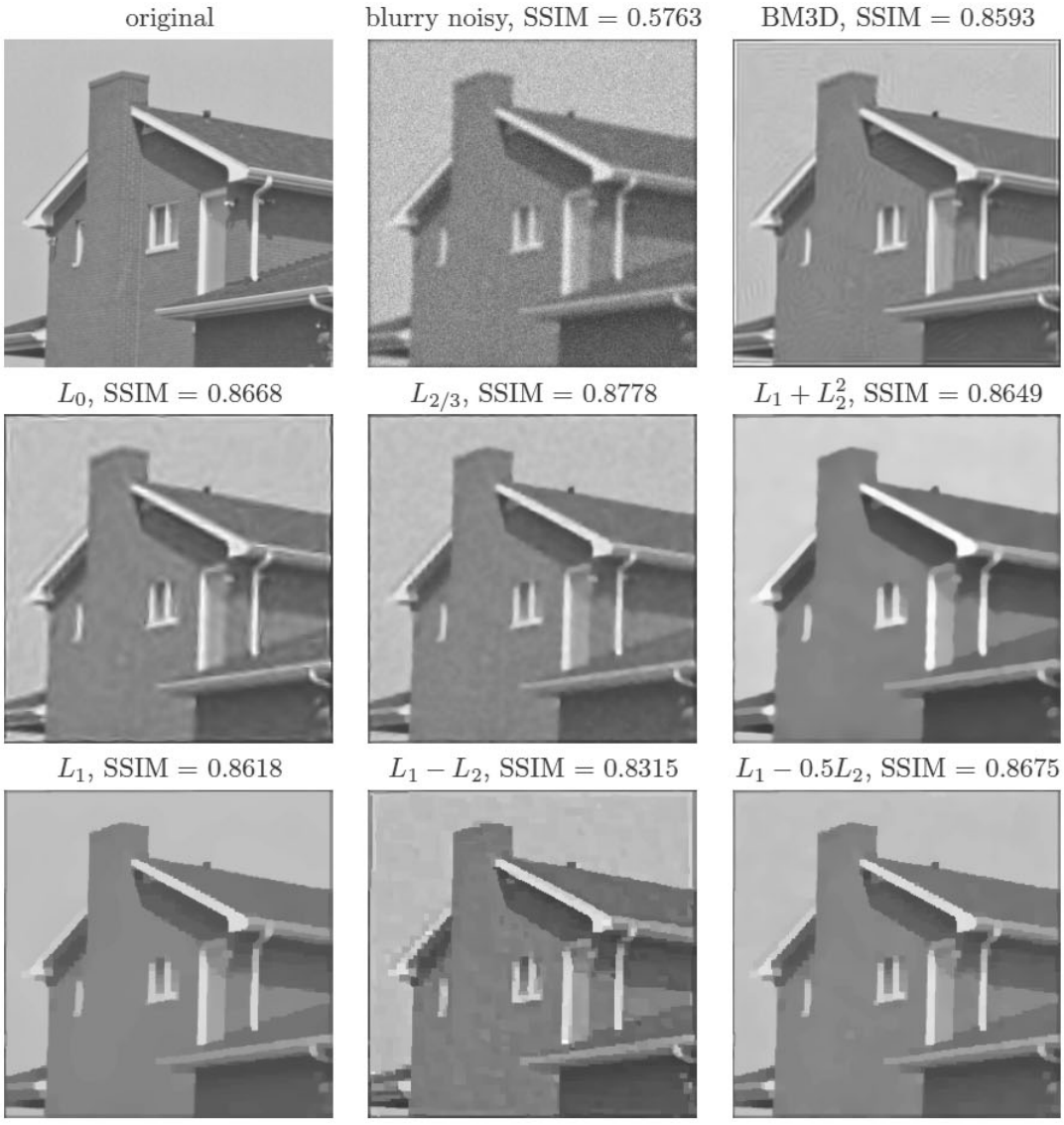

Figure 10. *Deblurring results with comparison to L_0 in [34], L_p for $p = 2/3$ in [25], $L_1 + L_2^2$ in [2], and the state-of-the-art deblurring method BM3D [13].*

Bregman iterations can be viewed as an optimization technique. Computing the optimality condition for each subproblem (4.4), we obtain

$$p^{n+1} - p^n + \mu A^T(Au^{n+1} - f) = 0. \tag{4.6}$$

Summing up to $n+1$, we have $p^{n+1} - \mu A^T(u^{n+1} - z^n)$ for $p^0 = 0$ and $z^{n+1} = z^n + (f - Au^n)$. It is the optimality condition for solving u^{n+1} from argmin $J(u) + \frac{\mu}{2}\|Au - z^n\|_2^2$. In short, the

