Journal of Inequalities and Applications
a SpringerOpen Journal

RESEARCH Open Access

A new reweighted l_1 minimization algorithm for image deblurring

Tiantian Qiao[1,2*], Boying Wu[2], Weiguo Li[1] and Alun Dong[1]

*Correspondence: qtthsnxf@126.com
[1]College of Science, China University of Petroleum, Changjiangxi Road 66, Qingdao, 266580, P.R. China
[2]Department of Mathematics, Harbin Institute of Technology, West Dazhi Street 92, Haerbin, 150001, P.R. China

Abstract

In this paper, a new reweighted l_1 minimization algorithm for image deblurring is proposed. The algorithm is based on a generalized inverse iteration and linearized Bregman iteration, which is used for the weighted l_1 minimization problem $\min_{u\in\mathbb{R}^n}\{\|u\|_\omega : Au = f\}$. In the computing process, the effective using of signal information can make up the detailed features of image, which may be lost in the deblurring process. Numerical experiments confirm that the new reweighted algorithm for image restoration is effective and competitive to the recent state-of-the-art algorithms.

Keywords: reweighted l_1 minimization; generalized inverse; linearized Bregman iteration; image deblurring

1 Introduction

Image deblurring is a fundamental problem in image processing, since many real-life problems can be modeled as deblurring problems [1]. In this paper, a new reweighted l_1 minimization algorithm for image deblurring is proposed. The algorithm is obtained based on a generalized inverse iteration and a linearized Bregman iteration.

Simply, we shall denote images as vectors in $\mathbb{R}^n$ by concatenating their columns. Let $u \in \mathbb{R}^n$ be the underlying image. Then the observed blurred image $f \in \mathbb{R}^n$ is given by

$$f = Au + \eta, \tag{1.1}$$

where $\eta \in \mathbb{R}^n$ is an additive noise and $A \in \mathbb{R}^{m\times n}$ is a linear blurring operator. This problem is ill-posed due to the large condition number of the matrix A. Any small perturbation on the observed blurred image f may cause the direct solution $A^{-1}f$, which is very difficult to obtain from the original image u [2]. This is a widely studied subject and many corresponding approaches have been developed, and one of them is to minimize some cost functionals [1]. The simplest method is a Tikhonov regularization, which minimizes an energy consisting of a data fidelity term and an l_2 norm regularization term. A is a convolution, which can solve the problem in the Fourier domain. In this case, the method is called a Wiener filter [3], this is a linear method, and the edges of restored image are usually smeared. To overcome this, a total variation (TV)-based regularization was proposed by Rudin *et al.* in [4], which is known as the ROF model. Due to its virtue of preserving edges, it is widely used in image processing, such as blind deconvolution, inpainting, and superresolution; see [1]. However, as we know, for the TV yields staircasing [5, 6], these TV-based methods do not preserve the fine structures, details, and textures. To avoid these drawbacks, nonlocal methods were proposed for denoising [7, 8], and then extended to deblurring [9]. Also, the Bregman iteration, introduced to image science [10], was shown to improve TV-based blind deconvolution [11–13]. Recently, a nonlocal TV regularization was invented based on graph theory [14] and applied to image deblurring [15]. Another approach for deblurring is the wavelet-based method, *etc.* [16].

Normally, the original image $u \in \mathbb{R}^n$ will be found by solving the following constrained minimization problem:

$$\min_{u \in \mathbb{R}^n} \{ J(u) : Au = f \}, \tag{1.2}$$

where $J(u)$ is a continuous convex function, and when $J(u)$ is strictly or strongly convex, the solution of (1.2) is unique.

This constrained optimization problem (1.2) arise in many applications, like in image compression, reconstruction, inpainting, segmentation, compressed sensing, *etc.* The problem (1.2) can be transformed into a linear programming problem, and then solved by a conventional linear programming solver in many cases. Recently, fixed-point continuation method [17] and Bregman iteration [18] are very popular. Specially, Bregman iterative regularization was proposed by Osher *et al.* [10]. In the past few years, a series of new methods have been developed, and among them, the linearized Bregman method [19–22] and the split Bregman method [23–26] got most attention.

Specially, when $J(u) = \|u\|_1$, the problem (1.2) becomes

$$\min_{u \in \mathbb{R}^n} \{ \|u\|_1 : Au = f \}. \tag{1.3}$$

Obviously, the problem (1.3) is an l_1-norm minimization problem. Since many practical problems related to the sparsity of the solution make the problem (1.3) stay on focus for years, like in signal processing, compressive sensing *etc.* [18, 19]. Similar to the problem (1.2), the problem (1.3) also can be transformed into a linear program and then solved by conventional linear programming solvers. However, such solvers are not tailored for the matrix A that is large-scale and completely dense. Fortunately, the problem (1.3) can be solved very effectively by the linearized Bregman method [19–22, 27]. The computing speed of its simplified form with soft threshold operator is faster [19, 21, 22]. The corresponding convergence analysis was discussed in [20].

