Restoration of Chopped and Nodded Images by Wavelet Frames

Raymond H. Chan* Lixin Shen† Zuowei Shen‡

Abstract

In infrared astronomy, the observed chopped and nodded image g can be viewed as the image obtained by passing the true image f through a highpass filter. Here we propose an iterative restoration algorithm by building up a tight frame wavelet system from a multiresolution analysis that has the highpass filter as one of the wavelet filters. To recover f, the low frequency information of f hidden in g is unfolded by a wavelet decomposition and reconstruction algorithm and combined with the given high frequency information in g. The main advantage of using our method to restore chopped and nodded images is that there are fewer artifacts as compared to the well-known projected Landweber method. Also the noise in the restored image is significantly reduced. Simulated and real images are tested to illustrate the efficiency of our method.

1 Introduction

Here we give a very brief introduction to the formation of chopped and nodded images. For details, we refer the readers to the papers by Bertero et. al [3, 2]. In thermal infrared astronomy the observed signal $s(x, y)$ at the point (x, y) on the detector plane is the superposition of a weak signal $f(x, y)$ coming from the celestial source and a large varying background $\eta(x, y, t)$ coming from the atmosphere and telescope optics, i.e.,

$$s(x, y) = f(x, y) + \eta(x, y, t).$$

To extract the celestial source f, we need to eliminate the effect of the background $\eta(x, y, t)$. A common approach is to apply a fast modulation of a secondary mirror ("chopping") together with the frequent beam switching ("nodding") techniques. Under suitable assumptions (see [3, 2]), the observed image $g(x, y)$ provides the second difference of the same signal

$$g(x, y) = -f(x, y - \Delta) + 2f(x, y) - f(x, y + \Delta), \tag{1}$$

where the quantity Δ is called the *chopping throw* or *chopping amplitude*. The image g is called the *chopped and nodded image*. To restore f, we need to solve the inversion problem (1) from the chopped and nodded image g.

If the chopping amplitude Δ is K times of the sampling distance in the detector plane, where K is an integer, then we can write (1) in the following discrete form:

$$g_{j,m} = -f_{j,m-K} + 2f_{j,m} - f_{j,m+K} \tag{2}$$

*(Email: rchan@math.cuhk.edu.hk) Department of Mathematics, the Chinese University of Hong Kong, Shatin, Hong Kong, China. Research supported in part by HKRGC Grant CUHK4243/01P and CUHK DAG 2060220.

†(Email: lshen@math.wvu.edu) Department of Mathematics, Armstrong Hall, P. O. Box 6310, West Virginia University, Morgantown, WV 26506. This work was supported by grant NSF-EPSCoR-0132740.

‡(Email: matzuows@nus.edu.sg) Department of Mathematics, National University of Singapore, 2 Science Drive 2, Singapore 117543. Research supported in part by several grants at the National University of Singapore.

where $g_{j,m}$ and $f_{j,m}$ are the samples of $g(x,y)$ and $f(x,y)$ respectively. For each j, let $\mathbf{g}_j$ be the vector whose mth entry is given by $g_{j,m}$, $m = 1, 2, \ldots, N$; and $\mathbf{f}_j$ be the vector with the nth entry given by $f_{j,n}$, $n = 1, 2, \ldots, N + 2K$. Then (2) can be written in matrix form:

$$\mathbf{g}_j = A\mathbf{f}_j \tag{3}$$

where the (m,n)th entry of A is given by $A(m,n) = -\delta_{m,n} + 2\delta_{m+K,n} - \delta_{m+2K,n}$ with $m = 1, 2, \ldots, N$; $n = 1, 2, \ldots, N + 2K$ and $\delta_{m,n} = 1$ if $m \neq n$ and $\delta_{n,n} = 1$. The matrix A is called the *imaging matrix.* In [3], (3) is solved by the projected Landweber method with positivity constraints. However, the restored images exhibit artifacts (called "jumps" and "ghosts") that are related to the mathematical structure of the imaging matrix A.