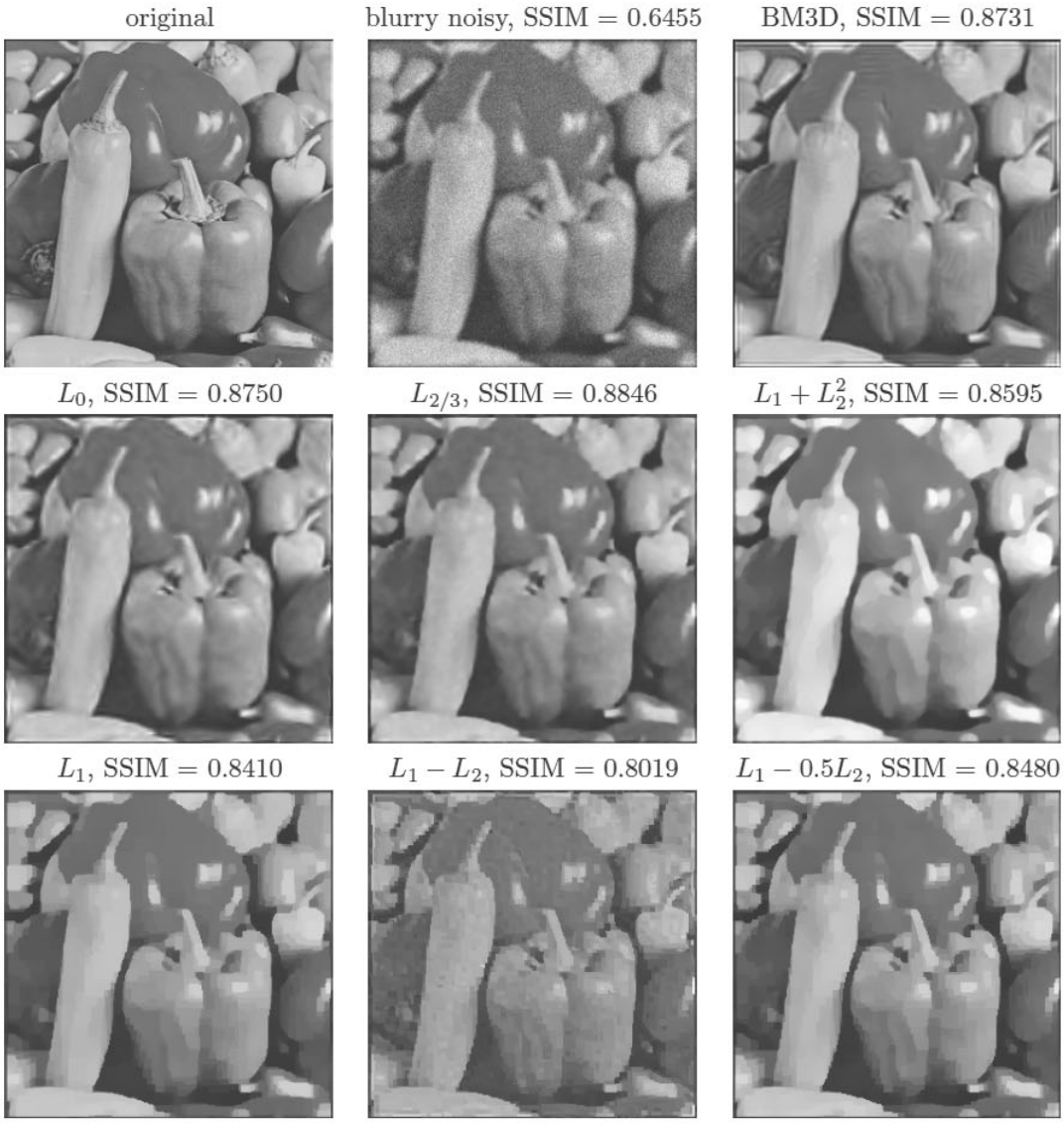

Figure 11. *Deblurring results with comparison to L_0 in [34], L_p for $p = 2/3$ in [25], $L_1 + L_2^2$ in [2], and the state-of-the-art deblurring method BM3D [13].*

Bregman iterations can be rewritten as

$$u^{n+1} = \operatorname{argmin}\ J(u) + \frac{\mu}{2}\|Au - z^n\|_2^2, \tag{4.7}$$

$$z^{n+1} = z^n + (f - Au^n). \tag{4.8}$$

The DCA for solving $L_1 - L_2$ minimization can be derived from the Bregman iterations in a similar way. Let p and q be the subgradient of anisotropic J_{ani} and isotropic J_{iso}, respectively.

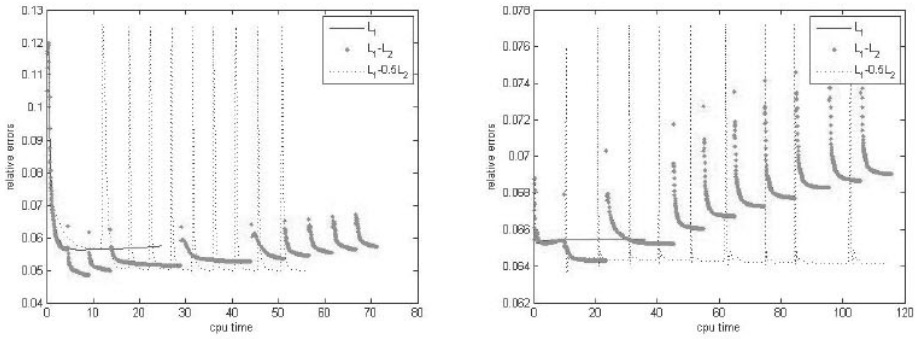

Figure 12. *The relative errors versus runtime for methods $L_1, L_1 - L_2, L_1 - 0.5L_2$ for deblurring examples in Figure 9 (left) and Figure 13 (right).*

