Computer Vision and Image Understanding xxx (2011) xxx–xxx

Contents lists available at SciVerse ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Composite splitting algorithms for convex optimization

Junzhou Huang [a,*], Shaoting Zhang [b], Hongsheng Li [c], Dimitris Metaxas [b]

[a] Department of Computer Science and Engineering, University of Texas at Arlington, USA
[b] Department of Computer Science, Rutgers University, USA
[c] Department of Computer Science and Engineering, Lehigh University,USA

ARTICLE INFO

Article history:
Received 21 April 2010
Accepted 6 June 2011
Available online xxxx

Keywords:
Convex optimization
Composite splitting
Compressive sensing
MR image reconstruction
Low-rank tensor completion

ABSTRACT

We consider the minimization of a smooth convex function regularized by the composite prior models. This problem is generally difficult to solve even if each subproblem regularized by one prior model is convex and simple. In this paper, we present two algorithms to effectively solve it. First, the original problem is decomposed into multiple simpler subproblems. Then, these subproblems are efficiently solved by existing techniques in parallel. Finally, the result of the original problem is obtained by averaging solutions of subproblems in an iterative framework. The proposed composite splitting algorithms are applied to the compressed MR image reconstruction and low-rank tensor completion. Numerous experiments demonstrate the superior performance of the proposed algorithms in terms of both accuracy and computation complexity.

1. Introduction

The composite prior models have been used in many fields, including sparse learning, computer vision and compressive sensing. For example, in compressive sensing, the linear combination of the total-variation (TV) norm and L1 norm is known as a powerful regularizer for the compressive Magnetic Resonance (MR) imaging [1–3] and has been widely used in recovering MR images. In practice, the constraints of these models are generally formulated as energy terms (data fidelity terms), which can be represented as a smooth convex function $f(x)$. The composite regularization terms are often used to generate solutions with multiple properties according to the composite priors. They can be represented by the linear combination of multiple convex functions $\{g_i(x)\}_{i=1,\ldots,m}$, which are possibly non-smooth.

In this paper, we propose two composite splitting algorithms to solve this problem which is formulated as the following equation:

$$\min_{x\in\mathbf{R}^p} F(x) \equiv f(x) + \sum_{i=1}^{m} g_i(B_i x) \tag{1}$$

where f is the loss function and $\{g_i\}_{i=1,\ldots,m}$ are the prior models, both of which are convex functions; $\{B_i\}_{i=1,\ldots,m}$ are orthogonal matrices. If the function f and $\{g_i\}_{i=1,\ldots,m}$ are well-structured, there are two classes of splitting algorithms to solve it: operator splitting and variable splitting algorithms.

The operator-splitting algorithm searches for an x to make that the sum of the maximal-monotone operators equal to zero. Forward–Backward schemes are widely used in operator-splitting algorithms [4–6]. These algorithms have been applied in sparse learning [7] and compressive MR imaging [2]. The Iterative Shrinkage-Thresholding Algorithm (ISTA) and Fast ISTA (FISTA) [8] are two important Forward–Backward methods. They have been successfully used in signal processing [8,9], matrix completion [10] and multi-task learning [11]. To handle the case of $m > 1$, Spingarn [12] reduces the sum of multiple maximal monotone operators to the sum of two maximal monotone operators by defining new subspaces and operators, and then applies a Douglas-Rachford splitting algorithm to solve the new problem. The general projective splitting methods are used to search for a point in the extended solution set [13].

The variable splitting algorithm is another choice to solve (1) based on the combination of alternating direction methods (ADM) under an augmented Lagrangian framework: (1) splitting the variable x into $m + 1$ variables by introducing m new variables, where each new variable corresponds to one of $\{g_i\}_{i=1,\ldots,m}$; (2) applying the augmented Lagrangian method to this problem for each variable; (3) minimizing the decomposed augmented-Lagrangian function by using the ADMs to iteratively obtain the solutions. It was firstly used to solve the numerical PDE problem in Refs. [14,15]. Tseng [16] and He et al. [17]. extended it to solve variational inequality problems. There has been significant interest from the field of compressive sensing [18,19], where L1 regularization is a key problem and can be efficiently solved by this type of algorithms [20–22]. An algorithm based on variable splitting was proposed for compound regularization problems [23]. Variable

* Corresponding author.
E-mail address: jzhuang@uta.edu (J. Huang).

1077-3142/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2011.06.011

splitting techniques are also used for the decoupling of deblurring and denoising in image restoration [24]. Similar splitting techniques are used in Refs. [23,24,3] with subtle differences. Despite the small difference between the two splitting strategies, the latter enables fast solutions that take advantage of the fast Fourier transform (FFT) while the former two methods cannot [3]. Wang et al. [25] incorporate the equality constraints into the objective function and apply an ADM to the new penalized function for obtaining better image reconstruction, which shows that the ADMs are very efficient for solving TV regularization problems. They were recently proposed to solve semi-definite programming (SDP) problems and outperform previous interior-point methods on some structured SDP problems [26,27]. They were also applied to solve the sparse covariance selection problem [28] and matrix decomposition problem [29]. The Multiple Splitting Algorithm (MSA) and Fast MSA (FaMSA) have been recently proposed to efficiently solve (1), while $\{g_i\}_{i=1,\ldots,m}$ are assumed to be smooth convex functions [30].

However, none of these algorithms can efficiently solve (1) with provable convergence complexity. Moreover, none of them can provide the complexity bounds of iterations for their problems, except ISTA/FISTA in Ref. [8] and MSA/FaMSA in Ref. [30]. Both ISTA and MSA are first order methods. Their complexity bounds are $O(1/\epsilon)$ for ϵ-optimal solutions. Their fast versions, FISTA and FaMSA, have complexity bounds $O(1/\sqrt{\epsilon})$, which are inspired by the seminal results of Nesterov and are optimal according to the conclusions of Nesterov [31,32]. However, both ISTA and FISTA are designed for simpler regularization problems and cannot be applied efficiently to the composite regularization problem in our formulation. While the MSA/FaMSA in Ref. [30] are designed to handle the case of $m \geqslant 1$ in (1), they assume that all $\{g_i\}_{i=1,\ldots,m}$ are smooth convex functions. It makes them unable to directly solve (1) as we would have to smooth the non-smooth function $\{g_i\}_{i=1,\ldots,m}$ first before applying them. Since the smooth parameters are related to ϵ, the FaMSA with complexity bound $O(1/\sqrt{\epsilon})$ requires $O(1/\epsilon)$ iterations to compute an ϵ-optimal solution, which makes it not optimal for this problem.

In this paper, we propose two splitting algorithms based on the combination of variable and operator splitting techniques. We decompose the hard composite regularization problem (1) into m simpler regularization subproblems by: (1) splitting the function $f(x)$ into m functions $f_i(x)$ (e.g., $f_i(x) = f(x)/m$); (2) splitting variable x into m variables $\{x_i\}_{i=1,\ldots,m}$; (3) performing operator splitting to minimize $h_i(x_i) = f_i(x_i) + g_i(B_i x_i)$ over $\{x_i\}_{i=1,\ldots,m}$ independently and (4) obtaining the solution x by the linear combination of $\{x_i\}_{i=1,\ldots,m}$. The methods include function splitting, variable splitting and operator splitting. We call them Composite Splitting Algorithms (CSA) and fast CSA (FCSA). Compared to ISTA and MSA, CSA is more general as it can efficiently solve composite regularization problems with m ($m \geqslant 1$) non-smooth functions. More importantly, our algorithms can effectively decompose the original hard problem into multiple simpler subproblems and efficiently solve them in parallel. Thus, the required CPU time is not longer than the time required to solve the most difficult subproblem using current parallel-processor techniques. Finally, we successfully apply the proposed algorithms in two practical applications: the compressed MR image reconstruction and low-rank tensor completion.

The paper is organized as follows. Section 2 briefly reviews the related algorithms. In Section 3, the composite splitting algorithm and its accelerated version are proposed to solve (1) and then applied to the compressed MR image reconstruction and low rank tensor completion. Numerous experiments and discussions are presented in Section 4. Finally, we provide our conclusions in Section 5.

2. Algorithm review

2.1. Notations

We provide a brief summary of the notation used throughout this paper.

Matrix Norm and Trace:

1. Frobenius norm: $\|X\|_F = \sqrt{\sum_{i,j} X_{i,j}^2}$.
2. Nuclear norm: $\|X\|_F$ and $\|X\|_*$.
3. L_1 and Total Variation norm: $\|X\|_1$ and $\|X\|_{TV}$.
4. Matrix inner product: $\langle X, Y\rangle = trace(X^H Y)$.

Gradient: $\nabla f(x)$ denotes the gradient of the function f at point x.

The proximal map: given a continuous convex function $g(x)$ and any scalar $\rho > 0$, the proximal map associated to function g is defined as follows [9,8]:

$$prox_\rho(g)(x) := \arg\min_u \left\{ g(u) + \frac{1}{2\rho}\|u - x\|^2 \right\} \tag{2}$$

ϵ-optimal Solution: Suppose x^* is an optimal solution to (1). $x \in \mathbf{R}^p$ is called an ϵ-optimal solution to (1) if $F(x) - F(x^*) \leqslant \epsilon$ holds.

2.2. ISTA and FISTA

The ISTA and FISTA consider the following optimization problem [8]:

$$\min\{F(x) \equiv f(x) + g(x), \quad x \in \mathbf{R^p}\} \tag{3}$$

Here, they make the following assumptions:

1. g: $\mathbf{R^p} \to \mathbf{R}$ is a continuous convex function, which is possibly non-smooth;
2. f: $\mathbf{R^p} \to \mathbf{R}$ is a smooth convex function of type $C^{1,1}$ and the continuously differential function with Lipschitz constant L_f: $\|\nabla f(x_1) - \nabla f(x_2)\| \leqslant L_f\|x_1 - x_2\|$ for every $x_1, x_2 \in \mathbf{R^p}$;
3. Problem (3) is solvable.

