

SIMULTANEOUS CARTOON AND TEXTURE INPAINTING

JIAN-FENG CAI

Department of Mathematics, University of California, Los Angeles, CA 90095

RAYMOND H. CHAN

Department of Mathematics, Chinese University of Hong Kong
Shatin, Hong Kong

ZUOWEI SHEN

Department of Mathematics, National University of Singapore
2 Science Drive 2, Singapore 117543

(Communicated by the associate editor name)

ABSTRACT. Real images usually have two layers, namely, cartoons (the piecewise smooth part of the image) and textures (the oscillating pattern part of the image). Both these two layers have sparse approximations under some tight frame systems such as framelet, translation invariant wavelet, curvelet, and local DCTs. In this paper, we solve image inpainting problems by using two separate tight frame systems which can sparsely represent cartoons and textures respectively. Different from existing schemes in the literature which are either analysis-based or synthesis-based sparsity priors, our minimization formulation balances these two priors. We also derive iterative algorithms to find their solutions and prove their convergence. Numerical simulation examples are given to demonstrate the applicability and usefulness of our proposed algorithms in image inpainting.

1. Introduction. The problem of inpainting is to fill-in the missing part in images. It is an interesting and important inverse problem. It arises, for example, in removing scratches in photos, in restoring ancient drawings, and in filling in the missing pixels of images transmitted through a noisy channel. We need to extract information such as edges and textures from the observed data to fill in the missing part such that shapes and patterns are consistent in the human vision.

One popular approach for image inpainting is the *PDE-based method*. The idea is to propagate the geometric information of the curves along the edges by specially designed differential operators. Since the PDE-based approaches are able to keep the edges, it performs very well for piecewise smooth images. As our approach is not along this line, we refer the readers to [2, 3, 4, 12, 13, 14, 31] for details.

Our approach is based on *tight frames* which is more related to those in [6, 11, 25, 26, 27, 36]. For simplicity, we denote images by vectors in \mathbb{R}^N by concatenating their columns. Let A be an M -by- N (with $M \geq N$) real-valued matrix such that

2000 *Mathematics Subject Classification.* Primary: 94A08; Secondary: 49N45, 68U10, 65T60.

Key words and phrases. Tight frame, Image inpainting, Cartoon and texture.

The first author is supported by the Wavelets and Information Processing Programme under a grant from DSTA, Singapore, and the second author is supported in part by HKRGC Grant CUHK 400508.

$A^T A = I$, where I is the identity matrix. Then, the rows of A form a *tight frame* in \mathbb{R}^N . Thus a collection of rows of vectors A in \mathbb{R}^N is a tight frame if for every vector $\mathbf{f} \in \mathbb{R}^N$,

$$\mathbf{f} = \sum_{\mathbf{y} \in A} \langle \mathbf{f}, \mathbf{y} \rangle \mathbf{y}.$$

The matrix A and A^T are called the *analysis* and *synthesis* operators respectively. Since the system A is linearly dependent in general, there are infinitely many ways to represent \mathbf{f} in terms of the tight frame A . The vector $A\mathbf{f}$ is called the *canonical coefficients sequence* representing $\mathbf{f} \in \mathbb{R}^N$ that is uniquely determined by \mathbf{f} . The canonical coefficient sequence $A\mathbf{f}$ has the minimal ℓ_2 -norm among all possible frame coefficient sequences that represent \mathbf{f} . The tight frame satisfies $A^T A\mathbf{f} = \mathbf{f}$. In general, $AA^T \neq I$. When $AA^T = I$, the system A is an orthonormal system. We remark that in the literature the tight frame here sometimes is also referred to as a *Parseval frame* or a tight frame with bound one.

Real images usually have two layers, referring to cartoons (the piecewise smooth part of the image) and textures (the oscillating pattern part of the image). Both these two layers have sparse approximations under some tight frame systems. Examples of tight frames that can sparsely approximate the cartoon part are curvelets proposed in [8, 9], orthonormal wavelets constructed in [19], and wavelet tight frames constructed by the unitary extension principle in [20, 34]. The basic assumption for tight frame based image inpainting is that real images have sparse approximations under the tight frame representations. The advantages of using redundant systems is that they lead to robust signal representation in which partial loss of the data can be tolerated without adverse effects.

Since tight frame systems are redundant systems, the mapping from the image \mathbf{f} to its coefficient is not one-to-one, i.e., the representation of \mathbf{f} in the frame domain is not unique. Therefore, there are two formulations for the sparse approximation of the underlying images, namely analysis-based and synthesis-based approaches. In the analysis-based approach, we assume that the analyzed coefficient $A\mathbf{f}$ can be sparsely approximated. This is usually reduced to minimizing a functional involving a term $\|A\mathbf{f}\|_1$. If one choose A to be the one level Haar translation-invariant wavelet system, the analysis-based approach approximate total-variation based approaches [35] for image processing. In the synthesis-based approach, the underlying image \mathbf{f} is assumed to be synthesized by a sparse coefficient α . In this formulation, it usually involves a term $\|\alpha\|_1$ in the minimization energy. The synthesis-based approaches include, for examples, [16, 26].

These two approaches for image inpainting were studied by many authors [16, 24, 25, 26]. In particular, [24] gives a comprehensive analysis on these two approaches. It was pointed out in [24] that there is a big gap between the analysis-based and synthesis-based approaches for image and signal processing. In this paper, we propose an approach that balances the analysis-based and synthesis-based approaches. It also fills the gap between the synthesis-based and analysis-based approaches in some sense. This approach is motivated by the inpainting algorithm based on framelets in [6, 11, 27], which balances the analysis-based and synthesis-based priors in the minimization functional.

The one tight frame based inpainting algorithm performs well only when the underlying images are piecewise smooth, which are sometimes referred to as cartoon images. However, real images usually consist of both cartoons and textures.

These two layers are of very different characteristics: cartoons are piecewise smooth functions whereas textures contain oscillating patterns. Therefore, these two different layers of images should be considered separately in image inpainting. One natural idea is to use two tight frame systems that can sparsely represent cartoons and textures separately, which was first proposed in [25, 37]. In particular, we use tight frames A_1 and A_2 , where A_1 can sparsely represent cartoons and A_2 can sparsely represent textures. Then, we extend the unified inpainting algorithm from one tight frame to two tight frames to handle images containing both cartoons and textures. Numerical algorithms based on proximal forward-backward splitting [18] are employed to solve the proposed minimization problems, and the convergence proofs are provided. Numerical simulation examples are also given to demonstrate the applicability and usefulness of our proposed algorithms in image inpainting. Other methods that decompose images into cartoon and textures with possible applications for image inpainting can be found in [1, 4, 7, 21, 33].

The remaining part of the paper is organized as follows. In Section 2, we present our tight frame based image inpainting. The algorithms for solving the models are proposed in Section 3. Finally, numerical simulation examples are given in Section 4 to demonstrate the applicability and usefulness of our algorithms in image inpainting.

2. Tight frame based inpainting. In this section, we present tight frame based image inpainting. To begin with, we introduce some notations. Let the original image \mathbf{f} be defined on the domain $\Omega = \{1, 2, \dots, N\}$ and the nonempty set $\Lambda \subsetneq \Omega$ be the given observed region. Then the observed (incomplete) image \mathbf{g} is

$$\mathbf{g}[i] = \begin{cases} \mathbf{f}[i] + \epsilon[i], & i \in \Lambda, \\ \text{arbitrary}, & i \in \Omega \setminus \Lambda, \end{cases} \quad (1)$$

where $\epsilon[i]$, the i -th coordinate of ϵ , are noise. The goal of image inpainting is to find \mathbf{f} from \mathbf{g} , i.e., find an image \mathbf{f} defined on Ω such that $P_\Lambda \mathbf{f} \approx P_\Lambda \mathbf{g}$, where P_Λ is the diagonal matrix defined by

$$P_\Lambda[i, j] = \begin{cases} 1, & i = j \in \Lambda, \\ 0, & \text{otherwise.} \end{cases}$$

When $\epsilon[i] = 0$ for all $i \in \Lambda$, \mathbf{f} is a solution of an interpolation. Otherwise, we seek a smooth solution \mathbf{f} such that $\|P_\Lambda(\mathbf{f} - \mathbf{g})\|$ is small.