Lagging the isotropic term gives us

$$p^{n+1} - p^n - \alpha(q^n - q^{n-1}) + \mu A^T(Au^{n+1} - f) = 0. \tag{4.9}$$

We apply the same summation technique as in (4.6) and obtain

$$p^{n+1} - \alpha q^n + \mu A^T(Au^{n+1} - z^{n+1}) = 0, \tag{4.10}$$

$$z^{n+1} = z^n + (f - Au^n) \tag{4.11}$$

for $p^0 = q^0 = z^0 = 0$. The subproblem (4.10) is equivalent to

$$u^{n+1} = \arg\min J_{ani}(u) - \alpha\langle q^n, u\rangle + \frac{\mu}{2}\|Au - z^n\|_2^2, \tag{4.12}$$

which looks very similar to applying the DCA for a constrained problem, (2.8), when $c = 0$. The algorithm derived from the Bregman iterations is summarized in Algorithm 3. Its difference from Algorithm 2 lies in the update of z and q. For Algorithm 2, z is updated with MaxBregmanOuter iterations, and then q is updated, while Algorithm 3 updates z and q simultaneously. The comparison between the Bregman and DCA iterations for solving such constrained nonconvex problems is a subject of further study.

4.2. Stopping criterion. We discuss the stopping conditions of Algorithms 1 and 2 for unconstrained and constrained problems, respectively. Both algorithms have an outer DCA loop, which iteratively updates q, and inner iterations for updating u. We use u^n and u_k to specify the outer and inner outputs of u and set the max inner/outer iterations to be 200 and 20, respectively, i.e., MaxBregman = 200 and MaxDCA = 20 in Algorithm 1.

The inner loop is easier to impose a proper stopping criterion for, because the inner loop solves a convex subproblem. Some standard stopping criteria are the relative error being small, the objective function being stagnant, or both, i.e.,

$$\frac{\|u_{k+1} - u_k\|}{\|u_k\|} < \epsilon_u \quad \text{and/or} \quad \frac{|F(u_{k+1}) - F(u_k)|}{|F(u_k)|} < \epsilon_F \tag{4.13}$$