Algorithm 1. ISTA

Input: $\rho = 1/L_f$, x_0
repeat
  for $k = 1$ **to** K **do**
    $x^k = prox_\rho(g)(x^{k-1} - \rho\nabla f(x^{k-1}))$
  end for
until Stop criterions

Algorithm 1 outlines the ISTA. Beck and Teboulle show that it terminates in $O(1/\epsilon)$ iterations with an ϵ-optimal solution in this case [9,8].

Theorem 2.1 (*Theorem 3.1 in Ref. [8]*). *Suppose $\{x_k\}$ is iteratively obtained by the algorithm of the ISTA, then, we have*

$$F(x^k) - F(x^*) \leqslant \frac{L_f\|x^0 - x^*\|^2}{2k}, \quad \forall x^* \in X_*$$

Algorithm 2 outlines the FISTA. Compared to ISTA, the increased computation costs come from the second step and third step in each iteration, which are almost negligible in large scale applications. Because of these advantages, the key idea of the FISTA is recently widely used in large scale applications, such as compressive sensing [8], image denoising and deblurring [9], matrix completion

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011

[10] and multi-task learning [11]. It has been proven that (Theorem 4.1 in Ref. [8]), with this acceleration scheme, the algorithm can terminate in $O(1/\sqrt{\epsilon})$ iterations with an ϵ-optimal solution instead of $O(1/\epsilon)$ for those of ISTA.

Theorem 2.2 (*Theorem 4.1 in Ref. [8]*). *Suppose* $\{x^k\}$ *and* $\{r^k\}$ *are iteratively obtained by the FISTA, then, we have*

$$F(x^k) - F(x^*) \leqslant \frac{2L_f\|x^0 - x^*\|^2}{(k+1)^2}, \quad \forall x^* \in X_*$$

Algorithm 2. FISTA

Input: $\rho = 1/L_f$, $r^1 = x^0$, $t^1 = 1$
repeat
  for $k = 1$ **to** K **do**
    $x^k = prox_\rho(g)(r^k - \rho\nabla f(r^k))$
    $t^{k+1} = \frac{1+\sqrt{1+4(t^k)^2}}{2}$
    $r^{k+1} = x^k + \frac{t^k-1}{t^{k+1}}(x^k - x^{k-1})$
  end for
until Stop criterions

The efficiency of the FISTA highly depends on the ability to quickly solve the first step $x^k = prox_\rho(g)(x_g)$, where $x_g = r^k - \rho\nabla f(r^k)$. For simpler regularization problems in some special cases, it is possible. For example, in Ref. [8], the FISTA can rapidly solve the $L1$ regularization problem with cost $\mathcal{O}(p\log(p))$ (where p is the dimension of x), since the second step $x^k = prox_\rho(\beta\|\Phi x\|_1)(x_g)$ has a close form solution while the wavelet transform Φ and its inverse can be computed with cost $\mathcal{O}(p\log(p))$. In Ref. [9], the FISTA is also used to quickly solve the TV regularization problem, since the step $x^k = prox_\rho(\alpha\|x\|_{TV})(x_g)$ can be computed with cost $\mathcal{O}(p)$ in limited iterations. However, the FISTA cannot efficiently solve the composite regularization problem (1), since no algorithm is available to efficiently solve the step:

$$\min_x F(x) \equiv \frac{1}{2}\|x - x_g\|^2 + \sum_{i=1}^m g_i(B_i x) \tag{4}$$

To solve (1), the key problem is thus to develop an efficient algorithm to solve (4). In the following section, we will show that a scheme based on composite splitting techniques can achieve that.

3. Composite splitting algorithms

In this section, we first propose the CSA and its accelerated version FCSA. They are then applied to the compressed MR image reconstruction and low-rank tensor completion, respectively.

3.1. Problem definition

Let us consider the following minimization problem:

$$\min_{x\in\mathbf{R}^p} F(x) \equiv f(x) + \sum_{i=1}^m g_i(B_i x) \tag{5}$$

with the following assumptions:

1. g_i: $\mathbf{R}^p \to \mathbf{R}$ is a continuous convex function for each $i \in \{1, \ldots, m\}$, which is possibly non-smooth;
2. f: $\mathbf{R}^p \to \mathbf{R}$ is a smooth convex function of type $C^{1,1}$ and the continuously differential function with Lipschitz constant L_f: $\|\nabla f(x_1) - \nabla f(x_2)\| \leqslant L_f\|x_1 - x_2\|$ for every $x_1, x_2 \in \mathbf{R}^p$;
3. $\{B_i \in \mathbf{R}^{p\times p}\}_{i=1,\ldots,m}$ are orthogonal matrices;
4. Problem (5) is solvable.

If $m = 1$, this problem degenerates to (3), which can be efficiently solved by FISTA. However, it may be challenging to solve (5) by ISTA/FISTA if $m > 1$. For example, let us assume $m = 2$, $g_1(x) = \|x\|_1$ and $g_2(x) = \|x\|_{TV}$. When $g(x) = g_1(x)$ in (3), the first step in Algorithm 2 has a closed form solution; When $g(x) = g_2(x)$, the first step in Algorithm 2 can also be solved iteratively in a few iterations [9]. However, if $g(x) = g_1(x) + g_2(x)$, the first step in Algorithm 2 is not easy to solve, which makes the computational complexity of each iteration too high to solve using FISTA in practice.

When $\{g_i\}_{i=1,\ldots,m}$ are smooth convex functions, this problem can be efficiently solved by the MSA/FaMSA. However, in our case, $\{g_i\}_{i=1,\ldots,m}$ may be non-smooth. Therefore, the MSA/FaMSA cannot be directly applied to solve this problem. It may be possible to smooth these non-smooth function first and then apply the FaMSA to solve it. However, in this case, the FaMSA with complexity bound $O(1/\sqrt{\epsilon})$ requires $O(1/\epsilon)$ iterations to compute an ϵ-optimal solution, which makes it not optimal for the first order methods [32].

In the following, we propose our algorithms that overcomes these difficulties. Our algorithms decompose the original problem (5) into m simpler regularization subproblems, where each of them is easier to solve using the FISTA.

3.2. Building blocks

The above discussion shows that, if a fast algorithm is available to solve (4), then the original composite regularization is efficiently solved by the FISTA, which obtains an ϵ-optimal solution in $\mathcal{O}(1/\sqrt{\epsilon})$ iterations. Actually, (4) can be considered as a denoising problem. We use composite splitting techniques to solve it: (1) splitting variable x into multiple variables $\{x_i\}_{i=1,\ldots,m}$; (2) performing operator splitting over each of $\{x_i\}_{i=1,\ldots,m}$ independently and (3) obtaining the solution x by a linear combination of $\{x_i\}_{i=1,\ldots,m}$. We call it Composite Splitting Denoising (CSD) method, which is outlined in Algorithm 3. Its validity is guaranteed by the following theorem:

Theorem 3.1. *Let* $\{x^j\}$ *be the sequence generated by the CSD. If* x^* *is the true solution of problem (4),* x^j *will strongly converge to* x^*.

Since the CSD algorithm is a special case of Algorithm 3.1 in Ref. [33], the reader can refer to Theorem 3.4 in Ref. [33] for the proof of this theorem. Similar proofs can also be found in Ref. [34].

Algorithm 3. CSD

Input: $\rho = 1/L$, α, β, $\{z_i^0\}_{i=1,\ldots,m} = x_g$
for $j = 1$ **to** J **do**
  for $i = 1$ **to** m **do**
    $x_i = \arg\min_x \frac{1}{2m\rho}\|x - z_i^{j-1}\|^2 + g_i(B_i x)$
  end for
  $x^j = \frac{1}{m}\sum_{i=1}^m x_i$
  for $i = 1$ **to** m **do**
    $z_i^j = z_i^{j-1} + x^j - x_i$
  end for
end for

It is worth mentioning that the decomposition techniques in the CSD are similar to those in the dual decomposition methods [35]. They are however still different. The CSD algorithm is a proximal decomposition algorithm for solving a denoising problem with several non-smooth regularization functions. It fully

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011

decomposes the original problem into several easier subproblems. Then, it makes use of each function individually via its own proximity operator and obtains the results of the original problem from the average of solutions of subproblems in an iterative framework.

It is also worth mentioning that the presence of the orthogonal matrices B_i does not make the problem harder. It still has a closed form solution for its proximity operator. Thus, each x_i can be computed from the proximity operator in a straightforward way easily. For example,

$$x_0 = \arg\min_x \left\{ \frac{1}{2\rho}\|x - z\|^2 + g(Bx) \right\}$$

where B is an orthogonal matrix. Suppose $y = Bx$, then, we know that

$$y_0 = \arg\min_y \left\{ \frac{1}{2\rho}\|y - Bz\|^2 + g(y) \right\}$$

This is a standard equation for the proximal operator $y_0 = prox_\rho(g)(Bz)$. Then $x_0 = B^T y_0 = B^T prox_\rho(g)(Bz)$. Thus, even with the orthogonal matrix B, we are still able to compute each x with a closed form solution.

3.3. Composite Splitting Algorithm (CSA)

Combining the CSD with ISTA, a new algorithm, CSA, is proposed to solve the composite regularization problem (5). In practice, a small iteration number J in the CSD is enough for the CSA to obtain good reconstruction results. Specifically, it is set as 1 in our algorithm. Numerous experimental results in the next section show that it is sufficient for real composite regularization problem.

Algorithm 4 outlines the proposed CSA. In each iteration, Algorithm 4 decomposes the original problem into m subproblems and solve them independently. For many problems in practice, these m subproblems are expected to be significantly easier to solve than the original joint problem. Another advantage of this algorithm is that the decomposed subproblems can be solved in parallel. Given x^{k-1}, the m subproblems to compute $\{y_i^k\}_{i=1,\ldots,m}$ are solved simultaneously in Algorithm 4. Although it assumes the Lipschitz constant L_f is known, this can be relaxed by using the backtracking technique in Ref. [8] to estimate L_f at each iteration.