In this paper, we introduce a new method to restore the chopped and nodded images. From (2), we can viewed g as the output obtained by passing f through the highpass filter

$$(-1, \underbrace{0, \ldots, 0}_{K-1}, 2, \underbrace{0, \ldots, 0}_{K-1}, -1). \tag{4}$$

An iterative restoration algorithm is proposed by considering the filter (4) as one of the tight wavelet filters. More precisely, we build up a tight frame wavelet system from a multiresolution analysis that has the highpass filter (4) as one of the wavelet filters. The chopped and nodded image g keeps most of the high frequency information of f and the low frequency information in f is folded by the highpass filter (4). Hence, to recover f, the low frequency information of f hidden in g will be unfolded and combined with the high frequency information to restore f. We will unfold the low frequency content iteratively using the wavelet decomposition and reconstruction algorithm.

There are many papers on the construction of wavelet systems from a given lowpass filter, see for instances, the references listed in [5, 9]. However, there appears to be no papers on building a wavelet system and its associated multiresolution analysis where one of the wavelet filters is a given highpass filter. The main advantage of using our method to restore chopped and nodded images is that there are fewer jumps and ghosts as compared to the projected Landweber method. Also the noise in the restored image can be significantly reduced by incorporating the wavelet thresholding denoising scheme. We will illustrate the efficiency of our method by testing it on simulated and real images.

The outline of the paper is as follows. In Section 2, we recall the projected Landweber method. In Section 3, we analyze the problem by tight frames and derive our algorithm in the L^2 space. In Section 4, we give the matrix form of the algorithm. Special implementation issues such as noise and artifacts reductions are considered in Section 5. Results on simulated and real images are presented in Section 6 and some extensions are given in Section 7.

2 The Projected Landweber Method

From the physical point of view, the brightness distribution of a celestial source must be non-negative. In [3], the *projected Landweber method* is used to find the non-negative solution of (3). It is defined as follows:

$$\mathbf{f}_j^{(n+1)} = \mathcal{P}_+ \left[\mathbf{f}_j^{(n)} + \nu A^T (\mathbf{g}_j - A\mathbf{f}_j^{(n)}) \right] \tag{5}$$

where $\mathcal{P}_+$ is the projection operator onto the set of non-negative vectors and ν is a relaxation parameter which satisfies $0 < \nu < 2/\lambda_1$ where λ_1 is the largest eigenvalue of $A^T A$. For a detail discussion and applications on the method, see [1, Chapter 6].

It was pointed out in [1, 3] that the projected Landweber method has a regularization property, known as *semiconvergence*: the iterates $\mathbf{f}_j^{(n)}$ first approach the required image, but the noise will be amplified when the number of iterations is more than a certain limit. Thus a stopping criterion, called the *discrepancy principle*, is introduced in order to obtain the best approximation of the required solution. However, the restored images always have two kinds of artifacts (see Figure 6 (b)) :

- Jumps in the brightness of the restored image at rows $\ell K + K_1$ and $(\ell+1)K$, $\ell = 0, 1, \ldots, q+1$, where q and K_1 are the quotient and remainder of N/K and N is the number of rows in the image $\mathbf{g}$ and K is the chopping amplitude.
- Multiple ghost images, spaced by K, of bright stars. They appear as dark images over a bright background or vice versa.

3 Tight Frame Analysis

From (4), we see that the chopped and nodded image can be viewed as passing the original image through a highpass filter $\sin^2(K\frac{\omega}{2})$ where K is the chopping amplitude. To simplify the discussion here, we consider only the case that K is odd. To make our main ideas work, we need to build up a tight frame wavelet system from a multiresolution analysis that has the highpass filter $\sin^2(K\frac{\omega}{2})$ as one of the wavelet filters. There are many papers devoted to the construction of wavelet systems from a given lowpass filter (see e.g. the references in [5, 9]), however, there are no papers that we are aware of on building a wavelet system and its associated multiresolution analysis with one of the wavelet filters is the given highpass filter. To this end, we make use of the unitary extension principle for tight frame constructions given in [11]. We start with the basics of tight frames and their constructions in Section 3.1 and then give our algorithm in Section 3.2.

As we have seen in (3), the recovering of the chopped and nodded images can be reduced to the restoration of one-dimensional signals (along every fixed j in (3)). We therefore consider the univariate setting only. However, it is straightforward to extend the analysis given here to the multivariate case.