2.1. One frame inpainting. We start by introducing the algorithm proposed in [6, 11, 27] which provides motivation for our proposed algorithms here. The algorithm is

$$\mathbf{f}^{(n+1)} = P_\Lambda \mathbf{g} + (I - P_\Lambda) A^T \mathbf{T}_\mathbf{u}(A \mathbf{f}^{(n)}), \quad (2)$$

where $\mathbf{T}_\mathbf{u}(\mathbf{x})$ is the soft-thresholding function [22] defined by

$$\mathbf{T}_\mathbf{u}(\mathbf{x}) = (t_{\mathbf{u}[1]}(\mathbf{x}[1]), \dots, t_{\mathbf{u}[i]}(\mathbf{x}[i]), \dots)^T \quad (3)$$

with

$$t_u(x) = \begin{cases} 0, & \text{if } |x| \leq u, \\ x - \text{sgn}(x)u, & \text{otherwise.} \end{cases} \quad (4)$$

Here \mathbf{u} is a vector of thresholding parameters. Let $\alpha^{(n)} = \mathbf{T}_\mathbf{u}(A \mathbf{f}^{(n)})$. Then $\mathbf{f}^{(n)}$ converges if and only if $\alpha^{(n)}$ converges. Let α be the limit of $\alpha^{(n)}$. When $\epsilon[i] = 0$

for all $i \in \Lambda$ in (1), then \mathbf{f} is the interpolation solution given by $P_\Lambda \mathbf{g} + (I - P_\Lambda)A^T \alpha$. Otherwise, if $\epsilon \neq 0$, \mathbf{f} is the smoothing solution given by $A^T \alpha$.

It was proven in [6] that the sequence $\alpha^{(n)} = \mathbf{T}_\mathbf{u}(A\mathbf{f}^{(n)})$ converges to a minimizer of

$$\min_{\alpha} \left\{ \frac{1}{2} \|P_\Lambda(\mathbf{g} - A^T \alpha)\|_2^2 + \frac{1}{2} \|(I - AA^T)\alpha\|_2^2 + \|\text{diag}(\mathbf{u})\alpha\|_1 \right\}. \quad (5)$$

As explained in [6], the role of the first term is data-fitting, i.e. to make sure that the recovered image is close enough to the known data \mathbf{g} , and the last two terms are regularization, i.e. to balance the sparsity and smoothness of the solution. It is well known that if the true solution is a piecewise smooth function, it has sparse approximation by framelet systems (see e.g. [5, 28]), which can be measured by $\|\alpha\|_0$, the number of nonzero terms in the sequence α . However, minimizing the ℓ_0 -norm is difficult. As shown in [23], the ℓ_1 -norm relaxes this difficult problem to the “nearest” convex problem and can still be a good measure of the sparsity of the sequence. Therefore, to increase the sparsity, one likes to minimize the (weighted) ℓ_1 -norm of α among all possible solutions, which is the term $\|\text{diag}(\mathbf{u})\alpha\|_1$ in (5).

While we like the penalty function to enhance the sparsity, we also like it to control the smoothness of the solution via a certain function norm. However, it is the canonical frame coefficients that are normally linked to the underlying function norm. For example, it was shown from framelet theory [5, 28] that the (weighted) ℓ_1 -norm of the canonical framelet coefficient sequence of a function is equivalent to its Besov norm in the space $B_{1,1}^\sigma$ under some mild conditions on the wavelets. Hence, we also require α to be close to some canonical coefficient sequence, i.e., to the range of A , so that we can be sure that the (weighted) ℓ_1 -norm of α is approximately linked to the Besov norm of the true function. Thus we also need to penalize the distance between α and the range of A . Note that the term $(I - AA^T)$ in (5) is the projection operator onto the kernel of A^T , i.e., the orthogonal complement of the range of A . In other words, the term $\|(I - AA^T)\alpha\|_2^2$ in (5) exactly penalizes the distance of α to the range of A .

In the formulation of (5), the weighting of the second term $\|(I - AA^T)\alpha\|_2^2$ is fixed to be $\frac{1}{2}$. It is natural to vary this weighting to obtain more flexibility. This amounts to the following unified framework for tight frame inpainting:

$$\min_{\alpha} \left\{ \frac{1}{2} \|P_\Lambda(\mathbf{g} - A^T \alpha)\|_2^2 + \frac{\kappa}{2} \|(I - AA^T)\alpha\|_2^2 + \|\text{diag}(\mathbf{u})\alpha\|_1 \right\}, \quad (6)$$

where $\kappa/2$ is the weighting of the second term. The parameter κ controls the distance of α to the range of A . As κ increases, the distance of α to the range of A decreases. This leads to changes in the following two aspects: (i) the smoothness of the underlying function increases since the link between the ℓ_1 -norm of α and the Besov norm is enhanced; (ii) α becomes less sparse because the ℓ_1 -norm of α will increase.

The formulation of (6) is also motivated from [24], where the gap between the synthesis-based sparsity and analysis-based sparsity priors are studied. When $\kappa = 0$, (6) is reduced to

$$\min_{\alpha} \left\{ \frac{1}{2} \|P_\Lambda(\mathbf{g} - A^T \alpha)\|_2^2 + \|\text{diag}(\mathbf{u})\alpha\|_1 \right\}. \quad (7)$$

Since the ℓ_1 -norm of α leads to sparsity, (7) amounts to the sparsest frame coefficient α , and then the recovered image is synthesized by $A^T \alpha$. Therefore, the

basic assumption in (7) is that the image is synthesized by a sparse frame coefficient. Hence, (7) is a synthesis-based approach, following the terminology in [24]. Synthesis-based methods were proposed in, for instances, [16, 25, 26, 37].

When $\kappa = \infty$, the distance $\|(I - AA^T)\alpha\|_2^2$ must be 0. This implies that α is in the range of A , so we can rewrite (6) as

$$\min_{\alpha \in \text{Range}(A)} \left\{ \frac{1}{2} \|P_\Lambda(\mathbf{g} - A^T \alpha)\|_2^2 + \|\text{diag}(\mathbf{u})\alpha\|_1 \right\}, \quad (8)$$

where $\text{Range}(A)$ stands for the range of A . Because α is in the range of A , for any α , there exists $\mathbf{f} \in \mathbb{R}^N$ such that $\alpha = A\mathbf{f}$. Therefore, (8) can be reformulated in terms of \mathbf{f} as the following

$$\min_{\mathbf{f}} \left\{ \frac{1}{2} \|P_\Lambda(\mathbf{g} - \mathbf{f})\|_2^2 + \|\text{diag}(\mathbf{u})A\mathbf{f}\|_1 \right\}. \quad (9)$$

Thus, the basic assumption in (9) is that the analyzed frame coefficient $A\mathbf{f}$ is sparse. Formulation (9) is called a analysis-based approach in [24]. Analysis-based approach leads to, for example, total variation inpainting in [13, 14, 15, 35].

From geometric viewpoint, the authors in [24] pointed out that there is a big gap between synthesis-based approaches (c.f. (7)) and analysis-based approaches (c.f. (8) and (9)). From the experiments performed in [24] on the synthetic signals, both approaches have their own favorable data sets, and it is hard to conclude which approach is superior than the other in general. From the experiments on the real-world images, it was shown that the analysis-based approach performs better than the synthesis-based one when the tight frame system is a overlapped local DCT. All these indicate the gap between these two approaches. Our formulation (6) sets up a bridge to link the synthesis-based and analysis-based approaches: When κ goes from 0 to ∞ , the synthesis-based and analysis-based approaches are the start point and destination respectively.

Finally, we would like to point out the relation between the case $\kappa = 1$ and the translation invariant wavelet [17]. When $\kappa = 1$, (6) is analogous to the translation invariant denoising in [17]. To see this, we further assume that $\Lambda = \Omega$, i.e., inpainting problem (1) becomes an image denoising one. Then, (6) with $\kappa = 1$ and $\Lambda = \Omega$ becomes

$$\min_{\alpha} \left\{ \frac{1}{2} \|\mathbf{g} - A^T \alpha\|_2^2 + \frac{1}{2} \|(I - AA^T)\alpha\|_2^2 + \|\text{diag}(\mathbf{u})\alpha\|_1 \right\}. \quad (10)$$

It is easy to verify that the solution of (10) is

$$\mathbf{f}^* = A^T \mathbf{T}_\mathbf{u} A \mathbf{g}. \quad (11)$$

The solution \mathbf{f}^* in (11) is just the translation invariant denoising in [17], whenever A is an undecimated wavelet system. Moreover, iteration (2) with $\Lambda = \Omega$ converges in one step, and it gives (11). Therefore, (6) with $\kappa = 1$ and $\Lambda = \Omega$ is the same as the translation invariant denoising. When $\Lambda \subset \Omega$ and $\kappa = 1$, we obtain algorithm (2). This algorithm can be seen as an extension of denoising scheme (11) to inpainting. In (2), denoising scheme (11) and replacement of the pixel values on Λ are performed alternatively.