3.4. Fast composite splitting algorithms

A fast version of CSA named FCSA is also proposed to solve (5), which is outlined in Algorithm 5. FCSA decomposes the difficult composite regularization problem into multiple simpler subproblems and solve them in parallel. Each subproblems can be solved by the FISTA, which requires only $O(1/\sqrt{\epsilon})$ iterations to obtain an ϵ-optimal solution.

Algorithm 4. CSA

Input: $\rho = 1/L$, x^0
repeat
  for $k = 1$ **to** K **do**
    for $i = 1$ **to** m **do**
      $y_i^k = prox_\rho(g_i)(B_i(x^{k-1} - \frac{1}{L}\nabla f_i(x^{k-1})))$
    end for
    $x^k = \frac{1}{m}\sum_{i=1}^m B_i^{-1} y_i^k$
  end for
until Stop criterions

Algorithm 5. FCSA

Input: $\rho = 1/L$, $t^1 = 1$ $r^1 = x^0$
repeat
  for $k = 1$ **to** K **do**
    for $i = 1$ **to** m **do**
      $y_i^k = prox_\rho(g_i)(B_i(r^k - \frac{1}{L}\nabla f_i(r^k)))$
    end for
    $x^k = \frac{1}{m}\sum_{i=1}^m B_i^{-1} y_i^k$
    $t^{k+1} = \frac{1+\sqrt{1+4(t^k)^2}}{2}$
    $r^{k+1} = x^k + \frac{t^k - 1}{t^{k+1}}(x^k - x^{k-1})$
  end for
until Stop criterions

In this algorithm, if we remove the acceleration step by setting $t^{k+1} \equiv 1$ in each iteration, we obtain the CSA. A key feature of the FCSA is its fast convergence performance borrowed from the FISTA. From Theorem 2.2, we know that the FISTA can obtain an ϵ-optimal solution in $\mathcal{O}(1/\sqrt{\epsilon})$ iterations.

Another key feature of the FCSA is that the cost of each iteration may be $\mathcal{O}(mp\log(p))$ under the following conditions: (1) the step $y_i^k = prox_\rho(g_i)(B_i(r^k - \frac{1}{L}\nabla f_i(r^k)))$ can be computed with the cost $\mathcal{O}(p\log(p))$ for some prior models g_i if B_i can be computed with $\mathcal{O}(p\log(p))$; (2) the step $x^k = \frac{1}{m}\sum_{i=1}^m B_i^{-1} y_i^k$ can also be computed with the cost of $\mathcal{O}(p\log(p))$ in these cases; and (3) other steps only involve adding vectors or scalars, thus cost only $\mathcal{O}(p)$ or $\mathcal{O}(1)$. Therefore, the total cost of each iteration in the FCSA can be $\mathcal{O}(mp\log(p))$ in many practical cases.

With these two key features, the FCSA efficiently solves the composite regularization problem (5) and obtains better results in terms of both accuracy and computational complexity. The experimental results in the next section demonstrate its superior performance on compressed MR image reconstruction and low-rank tensor completion.

3.5. Compressed MR image reconstruction

Specifically, we apply the CSA and FCSA to solve the compressed MR image reconstruction problem in compressive sensing [1]:

$$\min_x F(x) \equiv \frac{1}{2}\|Rx - b\|^2 + \alpha\|x\|_{TV} + \beta\|\Phi x\|_1 \tag{6}$$

where R is a partial Fourier transform, Φ is the wavelet transform, b is the under-sampled Fourier measurements, α and β are two positive parameters.

This model has been shown to be one of the most powerful models for the compressed MR image recovery [1]. However, since $\|x\|_{TV}$ and $\|\Phi x\|_1$ are both non-smooth in x, this problem is significantly more difficult to solve than any of those with a single non-smooth term such as the L1 regularization problem or a total variation regularization problem. In this case, the FISTA can efficiently solve the L1 regularization problem [8], since the first step $x^k = prox_\rho(\|\Phi x\|_1)(r^k - \rho\nabla f(r^k))$ has a closed form solution in Algorithm 2. The FISTA can also efficiently solve the total variation regularization problem [9], since the first step $x^k = prox_\rho(\|x\|_{TV})(r^k - \rho\nabla f(r^k))$ can be computed quickly with limited iterations in Algorithm 2. However, the FISTA cannot efficiently solve the joint L1 and TV regularization problem (6), since $x^k = prox_\rho(\alpha\|x\|_{TV} + \beta\|\Phi x\|_1)(r^k - \rho\nabla f(r^k))$ cannot be computed in a short time.

The Conjugate Gradient (CG) [1] has been applied to problem (6) and it converges very slowly. The computational complexity

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011

has been the bottleneck that made (6) impractical in the past [1]. To use this model for practical MR image reconstruction, Ma et al. propose a fast algorithm based on the operator splitting technique [2], which is called TVCMRI. In [3], a variable splitting method (RecPF) is proposed to solve the MR image reconstruction problem. Both of them can replace iterative linear solvers with Fourier domain computations, which provide substantial improvement in terms of time complexity. To the best of our knowledge, they are two of the fastest algorithms to solve problem (6) so far. By contrast, the CSA and FCSA directly attack the joint L1 and total variation norm regularization problem by transforming it to the L1 regularization and TV norm regularization subproblems, which can be efficiently solved. Algorithm 6 outlines the proposed FCSA for the compressive MR image reconstruction.

It is clear that the FCSA-MRI has the cost $\mathcal{O}(n\log(n))$ for each iteration, as confirmed by the following observations. The steps 4–6 only involve adding vectors or scalars, thus cost only $\mathcal{O}(p)$ or $\mathcal{O}(1)$. In step 1, $\nabla f(r^k) = R^T(R\,r^k - b)$ since $f(r^k) = \frac{1}{2}\|Rr^k - b\|^2$ in this case. Thus, this step only costs $\mathcal{O}(p\log(p))$. As introduced above, the step $x_1 = prox_\rho(2\alpha\|x\|_{TV})(x_g)$ can be computed quickly in limited iterations with cost $\mathcal{O}(p)$ [9]. The step $x_2 = prox_\rho(2\beta\|\Phi x\|_1)(x_g)$ has a closed form solution and can be computed with cost $\mathcal{O}(p\log(p))$. Thus, the total cost of each iteration in the FCSA is $\mathcal{O}(p\log(p))$.

In the next section, we compare our CSA and FCSA with other reconstruction algorithms, such as the TVCMRI and RecPF [2,3]. The results show that the FCSA is significantly more efficient than the TVCMRI and RecPF.

Algorithm 6. FCSA-MRI

Input: $\rho = 1/L$, α, β, $t^1 = 1$, $r^1 = x^0$
for $k = 1$ **to** K **do**
  $x_g = r^k - \rho\nabla f(r^k)$
  $x_1 = prox_\rho(2\alpha\|x\|_{TV})(x_g)$
  $x_2 = prox_\rho(2\beta\|\Phi x\|_1)(x_g)$
  $x^k = (x_1 + x_2)/2$
  $t^{k+1} = \left(1 + \sqrt{1 + 4(t^k)^2}\right)\Big/2$
  $r^{k+1} = x^k + \frac{t^k - 1}{t^{k+1}}(x^k - x^{k-1})$
end for

3.6. Low-rank tensor completion

In this section, we apply the proposed FCSA to the low rank tensor completion problem. This problem has attracted attention recently [36,37,10,38]. It is formulated as follows:

$$\min_X F(X) \equiv \frac{1}{2}\|\mathcal{A}(X) - b\|^2 + \alpha\|X\|_* \tag{7}$$

where $X \in \mathbb{R}^{p\times q}$ is a unknown matrix, $\mathcal{A} : \mathbb{R}^{p\times q} \to \mathbb{R}^n$ is the linear map, and $b \in \mathbb{R}^n$ is the observation. The nuclear norm is defined as $\|X\|_* = \sum_i \sigma_i(X)$, where $\sigma_i(X)$ is the singular value of the matrix X. The accelerated proximal gradient (APG) scheme in the FISTA has been used to solve (7) in Ref. [10]. In most cases, the APG gains the best performance compared with other methods, since it can obtain an ϵ-optimal solution in $O(1/\sqrt{\epsilon})$ iterations.

The tensor completion problem can be defined similarly. We use the 3-mode tensor as an example for the low-rank tensor completion. It is straightforward to extend to the n-mode tensor completion. The 3-mode tensor completion can be formulated as follows [39]:

$$\min_{\mathcal{X}} F(\mathcal{X}) \equiv \frac{1}{2}\|\mathcal{A}(\mathcal{X}) - b\|^2 + \sum_{i=1}^{m}\alpha_i\|B_i\mathcal{X}\|_* \tag{8}$$

where $\mathcal{X} \in \mathbb{R}^{p\times q\times m}$ is the unknown 3-mode tensor, $\mathcal{A} : \mathbb{R}^{p\times q\times m} \to \mathbb{R}^n$ is the linear map, and $b \in \mathbb{R}^n$ is the observation. B_1 is the "unfold" operation along the 1-mode on a tensor $\mathcal{X}$, which is defined as $B_1\mathcal{X} := X_{(1)} \in \mathbb{R}^{p\times qm}$; B_2 is the "unfold" operation along the 2-mode on a tensor $\mathcal{X}$, which is defined as $B_2\mathcal{X} := X_{(2)} \in \mathbb{R}^{q\times pm}$; B_3 is the "unfold" operation along the 3-mode on a tensor $\mathcal{X}$, which is defined as $B_3\mathcal{X} := X_{(3)} \in \mathbb{R}^{m\times pq}$. The opposite operation "fold" is defined as $B_i^T X_i = \mathcal{X}$ where $i = 1, 2, 3$.

Generally, it is much harder to solve the tensor completion problem than the matrix completion because of the composite regularization. The solvers in Ref. [10] cannot be used to efficiently solve (8). In Ref. [39], a relaxation technique is used to separate the dependant relationships and the block coordinate descent (BCD) method is used to solve the low-rank tensor completion problem. As far as we know, it is the best method for the low-rank tensor completion so far. However, it converges very slowly due to the convergence properties of the BCD.