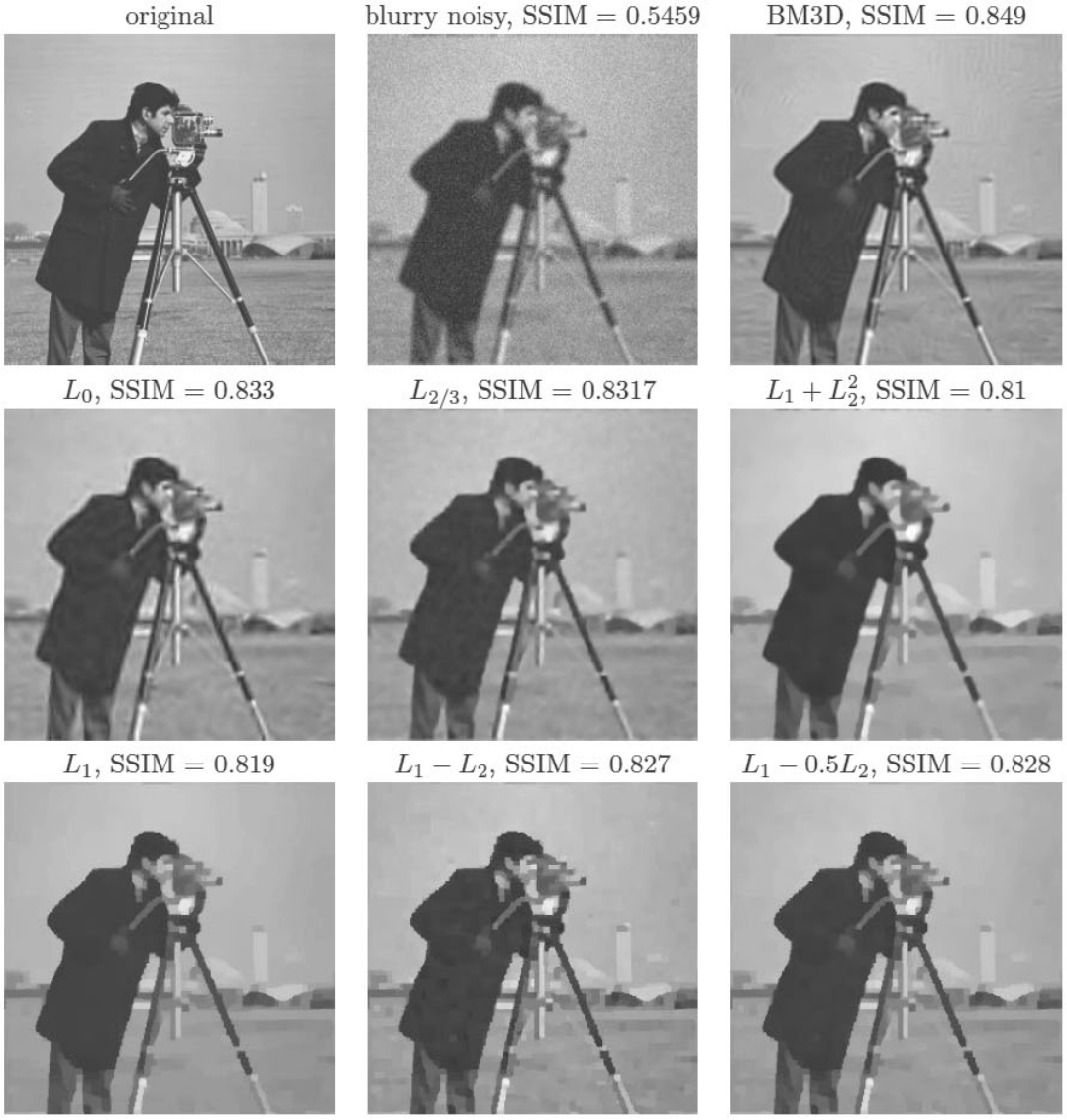

Figure 13. *Deblurring results with comparison to L_0 in [34], L_p for $p = 2/3$ in [25], $L_1 + L_2^2$ in [2], and the state-of-the-art deblurring method BM3D [13].*

with predefined tolerance values ϵ_u, ϵ_F. In this paper, we choose to stop the inner iteration when the relative error is smaller than $1e^{-6}$.

As for the outer iterations, Figures 8, 12, and 15 show that the relative error develops an oscillatory pattern, and Figure 7 suggests that the DCA sequence usually converges in a few outer iterations (10–20). One can estimate the onset time t_b of the oscillation stage of the error based on training images. In the denoising (deblurring) example, $t_b = 2$ $(= 10)$. Hence,

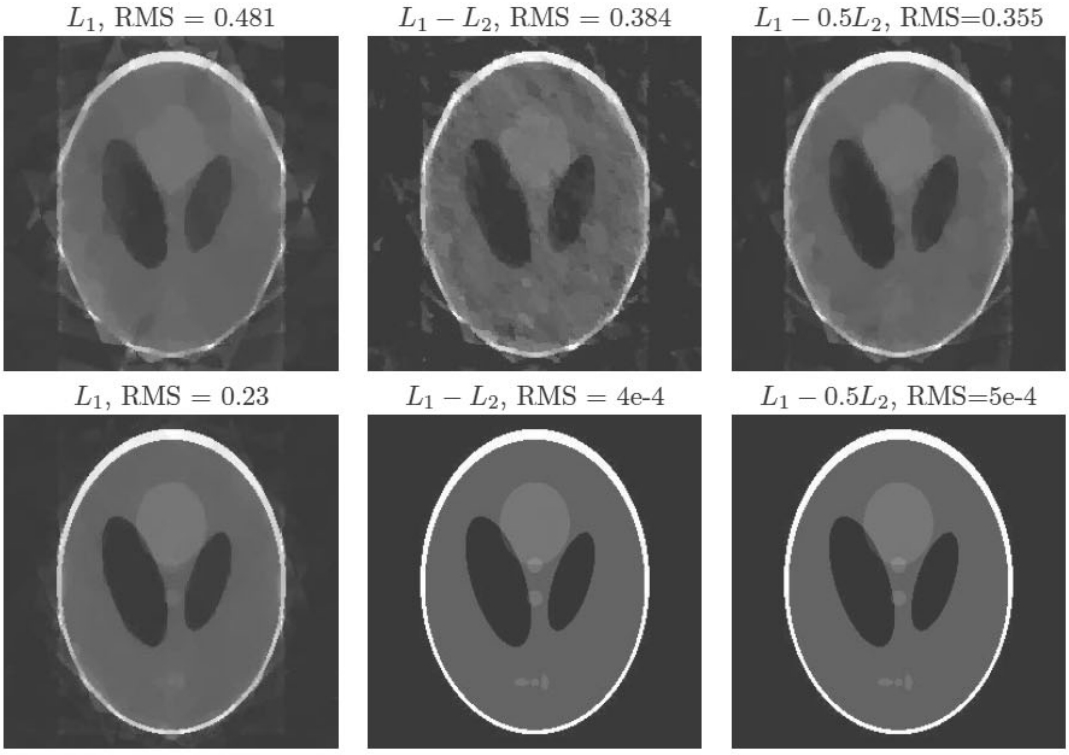

Figure 14. *MRI reconstruction using seven (top) and eight projections (bottom). The RMS is provided for comparison.*

a good stopping time for the outer iteration is at the end of an inner loop when the CPU time exceeds t_b.

More generally, if the error does not follow a clear oscillatory pattern, one could inject random perturbations with slowly reduced magnitudes to steer away from unstable stationary points or directions to help convergence toward the ground truth [22]. This approach is closely related to simulated annealing [18, 24].