2.2. Simultaneously cartoon and texture inpainting. As shown in [6], the approach (5), or generally (6), performs well when the underlying solution is piecewise smooth, which is sometimes referred to as cartoon images. However, real

images usually consist of both cartoons and textures. These two layers are of very different characteristics: cartoons are piecewise smooth functions whereas textures contain oscillating patterns. Therefore, these two different layers of images should be considered separately in image inpainting. One natural idea is to use two tight frame systems that can sparsely represent cartoons and textures separately as suggested by [16, 24, 25, 26]. This motivates our new algorithm which we are going to describe below.

Assume that the tight frame system A_i , $i = 1, 2$, can sparsely approximate cartoons and textures respectively. Then the inpainting problem is solved in the transform domain separately under the two systems, i.e. we find sequences α_i , $i = 1, 2$, such that $\mathbf{f}_1 = A_1^T \alpha_1$ represents the cartoon content of the underlying image, $\mathbf{f}_2 = A_2^T \alpha_2$ represents the texture content of the image, and $\mathbf{f} = \mathbf{f}_1 + \mathbf{f}_2$ represents the whole image. The next issue is what kind of properties the solution \mathbf{f} should have. Motivated by (6), here we suggest to develop an algorithm minimizing something similar to (6), i.e., we require α_i , $i = 1, 2$, to be a minimizer of

$$\min_{\alpha_1, \alpha_2} \left\{ \frac{1}{2} \|P_\Lambda(\mathbf{g} - \sum_{i=1}^2 A_i^T \alpha_i)\|_2^2 + \sum_{i=1}^2 \frac{\kappa}{2} \|(I - A_i A_i^T) \alpha_i\|_2^2 + \sum_{i=1}^2 \|\text{diag}(\mathbf{u}_i) \alpha_i\|_1 \right\}. \quad (12)$$

We remark that we can use different κ 's for A_1 and A_2 in (12). However, for the simplicity of notations, we use one κ for the two tight frames. The results in the remaining part can be generated straightforwardly to different κ 's.

The role of each term in (12) is as follows. The first term $\|P_\Lambda(\mathbf{g} - \sum_{i=1}^2 A_i^T \alpha_i)\|_2^2$ is the data fidelity. It measures the error between the observed image and the restored image restricted on Λ . The third term $\|\text{diag}(\mathbf{u}_i) \alpha_i\|_1$ penalizes the weighted ℓ_1 -norm of the tight frame coefficients α_1 and α_2 , hence ensures the sparsity of the tight frame coefficients. As pointed out above, the second term $\|(I - A_i A_i^T) \alpha_i\|_2^2$ measures how close α_i is to some canonical coefficient sequence, i.e. to the range of A_i . Hence, as discussed before, we see that (12) is a trade-off between the smoothness and sparsity of the solution.

One may challenge why we do not construct a new tight frame A by stacking A_1 and A_2 together, i.e., $A = \frac{1}{\sqrt{2}}[A_1^T \ A_2^T]^T$, and then use the one tight frame inpainting (6). This is not a good strategy for simultaneously cartoon and texture inpainting unless κ is small enough. The reason is as follows. Assume that we define $A = \frac{1}{\sqrt{2}}[A_1^T \ A_2^T]^T$ and solve (6). Write $\alpha = [\alpha_1^T, \alpha_2^T]^T$, where α_1 and α_2 are the coefficient corresponding to A_1 and A_2 respectively. Since κ is not small, the second term $\|(I - AA^T) \alpha\|_2^2$ in (6) is small. This means that α is close to the range of $A = \frac{1}{\sqrt{2}}[A_1^T \ A_2^T]^T$. Therefore, there exists an image $\bar{\mathbf{f}}$ such that $\alpha_1 \approx A_1 \bar{\mathbf{f}}$ and $\alpha_2 \approx A_2 \bar{\mathbf{f}}$. Hence, the cartoon part of inpainted image is $A_1^T \alpha_1 \approx \bar{\mathbf{f}}$, and the texture part is $A_2^T \alpha_2 \approx \bar{\mathbf{f}}$. Consequently, the cartoon part and the texture part of the inpainted image will be close to each others. However, these two parts are assumed to have different characteristics. We see therefore that it is not a good approach to combine A_1 and A_2 together to form a new tight frame A , and then apply (6).

When $\kappa = 0$, problem (12) becomes

$$\min_{\alpha_1, \alpha_2} \left\{ \frac{1}{2} \|P_\Lambda(\mathbf{g} - \sum_{i=1}^2 A_i^T \alpha_i)\|_2^2 + \sum_{i=1}^2 \|\text{diag}(\mathbf{u}_i) \alpha_i\|_1 \right\}. \quad (13)$$

This reveals that (12) with $\kappa = 0$ is a synthesis-based approach. Therefore, (13) is able to find sparsest α_i (sparsity comes from minimizing the term $\|\text{diag}(\mathbf{u}_i)\alpha_i\|_1$ as pointed out above). However, the ℓ_1 norm of the tight frame coefficients do not link to any norm of the cartoon and texture components of the underlying image.

When $\kappa = \infty$, we must have that $\|(I - A_i A_i^T)\alpha_i\|_2^2 = 0$ whenever the minimum is reached. This implies that α_i must be canonical, and α_i must be in the range of A_i instead of in \mathbb{R}^M . In this case, problem (12) is led to

$$\min_{\alpha_i \in \text{Range}(A_i), i=1,2.} \left\{ \frac{1}{2} \|P_\Lambda(\mathbf{g} - \sum_{i=1}^2 A_i^T \alpha_i)\|_2^2 + \sum_{i=1}^2 \|\text{diag}(\mathbf{u}_i)\alpha_i\|_1 \right\}. \quad (14)$$

We can rewrite (14) in terms of the cartoon component \mathbf{f}_1 and the texture component \mathbf{f}_2 as the following

$$\min_{\mathbf{f}_1, \mathbf{f}_2} \left\{ \frac{1}{2} \|P_\Lambda(\mathbf{g} - \sum_{i=1}^2 \mathbf{f}_i)\|_2^2 + \sum_{i=1}^2 \|\text{diag}(\mathbf{u}_i)A_i \mathbf{f}_i\|_1 \right\}. \quad (15)$$

This means that (12) with $\kappa = \infty$ is an analysis-based approach. Note that the solution of (14) seeks both the sparsity of α_i and certain smoothness of \mathbf{f}_i by requiring α_i to be in the range of A_i . Since there is only one unique canonical representation of any given function, (14) is actually giving up the possible extra sparsity of the underlying solution brought in by the redundancy. Again, this is a trade-off between sparsity and regularity. Comparing with (13) and (14), (14) is a minimization problem with constraints, which, as we will see, is more difficult to solve. In contrast, (13) exploits the additional sparsity the redundant system may bring in, and at the same time, controls the smoothness of the solution by penalizing the distance of the solution to the canonical frame coefficients.

Both models (13) and (15) are proposed in [25] for cartoon and texture image inpainting. In fact, the authors of [25] proposed more general models with an additional TV norm penalty term in both (13) and (15). We can formulate our model here with the TV norm penalty term as well. However, since the analysis and algorithm development are essentially the same, we focus our discussions without the TV norm penalty term.

Finally, we remark that, when A_i are orthonormal systems, then the minimization problems (12) with any κ are all reduced to the same problem

$$\min_{\alpha_1, \alpha_2} \left\{ \frac{1}{2} \|P_\Lambda(\mathbf{g} - \sum_{i=1}^2 A_i^T \alpha_i)\|_2^2 + \sum_{i=1}^2 \|\text{diag}(\mathbf{u}_i)\alpha_i\|_1 \right\},$$

since in this case $A_i A_i^T = I$ and the range of A_i is \mathbb{R}^M . We further note that any solution of (13) that falls in the ranges of A_i is also a solution of (12) and (14). In fact, to find a solution of (14) is to find a solution of (12) that is in the range of A_i .