The proposed FCSA can be directly used to efficiently solve (8). Algorithm 7 outlines the proposed algorithm for the low-rank tensor completion (LRTC). As opposed to the BCD method for LRTC using relaxation techniques [39], the FCSA directly attacks the composite matrix nuclear norm regularization problem by transforming it to multiple matrix nuclear norm regularization subproblems, which can be efficiently solved in parallel. In the following, we compare the proposed FCSA and BCD for the low rank tensor completion. They are named FCSA-LRTC and CBD-LRTC respectively. The results show that the FCSA is more efficient than the BCD for the LRTC problem.

Algorithm 7. FCSA-LRTC

Input: $\rho = 1/L_f$, $\mathcal{R}^1 = \mathcal{X}^0$, $t^1 = 1$
repeat
  for $k = 1$ **to** K **do**
    for $i = 1$ **to** m **do**
      $X_{(i)}^k = prox_\rho(\alpha_i\|X_{(i)}\|_*)(R_{(i)}^k - \rho\mathcal{A}^*(\mathcal{A}(R_{(i)}^k - b)))$
    end for
    $\mathcal{X}^k = \frac{1}{m}\sum_{i=1}^m B_i^T X_{(i)}^k$
    $t^{k+1} = \frac{1+\sqrt{1+4(t^k)^2}}{2}$
    $\mathcal{R}^{k+1} = \mathcal{X}^k + \frac{t^k - 1}{t^{k+1}}(\mathcal{X}^k - \mathcal{X}^{k-1})$
  end for
until Stop criterions

4. Experiments

The reproducible code for the experiments in this paper can be downloaded from http://paul.rutgers.edu/jzhuang/Submission/CVIUCode.rar.

4.1. Compressed MR image reconstruction

4.1.1. Experimental setup

Suppose a MR image x has n pixels, the partial Fourier transform R in problem (6) consists of m rows of a $n \times n$ matrix corresponding to the full 2D discrete Fourier transform. The m selected rows correspond to the acquired b. The sampling ratio is defined as m/n. The scanning duration is shorter if the sampling ratio is smaller. In MR imaging, we have certain freedom to select rows, which correspond to certain frequencies. In the following experiments, we select the frequencies in the following manner. In the k-space, we randomly obtain more samples in lower frequencies and less samples in higher frequencies. This sample scheme has been widely

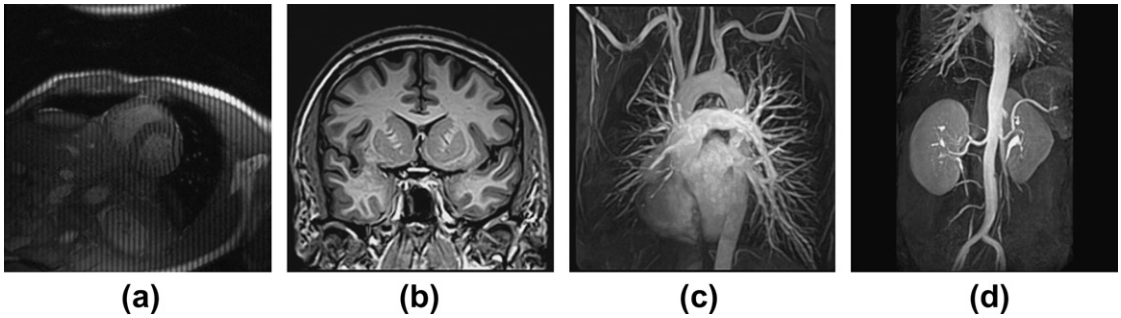

Fig. 1. MR images: (a) Cardiac. (b) Brain. (C) Chest. (d) Artery.

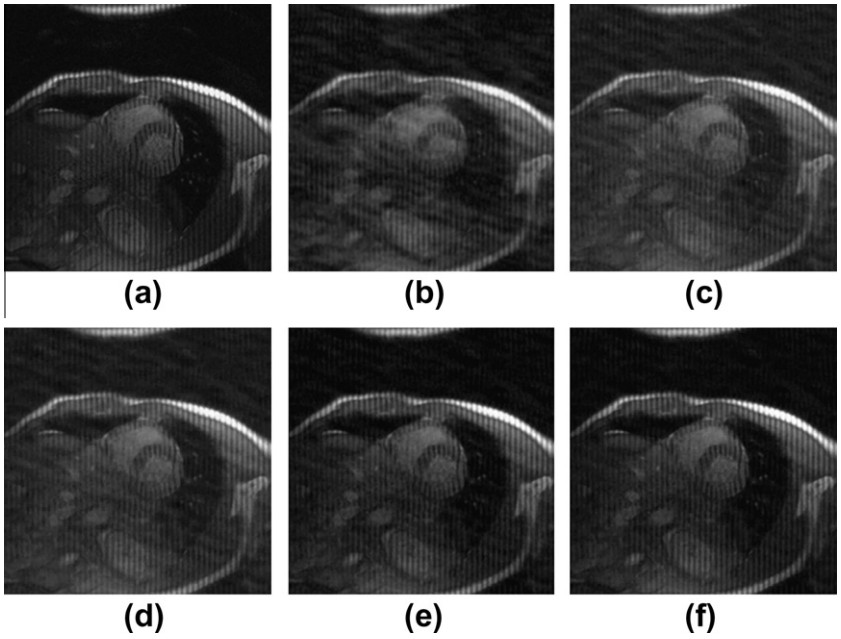

Fig. 2. Cardiac MR image reconstruction from 20% sampling (a) original image; (b–f) are the reconstructed images by the CG [1], TVCMRI [2], RecPF [3], CSA and FCSA. Their SNR are 9.86, 14.43, 15.20, 16.46 and 17.57 (db). Their CPU time are 2.87, 3.14, 3.07, 2.22 and 2.29 (s).

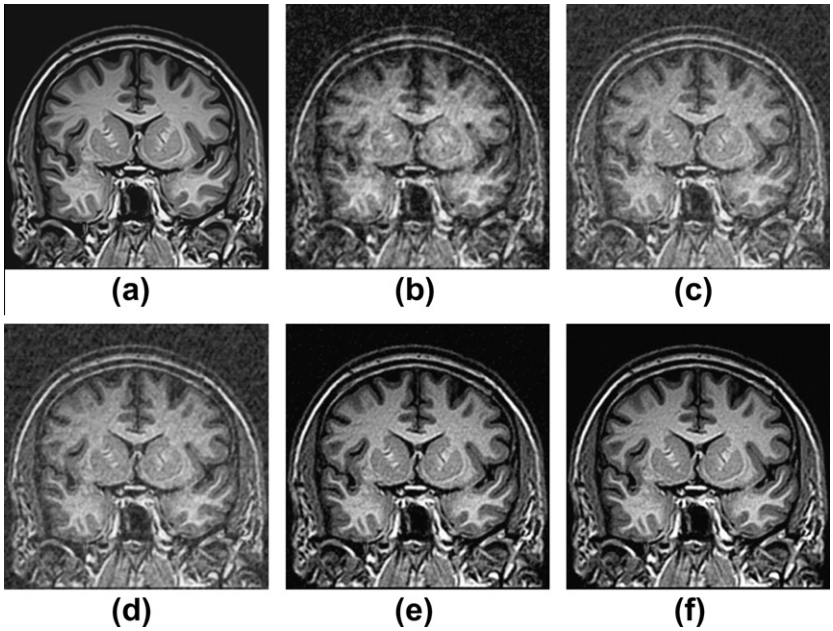

Fig. 3. Brain MR image reconstruction from 20% sampling (a) original image; (b–f) are the reconstructed images by the CG [1], TVCMRI [2], RecPF [3], CSA and FCSA. Their SNR are 8.71, 12.12, 12.40, 18.68 and 20.35 (db). Their CPU time are 2.75, 3.03, 3.00, 2.22 and 2.20 (s).

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011

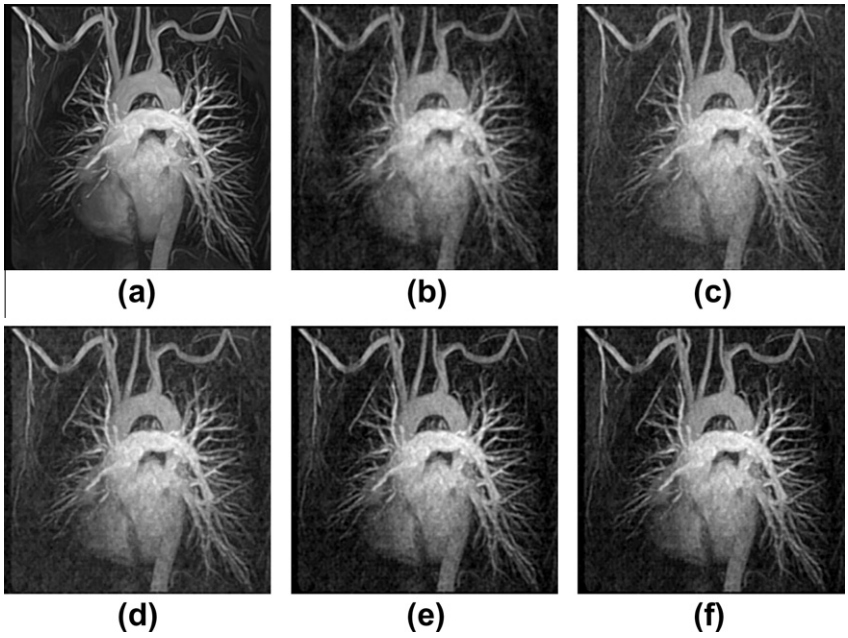

Fig. 4. Chest MR image reconstruction from 20% sampling (a) original image; (b–f) are the reconstructed images by the CG [1], TVCMRI [2], RecPF [3], CSA and FCSA. Their SNR are 11.80, 15.06, 15.37, 16.53 and 16.07 (db). Their CPU time are 2.95, 3.03, 3.00, 2.29 and 2.234 (s).