3.1 Filter Design

Given a finite set $\Psi \subset L^2(\mathbb{R})$, the *wavelet system* generated by Ψ is defined as

$$X = \{\psi(2^k x - j) : \psi \in \Psi, k, j \in \mathbb{Z}\}. \tag{6}$$

A system $X \subset L^2(\mathbb{R})$ with countably many elements is a *tight frame* if for all $f \in L^2(\mathbb{R})$,

$$\|f\|^2 = \sum_{h \in X} |\langle f, h\rangle|^2.$$

To use the unitary extension principle [11], we start with a refinable function. A compactly supported function $\phi \in L^2(\mathbb{R})$ is *refinable* (a scaling function) with refinement mask (lowpass filter) τ_ϕ if it satisfies

$$\widehat{\phi}(2\cdot) = \tau_\phi \widehat{\phi},$$

where $\widehat{\phi}$ is the Fourier transform of ϕ, and τ_ϕ is a 2π trigonometric polynomial with $\tau_\phi(0) = 1$. Let $a(j)$ be the Fourier coefficients of the trigonometric polynomial τ_ϕ, then the above refinement equation can be written as:

$$\phi = 2\sum_{j \in \mathbb{Z}} a(j)\phi(2 \cdot -j).$$

Normally, the sequence $\{a(j)\}$ is called the refinement mask, but we also call the corresponding Fourier series the refinement mask.

For a given refinable function, we define V_0 to be the closure of the linear combination of ϕ and its integer translates (shifts) and $V_k := \{f(2^k\cdot) : f \in V_0, \quad k \in \mathbb{Z}\}$. Then, (i) $V_k \subset V_{k+1}$, (ii) $\cap_{k\in\mathbb{Z}} V_k = \{0\}$, and (iii) $\cup_{k\in\mathbb{Z}} V_k$ is dense in $L^2(\mathbb{R})$ (see e.g. [6], [8]). The sequence of spaces V_k, $k \in V_k$, is called a multiresolution analysis whenever it satisfies (i)–(iii).

To make the wavelets so constructed to be of any practical use, we need a smoothness assumption on the refinable function. Here, we assume that the Fourier transform of $\widehat{\phi}$ decays at $\pm\infty$ with the rate of $1/2+\delta$, $\delta > 0$, i.e.

$$|\widehat{\phi}(\omega)| \le C\left(1+|\omega|^{-(1/2+\delta)}\right)$$

for some $C < \infty$. For a given compactly supported refinable function whose Fourier transform satisfies this decay condition, the construction of tight frame wavelet systems is to find a finite set Ψ in V_1 such that the wavelet system generated by Ψ as defined in (6) forms a tight frame of $L^2(\mathbb{R})$. Recall that an arbitrary $\psi \in V_1$ can be represented in the Fourier domain as

$$\widehat{\psi}(2\cdot) = \tau_\psi \widehat{\phi},$$

for some 2π-periodic τ_ψ (see e.g. [11]). The unitary extension principle of [11] says that the wavelet system form a tight frame in $L^2(\mathbb{R})$ provided that τ_ϕ together with τ_ψ, $\psi \in \Psi$, satisfy the following conditions

$$\tau_\phi(\omega)\overline{\tau_\phi(\omega+\gamma\pi)} + \sum_{\psi\in\Psi} \tau_\psi(\omega)\overline{\tau_\psi(\omega+\gamma\pi)} = \delta_{\gamma,0}, \quad \text{for } \gamma = 0, 1 \tag{7}$$

for ω a.e. in $\mathbb{R}$. The sequences of Fourier coefficients of τ_ψ, as well as τ_ψ itself, are called wavelet filters.

In order to include $\sin^2\left(K\frac{\omega}{2}\right)$ as one of the wavelet filters, we choose

$$\tau_\phi = \tau_0 := \cos^2\left(K\frac{\omega}{2}\right)$$

as our refinement mask. The corresponding refinable function is a piecewise linear spline with $1/K$ at origin and 0 at $\pm K$, i.e.

$$\phi(x) = \frac{1}{K} - \frac{|x|}{K^2} \tag{8}$$

see Figure 1 (left). Then we define the wavelet filters

$$\tau_1(\omega) = -\sqrt{-2}\sin\left(K\frac{\omega}{2}\right)\cos\left(K\frac{\omega}{2}\right), \quad \text{and} \quad \tau_2(\omega) = \sin^2\left(K\frac{\omega}{2}\right).$$