4.3. Parameter estimation. Let us derive the value of α based on the gradient distribution. Suppose that the gradient value $D_x u$ follows the distribution [25], $\frac{p}{2\Gamma(\frac{1}{p})} e^{-|x|^p}$, where $\Gamma(t) = \int_0^{+\infty} x^{t-1} e^{-x}$. It is Gaussian distribution for $p = 2$, Laplacian distribution for $p = 1$, and hyper-Laplacian for $0 < p < 1$. We have

$$E_1 = E|D_x u| = \frac{p}{2\Gamma\left(\frac{1}{p}\right)} \int_{-\infty}^{+\infty} e^{-|x|^p} |x| \mathrm{d}x = \frac{1}{\Gamma(\frac{1}{p})} \int_0^{+\infty} e^{-t} t^{\frac{2}{p}-1} \mathrm{d}t = \frac{\Gamma(\frac{2}{p})}{\Gamma(\frac{1}{p})}, \tag{4.14}$$

$$E_2 = E|D_x u|^2 = \frac{p}{2\Gamma\left(\frac{1}{p}\right)} \int_{-\infty}^{+\infty} e^{-|x|^p} |x|^2 \mathrm{d}x = \frac{1}{\Gamma(\frac{1}{p})} \int_0^{+\infty} e^{-t} t^{\frac{3}{p}-1} \mathrm{d}t = \frac{\Gamma(\frac{3}{p})}{\Gamma(\frac{1}{p})}. \tag{4.15}$$

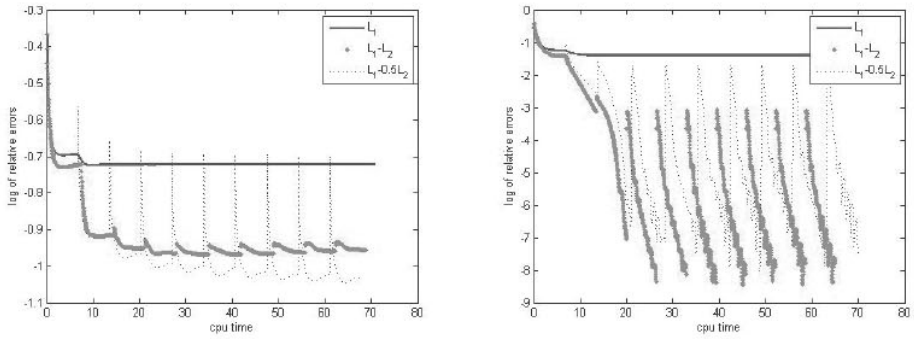

Figure 15. *The logarithm of relative errors versus runtime for methods $L_1, L_1 - L_2, L_1 - 0.5L_2$ in MRI reconstruction problem using seven (left) and eight (right) projections. All are solved under constrained formulation.*

Algorithm 3. For solving constrained problem (2.8) using the Bregman method.

Define $u = q_x = q_y = 0, z = f$ and MaxDCA, MaxBregman
for 1 **to** MaxDCA **do**
  $b_x = b_y = 0$
  for 1 **to** MaxBregman **do**
    $u = (\mu A^T A - \lambda \triangle)^{-1}(\mu \mathcal{A} z + \lambda D_x^T(d_x - b_x) + \lambda D_y^T(d_y - b_y))$,
    $d_x = \text{shrink}(D_x u + b_x + \alpha q_x/\lambda, 1/\lambda)$,
    $d_y = \text{shrink}(D_y u + b_y + \alpha q_y/\lambda, 1/\lambda)$,
    $b_x = b_x + D_x u - d_x$,
    $b_y = b_y + D_y u - d_y$
  end for
  $z = z + f - Au$,
  $(q_x, q_y) = (D_x u, D_y u)/\sqrt{|D_x u|^2 + |D_y u|^2}$
end for

As α is a weighting parameter to balance the anisotropic and isotropic TV terms, it can be estimated using the ratio of E_1 and $\sqrt{E_2}$, i.e.,

$$\alpha = \frac{E_1}{\sqrt{E_2}} = \frac{\Gamma(2/p)}{\sqrt{\Gamma(3/p)\Gamma(1/p)}}. \tag{4.16}$$

Table 3 lists the values of α based on gradient distributions for $p = 0.5, 1, 2$. We analyze the gradient distribution in Figure 16, which shows that the distribution of image gradient data matches the $p = 1/2$ distribution better than classical Gaussian ($p = 2$) or Laplacian ($p = 1$) distribution. This observation is consistent with the choice of the hyper-Laplacian [4, 25] for image processing ($p \in [0.5, 0.8]$). In the rest of the paper, we shall fix the weighting coefficient $\alpha = 1/2$ to approximate the desired value in Table 3.

As for the parameter c in (2.8), theoretically we need c to be positive so that strong convexity leads to the proof of DCA convergence (Lemma 2.1). Without strong convexity at

Table 3
The value of α based on the gradient distribution.

p	α
0.5	0.5477
1	0.7071
2	0.7979

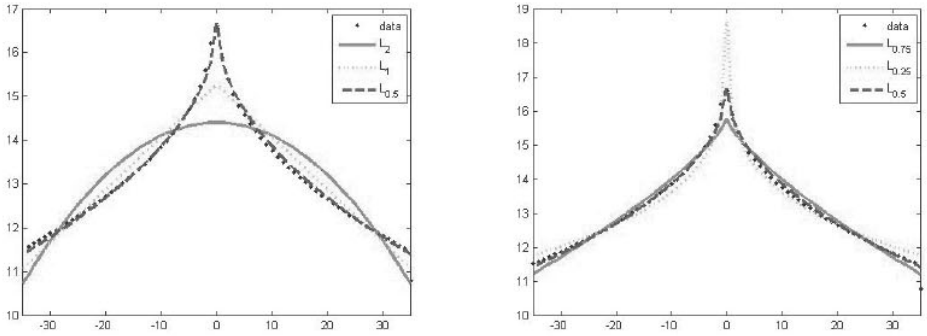

Figure 16. *The plot of log probability versus gradient in comparison with different distributions, indicating that the gradient distribution of a large natural image dataset matches $L_{1/2}$ or the $p = 1/2$ hyper-Laplacian distribution better than classical Gaussian or Laplacian distribution.*

$c = 0$, we can only show that the objective function is monotonically nonincreasing, while we are unable to get that $\|u^{n+1} - u^n\|$ converges to zero. In practice, we observe numerically that the algorithm still converges if $c = 0$ (refer to Figure 7), and the algorithm converges slowly if c is large, so we choose $c = 0$ in experiments. The convergence analysis without strong convexity is left to future exploration.