3. Algorithms. In this section, we develop an algorithm that converges to a minimizer of (12) with $0 \leq \kappa \leq \infty$.

3.1. Proximal forward-backward splitting. Our main tool for developing our algorithm is based on the proximal forward-backward splitting in [18, 38, 39] from

the theory of optimization and convex analysis in [29]. The main idea of the method is that: in order to find a minimizer of

$$\min_{\mathbf{x}} \{L_1(\mathbf{x}) + L_2(\mathbf{x})\}, \quad (16)$$

one uses the iteration

$$\mathbf{x}^{(n+1)} = \text{prox}_{\gamma L_1}(\mathbf{x}^{(n)} - \gamma \nabla L_2(\mathbf{x}^{(n)})) + \mathbf{z}^{(n)}, \quad (17)$$

where γ is a scalar, $\mathbf{z}^{(n)}$ is a sequence, and ‘‘prox’’ is the proximal operator defined by

$$\text{prox}_{\gamma \varphi}(\mathbf{x}) := \arg \min_{\mathbf{y}} \left\{ \frac{1}{2} \|\mathbf{x} - \mathbf{y}\|_2^2 + \gamma \varphi(\mathbf{y}) \right\}, \quad (18)$$

for any proper, convex and lower semi-continuous function φ . The following convergence theorem is an immediate consequence of Proposition 3.1(i) and Theorem 3.4 in [18]. Since we are working in the finite dimension space \mathbb{R}^M , the convergence is in the Euclidean norm of \mathbb{R}^M .

Proposition 1. *Suppose that L_1 and L_2 satisfy*

1. $[L_1(\mathbf{x}) + L_2(\mathbf{x})]$ is coercive, i.e., whenever $\|\mathbf{x}\|_2 \rightarrow +\infty$, $[L_1(\mathbf{x}) + L_2(\mathbf{x})] \rightarrow +\infty$;
2. L_1 is a proper, convex, lower semi-continuous function; and
3. L_2 is a proper, convex, differentiable function with a $1/b$ -Lipshitz continuous gradient:

$$\|\nabla L_2(\mathbf{x}) - \nabla L_2(\mathbf{y})\|_2 \leq \frac{1}{b} \|\mathbf{x} - \mathbf{y}\|_2, \quad \forall \mathbf{x}, \mathbf{y}, \quad (19)$$

and $0 < \gamma < 2b$; and

4. $\sum_{n=0}^{\infty} \|\mathbf{z}^{(n)}\|_2 < \infty$.

Then there exists at least one minimizer of (16), and for any initial guess $\mathbf{x}^{(0)}$, the iteration (17) converges to a minimizer of (16).

3.2. Minimization algorithms. In this subsection, we will derive convergent algorithms for the minimization problem (12). The assumption we put on the A_i 's are minimal: that they are tight frame systems which can sparsely approximate cartoons and textures respectively. Examples of tight frames that can sparsely approximate piecewise smooth contents in images are curvelets proposed in [8, 9], orthonormal wavelets constructed in [19], and wavelet tight frames constructed by the unitary extension principle in [34]. Examples of tight frames that can sparsely approximate textures are local discrete cosine transform and Gabor transform, see [25, 30, 32]. As pointed out in [25, 37], it is difficult to find a pair of dictionaries so that it can separate all images in general; but the experimental results there show the possibility of finding a pair of properly chosen dictionaries that can separate a relatively large class of images. Note that in our proofs, we will assume A_i to be real matrices for simplicity, but the proofs can be modified to include cases where A_i are complex.

When $\kappa = \infty$, since (12) is transferred from an unconstrained minimization problem to a constrained one, we separate the algorithm into two cases, i.e., the case of $0 \leq \kappa < \infty$ and the case of $\kappa = \infty$.

3.2.1. *Algorithm for finite κ .* To develop an algorithm that converges to a solution of (12) with $0 \leq \kappa < \infty$ by employing the proximal forward-backward splitting proposed in [18], we denote the energy in (12) by

$$F(\alpha_1, \alpha_2) = \frac{1}{2} \|P_\Lambda(\mathbf{g} - \sum_{i=1}^2 A_i^T \alpha_i)\|_2^2 + \sum_{i=1}^2 \frac{\kappa}{2} \|(I - A_i A_i^T) \alpha_i\|_2^2 + \sum_{i=1}^2 \|\text{diag}(\mathbf{u}_i) \alpha_i\|_1. \quad (20)$$

Then we split $F(\alpha_1, \alpha_2)$ as $F(\alpha_1, \alpha_2) = L_1(\alpha_1, \alpha_2) + L_2(\alpha_1, \alpha_2)$, where

$$L_1(\alpha_1, \alpha_2) = \sum_{i=1}^2 \|\text{diag}(\mathbf{u}_i) \alpha_i\|_1, \quad (21)$$

and

$$L_2(\alpha_1, \alpha_2) = \frac{1}{2} \|P_\Lambda(\mathbf{g} - \sum_{i=1}^2 A_i^T \alpha_i)\|_2^2 + \sum_{i=1}^2 \frac{\kappa}{2} \|(I - A_i A_i^T) \alpha_i\|_2^2. \quad (22)$$

In order to apply the iteration (17), we first derive $\text{prox}_{\gamma L_1}$ and ∇L_2 . For $\text{prox}_{\gamma L_1}$, by definition,

$$\text{prox}_{\gamma L_1}(\alpha_1, \alpha_2) = \arg \min_{\beta_1, \beta_2} \sum_{i=1}^2 \left\{ \frac{1}{2} \|\alpha_i - \beta_i\|_2^2 + \gamma \|\text{diag}(\mathbf{u}_i) \beta_i\|_1 \right\}.$$

Following e.g. [18], we have

$$\text{prox}_{\gamma L_1}(\alpha) = \begin{bmatrix} \mathbf{T}_{\gamma \mathbf{u}_1}(\alpha_1) \\ \mathbf{T}_{\gamma \mathbf{u}_2}(\alpha_2) \end{bmatrix}. \quad (23)$$

Here only the soft-thresholding operator is required, so it is very cheap to compute $\text{prox}_{\gamma L_1}$, see (3) and (4). Obviously,

$$\frac{\partial L_2}{\partial \alpha_i} = \kappa \alpha_i - A_i \left(\kappa A_i^T \alpha_i + P_\Lambda(\mathbf{g} - \sum_{i=1}^2 A_i^T \alpha_i) \right). \quad (24)$$

Substituting (23) and (24) into (17) and choosing $\mathbf{z}^{(n)} = 0$ for all n , we obtain the following iteration:

$$\begin{cases} \alpha_1^{(n+1)} = \mathbf{T}_{\gamma \mathbf{u}_1} \left((1 - \kappa_1 \gamma) \alpha_1^{(n)} + \gamma A_1 (\kappa_1 A_1^T \alpha_1^{(n)} + P_\Lambda(\mathbf{g} - A_1^T \alpha_1^{(n)} - A_2^T \alpha_2^{(n)})) \right), \\ \alpha_2^{(n+1)} = \mathbf{T}_{\gamma \mathbf{u}_2} \left((1 - \kappa_2 \gamma) \alpha_2^{(n)} + \gamma A_2 (\kappa_2 A_2^T \alpha_2^{(n)} + P_\Lambda(\mathbf{g} - A_1^T \alpha_1^{(n)} - A_2^T \alpha_2^{(n)})) \right). \end{cases} \quad (25)$$

We remark here, when $\Lambda = \Omega$, this algorithm will give \mathbf{f}_1 and \mathbf{f}_2 that separate the cartoon and texture parts of the original image \mathbf{f} as discussed in [37]. This is the case for all other algorithms in this paper and we will not emphasize it again.

We have the following convergence result for iteration (25).