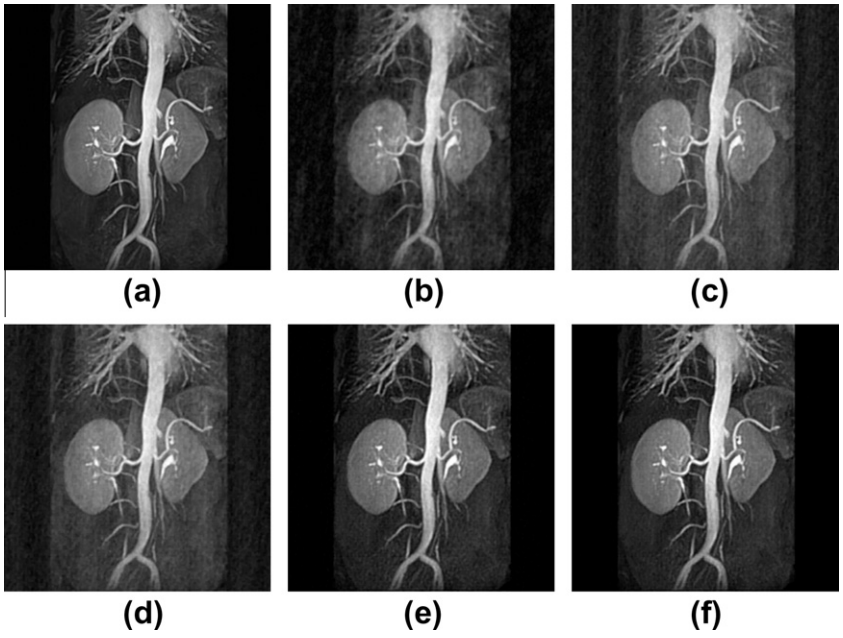

Fig. 5. Artery MR image reconstruction from 20% sampling (a) original image; (b–f) are the reconstructed images by the CG [1], TVCMRI [2], RecPF [3], CSA and FCSA. Their SNR are 11.73, 15.49, 16.05, 22.27 and 23.70 (db). Their CPU time are 2.78, 3.06, 3.20, 2.22 and 2.20 (s).

used for compressed MR image reconstruction [1–3]. Practically, the sampling scheme and speed in MR imaging also depend on physical and physiological limitations [1].

All experiments are conducted on a 2.4 GHz PC in Matlab environment. We compare the CSA and FCSA with the classic MR image reconstruction method based on the CG [1]. We also compare them with two of the fastest MR image reconstruction methods, TVCMRI[1] [2] and RecPF[2] [3]. For fair comparisons, we download the codes from their websites and carefully follow their experiment setup. For example, the observation measurement b is synthesized as $b = Rx + \mathbf{n}$, where $\mathbf{n}$ is the Gaussian white noise with standard deviation $\sigma = 0.01$. The regularization parameter α and β are set as 0.001 and 0.035. R and b are given as inputs, and x is the unknown target. For quantitative evaluation, we compute the Signal-to-Noise Ratio (SNR) for each reconstruction result.

4.1.2. Visual comparisons

We apply all methods on four 2D MR images: cardiac, brain, chest and artery respectively. Fig. 1 shows these images. For

[1] http://www.columbia.edu/sm2756/TVCMRI.htm.
[2] http://www.caam.rice.edu/optimization/L1/RecPF/.

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011

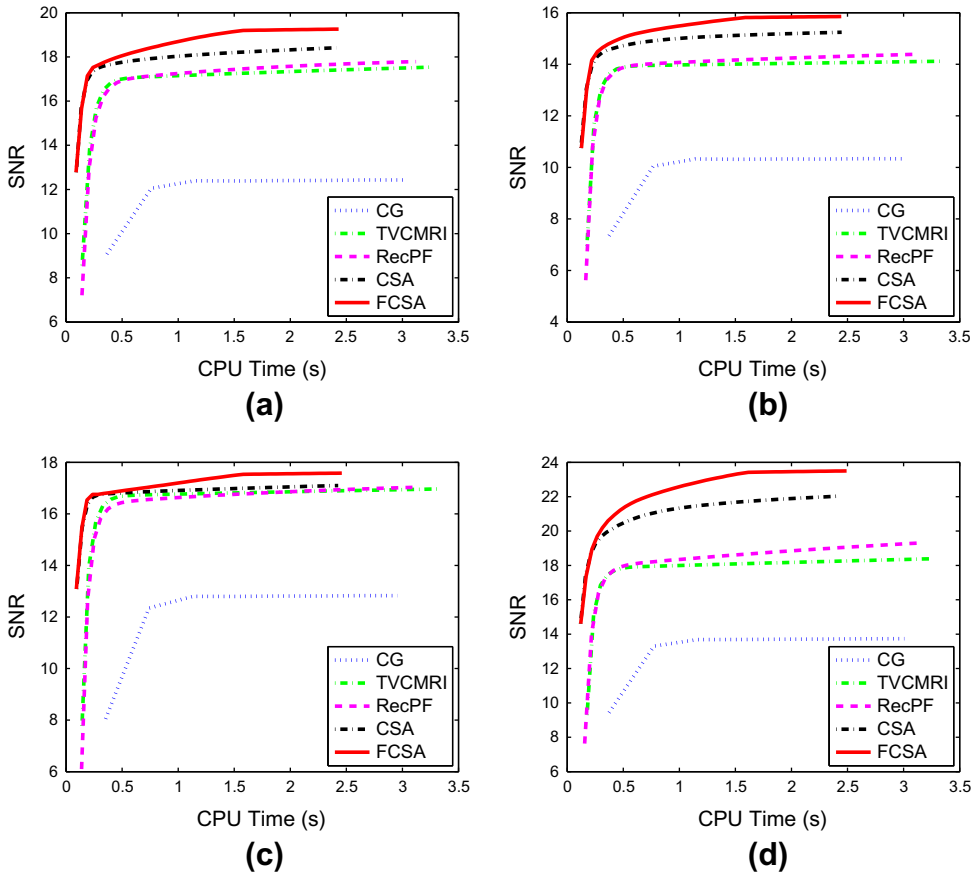

Fig. 6. Performance comparisons (CPU time vs. SNR) on different MR images: (a) Cardiac image. (b) Brain image. (C) Chest image. (d) Artery image.

Table 1
Comparisons of the SNR (db) over 100 runs.

	CG [1]	TVCMRI [2]	RecPF [3]	CSA	FCSA
Cardiac	12.43 ± 1.53	17.54 ± 0.94	17.79 ± 2.33	18.41 ± 0.73	19.26 ± 0.78
Brain	10.33 ± 1.63	14.11 ± 0.34	14.39 ± 2.17	15.25 ± 0.23	15.86 ± 0.22
Chest	12.83 ± 2.05	16.97 ± 0.32	17.03 ± 2.36	17.10 ± 0.31	17.58 ± 0.32
Artery	13.74 ± 2.28	18.39 ± 0.47	19.30 ± 2.55	22.03 ± 0.18	23.50 ± 0.20

Table 2
Comparisons of the CPU time (s) over 100 runs.

	CG [1]	TVCMRI [2]	RecPF [3]	CSA	FCSA
Cardiac	2.82 ± 0.16	3.16 ± 0.10	2.97 ± 0.12	2.27 ± 0.08	2.30 ± 0.08
Brain	2.81 ± 0.15	3.12 ± 0.15	2.95 ± 0.10	2.27 ± 0.12	2.31 ± 0.13
Chest	2.79 ± 0.16	3.00 ± 0.11	2.89 ± 0.07	2.21 ± 0.06	2.26 ± 0.07
Artery	2.81 ± 0.17	3.04 ± 0.13	2.94 ± 0.09	2.22 ± 0.07	2.27 ± 0.13

comparison purposes, they have the same size of 256×256. The sample ratio is set to be approximately 20%. To perform fair comparisons, all methods run 50 iterations, except that the CG runs only 8 iterations due to its higher computational complexity.

Figs. 2–5 show the visual comparisons of the reconstructed 2results by different methods. The FCSA always obtains the best visual effects on all MR images in less CPU time. The CSA is always

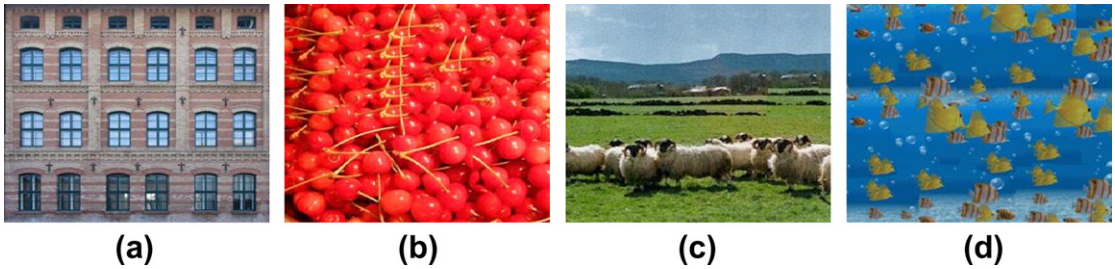

Fig. 7. MR images: (a) Window. (b) Cherry. (C) Sheep. (d) Fish.