In this paper we highlight numerical computation of coefficient in sparse reconstruction methods for image deblurring, described by an operator $\Phi : X \to Y$ between Hilbert spaces X and Y. We seek sparse solutions in an orthogonal basis $\{\psi_j\}_{j\in N}$. The standard approach is the weighted ℓ_1 minimization (1.3):

$$\min_{u \in \ell^2(N) \cap \ell^1_\omega(N)} \left\{ \frac{1}{2} \left\| \sum_j u_j \Phi \psi_j - f \right\|^2 + \alpha \sum_j \omega_j |u_j| \right\}. \tag{1.4}$$

Here $\ell^1_\omega(N)$ denotes the space of coefficients u_j such that $\sum_j \omega_j |u_j| < \infty$. In order to simplify the notation we introduce the operator $A : \ell^2(N) \to Y$, $(u_j) \to \sum_j u_j \Phi \psi_j$. Moreover, we will assume that $\{\omega_j\}_{j\in\mathbb{N}}$ entail positive weights and there is a constant $\omega_0 > 0$ such that $\omega_j \geq \omega_0$ for all $j \in N$. Hence $\sum_j \omega_j |u_j|$ is really a norm on $\ell^1(N)$, denoted by $\|u\|_\omega$. Then the ℓ_1 minimization can be rewritten as

$$\min_{u\in\ell^2(N)\cap\ell^1_\omega(N)}\left\{\alpha\|u\|_\omega+\frac{1}{2}\|Au-f\|^2\right\}. \tag{1.5}$$

Naturally one can set $\omega_{k+1}(i)=\frac{1}{|u_k(i)|}$. Then we can see the weighted ℓ_1 norm as a kind of approximation to ℓ_0 norm, but we can easily note that when $u_k(i)=0$, $\omega_{k+1}(i)$ is not well defined. The good news is we can regularize it as $\omega_{k+1}(i)=\frac{1}{|u_k(i)|+\epsilon}$, where $\epsilon>0$ is a small number [28]. So in this paper we set

$$\omega_{k+1}(i)=\frac{1}{|u_k(i)|+\epsilon}.$$

On this basis, the authors propose a new reweighted l_1 minimization method to solve the problem (1.5) and illustrate by numerical experiments.

The rest of the paper is organized as follows. In Section 2, we summarize the existing methods for solving the constrained problem (1.3). In Section 3, the generalized shrinkage operator is proposed. The new algorithm is proposed in Section 4. Numerical results are shown in Section 5. Finally, we draw some conclusions in Section 6.

2 Preliminaries

2.1 Generalized inverse

We are interested in the iterative formula of the generalized inverse, because it is used by our new algorithm. Therefore, before we give a detailed discussion, we first give some definitions and lemmas.

Definition 2.1 [29] Let $A\in\mathbb{C}^{m\times n}$, then X is called the pseudoinverse of A and denoted by $A^\dagger$. If X satisfies the following properties, *i.e.*, the Moore-Penrose conditions:

$$\begin{aligned}&1.\quad AXA=A,\\&2.\quad XAX=X,\\&3.\quad (AX)^*=AX,\\&4.\quad (XA)^*=XA.\end{aligned} \tag{2.1}$$

Remark 2.1 The inner inverse is not unique. In general, the set of the inner inverses of the matrix A is denoted A^-.

Definition 2.2 [29] Let $A,B\in\mathbb{C}^{n\times m}$, the set

$$\mu(A,B)=\left\{X|X=AYB, Y\in\mathbb{C}^{m\times n}\right\} \tag{2.2}$$

is called the range of (A,B).

Lemma 2.1 [30] *Let* $A\in\mathbb{C}^{m\times n}\neq 0$; *if initial matrix* V_0 *satisfies*

$$V_0\in\mu\left(A^*,A^*\right), \tag{2.3}$$

$$\rho(I-AV_0)<1, \tag{2.4}$$

where I is an identity matrix with the same dimension as matrix A and A^* is the conjugate transpose of matrix A. Then the sequence $\{V_q\}_{q\in\mathbb{N}}$ generated by

$$V_{q+1} = V_q + V_0(I - AV_q), \quad q = 1, 2, \ldots \tag{2.5}$$

is convergent to $A^\dagger$.

2.2 Linearized Bregman iteration

The Bregman distance [31], based on the convex function J, between points u and v, is defined by

$$D_J^p(u,v) = J(u) - J(v) - \langle p, u - v\rangle, \tag{2.6}$$

where $p \in \partial J(v)$ is an element in the subgradient set of J at the point v. In general $D_J^p(u,v) \neq D_J^p(v,u)$ and the triangle inequality is not satisfied, so $D_J^p(u,v)$ is not a distance in the usual sense. For details, see [31].