Theorem 3.1. *Assume that $0 \leq \kappa < \infty$ and $0 < \gamma < 1/\max\{1, \kappa\}$ in (25). Then the sequence $\alpha^{(n)} = [\alpha_1^{(n)}, \alpha_2^{(n)}]$ generated by (25) converges to a solution of (12).*

Proof. We have to check that Conditions 1–4 in Proposition 1 are satisfied. Since all the entries of \mathbf{u}_1 and \mathbf{u}_2 are assumed to be positive, it is obvious that when $\|[\alpha_1^T, \alpha_2^T]^T\|_2 \rightarrow \infty$, $L_1(\alpha_1, \alpha_2) \rightarrow +\infty$. Therefore, L_1 is coercive which means that $F(\alpha_1, \alpha_2) = L_1(\alpha_1, \alpha_2) + L_2(\alpha_1, \alpha_2)$ is also coercive. It is obvious that both $L_1(\alpha_1, \alpha_2)$ and $L_2(\alpha_1, \alpha_2)$ are proper, convex and continuous functions. Therefore, Conditions 1 and 2 in Proposition 1 are verified. Since we have chosen $\mathbf{z}^{(n)} = 0$ for

all n , Condition 4 is fulfilled. Thus, the convergence of (25) follows if we can show that $\nabla L_2(\alpha_1, \alpha_2)$ is Lipschitz continuous with a certain Lipschitz constant so that we can choose a proper γ in (25).

For this, since L_2 is quadratic, we only need to estimate the norm of the Hessian of L_2

$$\nabla^2 L_2 = \begin{bmatrix} \kappa(I - A_1 A_1^T) + A_1 P_\Lambda A_1^T & A_1 P_\Lambda A_2^T \\ A_2 P_\Lambda A_1^T & \kappa(I - A_2 A_2^T) + A_2 P_\Lambda A_2^T \end{bmatrix}.$$

This is done by the matrix decomposition

$$\nabla^2 L_2 = \kappa I + \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} P_\Lambda - \kappa I & P_\Lambda \\ P_\Lambda & P_\Lambda - \kappa I \end{bmatrix} \begin{bmatrix} A_1^T & 0 \\ 0 & A_2^T \end{bmatrix}.$$

It, together with the following equations

$$\left\| \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \right\|_2 = 1, \quad \left\| \begin{bmatrix} P_\Lambda - \kappa I & P_\Lambda \\ P_\Lambda & P_\Lambda - \kappa I \end{bmatrix} \right\|_2 \leq \max\{1 + |1 - \kappa|, \kappa\},$$

implies that

$$\|\nabla^2 L_2\|_2 \leq \kappa + \max\{1 + |1 - \kappa|, \kappa\} = \max\{2, 2\kappa\}.$$

Therefore, the Lipschitz constant in (19) is $1/b = \max\{2, 2\kappa\}$. Hence, $0 < \gamma < 1/\max\{1, \kappa\}$ guarantees the convergence of (25). \square

3.2.2. Algorithm for $\kappa = \infty$. When $\kappa = \infty$, (12) becomes (15). Denote the energy in (15) as

$$E(\mathbf{f}_1, \mathbf{f}_2) = \frac{1}{2} \|P_\Lambda(\mathbf{g} - \mathbf{f}_1 - \mathbf{f}_2)\|_2^2 + \sum_{i=1}^2 \|\text{diag}(\mathbf{u}_i) A_i \mathbf{f}_i\|_1.$$

As before, we split $E(\mathbf{f}_1, \mathbf{f}_2)$ as $E(\mathbf{f}_1, \mathbf{f}_2) = L_1(\mathbf{f}_1, \mathbf{f}_2) + L_2(\mathbf{f}_1, \mathbf{f}_2)$, where

$$L_1(\mathbf{f}_1, \mathbf{f}_2) = \sum_{i=1}^2 \|\text{diag}(\mathbf{u}_i) A_i \mathbf{f}_i\|_1, \tag{26}$$

and

$$L_2(\mathbf{f}_1, \mathbf{f}_2) = \frac{1}{2} \sum_{i=1}^2 \|P_\Lambda(\mathbf{g} - \mathbf{f}_1 - \mathbf{f}_2)\|_2^2. \tag{27}$$

The gradient of L_2 in (27) is given by $\frac{\partial L_2}{\partial \mathbf{f}_i} = -P_\Lambda(\mathbf{g} - \sum_{j=1}^2 \mathbf{f}_j)$. By (17) with $\mathbf{z}^{(n)} = 0$, we get the following iterative algorithm to find a solution of (15):

$$\begin{cases} \mathbf{f}_1^{(n+1)} = \text{prox}_{\gamma \|\text{diag}(\mathbf{u}_1) A_1 \cdot\|_1} \left(\mathbf{f}_1^{(n)} + \gamma P_\Lambda(\mathbf{g} - \mathbf{f}_1^{(n)} - \mathbf{f}_2^{(n)}) \right), \\ \mathbf{f}_2^{(n+1)} = \text{prox}_{\gamma \|\text{diag}(\mathbf{u}_2) A_2 \cdot\|_1} \left(\mathbf{f}_2^{(n)} + \gamma P_\Lambda(\mathbf{g} - \mathbf{f}_1^{(n)} - \mathbf{f}_2^{(n)}) \right). \end{cases} \tag{28}$$

Unlike the cases in Section 3.2.1, $\text{prox}_{\gamma \|\text{diag}(\mathbf{u}_i) A_i \cdot\|_1}$ here is not equivalent to a simple soft-thresholding alone, since for a given vector in the range of A_i , one cannot guarantee that after the soft-thresholding, it will still be in the range of A_i . Moreover, there is not an explicit expression for $\text{prox}_{\gamma \|\text{diag}(\mathbf{u}_i) A_i \cdot\|_1}$. Hence, one has to employ a special iterative algorithm to find $\text{prox}_{\gamma \|\text{diag}(\mathbf{u}_i) A_i \cdot\|_1}$.

Here we develop an algorithm for finding $\text{prox}_{\gamma \|\text{diag}(\mathbf{u}_i)A_i \cdot \|_1}(\mathbf{h}_i)$, $i = 1, 2$, in (28). Our algorithm is based on the one proposed in [10, 18] using duality formulation. Recall that

$$\text{prox}_{\gamma \|\text{diag}(\mathbf{u}_i)A_i \cdot \|_1}(\mathbf{h}_i) = \arg \min_{\mathbf{f}_i} \left\{ \frac{1}{2} \|\mathbf{h}_i - \mathbf{f}_i\|_2^2 + \gamma \|\text{diag}(\mathbf{u}_i)A_i \mathbf{f}_i\|_1 \right\}.$$

Notice that

$$\|\text{diag}(\mathbf{u}_i)A_i \mathbf{f}_i\|_1 = \sup_{\mathbf{g} \in \mathcal{W}} \{\mathbf{g}^T \text{diag}(\mathbf{u}_i)A_i \mathbf{f}_i\}, \quad (29)$$

where $\mathcal{W} = \{\mathbf{p} : \|\mathbf{p}\|_\infty \leq 1\}$. It was shown in Example 2.17 of [18] that the proximal operator is

$$\text{prox}_{\gamma \|\text{diag}(\mathbf{u}_i)A_i \cdot \|_1}(\mathbf{h}_i) = \mathbf{h}_i - P_{\gamma A_i^T \text{diag}(\mathbf{u}_i)\mathcal{W}}(\mathbf{h}_i),$$

where $P_{\gamma A_i^T \text{diag}(\mathbf{u}_i)\mathcal{W}}(\mathbf{h}_i)$ is the unique projection of \mathbf{h}_i onto the closed convex set $\gamma A_i^T \text{diag}(\mathbf{u}_i)\mathcal{W}$. Our iteration for computing the projection is obtained by generalizing the algorithm in [10], and is given below:

$$\begin{cases} \mathbf{y}^{(m)} = \mathbf{h}_i - \gamma A_i^T \text{diag}(\mathbf{u}_i)\mathbf{p}^{(m)}, \\ \mathbf{p}^{[k]^{(m+1)}} = \frac{\mathbf{p}^{[k]^{(m)}} + \tau (\gamma \text{diag}(\mathbf{u}_i)A_i \mathbf{y}^{(m)}) [k]}{1 + \tau |(\gamma \text{diag}(\mathbf{u}_i)A_i \mathbf{y}^{(m)}) [k]|}. \end{cases} \quad (30)$$

where τ is a step parameter. We have the following theorem for the convergence of (30), which generalized Theorem 3.1 in [10] from a discrete difference operator to the matrix $A_i^T \text{diag}(\mathbf{u}_i)$. The proof can be obtained by mimicking the proof of Theorem 3.1 in [10], and we omit the details here.