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011

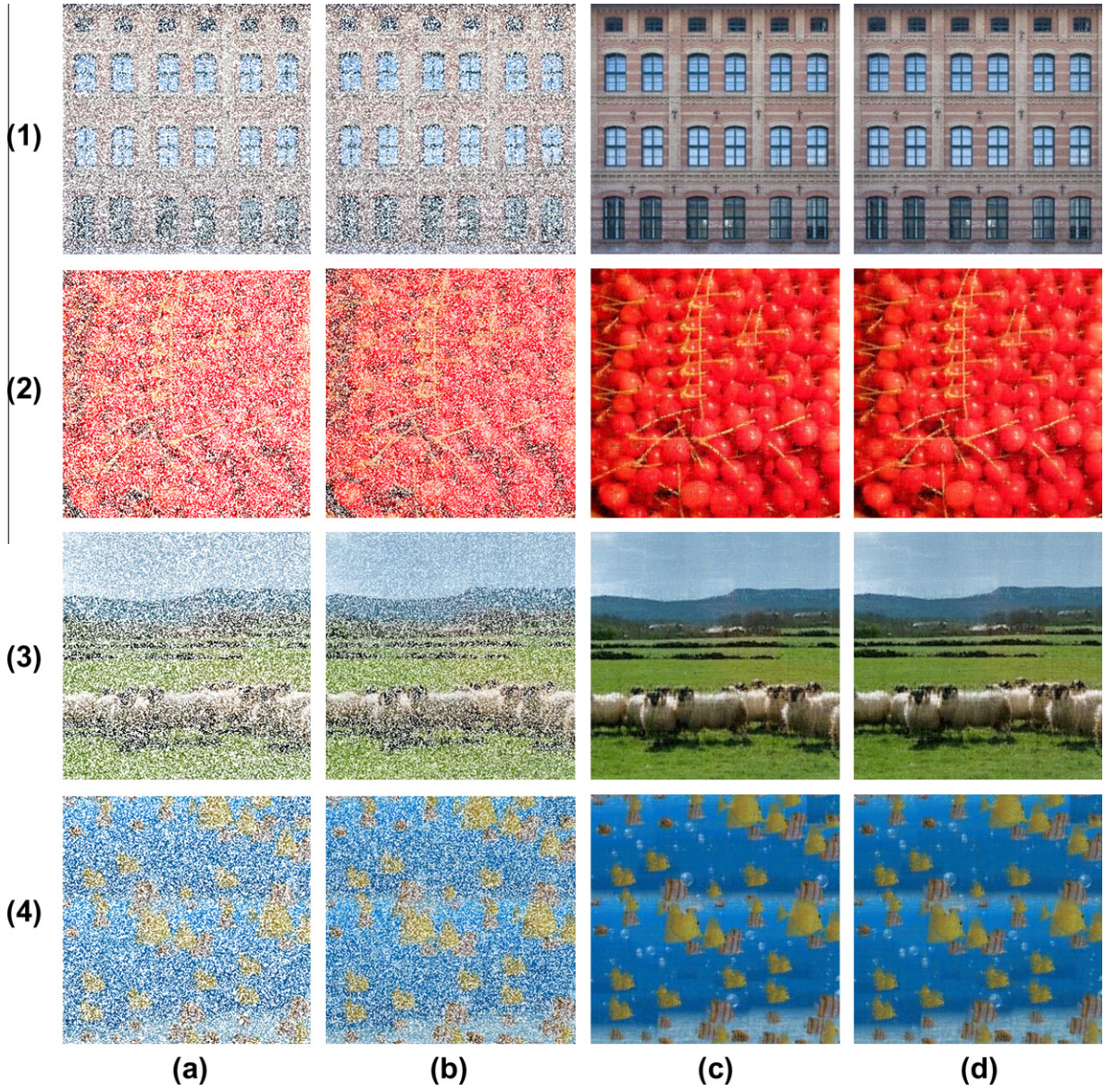

Fig. 8. Comparisons in terms of visual effects. Color images are: (1) Window. (2) Cherry. (3) Sheep. (4) Fish. Columns (a–d) correspond to the images before completion, the obtained results by the CGD-LRTC [39], APG-LRMC [10] and FCSA-LRTC, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

inferior to the FCSA, which shows the effectiveness of acceleration steps in the FCSA for the MR image reconstruction. The classical CG [1] is worse than the others because of its higher cost in each iteration, the RecPF is slightly better than the TVCMRI, which is consistent with observations in Refs. [2,3].

4.1.3. CPU time and SNRs

Fig. 6 gives the performance comparisons between different methods in terms of the CPU time over the SNR. Tables 1 and 2 tabulate the SNR and CPU Time by different methods, averaged over 100 runs for each experiment, respectively. The FCSA always obtains the best reconstruction results on all MR images by achieving the highest SNR in less CPU time. The CSA is always inferior to the FCSA, which shows the effectiveness of the acceleration steps in the FCSA for the MR image reconstruction. While the classical CG [1] is worse than others because of its higher cost in each iteration, the RecPF is slightly better than the TVCMRI, which is consistent with observations in Refs. [2,3].

Table 3
Comparisons of the CPU time and RSE for random missing.

	CGD-LRTC [39]		APG-LRMC [10]		FCSA-LRTC	
	Time (s)	RSE	Time (s)	RSE	Time (s)	RSE
Window	129.79	0.4917	105.08	0.1031	132.82	0.0641
Cherry	135.23	0.6154	102.81	0.3613	133.39	0.1014
Sheep	130.28	0.6137	101.37	0.1534	128.03	0.0908
Fish	139.11	0.7009	101.47	0.1566	129.21	0.0883

4.2. Low-rank tensor completion

4.2.1. Experiment setup

Suppose a color image $\mathcal{X}$ with low rank has the size of $h \times w \times d$, where h, w, d denote its height, width and color channel respectively. When the color values of some pixels are missing in the image, the tensor completion is conducted to recover the missed values. Suppose the color values of q pixels are missing in the image, the sampling ratio is defined as $(h \times w \times d - q \times d)/$

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011

($h \times w \times d$). The known color values in the image are called the samples for tensor completion. We randomly obtain these samples or designate the samples before the tensor completion [39].

All experiments are conducted on a 2.4 GHz PC in Matlab environment. We compare the proposed FCSA-LRTC with the CGD-LRTC [39] for the tensor completion problem. To show the advantage of

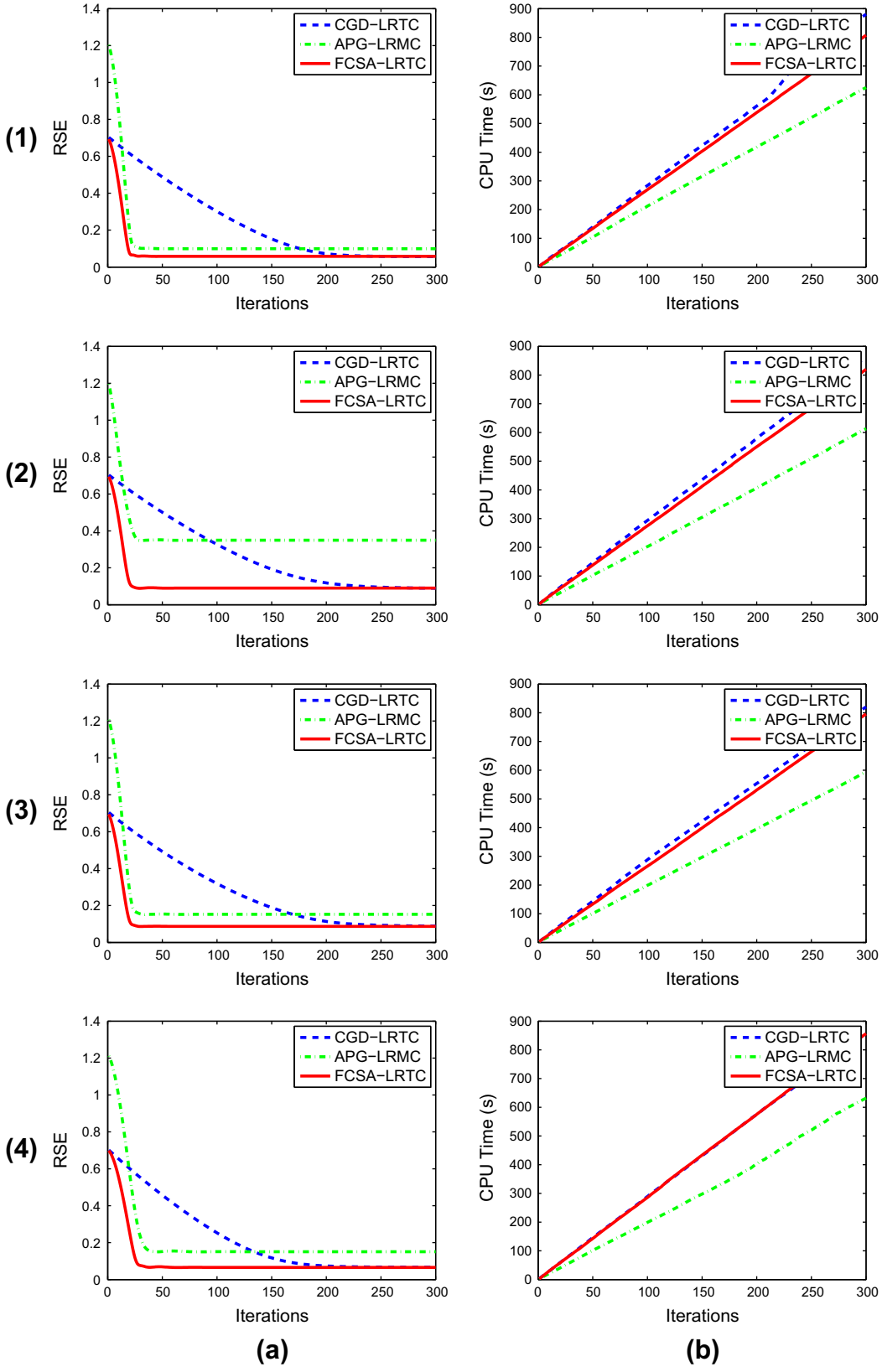

Fig. 9. comparisons in terms of RSE and CPU time. Color images are: (1) Window. (2) Cherry. (3) Sheep. (4) Fish. Column (a and b) correspond to the comparisons in terms of iterations vs. RSE and iterations vs. CPU time (s), respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011