4.4. Sparse gradients. We examine the sparsity of the gradient vectors Du of the results obtained in the denoising and deblurring experiments. Define a gradient vector to be *nonsparse* if both $D_x u$ and $D_y u$ at that pixel are larger than 0.01. Then we can calculate the percentage of nonsparse gradient vectors over the total number of pixels. The sparsity percentage of all testing images is recorded in Tables 4 and 5 for denoising and deblurring examples, respectively. In the denoising case, the L_0 and L_1 norms yield the least nonsparse gradient vectors, though the reconstructed images look oversmoothed with lower SSIM values. As for a more difficult deblurring problem, the methods of BM3D, L_0, $L_{2/3}$, and $L_1 + L_2^2$ do not promote sparsity, while $L_1 - 0.5L_2$ produces more 1-sparse gradients, and L_1 is comparable in this regard. The sparsity of $L_1 - L_2$ is always worse than that of $L_1 - 0.5L_2$, possibly due to the unstable behavior of the algorithm, as illustrated in Figures 8 and 12.

5. Conclusion. We proposed a weighted difference of anisotropic and isotropic total variation (TV) as a regularization term for image processing applications. We presented a difference of convex algorithm (DCA) for both the constrained and unconstrained formulations. We proved the convergence of the algorithm to ensure that each limiting point is a stationary point and the values of the objective function monotonically decrease. The behavior of the

Table 4

The percentages (%) of nonsparse gradient vectors Du of the denoising results obtained with L_0, $L_1 + L_2^2$, L_1, $L_1 - L_2$, and $L_1 - 0.5L_2$ regularization terms in comparison to Du of the original image.

	Figure 3 Shapes	Figure 4 Peppers	Figure 5 House	Figure 6 Lena
Original	1.65	28.26	18.94	29.28
L_0	84.95	**13.53**	**6.27**	**14.12**
$L_1 + L_2^2$	6.70	23.86	10.77	24.97
L_1	2.73	16.01	7.29	15.72
$L_1 - L_2$	2.10	19.71	9.63	25.62
$L_1 - 0.5L_2$	**1.90**	17.58	7.84	21.64

Table 5

The percentages (%) of nonsparse gradient vectors Du of the deblurring results obtained with BM3D, L_0, $L_{2/3}$, $L_1 + L_2^2$, L_1, $L_1 - L_2$, and $L_1 - 0.5L_2$ regularization terms in comparison to Du of the original image.

	Figure 9 Binary	Figure 10 House	Figure 11 Peppers	Figure 13 Cameraman
Original	0.38	18.94	28.26	22.99
BM3D	6.68	13.44	27.23	17.88
L_0	8.50	14.21	28.93	18.23
$L_{2/3}$	41.09	10.67	21.24	12.71
$L_1 + L_2^2$	8.47	10.73	23.28	11.65
L_1	1.36	3.35	7.72	**3.84**
$L_1 - L_2$	0.83	3.85	8.32	4.91
$L_1 - 0.5L_2$	**0.80**	**3.10**	**7.09**	3.97

iterations was observed numerically to be oscillatory around the ground truth. The deviation occurs at the beginning of outer loops of the DCA. A stopping criterion was introduced based on such an oscillatory pattern of the errors.

In the numerical experiments, we examined three particular applications: image denoising, deblurring, and MRI reconstruction. By design, our method works particularly well for piecewise constant images. For natural images, it improved the classical TV model and is comparable to the state-of-the-art methods. In future work, we plan to carry out a detailed comparison between the DCA and Bregman methods, analyze convergence without strong convexity, accelerate the algorithm, and further study the error pattern and the resulting stopping criterion for other imaging science problems.

Appendix. We want to prove that if $f(u_x, u_y) = |u_x| + |u_y|, g(u_x, u_y) = \sqrt{u_x^2 + u_y^2}$ and $\alpha \in (0, 1)$, then

$$\partial(f - \alpha g) \subseteq \partial f - \alpha \partial g. \tag{A.1}$$

Proof. The discontinuity of both f and g is at $(0, 0)$, which means that we only need to demonstrate (A.1) at the origin. Let (h_x, h_y) be the subgradient of function $(f - \alpha g)$ at zero. By the subgradient's definition, we have

$$|u_x| + |u_y| - \alpha\sqrt{u_x^2 + u_y^2} \geqslant h_x u_x + h_y u_y \quad \forall (u_x, u_y). \tag{A.2}$$

It suffices to discuss the case where one of u_x, u_y is equal to zero. Without loss of generality, $u_x = 0$. Then (A.2) reduces

$$(1-\alpha)|u_y| \geqslant h_y u_y \quad \forall u_y, \tag{A.3}$$

and hence $h_y \in [-1+\alpha, 1-\alpha]$. Similarly, we have $h_x \in [-1+\alpha, 1-\alpha]$ if $u_y = 0$. When (u_x, u_y) is along the x-axis or y-axis, the corresponding subgradient set is $S_0 = [-1+\alpha, 1-\alpha] \times [-1+\alpha, 1-\alpha]$, which shows that $\partial(f-\alpha g) \subseteq S_0$.