Theorem 3.2. *Let $\mathbf{y}^{(m)}$ be defined as (30). Assume that $\tau \leq 1/(\gamma^2 \|\mathbf{u}_i\|_\infty^2)$ and $\mathbf{p}^{(0)} \in \mathcal{W}$. Then $\lim_{m \rightarrow \infty} \mathbf{y}^{(m)} = \text{prox}_{\gamma \|\text{diag}(\mathbf{u}_i)A_i \cdot \|_1}(\mathbf{h}_i)$.*

Putting (28) and (30) together, the complete algorithm for finding a minimizer of (15) emerges:

$$\begin{cases} \mathbf{h}_1^{(n)} = \mathbf{f}_1^{(n)} + \gamma P_\Lambda(\mathbf{g} - \mathbf{f}_1^{(n)} - \mathbf{f}_2^{(n)}), \\ \mathbf{f}_1^{(n+1)} = P_1^{M_{n,1}}(\mathbf{h}_1^{(n)}), \\ \mathbf{h}_2^{(n)} = \mathbf{f}_2^{(n)} + \gamma P_\Lambda(\mathbf{g} - \mathbf{f}_1^{(n)} - \mathbf{f}_2^{(n)}), \\ \mathbf{f}_2^{(n+1)} = P_2^{M_{n,2}}(\mathbf{h}_2^{(n)}). \end{cases} \quad (31)$$

where $P_i^{M_{n,i}}(\mathbf{h}_i^{(n)})$ is the vector $\mathbf{y}^{(M_{n,i})}$ obtained from (30) by replacing \mathbf{h}_i by $\mathbf{h}_i^{(n)}$.

Theoretically, the inner iteration number $M_{n,i}$ should be infinity so that $P_1^{M_{n,1}}(\mathbf{h}_1^{(n)})$ is equal to $\text{prox}_{\gamma \|\text{diag}(\mathbf{u}_i)A_i \cdot \|_1}(\mathbf{h}_i^{(n)})$. Then, (31) will converge to a solution of (15) with the proper choices of γ and τ given by Theorems 3.2. However, in practical computations, $M_{n,i}$ can only be finite numbers. Due to Theorem 3.2, $P_1^{M_{n,1}}(\mathbf{h}_1^{(n)})$ can be arbitrarily close to $\text{prox}_{\gamma \|\text{diag}(\mathbf{u}_i)A_i \cdot \|_1}(\mathbf{h}_i^{(n)})$ by chosen a sufficiently large finite number. We have the following convergence result for algorithm (31) for a special choice of $M_{n,i}$.

Theorem 3.3. *Assume that $0 < \gamma < 1$ and $\tau \leq 1/(\gamma^2 \|\mathbf{u}_i\|_\infty^2)$. Let $M_{n,i}$, $n \geq 0$, $i = 1, 2$, be finite integers such that*

$$\sum_{n=0}^{+\infty} \|P_i^{M_{n,i}}(\mathbf{h}_i^{(n)}) - \text{prox}_{\gamma \|\text{diag}(\mathbf{u}_i)A_i \cdot \|_1}(\mathbf{h}_i^{(n)})\|_2 \leq +\infty, \quad i = 1, 2. \quad (32)$$

Then $\mathbf{f}^{(n)} = [(\mathbf{f}_1^{(n)})^T, (\mathbf{f}_2^{(n)})^T]^T$ generated by (31) converges to a solution of (15).

Proof. Again we prove the convergence of (31) by checking Conditions 1–4 in Proposition 1. We rewrite (31) into the form

$$\begin{cases} \mathbf{f}_1^{(n+1)} = \text{prox}_{\gamma \|\text{diag}(\mathbf{u}_1)A_1\cdot\|_1}(\mathbf{f}_1^{(n)} + \gamma P_\Lambda(\mathbf{g} - \mathbf{f}_1^{(n)} - \mathbf{f}_2^{(n)})) \\ \quad + (P_1^{M_{n,1}}(\mathbf{h}_1^{(n)}) - \text{prox}_{\gamma \|\text{diag}(\mathbf{u}_1)A_1\cdot\|_1}(\mathbf{h}_1^{(n)})) \\ \mathbf{f}_2^{(n+1)} = \text{prox}_{\gamma \|\text{diag}(\mathbf{u}_2)A_2\cdot\|_1}(\mathbf{f}_2^{(n)} + \gamma P_\Lambda(\mathbf{g} - \mathbf{f}_1^{(n)} - \mathbf{f}_2^{(n)})) \\ \quad + (P_2^{M_{n,2}}(\mathbf{h}_2^{(n)}) - \text{prox}_{\gamma \|\text{diag}(\mathbf{u}_2)A_2\cdot\|_2}(\mathbf{h}_2^{(n)})). \end{cases}$$

By (32), Condition 4 is satisfied. Since all the entries of \mathbf{u}_1 and \mathbf{u}_2 are assumed to be positive, it is obvious that if $\|[\mathbf{f}_1^T, \mathbf{f}_2^T]^T\|_2 \rightarrow \infty$, then $L_1(\mathbf{f}_1, \mathbf{f}_2)$ and hence $E(\mathbf{f}_1, \mathbf{f}_2)$ both go to infinity. Clearly, both $L_1(\mathbf{f}_1, \mathbf{f}_2)$ and $L_2(\mathbf{f}_1, \mathbf{f}_2)$ are proper, convex and continuous functions. Therefore, Conditions 1 and 2 in Proposition 1 are verified. It remains to establish Condition 3, i.e., to estimate the Lipschitz constant for ∇L_2 .

Since L_2 is a quadratic function, the Lipschitz constant is the norm of the Hessian $\nabla^2 L_2$. It can be easily checked that

$$\|\nabla^2 L_2\|_2 = \left\| \begin{bmatrix} P_\Lambda & P_\Lambda \\ P_\Lambda & P_\Lambda \end{bmatrix} \right\|_2 = 2.$$

Therefore, the Lipschitz constant in (19) is $1/b = 2$. This means that Condition 3 is verified and $0 < \gamma < 1$ leads to the convergence of (31). \square

4. Numerical illustration. In this section, we use a numerical example to illustrate the applicability and usefulness of our algorithms. We will see that the tight frame based image inpainting with two tight frame systems outperform that with only one tight frame system for images having both cartoon and texture. The test image is the 512×512 “Barbara” image, which has both cartoon and texture. The original image and the observed incomplete image are shown in Figure 1. The width of the unknown region is between 5–10 pixels. As pointed out in [25, 37], it is difficult to find a pair of dictionaries so that it can separate all images in general. Here, we use two simple tight frame systems. The piecewise linear polynomial tight frame system constructed by [34] is used to represent the cartoon part and redundant local discrete cosine tight frame to represent the texture part. More precisely, to represent the cartoon part, we choose A_1 to be the tight frame generated from the framelets filters derived from piecewise linear B-spline (c.f. [6, 20]). The filters are

$$\frac{1}{4}[1, 2, 1], \quad \frac{\sqrt{2}}{4}[1, 0, -1], \quad \frac{1}{4}[-1, 2, -1]$$

The local discrete cosine transform with redundancy are chosen as A_2 to represent the texture part. All the image gray levels in our experiments are scaled into $[0, 1]$ by dividing them by 255. Throughout the whole experiments, the results of (12) with finite κ is computed by (25), and infinite κ by (28).

To ensure that the algorithms converge, according to Theorems 3.1 and 3.3, we choose $\gamma = 0.5/\max\{1, \kappa\}$ in algorithm (25), and $\gamma = 0.5$ in algorithm (31). For the inner iterations in the algorithm (30), to save the computation time, we fix the number of inner iterations in each outer iteration to be 10, i.e., $M_{n,i} = 10$ for all n and i . The step length τ in (28) is chosen to be its upper bound $1/(\gamma^2 \|\mathbf{u}\|_\infty^2)$. To accelerate the convergence of the algorithm, we use a continuation technique for the thresholding parameters \mathbf{u}_1 and \mathbf{u}_2 . First, we choose a bigger thresholding parameter \mathbf{u}_1 and \mathbf{u}_2 , and perform the algorithms (25) and (31) until



FIGURE 1. The left is the original “Barbara” image, and the right is the observed incomplete image.

they converges. Then we decrease \mathbf{u}_1 and \mathbf{u}_2 by multiplying them by a factor. The procedure is continued until the thresholding parameters \mathbf{u}_1 and \mathbf{u}_2 reach the prescribed values.