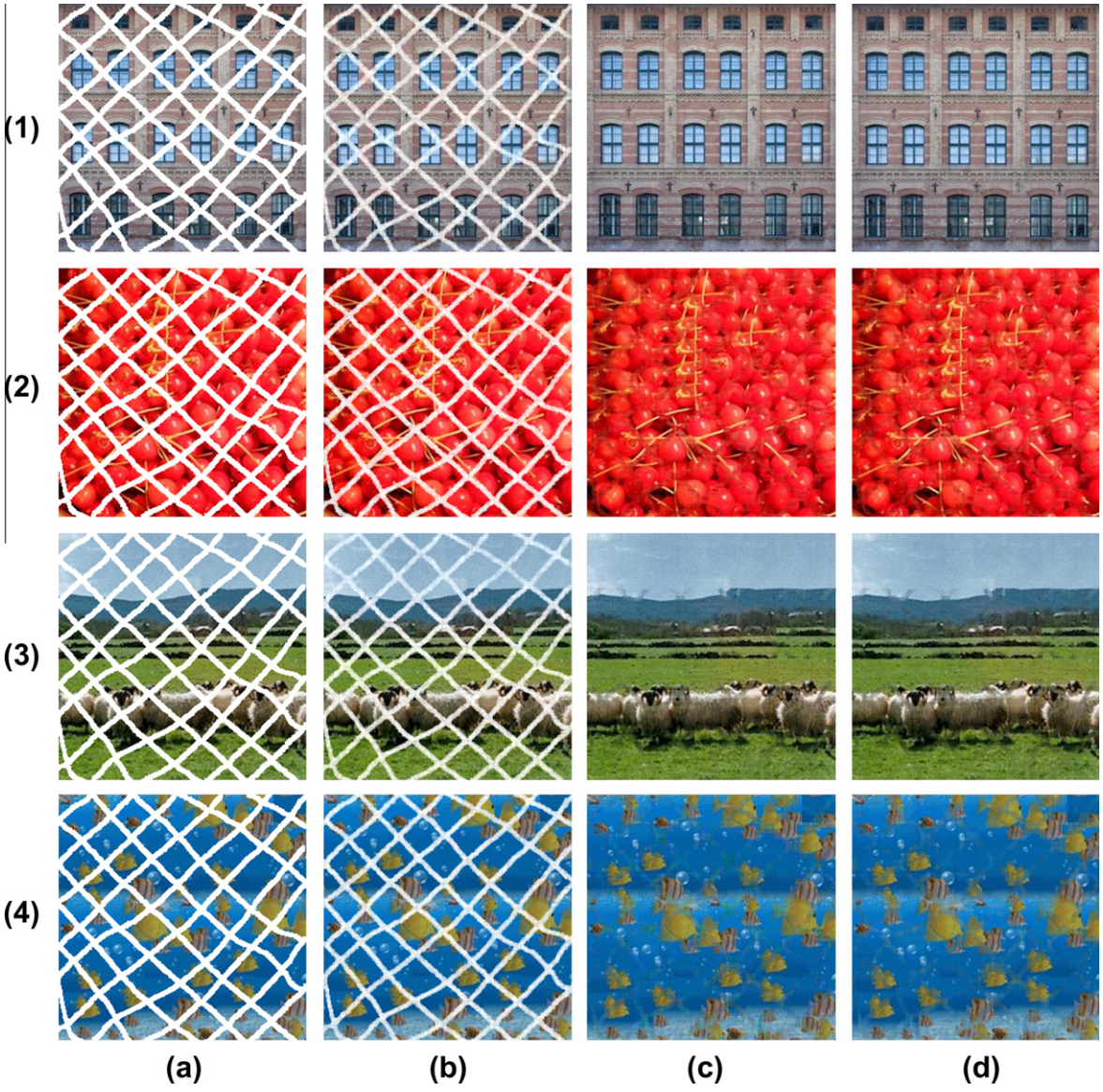

Fig. 10. Comparisons in terms of visual effects. Color images are: (1) Window. (2) Cherry. (3) Sheep. (4) Fish. Columns (a–d) correspond to the images before completion, the obtained results by the CGD-LRTC [39], APG-LRMC [10] and FCSA-LRTC, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 4
Comparisons of the CPU time and RSE for occlusion missing.

	CGD-LRTC [39]		APG-LRMC [10]		FCSA-LRTC	
	Time (s)	RSE	Time (s)	RSE	Time (s)	RSE
Window	133.21	0.3843	100.98	0.0962	133.56	0.0563
Cherry	134.39	0.5583	102.43	0.3201	134.65	0.1069
Sheep	134.96	0.5190	101.33	0.1784	131.23	0.1017
Fish	136.29	0.5886	99.89	0.2234	135.31	0.1056

the LRTC over the low rank matrix completion (LRMC), we also compare the proposed FCSA-LRTC with the APG based LRMC method (APG-LRMC) [10]. As introduced in the above section, the APG-LRMC is not able to solve the tensor completion problem (8) directly. For comparisons, we approximately solve (8) by using the APG-LRMC to conduct the LRMC in d color channels independently. For quantitative evaluation, we compute the Relative Square Error (RSE) for each completion result. The RSE is defined as $\|\mathcal{X}_c - \mathcal{X}\|_F / \|\mathcal{X}\|_F$, where $\mathcal{X}_c$ and $\mathcal{X}$ are the completed image and ground-truth image respectively.

We apply all methods on four color images: window, cherry, sheep and fish respectively. Fig. 7 shows these images. For convenience, they have the same size of 256×256. The regularization parameter α are set as [100, 100, 0] in all three methods for these images.

4.2.2. Random missing

We apply three methods on four 2D color images respectively. The missing samples are random and have white color in the images. The sample ratio is set to be approximately 50%. To perform fair comparisons, all methods run 50 iterations. Fig. 8 shows the visual comparisons of the completion results. The visual effects obtained by the FCSA-LRTC are better than those of the CGD-LRTC [39] and slightly better than those obtained by the APG-LRMC [10]. Table 3 tabulates the RSE and CPU Time by different methods on different color images. The FCSA-LRTC always obtains the smallest RSE in all color images, which illustrates its good performance for low-rank tensor completion.

Fig. 9 gives the performance comparisons among different methods in terms of iterations vs. RSE and CPU time. To test the

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011

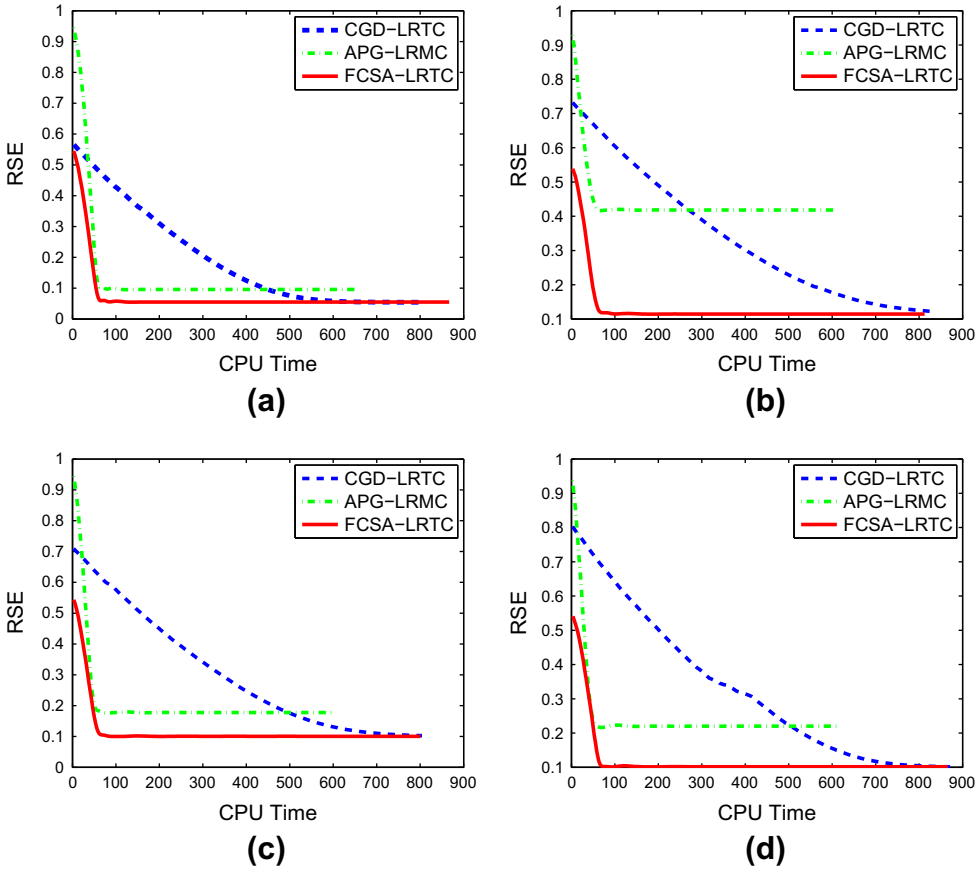

Fig. 11. Performance comparisons in terms of CPU Time and RSE: (a) Window image. (b) Cherry image. (c) Sheep image. (d) Fish image.

completion accuracy, all three methods are run over 300 iterations. The FCSA-LRTC always obtains the best completion results on all color images by achieving the lowest RSE, which shows the effectiveness of the composite splitting techniques in the FCSA for LRTC on color images. The CGD-LRTC [39] is worse than the FCSA-LRTC in terms of the convergence performance because of the slow convergence property of the CBD method, although the former can still obtain similar completion results with enough iterations. The APG-LRMC converges quickly because of the effective acceleration scheme. However, the completion accuracy is worse than those of the FCSA-LRTC and CGD-LRTC even with enough iterations, which shows the advantages of the LRTC over the LRMC for the tensor data. The reason is that, the APG-LRMC solves the tensor completion problem by conducting the LRMC independently in different color channel, which ignores the dependency between different color channels in color images.

4.2.3. Occlusion missing

In this experiment, the missing samples are not randomly distributed. We apply three methods on four 2D color images respectively. To perform fair comparisons, all methods run over 50 iterations. Fig. 10 shows the visual comparisons of the completion results. In this case, the visual effects obtained by the FCSA-LRTC are also better than those of the CGD-LRTC [39] and slightly better than those obtained by the APG-LRMC [10]. Table 4 lists the RSE and CPU Time by different methods on different color images. The FCSA-LRTC always obtains the smallest RSE in all color images, which shows its good performance for the low-rank tensor completion.