To solve (1.3), in [19] the *linearized Bregman iteration* is generated by

$$\begin{cases} u^{k+1} = \arg\min_u\{\mu D_J^{p^k}(u,u^k) + \frac{1}{2\delta}\|u - (u^k - \delta A^T(Au^k - f))\|^2\}, \\ p^{k+1} = p^k - \frac{1}{\mu\delta}(u^{k+1} - u^k) - \frac{1}{\mu}A^T(Au^k - f), \quad p^k \in \partial J(u^k), \end{cases} \tag{2.7}$$

where δ is a constant and $p^0 = u^0 = 0$. Hereafter, we use $\|\cdot\| = \|\cdot\|_2$ to denote the l_2 norm.

When $J(u) = \|u\|_1$, algorithm (2.7) can be rewritten as

$$\begin{cases} v^{k+1} = v^k + A^T(f - Au^k), \\ u^{k+1} = \delta T_\mu(v^{k+1}), \end{cases} \tag{2.8}$$

where $u^0 = v^0 = 0$, and

$$T_\lambda(\omega) := \bigl[t_\lambda(\omega(1)), t_\lambda(\omega(2)), \ldots, t_\lambda(\omega(n))\bigr]^T \tag{2.9}$$

is the soft thresholding operator [18] with

$$t_\lambda(\xi) = \begin{cases} 0, & |\xi| \leq \lambda, \\ \operatorname{sgn}(\xi)(|\xi| - \lambda), & |\xi| > \lambda. \end{cases} \tag{2.10}$$

Namely, the algorithm (2.8) is called an A^T *linearized Bregman iteration.*

Subsequently, when A is any matrix, the constraint condition $Au = f$ of the problem (1.3) is not satisfied. So the conditions will be extended to solve the least-squares problem $\min_{u\in\mathbb{R}^n} \|Au - f\|^2$, and the algorithm becomes the following $A^\dagger$ *linearized Bregman iteration* [22]:

$$\begin{cases} f^{k+1} = f^k + f - Au^k, \\ u^{k+1} = \delta T_\mu(A^\dagger f^{k+1}), \end{cases} \tag{2.11}$$

where $A^\dagger$ is generalized inverse of matrix A.

3 The generalized shrinkage operator

Theorem 3.1 $T_\mu(v) = \arg\min_{u\in\mathbb{R}^n}\{\mu\|u\|_1 + \frac{1}{2}\|u-v\|^2\}$.

Proof Let $f(u) = \mu\|u\|_1 + \frac{1}{2}\|u-v^k\|^2 = \mu\sum_{i=1}^n |u_i| + \frac{1}{2}\sum_{i=1}^n (v_i^k - u_i)^2$, then we have

$$\frac{\partial f(u)}{\partial u_i} = \begin{cases} \mu + u_i - v_i^k, & u_i > 0, \\ -\mu + u_i - v_i^k, & u_i < 0. \end{cases} \tag{3.1}$$

Case 1: $v_i^k > \mu > 0$.

(1) If $u_i > 0$, and notice that $\frac{\partial f(u)}{\partial u_i} = 0$ then $u_i = v_i^k - \mu > 0$, for this case $f(u)$ gets its minimum at point $u_i = v_i^k - \mu$ along the direction e_i and the minimum is

$$f(u)|_{u_i=v_i^k-\mu} = \mu\left(v_i^k - \mu\right) + \frac{1}{2}\mu^2 + \delta_1\ (>0) = \Delta_1 + \delta_1. \tag{3.2}$$

(2) If $u_i < 0$, and notice that $\frac{\partial f(u)}{\partial u_i} = u_i - v_i^k - \mu < 0$, again we find that $f(u)$ decreases along the direction e_i:

$$f(u)|_{u_i=0} = \frac{1}{2}\left(v_i^k\right)^2 + \delta_1\ (>0) = \Delta_2 + \delta_1. \tag{3.3}$$

Since $\Delta_2 - \Delta_1 = \frac{1}{2}(v_i^k)^2 - (\mu v_i^k - \frac{1}{2}\mu^2) = \frac{1}{2}(v_i^k - \mu)^2 > 0$, along the direction e_i we find that the minimizer of $f(u)$ is $u_i = v_i^k - \mu$.

Case 2: $v_i^k < -\mu < 0$.

(1) If $u_i > 0$, since $\frac{\partial f(u)}{\partial u_i} = u_i - v_i^k + \mu > 0$, $f(u)$ increases along the direction e_i:

$$f(u)|_{u_i=0} = \frac{1}{2}\left(v_i^k\right)^2 + \delta_3 = \Delta_3 + \delta_3. \tag{3.4}$$

(2) If $u_i < 0$, since $\frac{\partial f(u)}{\partial u_i} = 0$ we have $u_i = v_i^k + \mu < 0$, the minimizer of $f(u)$ along the direction e_i is $u_i = v_i^k + \mu$ and the corresponding minimum is

$$f(u)|_{u_i=v_i+\mu} = -\mu\left(v_i^k + \mu\right) + \frac{1}{2}\mu^2 + \delta_3 = \Delta_4 + \delta_3. \tag{3.5}$$

Since $\Delta_3 - \Delta_4 = \frac{1}{2}(v_i^k)^2 + \mu(v_i^k + \mu) - \frac{1}{2}\mu^2 = \frac{1}{2}(v_i^k + \mu)^2 > 0$, we can get the minimizer of $f(u)$ at $u_i = v_i^k + \mu$ along the direction e_i.