To compare the image quality quantitatively, we use peak signal to noise ratio (PSNR), which is defined as $\text{PSNR} = 20 \log_{10} \frac{N}{\|\mathbf{f}^{(n)} - \mathbf{f}\|_2}$. It is clear that larger PSNR means better image quality. The parameters in algorithms (25) and (31) are chosen such that the algorithms give out the best restored images in the sense of best PSNR.

The restored images by (12) with $\kappa = 0$, $\kappa = 1$ and $\kappa = \infty$ are shown in Figure 2. The restored image of the algorithm (2) in [6], where only piecewise linear polynomial tight frame system is used, are shown in Figure 2 also. By comparing the simultaneously cartoon and texture inpainting algorithms in this paper with the only cartoon inpaiting algorithm (2) given in [6], we see that the former ones can give much better result than the latter one. By simultaneously cartoon and texture inpainting algorithms, we gain 2-3dBs PSNR of the restored images, and much better visual qualities. (Note that the textures are recovered very well in Figure 2(e)–(g), but not in Figure 2(h).). This shows the applicability and usefulness of our proposed algorithms in various applications.

As the comparison of the simultaneously cartoon and texture inpainting (12) with different κ , we see from Figure 2 that the differences between the restored images with different κ are insignificant: the PSNR differences of Figures 2(a)–(c) are within 0.8dB, and there are no significant visual differences in the corresponding zoomed images Figure 2(e)–(g). Though $\kappa = \infty$ leads to best PSNR, the algorithm is complicated and converges slowly. In Figure 3, we plot the PSNR of the restored images and the numbers of steps of iterations versus the parameter κ in (12). From this figure, we see that, as κ increases, the PSNR of restored images also increases. The increase rate in PSNR is very small when κ is large. On the other hand, if κ is very large, the step size γ is small, hence the algorithm is slow. Therefore, to balance the computational speed and the PSNR of restored image, we shall choose a medium-valued κ .

Finally, the cartoons and the textures of the restored images in Figure 2(a)–(c) are shown in Figure 4. We see that our algorithms can separate the cartoons and textures. Comparing the images in Figure 4, we see that $\kappa = 0$ separates the cartoons and textures not as well as $\kappa = 1$ and $\kappa = \infty$ do: the cartoon part by $\kappa = 0$ is blurred. There is no significant difference between the results by $\kappa = 1$ and $\kappa = \infty$. Taking the computational speed into account, again, we recommend to use a medium-valued κ .

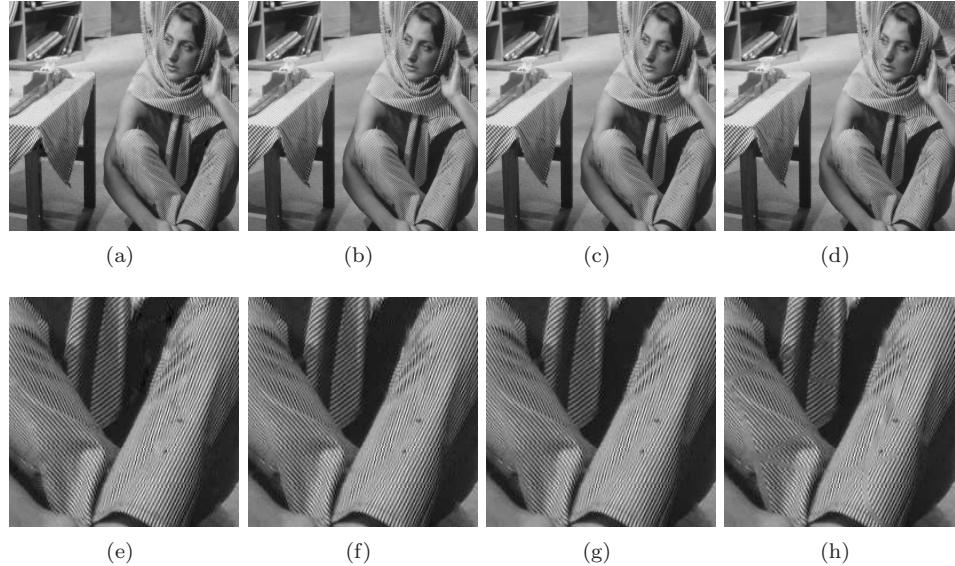


FIGURE 2. The results of tight frame based inpainting algorithms. The upper row shows the restored images by (*from left to right*) (12) with $\kappa = 0$, $\kappa = 1$, $\kappa = \infty$ and by the algorithm in [6] (i.e.(5)). The PSNR values are 33.98dB, 34.47dB, 34.75dB and 31.89dB respectively. The bottom row shows the corresponding zoomed images.

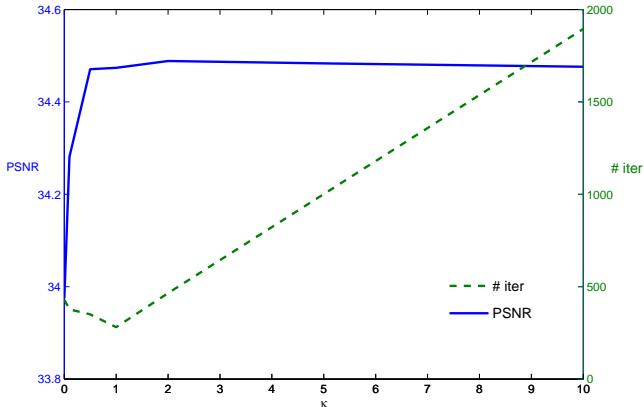


FIGURE 3. PSNR of restored image and numbers of iteration steps versus κ in (12).

It should be pointed out that when the missing region is sufficiently large, the performance is not so good. In the example shown in Figure 5, the missing regions are 48×48 square patches. The algorithm recovers some missing content such as textures, but the edges are smeared.

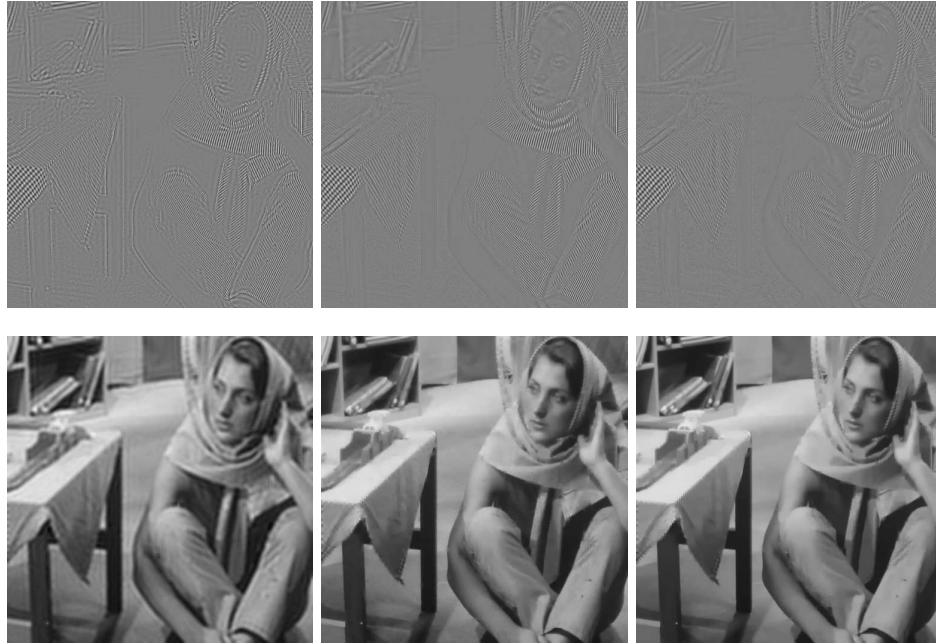


FIGURE 4. The textures (*upper row*) and the cartoons (*bottom row*) of the restored images shown in Figure 2(a), (b) and (c).



FIGURE 5. An example of image inpainting with large missing regions.