Fig. 11 gives the performance comparisons in terms of the CPU time over the RSE for different methods on four color images. To test the completion accuracy, all 3 methods run 300 iteration. The FCSA-LRTC always obtains the best completion results on all color images by achieving the lowest RSE in the least CPU Time, which shows the effectiveness of the composite splitting techniques in the FCSA for LRTC on color images. The CGD-LRTC [39] is worse than the FCSA-LRTC in terms of the convergence performance because of the slow convergence property of the CBD method, although the former can also obtain good completion results with enough iterations. The APG-LRMC converges quickly because of the effective acceleration scheme. However, the completion accuracy is worse than those of the FCSA-LRTC and CGD-LRTC even with enough iterations. These results further show the effective scheme of the composite splitting and the advantages of the LRTC over the LRMC for the tensor data. The reconstruction performance of the FCSA-LRTC is the best in terms of both completion accuracy and computational complexity, which further demonstrates the effectiveness and efficiency of the FCSA for the low rank tensor completion.

4.3. Discussion

All of the above experimental results have validated the effectiveness and efficiency of the proposed composite splitting algorithms for convex optimization. Our main contributions are as follows:

- We propose a general scheme to solve the composite regularization problem. It transforms the original harder composite regularization problem into multiple simpler subproblems, which leads to higher accuracy and lower computation complexity. Due to its general properties, it can be widely used in different applications for convex optimization and is well adapted for large scale data in practice.

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011

- The proposed FCSA can be used to efficiently reconstruct the compressed MR images. It minimizes a linear combination of three terms corresponding to a least square data fitting, total variation (TV) and L1 norm regularization. The computational complexity of the FCSA is only $\mathcal{O}(n\log(n))$ in each iteration. It also has strong convergence properties. It has been shown to signigicantly outperform the classic CG methods [1] and two state-of-the-art methods (TVCMRI [2] and RecPF [3]) in terms of both accuracy and complexity.
- The proposed FCSA can also be efficiently applied to conduct low rank tensor completion. It minimizes a linear combination of a least square data fitting term and a composite nuclear norm regularization. Experiments are conducted on several color images with different sample schemes. It has been shown to significantly outperform the state-of-the-art low-rank tensor completion method [39] in terms of both accuracy and computation complexity. Its completion accuracy is also better than that of the APG-LRMC [10]. These results further show the effective scheme of the composite splitting and the advantages of the LRTC over the LRMC for the tensor data.
- The step size in the ISTA and the FISTA is designed according to the inverse of the Lipschitz constant L_f. Actually, using larger values is known to be a way of obtaining faster versions of the algorithm [40]. Future work will study the combination of this techniques with the CSD or FCSA together. It is expected to further accelerate the optimization for this kind of problems.

5. Conclusion

In this paper, we proposed the composite splitting algorithms based on splitting techniques and accelerated gradient descent. The proposed algorithms decompose a hard composite regularization problem into multiple simpler subproblems and efficiently solve them. This is well adapted for practical applications involving large-scale data optimization. The computational complexities of the proposed algorithms are very low in each iteration. The promising numerical results for the compressed MR image reconstruction and low-rank tensor completion validate the advantages of the proposed algorithms. Future work will include using the proposed algorithms on more application and on large scale data.

References

[1] D.D.M. Lustig, J. Pauly, Sparse MRI: the application of compressed sensing for rapid MR imaging, in: Magnetic Resonance in Medicine, 2007.
[2] S. Ma, W. Yin, Y. Zhang, A. Chakraborty, An efficient algorithm for compressed MR imaging using total variation and wavelets, in: Proceedings of CVPR, 2008.
[3] J. Yang, Y. Zhang, W. Yin, A fast alternating direction method for TVL1-L2 signal reconstruction from partial fourier data, IEEE J. Sel. Top. Signal Process. 4 (2) (2010) 288–297.
[4] D. Gabay, Applications of the method of multipliers to variational inequalities, in: Augmented Lagrange Methods: Applications to the Solution of Boundary-Valued Problems, North Holland, Amsterdam, 1983, pp. 299–331.
[5] P.L. Combettes, V.R. Wajs, Signal recovery by proximal forward-backward splitting, SIAM J. Multiscale Model. Simul. 19 (2008) 1107–1130.
[6] P. Tseng, A modified forward-backward splitting method for maximal monotone mappings, SIAM J. Control Optim. 38 (2000) 431–446.
[7] E.T. Hale, W. Yin, Y. Zhang, Fixed-point continuation for L1-minimization: methodology and convergence, SIAM J. Optim. 19 (2008) 1107–1130.
[8] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM J. Imaging Sci. 2 (1) (2009) 183–202.
[9] A. Beck, M. Teboulle, Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems, IEEE Trans. Image Process. 18 (113) (2009) 2419–2434.
[10] K.-C. Toh, S. Yun, An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems, Pacific J. Math. 6 (2010) 615–640.
[11] S. Ji, J. Ye, An accelerated gradient method for trace norm minimization, in: Proceedings of ICML, 2009.
[12] J.E. Spingarn, Partial inverse of a monotone operator, Appl. Math. Optim. 10 (1983) 247–265.
[13] J. Eckstein, B.F. Svaiter, General projective splitting methods for sums of maximal monotone operators, SIAM J. Control Optim. 48 (2009) 787–811.
[14] D. Gabay, B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite-element approximations, Comput. Math. Appl. 2 (1976) 17–40.
[15] R. Glowinski, P.L. Tallec, Augmented Lagrangian and operator-splitting methods in nonlinear mechanics, SIAM Stud. Appl. Math. 9 (1989).
[16] P. Tseng, Applications of a splitting algorithm to decomposition in convex programming and variational inequalities, SIAM J. Control Optim. 29 (1991) 119–138.
[17] B.S. He, L.-Z. Liao, D. Han, H. Yang, A new inexact alternating direction method for monotone variational inequalities, Math. Program. 92 (2002) 103–118.
[18] E.J. Candes, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information, IEEE Trans. Inform. Theory 52 (2006) 489–509.
[19] D. Donoho, Compressed sensing, IEEE Trans. Inform. Theory 52 (4) (2006) 1289–1306.
[20] T. Goldstein, S. Osher, The Split Bregman Algorithm for L1 Regularized Problems, Tech. Rep., UCLA, CAM Report 08-29, 2008.
[21] M. Afonso, J. Bioucas-Dias, M. Figueiredo, Fast image recovery using variable splitting and constrained optimization, in: IEEE Transactions on Image Processing, Submitted, 2009.
[22] J. Yang, Y. Zhang, Alternating Direction Algorithms for L1-Problems in Compressive Sensing, Tech. Rep., Rice University, TR 09-37, 2009.
[23] J.M. Bioucas-Dias, M. Figueiredo, An iterative algorithm for linear inverse problems with compound regularizers, in: Proceedings of ICIP, 2008, pp. 685–688.
[24] Y.-W. Wen, M.K. Ng, W.-K. Ching, Iterative algorithms based on decoupling of deblurring and denoising for image restoration, SIAM J. Sci. Comput. 30 (5) (2008) 2655–2674.
[25] Y. Wang, J. Yang, W. Yin, Y. Zhang, A new alternating minimization algorithm for total variation image reconstruction, SIAM J. Imaging Sci. 1 (2008) 248–272.
[26] Z. Wen, D. Goldfarb, W. Yin, Alternating Direction Augmented Lagrangian Methods for Semidefinite Programming, Tech. Rep., Columbia University, New York, 2009.
[27] J. Malick, J. Povh, F. Rendl, A. Wiegele, Regularization methods for semidefinite programming, SIAM J. Optim. 20 (2009) 336–356.
[28] X. Yuan, Alternating direction methods for sparse covariance selection, preprint, 2009.
[29] X. Yuan, J. Yang, Sparse and low rank matrix decomposition via alternating direction methods, preprint, 2009.
[30] D. Goldfarb, S. Ma, Fast Multiple Splitting Algorithms for Convex Optimization, Tech. Rep., Department of IEOR, Columbia University, New York, 2009.
[31] Y.E. Nesterov, A method for solving the convex programming problem with convergence rate o($1/k^2$), Dokl. Akad. Nauk SSSR 269 (3) (1983) 543–547.
[32] Y.E. Nesterov, Gradient Methods for Minimizing Composite Objective Function, Tech. Rep., 2007. <http://www.ecore.beDPs/dp1191313936.pdf>.
[33] P.L. Combettes, J.-C. Pesquet, A proximal decomposition method for solving convex variational inverse problems, Inverse Problems 24 (2008) 1–27.
[34] P.L. Combettes, Iterative construction of the resolvent of a sum of maximal monotone operators, J. Convex Anal. 16 (2009) 727–748.
[35] D.P. Bertsekas, Nonlinear Programming, second ed., Athena Scientific, Belmont, Massachusetts, 1999.
[36] S. Ma, D. Goldfarb, L. Chen, Fixed point and Bregman iterative methods for matrix rank minimization, Math. Program. (2009) 1–33.
[37] J.-F. Cai, E.J. Candes, Z. Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optim. 20 (2010) 1956–1982.
[38] J. Yang, X. Yuan, An inexact alternating direction method for trace norm regularized least squares problem, preprint, 2010.
[39] J. Liu, P. Musialski, P. Wonka, J. Ye, Tensor completion for estimating missing values in visual data, in: Proceedings of ICCV, 2009.
[40] S. Wright, R. Nowak, M. Figueiredo, Sparse reconstruction by separable approximation, IEEE Trans. Signal Process. 57 (7) (2009) 2479–2493.

Please cite this article in press as: J. Huang et al., Composite splitting algorithms for convex optimization, Comput. Vis. Image Understand. (2011), doi:10.1016/j.cviu.2011.06.011