Case 3: $-\mu \le v_i^k \le \mu$.

(1) If $u_i > 0$, since $\frac{\partial f(u)}{\partial u_i} = u_i - v_i^k + \mu > 0$, $f(u)$ increases along the direction e_i:

$$f(u)|_{u_i=0} = \frac{1}{2}\left(v_i^k\right)^2 + \delta. \tag{3.6}$$

(2) If $u_i < 0$, since $\frac{\partial f(u)}{\partial u_i} = u_i - v_i^k - \mu < 0$, $f(u)$ decreases along the direction e_i:

$$f(u)|_{u_i=0} = \frac{1}{2}\left(v_i^k\right)^2 + \delta, \tag{3.7}$$

when $u_i = 0$, the minimum of $f(u)$ along the direction e_i is $f(u) = \frac{1}{2}(v_i^k)^2 + \delta$.

In conclusion, we have the following soft shrinkage operator:

$$t_\mu(\xi) = \begin{cases} 0, & |\xi| \le \mu, \\ \operatorname{sgn}(\xi)(|\xi| - \mu), & |\xi| > \mu. \end{cases} \tag{3.8}$$

The minimizer of the minimization problem is given by

$$\begin{aligned} u &= \arg\min\left\{\mu|u| + \frac{1}{2}\left(u - v^k\right)^2 \middle| u \in \mathbb{R}^n, v^k \in \mathbb{R}^n\right\} \\ &= \begin{cases} v_i^k - \mu, & v_i^k > \mu > 0, \\ 0, & -\mu \le v_i^k \le \mu, \\ v_i^k + \mu, & v_i^k < -\mu < 0 \end{cases} \\ &= \left[t_\mu(\omega_1), t_\mu(\omega_2), \dots, t_\mu(\omega_n)\right]^T \\ &= T_\mu\left(v^k\right). \end{aligned} \tag{3.9}$$

□

The unknown variable u is component-wise separable in the problem

$$u = \arg\min_{u \in \ell^2(N) \cap \ell^1_\omega(N)} \left\{\mu\|u\|_\omega + \frac{1}{2}\|u - v\|^2\right\} \tag{3.10}$$

for any $v \in \ell^2(N) \cap \ell^1_\omega(N)$ and $\omega > 0$. Then each of its components u_i can be independently obtained by the shrinkage operation, which is also referred as soft thresholding [32]:

$$u_i = T_{\mu\omega_i}(v_i) = \operatorname{shrink}(v_i, \mu\omega_i), \quad i = 1, 2, \dots. \tag{3.11}$$

For v_i, ω_i and $\mu \in R$, we define $u_i \in R$

$$\begin{aligned} u_i &= \operatorname{shrink}(v_i, \mu\omega_i) := \operatorname{sgn}(v_i) \max\left\{|v_i| - \mu\omega_i, 0\right\} \\ &= \begin{cases} v_i - \mu\omega_i, & v_i > \mu\omega_i, \\ 0, & -\mu\omega_i \le v_i \le \mu\omega_i, \\ v_i + \mu\omega_i, & v_i < -\mu\omega_i. \end{cases} \end{aligned} \tag{3.12}$$

The generalized shrinkage operator leads to the sparse solution and removes noises. Hence, the algorithm with the generalized shrinkage operator converges to a sparse solution and is robust to noises.

4 The new reweighted l_1 minimization algorithm

The sequence $\{u^k\}$ given by $A^\dagger$ *linearized Bregman iteration* converges to an optimal solution of the problem (1.3). The computation of generalized inverse $A^\dagger$ is time consuming; to overcome this, a method called *chaotic iterative algorithm* is proposed combined with (2.5). In this algorithm we just need matrix-vector multiplication, so the generalized inverse $A^\dagger$ can be computed efficiently. In order to understand the algorithm better, we give a brief description of this method as follows:

$$\begin{cases} f^{k+1} = f^k + (f - Au^k), \\ y^{k+1} = y^k + V_0 f^{k+1} - V_0(Ay^k), \quad k = 0, 1, 2, \dots, \\ u^{k+1} = \delta T_\mu(y^{k+1}), \end{cases} \tag{4.1}$$

where $y^0 = V_0 f^0$, $V_0 = \alpha A^*$ and $0 < \alpha < \frac{2}{\|A\|^2}$. The corresponding sequence $\{u^k\}$ also converges to an optimal solution of the problem (1.3).