On the other hand, we know that $\partial f(0,0) = [-1,1] \times [-1,1]$ and $\partial g(0,0)$ is a unit ball (see (2.19)), so the set of $\partial f - \alpha \partial g$ is

$$S = \bigcup_{a,b\in[-1,1]} \Big\{(x,y)|(a-x)^2 + (b-y)^2 \leqslant \alpha^2\Big\}. \tag{A.4}$$

In other words, the set S consists of all the circles of radius α, each centered inside $[-1,1] \times [-1,1]$, and hence $[-1,1] \times [-1,1]$ is in S.

In summary, we get $\partial(f-\alpha g) \subseteq S_0 \subset [-1,1] \times [-1,1] \subset S = \partial f - \alpha \partial g$. The equality holds for nondegenerate points where $(u_x, u_y) \neq (0,0)$. ■

In fact, we can show that the subgradient of $(f-\alpha g)$ is indeed the set $[-1+\alpha, 1-\alpha] \times [-1+\alpha, 1-\alpha]$ with additional discussion on (u_x, u_y) located at different quadrants. The details are omitted here.

Acknowledgments. We would like to thank the anonymous referees for their useful suggestions, which significantly clarified the presentation of the paper. J. Xin would like to thank Profs. Krishna Nayak and Angel Pineda for their hospitality during a visit to USC in March of 2014 and for their suggestion to consider a weighted variant of $L_1 - L_2$ for compressed sensing and the SSIM measure for image quality evaluation.

REFERENCES

[1] L. Bregman, *The relaxation method of finding the common points of convex sets and its application to the solution of problems in convex programming*, USSR Comp. Math. Math. Phys., 7 (1967), pp. 200–217.

[2] X. Cai, R. Chan, and T. Zeng, *A two-stage image segmentation method using a convex variant of the Mumford–Shah model and thresholding*, SIAM J. Imaging Sci., 6 (2013), pp. 368–390.

[3] E. Candes, J. Romberg, and T. Tao, *Stable signal recovery from incomplete and inaccurate measurements*, Comm. Pure Appl. Math., 59 (2006), pp. 1207–1223.

[4] W. Cao, J. Sun, and Z. Xu, *Fast image deconvolution using closed-form thresholding formulas of $lq(q = 1/2, 2/3)$ regularization*, J. Vis. Commun. Image Represent., 24 (2013), pp. 31–41.

[5] A. Chambolle, *An algorithm for total variation minimization and applications*, J. Math. Imaging Vision, 20 (2004), pp. 89–97.

[6] A. Chambolle and P.-L. Lions, *Image recovery via total variation minimization and related problems*, Numer. Math., 76 (1997), pp. 167–188.

[7] T. F. Chan and J. Shen, *Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods*, SIAM, Philadelphia, 2005.

[8] T. F. Chan and J. Shen, *Mathematical models for local nontexture inpainting*, SIAM J. Appl. Math., 62 (2002), pp. 1019–1043.

[9] T. F. Chan and C. K. Wong, *Total variation blind deconvolution*, IEEE Trans. Image Process., 7 (1998), pp. 370–375.