Using the fact that K is odd, it is easy to check that the masks, τ_0, τ_1, and τ_2, defined above satisfy (7). The Fourier coefficients of the three masks are:

$$a = \left\{\frac{1}{4}, \underbrace{0,\ldots,0}_{K-1}, \frac{1}{2}, \underbrace{0,\ldots,0}_{K-1}, \frac{1}{4}\right\}, \tag{9}$$

$$b_1 = \left\{-\frac{\sqrt{2}}{4}, \underbrace{0,\ldots,0}_{2K-1}, \frac{\sqrt{2}}{4}\right\}, \tag{10}$$

$$b_2 = \left\{-\frac{1}{4}, \underbrace{0,\ldots,0}_{K-1}, \frac{1}{2}, \underbrace{0,\ldots,0}_{K-1}, -\frac{1}{4}\right\} \tag{11}$$

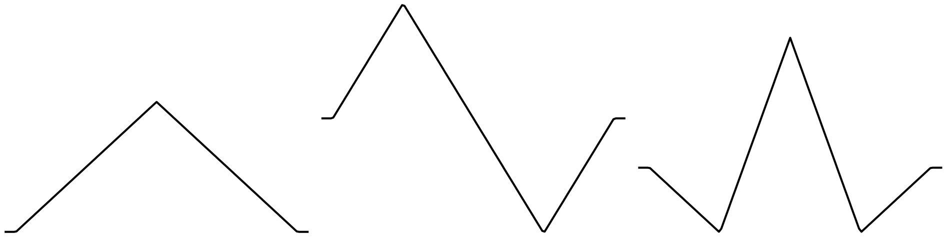


Figure 1: The scaling function (left), the first wavelet (center), and the second wavelet (right).

and the wavelet functions supported on $[-K, K]$ are:

$$\psi_1(x) = \begin{cases} \frac{\sqrt{2}}{2K} - \frac{\sqrt{2}|2x+K|}{2K^2} & \text{if } -K \le x \le 0, \\ -\frac{\sqrt{2}}{2K} + \frac{\sqrt{2}|2x+K|}{2K^2} & \text{if } 0 \le x \le K; \end{cases} \qquad \psi_2(x) = \begin{cases} -\frac{1}{K} - \frac{x}{K^2} & \text{if } -K \le x \le -\frac{K}{2}, \\ \frac{1}{K} + \frac{2x}{K^2} & \text{if } -\frac{K}{2} \le x \le 0, \\ \frac{1}{K} - \frac{2x}{K^2} & \text{if } 0 \le x \le \frac{K}{2}, \\ -\frac{1}{K} + \frac{x}{K^2} & \text{if } \frac{K}{2} \le x \le K; \end{cases} \tag{12}$$

see Figure 1 (center) and (right). Clearly b_2 matches the filter in (4).

Remark 1. *When K is even, a tight frame wavelet system can be constructed with $\sin^2(K\frac{\omega}{2})$ as one of the highpass filters by using the method suggested in [12].*

3.2 Analysis

Since V_k, $k \in \mathbb{Z}$, forms a multiresolution analysis of $L^2(\mathbb{R})$, we can assume without loss of generality that the pixel values of the original image are the coefficients of a function f in V_k for some k. From (4), we see that the chopped and nodded image can be viewed as passing the true image through the highpass filter $\sin^2(K\frac{\omega}{2}) = \tau_2(\omega)$. Thus the pixel values of the chopped and nodded images can be considered as the coefficients of a function g in

$$W_{k-1,2} := \operatorname{span}\{\psi_2(2^{k-1}(\cdot - j/2^k)) : \quad j \in \mathbb{Z}\},$$

i.e. g is represented by $\psi_2(2^{k-1} \cdot -j/2)$ with $j \in \mathbb{Z}$. The chopped and nodded image g keeps most of the high frequency information of f (the component represented by wavelet ψ_2) and the low frequency information in f is folded by the highpass filter b_2. Hence, to recover f, the low frequency information of f hidden in the chopped and nodded images will be unfolded and combined with the high frequency information to restore f. We will unfold the low frequency content iteratively using the wavelet decomposition and reconstruction algorithms. For simplicity, we give the analysis for the case $k = 1$. A similar analysis can be carried out for general k.