Here we first study an iteratively reweighted least-squares (IRLS) method [33] for robust statistical estimation. Considering a regression problem $Ax = b$ where the observation matrix A is underdetermined; it was noticed as regards a standard least-squares regression, in which $\|r\|_2$ is minimized where $r = Ax - b$ is the residual vector. To overcome the problem of lacking of robustness of the algorithm, IRLS was proposed as an iterative method to

$$\min_x \sum_i \rho\big(r_i(x)\big), \tag{4.2}$$

where $\rho(\cdot)$ is a penalty function such as the ℓ_1 norm. This minimization can be accomplished by solving a sequence of weighted least-squares problems where the weights $\{w_i\}$ depend on the previous residuals $w_i = \rho'(r_i)/r_i$. The typical choice of ρ is inversely proportional to the residual, so that the large residuals will be penalized less in the subsequent iterations. Then an IRLS involving an iteratively reweighted ℓ_2-norm can be better approximated by an ℓ_1-like criterion. Inspired by the above idea, in order to better approximate an ℓ_0-like criterion [34], our algorithm involves the iteratively reweighted ℓ_1-norm.

Since that reweighted minimization can enhance the sparsity and the chaotic iterative algorithm can reduce the computational complexity of the generalized inverse $A^\dagger$, we iteratively solve the following weighted ℓ_1 minimization problem:

$$\min_u \big\{ \|u\|_\omega : Au = f \big\}. \tag{4.3}$$

We refine the chaotic iterative algorithm, and obtain a new *reweighted l_1 minimization algorithm* as follows:

$$\begin{cases} f^{k+1} = f^k + (f - Au^k), \\ y^{k+1} = y^k + V_0 f^{k+1} - V_0(Ay^k), \\ u^{k+1} = \delta T_{\mu\omega^k}(y^{k+1}), \\ \omega_i^{k+1} = 1/(|u_i^{k+1}| + \epsilon), \quad i = 1, \ldots, n, \end{cases} \quad k = 0, 1, 2, \ldots, \tag{4.4}$$

where $y^0 = V_0 f^0$, $V_0 = \alpha A^*$, and $0 < \alpha < \frac{2}{\|A\|^2}$.

5 Numerical experiments

In this section, we test the reweighted l_1 minimization algorithm for the problem (4.3). We used Word image. Here Word is a 256×256 sparse image. In our experiments we tested several kinds of blurring kernels including disk, Gaussian, and motion. We compare different algorithms through both visual effects and quality measurements. Here, the quality of restoration is measured by the signal-to-noise ratio (SNR), defined by

$$\mathrm{SNR} = 10 \times \ln \frac{\sum_{i=1}^m \sum_{j=1}^n (u^*(i,j) - \mathrm{mean}(u^*))^2}{\sum_{i=1}^m \sum_{j=1}^n (u^*(i,j) - u^0(i,j) - \mathrm{mean}(u^* - u^0))^2}, \tag{5.1}$$

where u^*, u^0, and mean$(\cdot)$ are the restored image, original image, and average operator, respectively.

Our code is written in MATLAB and run on a Windows PC with a Intel(R) Core(TM) 2 Duo CPU T8100 @ 2.10 GHz 2.10 GHz and 1.5 GB memory. The MATLAB version is 7.1.

Reweighted l_1 minimization algorithm:

Step 1. Set $u^0 = 0, f^0 = 0, y^0 = V_0 f^0$, $V_0 = \alpha A^T$, $0 < \alpha < \frac{2}{\|A\|_2^2}$, $0 < \delta < 1$, μ = parameter.

Step 2. The sequence $\{u^k\}_{k\in\mathbb{N}}$ generated by (4.4).

Step 3. Until $\frac{\|u^{k+1}-u^k\|}{\|u^k\|} < \epsilon$.

We demonstrate the performance of the reweighted l_1 minimization algorithm, the chaotic iterative algorithm, the A^T Bregman iteration, and the $A^\dagger$ Bregman iteration with pinv(A) in MATLAB.

In the first experiment, the images we used were blurred with a 'disk' kernel of hsize = 15. The blurry and restored images are presented in Figure 1. By comparing these three algorithms, it is clear that the reweighted l_1 minimization algorithm performs better in terms of SNR than the chaotic iterative algorithm, and the A^T Bregman iteration lemma is a little slower than the chaotic iterative algorithm and the A^T Bregman iteration, which is still acceptable.

In the second experiment the images were blurred with a 'Gaussian' kernel of hsize = 7. The results are shown in Figure 2. The comparison of the restored effect and the computing time is basically the same as the first one.

In the third experiment we used a part of the Word image blurred with a 3×5 'motion' kernel to better show the local information of the recovered image. The restored small sparse Word images after using the reweighted l_1 minimization algorithm, the chaotic iterative algorithm, the A^T Bregman iteration, and the $A^\dagger$ Bregman iteration are plotted in Figure 3. Again we obtain a similar conclusion to the above experiments.