5. Conclusions. In this paper, we have considered minimization problems in image inpainting that simultaneously restore the cartoon and texture parts of the image. By using proximal forward-backward splitting, we have proposed algorithms that solve the minimization problems, and established their convergence. Numerical examples are given to illustrate the applicability and usefulness of the algorithms.

REFERENCES

- [1] J.F. AUJOL, G. GILBOA, T. CHAN, AND S. OSHER, *Structure-texture image decomposition-modeling, algorithms, and parameter selection*, International Journal of Computer Vision, 67 (2006), pp. 111–136.

- [2] M. BERTALMÍO, A. BERTOZZI, AND G. SAPIRO, *Navier stokes, fluid-dynamics and image and video inpainting*, in Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001, pp. 355–362.
- [3] M. BERTALMÍO, G. SAPIRO, V. CASELLES, AND C. BALLESTER, *Image inpainting*, in Proceedings of the 27th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co., New York, 2000, pp. 417–424.
- [4] M. BERTALMIO, L. VESE, G. SAPIRO, AND S. OSHER, *Simultaneous structure and texture image inpainting*, IEEE Transactions on Image Processing, 12 (2003), pp. 882–889.
- [5] L. BORUP, R. GRIBONVAL, AND M. NIELSEN, *Bi-framelet systems with few vanishing moments characterize Besov spaces*, Appl. Comput. Harmon. Anal., 17 (2004), pp. 3–28.
- [6] J.-F. CAI, R. H. CHAN, AND Z. SHEN, *A framelet-based image inpainting algorithm*, 2007. Appl. Comput. Harmon. Anal., 24 (2008), pp. 131–149.
- [7] J.-F. CAI, S. OSHER, AND Z. SHEN, *Split Bregman Methods and Frame Based Image Restoration* Multiscale Model. Simul., 8 (2009), pp. 337–369.
- [8] E. J. CANDÈS AND D. L. DONOHO, *Curvelets - a surprisingly effective nonadaptive representation for objects with edges*, in Curves and Surfaces, Vanderbilt University Press, Nashville, TN, 2000, pp. 105–120.
- [9] E. J. CANDÈS AND D. L. DONOHO, *New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities*, Comm. Pure Appl. Math., 57 (2004), pp. 219–266.
- [10] A. CHAMBOLLE, *An algorithm for total variation minimization and applications*, J. Math. Imaging Vision, 20 (2004), pp. 89–97. Special issue on mathematics and image analysis.
- [11] R. H. CHAN, L. SHEN, AND Z. SHEN, *A framelet-based approach for image inpainting*, 2005. Research Report 2005-04(325), Department of Mathematics, The Chinese University of Hong Kong.
- [12] T. F. CHAN, S. H. KANG, AND J. SHEN, *Euler's elastica and curvature-based inpainting*, SIAM J. Appl. Math., 63 (2002), pp. 564–592 (electronic).
- [13] T. F. CHAN AND J. SHEN, *Non-texture inpainting by curvature-driven diffusions(cdd)*, Journal of Visual Communication and Image Representation, 12 (2001), pp. 436–449.
- [14] T. F. CHAN AND J. SHEN, *Mathematical models for local nontexture inpaintings*, SIAM J. Appl. Math., 62 (2001/02), pp. 1019–1043 (electronic).
- [15] T. F. CHAN AND J. SHEN, *Image processing and analysis*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2005. Variational, PDE, wavelet, and stochastic methods.
- [16] S. S. CHEN, D. L. DONOHO, AND M. A. SAUNDERS, *Atomic decomposition by basis pursuit*, SIAM Rev., 43 (2001), pp. 129–159 (electronic). Reprinted from SIAM J. Sci. Comput. **20** (1998), no. 1, 33–61 (electronic).
- [17] R. R. COIFMAN AND D. L. DONOHO, *Translation-invariant de-noising*, in Wavelets and Statistics, A. Antoniadis and G. Oppenheim, eds., vol. 103 of Lecture Notes in Statistics, New York, 1995, Springer-Verlag, pp. 125–150.
- [18] P. L. COMBETTES AND V. R. WAJS, *Signal recovery by proximal forward-backward splitting*, Multiscale Model. Simul., 4 (2005), pp. 1168–1200 (electronic).
- [19] I. DAUBECHIES, *Ten lectures on wavelets*, vol. 61 of CBMS-NSF Regional Conference Series in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1992.
- [20] I. DAUBECHIES, B. HAN, A. RON, AND Z. SHEN, *Framelets: MRA-based constructions of wavelet frames*, Appl. Comput. Harmon. Anal., 14 (2003), pp. 1–46.
- [21] I. DAUBECHIES, AND G. TESCHKE, *Variational image restoration by means of wavelets: Simultaneous decomposition, deblurring, and denoising*, Appl. Comput. Harmon. Anal., 19 (2005), pp. 1–16.
- [22] D. L. DONOHO, *De-noising by soft-thresholding*, IEEE Trans. Inform. Theory, 41 (1995), pp. 613–627.
- [23] D. L. DONOHO, *For most large underdetermined systems of equations, the minimal l_1 -norm near-solution approximates the sparsest near-solution*, Comm. Pure Appl. Math., 59 (2006), pp. 907–934.
- [24] M. ELAD, P. MILANFAR, AND R. RUBINSTEIN, *Analysis versus synthesis in signal priors*, Inverse Problems, 23 (2007), pp. 947–968.

- [25] M. ELAD, J.-L. STARCK, P. QUERRE, AND D. L. DONOHO, *Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA)*, Appl. Comput. Harmon. Anal., 19 (2005), pp. 340–358.
- [26] M. FADILI AND J.-L. STARCK, *Sparse representations and bayesian image inpainting*, in Proc. SPARS’05, Vol. I, Rennes, France, 2005.
- [27] O. G. GULERYUZ, *Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising – part II: Adaptive algorithms*, IEEE Trans. Image Process., 15 (2006), pp. 555–571.
- [28] B. HAN AND Z. SHEN, *Dual wavelet frames and Riesz bases in Sobolev spaces*, Constructive Approximation, 29 (2009), pp. 369–406.
- [29] J.-B. HIRIART-URRUTY AND C. LEMARÉCHAL, *Convex analysis and minimization algorithms. I*, vol. 305 of Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences], Springer-Verlag, Berlin, 1993. Fundamentals.
- [30] S. MALLAT, *A wavelet tour of signal processing*, Academic Press Inc., San Diego, CA, 1998.
- [31] S. MASNOU, J.M. MOREL, *Level lines based disocclusion*, in Proceedings of 1998 International Conference on Image Processing, 1998, pp. 259–263.
- [32] Y. MEYER, *Oscillating patterns in image processing and nonlinear evolution equations*, vol. 22 of University Lecture Series, American Mathematical Society, Providence, RI, 2001. The fifteenth Dean Jacqueline B. Lewis memorial lectures.
- [33] S. OSHER, A. SOLE AND L. VESE, *Image Decomposition and Restoration Using Total Variation Minimization and the H^{-1} Norm*, Multiscale Modeling and Simulation, 1 (2003), pp. 349–370.
- [34] A. RON AND Z. SHEN, *Affine systems in $L_2(\mathbf{R}^d)$: the analysis of the analysis operator*, J. Funct. Anal., 148 (1997), pp. 408–447.
- [35] L. RUDIN, S. OSHER, AND E. FATEMI, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), pp. 259–268.
- [36] I. SELESNICK, R. VAN SLYKE, AND O. GULERYUZ, *Pixel recovery via ℓ_1 minimization in the wavelet domain*, in Proceedings of 2004 International Conference on Image Processing, IEEE, Singapore, 2004, pp. 1819–1822.
- [37] J.-L. STARCK, M. ELAD, AND D. L. DONOHO, *Image decomposition via the combination of sparse representations and a variational approach*, IEEE Trans. Image Process., 14 (2005), pp. 1570–1582.
- [38] P. TSENG, *Applications of a splitting algorithm to decomposition in convex programming and variational inequalities*, SIAM J. Control Optim., 29 (1991), pp. 119–138.
- [39] P. TSENG, *A modified forward-backward splitting method for maximal monotone mappings*, SIAM J. Control Optim., 38 (2000), pp. 431–446.

E-mail address: cai@math.ucla.edu

E-mail address: rchan@math.cuhk.edu.hk

E-mail address: matzuows@nus.edu.sg