Let $f \in V_1$. Then,

$$f = 2\sum_{j\in\mathbb{Z}} \langle f, \phi(2 \cdot -j)\rangle \phi(2 \cdot -j) := \sqrt{2}\sum_{j\in\mathbb{Z}} v(j)\phi(2 \cdot -j). \tag{13}$$

The numbers $v(j)$, $j \in \mathbb{Z}$, are the pixel values of the original image we are seeking, and they form the discrete representation of f under the basis $\sqrt{2}\phi(2 \cdot -j)$, $j \in \mathbb{Z}$. Comparing (4) and (11), we see that the given data set $(b_2 * v)(j)$ are the pixel values of the chopped and nodded image. By using the refinability of ϕ, $b_2 * v$ is the coefficient sequence of g represented by $\psi_2(\cdot - j/2)$, $j \in \mathbb{Z}$, in $W_{0,2}$. We call this g the *chopped and nodded function* and it is given by

$$g := \sum_{j\in\mathbb{Z}} (b_2 * v)(j)\psi_2(\cdot - j/2).$$

This function can be obtained precisely once $b_2 * v$ is given.

Here we provide an iterative algorithm to recover v. At step $(n+1)$ of the algorithm, it improves the low frequency components of f by updating the low-frequency components of the previous step. The algorithm is presented in the Fourier domain where the problem becomes: given $\widehat{b_2 * v} = \tau_2 \widehat{v}$, find $\widehat{v}$. The algorithm makes use of the fact that $\overline{\tau_0}\tau_0 + \overline{\tau_1}\tau_1 + \overline{\tau_2}\tau_2 = 1$, see (7).

Algorithm 1.

(i) Choose $\widehat{v}_0 \in L^2[-\pi, \pi]$;

(ii) Iterate on n until convergence:

$$\widehat{v}_{n+1} = \overline{\tau_0}\tau_0 \widehat{v}_n + \overline{\tau_1}\tau_1 \widehat{v}_n + \overline{\tau_2}\widehat{b_2 * v}. \tag{14}$$

We remark that the first term in the right hand side of (14) represents the approximation of the low-frequency components of f whereas the second term improves the high frequency approximation.

Given $\widehat{v}_n$, f_n is defined via its Fourier transform as:

$$\widehat{f_n}(\cdot) := \widehat{v}_n(\cdot/2)\widehat{\phi}(\cdot/2) \in V_1(\phi). \tag{15}$$

Strictly speaking, $\widehat{f_n}$ and $\widehat{\phi}$ are the Fourier transform of f and ϕ respectively, whereas $\widehat{v}_n$ is the symbol of the sequence v_n. But here to simplify the notations, we will use the same "hat" notation to denote both. In the absence of noise, we can show that the functions f_n converge to the true function f in (13).

Proposition 1. *Let ϕ be the refinable function given in* (8) *with the refinement mask a given in (9), and let ψ_1, ψ_2 be the two wavelets defined in* (12) *with masks b_1 and b_2 given in (10) and (11). Then, the sequence $\widehat{v}_n$ defined in (14) converges to $\widehat{v}$ in the L^2-norm for any arbitrary $\widehat{v}_0 \in L^2[-\pi, \pi]$. In particular, f_n in (15) converges to f in (13) in the L^2-norm.*

The proof can be modified readily from the proof of Proposition 3.1 given in [4] where, instead of a highpass filter, a lowpass filter is given. In the presence of noise, we will add to Algorithm 1 a wavelet thresholding denoising scheme like we did in [4]. This will be done in Section 5.

Remark 2. *Algorithm 1 and Proposition 1 are closely related to the corresponding algorithm and proposition in [4] except that we use a tight frame wavelet system here instead of a biorthogonal wavelet system as in [4]. In fact, if $K > 1$, there is no available method to construct biorthogonal or orthogonal wavelet systems having $\sin^2(K\frac{\omega}{2})$ as one of the wavelet masks. However, thanks to the unitary extension principle [11], we can construct a tight frame wavelet system. As we will see later, tight frame wavelet systems are more nature here, since they provide a new interpretation of existing numerical methods such as the Landweber method. This permits us to build into our method powerful denoising and artifact removing schemes based on wavelet theory. Furthermore, the redundant nature of tight frame systems allows us to extract more information contents from the available redundant information. In our forthcoming work, the wavelet frames idea will be extended to high resolution image reconstruction applications where we are given a lowpass filter. The tight frame wavelet system built up on the given lowpass filter via the unitary extension principle of [11] can provide a nature approach to the problem and better numerical results.*

4 Matrix Form of Algorithm 1

In this section, we express Algorithm 1 in the time domain, i.e. in a matrix form ready for numerical computations. To derive it, we need to consider the boundary conditions, i.e. assumptions of the true image outside the region of interest. For simplicity, we use symmetric (reflective) extension here, see [10]. For other extensions, the derivation is similar. In fact, for the infrared images that we are interested here, the difference between symmetric and periodic boundary extensions is small.