In fact, the complexity analysis also shows comparative results of several methods. Set the same loop number is K. So, the workload of the $A^\dagger$ algorithm (2.11) is two parts. They are the workload of the $A^\dagger$ and the loop of the (2.11). The workload is $O(n^3)$ during the

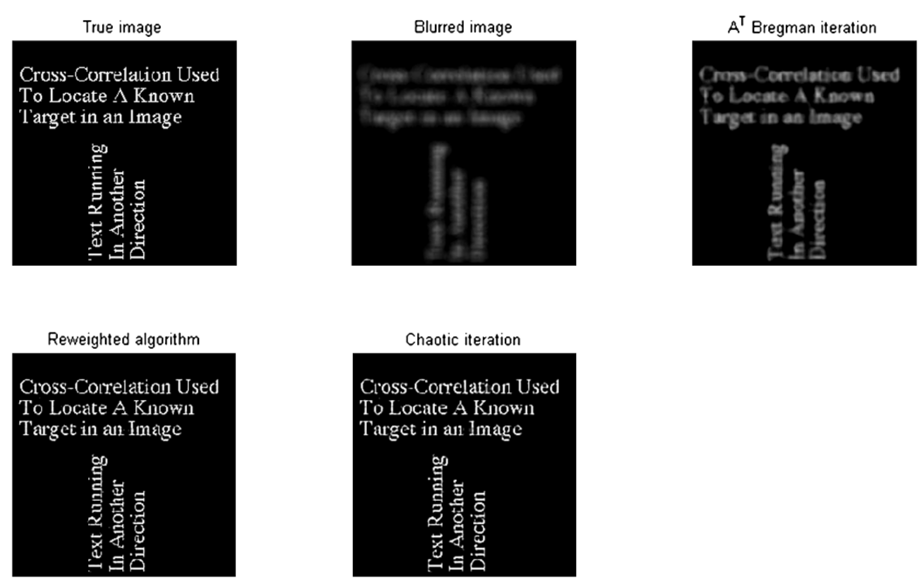

Figure 1 Deblurring results of 256 × 256 sparse Word image convolved by a 15 × 15 disk kernel generated by the MATLAB command fspecial('disk', 7). Upper left: original image; upper middle: blurred image. The other three are reconstructed images, respectively, by an A^T Bregman iteration, a reweighted ℓ_1 minimization algorithm, and a chaotic iteration.

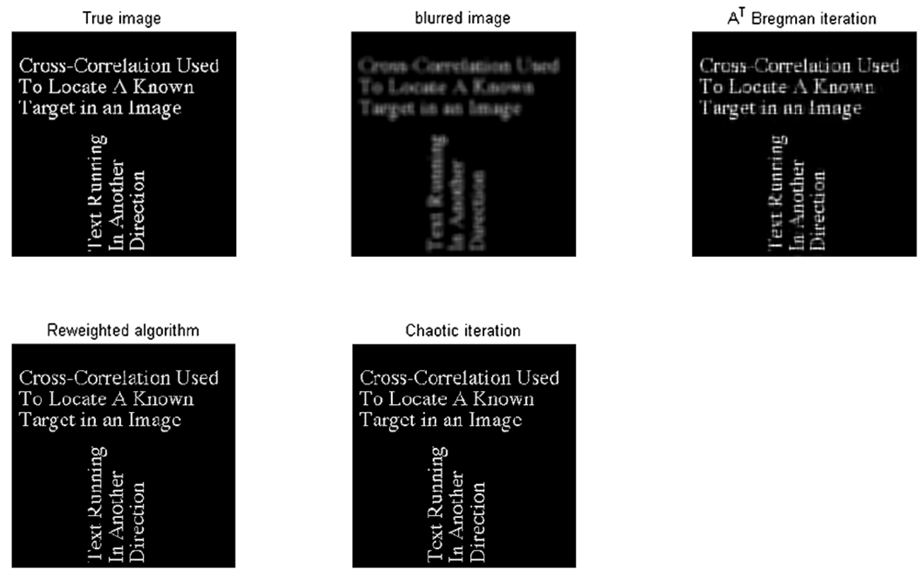

Figure 2 Deblurring results of 256 × 256 sparse Word image convolved by a 7 × 7 Gaussian kernel generated by the MATLAB command fspecial('Gaussian', 7, 15). Upper left: original image; upper middle: blurred image. The other three are reconstructed images, respectively, by the A^T Bregman iteration, the reweighted ℓ_1 minimization algorithm, and the chaotic iteration.