[10] R. Chartrand, *Exact reconstruction of sparse signals via nonconvex minimization*, IEEE Trans. Signal Process., 10 (2007), pp. 707–710.
[11] Y. Chen, W. W. Hager, M. Yashtini, X. Ye, and H. Zhang, *Bregman operator splitting with variable stepsize for total variation image reconstruction*, Comput. Optim. Appl., 54 (2013), pp. 317–342.
[12] R. Choksi, Y. van Gennip, and A. Oberman, *Anisotropic total variation regularized l^1-approximation and denoising/deblurring of 2D bar codes*, Inverse Probl. Imaging, 3 (2011), pp. 591–617.
[13] K. Dabov, A. Foi, and K. Egiazarian, *Image restoration by sparse 3D transform-domain collaborative filtering*, in Proceedings of SPIE Electronic Imaging, San Jose, CA, 2008, 6812-07.
[14] D. L. Donoho, *Compressed sensing*, IEEE Trans. Inform. Theory, 52 (2006), pp. 1289–1306.
[15] S. Esedoḡlu and S. J. Osher, *Decomposition of images by the anisotropic Rudin-Osher-Fatemi model*, Comm. Pure Appl. Math, 57 (2003), pp. 1609–1626.
[16] E. Esser, Y. Lou, and J. Xin, *A method for finding structured sparse solutions to nonnegative least squares problems with applications*, SIAM J. Imaging Sci., 6 (2013), pp. 2010–2046.
[17] D. J. Eyre, *An Unconditionally Stable One-Step Scheme for Gradient Systems*, manuscript, 1998.
[18] S. Geman and D. Geman, *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*, IEEE Trans. Pattern Anal. Machine Intell., 6 (1984), pp. 721–741.
[19] T. Goldstein and S. Osher, *The split Bregman method for L1-regularized problems*, SIAM J. Imaging Sci., 2 (2009), pp. 323–343.
[20] A. Hantoute, M. A. López, and C. Zălinescu, *Subdifferential calculus rules in convex analysis: A unifying approach via pointwise supremum functions*, SIAM J. Optim., 19 (2008), pp. 863–882.
[21] L. He, A. Marquina, and S. Osher, *Blind deconvolution using TV regularization and Bregman iteration*, Internat. J. Imaging Syst. Tech., 15 (2005), pp. 74–83.
[22] Q. He and J. Xin, *A randomly perturbed INFOMAX algorithm for blind source separation*, in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2013, IEEE, Washington, DC, pp. 3218–3222.
[23] N. Hurley and S. Rickard, *Comparing measures of sparsity*, IEEE Trans. Inform. Theory, 55 (2009), pp. 4723–4741.
[24] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, *Optimization by simulated annealing*, Science, 220 (1983), pp. 671–680.
[25] D. Krishnan and R. Fergus, *Fast image deconvolution using hyper-Laplacian priors*, in Advances in Neural Information Processing Systems (NIPS), Vancouver, Canada, 2009, pp. 1033–1041.
[26] D. Krishnan, T. Tay, and R. Fergus, *Blind deconvolution using a normalized sparsity measure*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Providence, RI), IEEE, Washington, DC, 2011, pp. 233–240.
[27] Y. Lou, P. Yin, Q. He, and J. Xin, *Computing sparse representation in a highly coherent dictionary based on difference of L_1 and L_2*, J. Sci. Comput., 64 (2015), pp. 178–196.
[28] M. Lysaker, S. Osher, and X. C. Tai, *Noise removal using smoothed normals and surface fitting*, IEEE Trans. Image Process., 13 (2004), pp. 1345–1357.
[29] A. Marquina, *Nonlinear inverse scale space methods for total variation blind deconvolution*, SIAM J. Imaging Sci., 2 (2009), pp. 64–83.
[30] A. Marquina and S. Osher, *Image super-resolution by TV-regularization and Bregman iteration*, J. Sci. Comput., 37 (2008), pp. 367–382.
[31] 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 Proceedings of the International Conference on Computer Vision, Vol. 2, 2001, pp. 416–423.
[32] D. Mumford and J. Shah, *boundary detection by minimizing functionals*, in Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, Canada, 1985, pp. 137–154.
[33] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, *An iterated regularization method for total variation-based image restoration*, Multiscale Model. Simul, 4 (2005), pp. 460–489.
[34] J. Portilla, *Image restoration through l0 analysis-based sparse optimization in tight frames*, in Proceedings of the IEEE International Conference on Image Processing (Cairo, Egypt), IEEE, Washington, DC, 2009, pp. 3909–3912.

[35] R. B. Potts, *Some generalized order-disorder transformations*, Proc. Cambridge Philos. Soc., 48 (1952), pp. 106–109.
[36] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Grundlehren Math. Wiss. 317, Springer-Verlag, Berlin, 1998.
[37] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), pp. 259–268.
[38] S. Setzer, *Split Bregman algorithm, Douglas-Rachford splitting and frame shrinkage*, in Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 5567, Springer, Berlin, 2009, pp. 464–476.
[39] M. Storath, A. Weinmann, and L. Demaret, *Jump-sparse and sparse recovery using Potts functionals*, IEEE Trans. Signal Process., 62 (2014), pp. 3654–3666.
[40] X. C. Tai and C. Wu, *Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model*, in Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 5567, Springer, Berlin, 2009, pp. 502–513.
[41] P. Tao and L. T. H. An, *A D.C. optimization algorithm for solving the trust-region subproblem*, SIAM J. Optim., 8 (1998), pp. 476–505.
[42] P. D. Tao and L. T. H. An, *Convex analysis approach to d.c. programming: Theory, algorithms and applications*, Acta Math. Vietnam., 22 (1997), pp. 289–355.
[43] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, *Image quality assessment: From error visibility to structural similarity*, IEEE Trans. Image Process., 13 (2004), pp. 600–612.
[44] C. Wu and X. C. Tai, *Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models*, SIAM J. Imaging Sci., 3 (2010), pp. 300–339.
[45] L. Xu, C. Lu, Y. Xu, and J. Jia, *Image smoothing via L_0 gradient minimization*, in Proceedings of the 2011 SIGGRAPH Asia Conference, ACM Trans. Graphics, 30 (2011), 174.
[46] Z. Xu, X. Chang, F. Xu, and H. Zhang, *$L_{1/2}$ regularization: A thresholding representation theory and a fast solver*, IEEE Trans. Neural Networks, 23 (2012), pp. 1013–1027.
[47] P. Yin, E. Esser, and J. Xin, *Ratio and difference of L_1 and L_2 norms and sparse representation with coherent dictionaries*, Commun. Inf. Syst., 14 (2014), pp. 87–109.
[48] P. Yin, Y. Lou, Q. He, and J. Xin, *Minimization of ℓ_{1-2} for compressed sensing*, SIAM J. Sci. Comput., 37 (2015), pp. A536–A563.