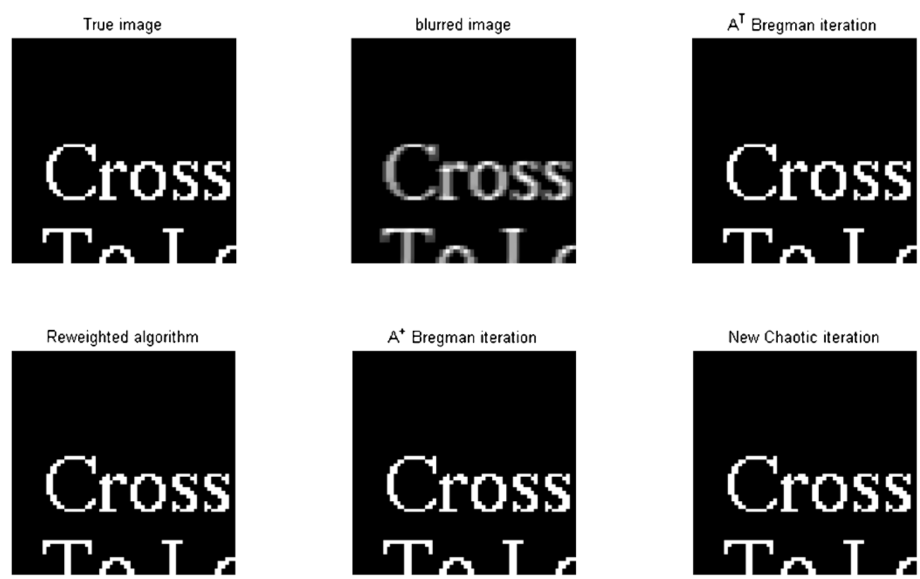

Figure 3 Deblurring results of 64 × 80 part of sparse Word image convolved by a 3 × 5 motion kernel generated by the MATLAB command fspecial('motion', 5, 7). Upper left: original image; upper middle: blurred image. The other four are reconstructed images, respectively, by the A^T Bregman iteration, the reweighted ℓ_1 minimization algorithm, the $A^\dagger$ Bregman iteration, and the chaotic iteration.

computation of $A = USV$, $A^\dagger = V^T S^\dagger U^T$, when $m < n$, because of the singular value decomposition involving multiplication of the matrix and matrix and eigenvalue calculation. The workload of the loop of the (2.11) is $O(m * n * K)$, because the loop only contains multiplication of matrix and vector. Therefore, the total workload of the $A^\dagger$ algorithm (2.11) is $O(n^3) + O(m * n * K)$. The workload of the chaotic iteration (4.1), the reweighted l_1

Table 1 The comparison of different algorithms

Algorithm	Image scale	Blur kernel	Time (s)	SNR
A^T Bregman iteration	256 × 256	15 × 15 'disk'	98.580627	2.285
$A^\dagger$ Bregman iteration			116.845442	9.4495
Chaotic iteration			8.8303	114.648685
A^T Bregman iteration	256 × 256	7 × 15 'Gaussian'	51.934199	5.6389
$A^\dagger$ Bregman iteration			63.003234	15.1254
Chaotic iteration			63.68804	13.2921
A^T Bregman iteration	64 × 80	3 × 5 'motion'	0.521046	26.5770
$A^\dagger$ Bregman iteration			17.214257	47.7984
Chaotic iteration			0.617180	62.4899
Reweighted ℓ_1 algorithm			0.631006	63.5906

minimization algorithm (4.4) and the A^T Bregman iteration (2.8) are $O(m * n * K)$, respectively. Obviously, $K < m \ll n$, the workload of the $A^\dagger$ algorithm (2.11) is bigger than the other three algorithms.

All the experiment data are listed in Table 1. In summary, for the restored quality of the three methods we have Reweighted > Chaotic > $A^\dagger \gg A^T$, while for the computing time the order of magnitude is about $1 : 1 : 10^2 : 1$. The numerical examples illustrate that the new reweighted l_1 minimization algorithm is fast and efficient for deblurring the image. It is a very useful method.

6 Conclusion

In this paper, we propose the reweighted l_1 minimization algorithm for image deblurring. Above all, we can see that the recovery of the image effect is obvious. Especially in the case of a large degree of blurring and difficult to recover details, it is stable and effective. In addition, we can improve the efficiency of this reweighted l_1 minimization algorithm combining with the 'kicking' technology. Because of the scale factor and efficiency of the algorithm $A^\dagger$, the new method proposed in this paper can be used in a parallel operation to get a better algorithm.

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

Acknowledgements
This research was partly supported by Fund of Oceanic Telemetry Engineering and Technology Research Center, State Oceanic Administration (grant no. 2012003), the NSFC (grant nos. 60971132,61101208) and Fundamental Research Funds for the Central Universities (grant no. 13CX02086A).

Received: 22 February 2013 Accepted: 16 March 2014 Published: 16 June 2014

References

1. Chan, TF, Shen, J: Image Processing and Analysis. SIAM, Philadelphia (2005)
2. Aubert, G, Kornprobst, P: Mathematical Problems in Image Processing, 2nd edn. Appl. Math. Sci., vol. 147. Springer, New York (2006)
3. Andrews, HC, Hunt, BR: Digital Image Restoration. Prentice Hall, Englewood Cliffs (1977)
4. Rudin, L, Osher, S, Fatemi, E: Nonlinear total variation based noise removal algorithms. Physica D **60**, 259-268 (1992)
5. Dobson, DC, Santosa, F: Recovery of blocky images from noise and blurred data. SIAM J. Appl. Math. **56**, 1181-1198 (1996)
6. Nikolova, M: Local strong homogeneity of a regularized estimator. SIAM J. Appl. Math. **61**, 633-658 (2000)
7. Buades, A, Coll, B, Morel, JM: A review of image denoising algorithms, with a new one. Multiscale Model. Simul. **4**, 490-530 (2005)
8. Tomasi, C, Manduchi, R: Bilateral filtering for gray and color images. In: Proceedings of the 1998 IEEE International Conference on Computer Vision, Bombay, India (1998)

9. Buades, A, Coll, B, Morel, JM: Image enhancement by non-local reverse heat equation. CMLA Tech. rep. 22 (2006)
10. Osher, S, Burger, M, Goldfarb, D, Xu, J, Yin, W: An iterative regularization method for total variation-based image restoration. Multiscale Model. Simul. **4**, 460-489 (2005)
11. He, L, Marquina, A, Osher, S: Blind deconvolution using TV regularization and Bregman iteration. Int. J. Imaging Syst. Technol. **15**, 74-83 (2005)
12. Marquina, A: Inverse scale space methods for blind deconvolution. UCLA-CAM-Report 06-36 (2006)
13. Marquina, A, Osher, S: Image super-resolution by TV-regularization and Bregman iteration. J. Sci. Comput. **37**, 367-382 (2008)
14. Gilboa, G, Osher, S: Nonlocal linear image regularization and supervised segmentation. Multiscale Model. Simul. **6**, 595-630 (2007)
15. Lou, Y, Zhang, X, Osher, S, Bertozzi, A: Image recovery via nonlocal operators. UCLA-CAM-Report 08-35 (2008)
16. Cofiman, RR, Donoho, DL: Translation-invariant de-noising. In: Antoniadis, A, Oppenheim, G (eds.) Wavelets and Statistics. Lecture Notes in Statistics, vol. 103. Springer, New York (1995)
17. Hale, E, Yin, W, Zhang, Y: A fixed-point continuation method for l_1-regularization with application to compressed sensing. CAAM-TR07-07 (2007)
18. Yin, W, Osher, S, Goldfarb, D, Darbon, J: Bregman iterative algorithms for l_1-minimization with applications to compressed sensing. SIAM J. Imaging Sci. **1**, 143-168 (2008)
19. Cai, J, Osher, S, Shen, Z: Linearized Bregman iterations for compressed sensing. Math. Comput. **78**(267), 1515-1536 (2009)
20. Cai, J, Osher, S, Shen, Z: Convergence of the linearized Bregman iteration for l_1-norm minimization. Math. Comput. **78**(268), 2127-2136 (2009)
21. Osher, S, Mao, Y, Dong, B, Yin, W: Fast linearized Bregman iteration for compressive sensing and sparse denoising. UCLA-CAM-Report 08-37 (2008)
22. Cai, J, Osher, S, Shen, Z: Linearized Bregman iterations for frame-based image deblurring. SIAM J. Imaging Sci. **2**(1), 226-252 (2009)
23. Goldstein, T, Osher, S: The split Bregman method for L_1-regularized problems. SIAM J. Imaging Sci. **2**(2), 323-343 (2009)
24. Cai, J, Osher, S, Shen, Z: Split Bregman method and frame based image restoration. Multiscale Model. Simul. **8**(2), 337-369 (2009)
25. Wu, C, Tai, X: Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models. SIAM J. Imaging Sci. **3**(3), 300-339 (2010)
26. Yang, Y, Möller, M, Osher, S: A dual split Bregman method for fast l_1 minimization. UCLA-CAM-Report 11-57 (2011)
27. Zhang, H, Cheng, L: A^- Linearized Bregman iteration algorithm. Math. Numer. Sin. **32**, 97-104 (2010) (in Chinese)
28. Candés, EJ, Wakin, MB, Boyd, SP: Enhancing sparsity by reweighted ℓ_1 minimization. J. Fourier Anal. Appl. **14**(5), 877-905 (2008)
29. Wang, G, Wei, Y, Qiao, S: Generalized Inverses: Theory and Computations. Science Press, Beijing (2004)
30. Wang, S, Yang, Z: Generalized Inverse Matrix and Its Applications. Beijing University of Technology Press, Beijing (1996)
31. Bregman, L: The relaxation method of finding the common points of convex sets and its application to the solution of problems in convex programming. USSR Comput. Math. Math. Phys. **7**(3), 200-217 (1967)
32. Donoho, D: De-noising by soft-thresholding. IEEE Trans. Inf. Theory **41**, 613-627 (1995)
33. Schlossmacher, EJ: An iterative technique for absolute deviations curve fitting. J. Am. Stat. Assoc. **68**, 857-859 (1973)
34. Zhao, YB, Li, D: Reweighted ℓ_1-minimization for sparse solutions to underdetermined linear systems. SIAM J. Optim. **22**(3), 1065-1088 (2012)

doi:10.1186/1029-242X-2014-238
Cite this article as: Qiao et al.: **A new reweighted l_1 minimization algorithm for image deblurring.** *Journal of Inequalities and Applications* 2014 **2014**:238.