We first define the following matrices. For a given sequence $h = (h(j))_{j=-J}^{J}$, we let $T(h)$ to be the $M \times M$ (with $M > 2J + 1$) Toeplitz matrix

$$T(h) := \begin{bmatrix} h(0) & \cdots & h(-J) & & 0 \\ \vdots & \ddots & \ddots & h(-J) & \\ h(J) & \ddots & \ddots & \ddots & h(-J) \\ & \ddots & \ddots & \ddots & \vdots \\ 0 & & h(J) & \cdots & h(0) \end{bmatrix}.$$

Define

$$T_\ell(h) := \left[\begin{array}{c|ccccc|c} & h(J) & h(J-1) & \cdots & h(2) & h(1) & \\ & & \ddots & \ddots & & h(2) & \\ 0_{J\times(M-J-1)} & & & \ddots & \ddots & \vdots & 0_{J\times 1} \\ & & & & \ddots & h(J-1) & \\ & & & & & h(J) & \\ \hline 0_{(M-J)\times(M-J-1)} & \multicolumn{5}{c|}{0_{(M-J)\times J}} & 0_{(M-J)\times 1} \end{array}\right] R,$$

and

$$T_r(h) = \left[\begin{array}{c|ccccc|c} 0_{(M-J)\times 1} & \multicolumn{5}{c|}{0_{(M-J)\times J}} & 0_{(M-J)\times(M-J-1)} \\ \hline & h(-J) & & & & & \\ & h(-J+1) & \ddots & & & & \\ 0_{J\times 1} & \vdots & \ddots & \ddots & & & 0_{J\times(M-J-1)} \\ & h(-2) & & \ddots & \ddots & & \\ & h(-1) & h(-2) & \cdots & h(-J+1) & h(-J) & \end{array}\right] R,$$

where R is the $M \times M$ reversal (anti-identity) matrix, i.e., it has 1's on its anti-diagonal and 0's elsewhere. They correspond to the symmetric boundary conditions on the left and on the right respectively. Finally let $S_+(h)$ and $S_-(h)$ be:

$$S_+(h) = T_\ell(h) + T(h) + T_r(h),$$

and

$$S_-(h) = -T_\ell(h) + T(h) - T_r(h).$$

For the masks a, b_1 and b_2 given in (9)–(11), their corresponding decomposition (and respectively reconstruction) matrices are:

$$H_0 = S_+(a), \quad H_1 = S_+(b_1), \quad H_2 = S_+(b_2) \tag{16}$$

(and

$$\widetilde{H}_0 = S_+(a), \quad \widetilde{H}_1 = S_-(-b_1), \quad \widetilde{H}_2 = S_+(b_2). \tag{17}$$

respectively) with $J = K$ and $M = N + 2K$. We note that since b_1 is antisymmetric, $H_1 \neq \widetilde{H}_1$. However, if periodical boundary conditions are used, they will be the same.

Using (7), it is easy to verify that

$$\widetilde{H}_0 H_0 + \widetilde{H}_1 H_1 + \widetilde{H}_2 H_2 = I, \tag{18}$$

where I is the identity. Notice that H_2 and the imaging matrix A in (3) are related by:

$$H_2 = \frac{1}{4}\left[\begin{array}{c} * \\ \hline A \\ \hline * \end{array}\right],$$

where $*$ here denotes non-zero matrices of size $K \times (N + 2K)$ (cf. (4) and (11)).

According to (14), the term $H_2\mathbf{f}^{(n)}$, which corresponds to $\tau_2\hat{v}_n$, should be replaced by $\mathbf{g}$ which corresponds to $\widehat{b_2 * v}$. However, the vector $\mathbf{g}$, which is of length N, is shorter than the vector $H_2\mathbf{f}^{(n)}$ which is of length $N + 2K$. Hence, we can only replace the central part of $H_2\mathbf{f}^{(n)}$ by $\mathbf{g}$, and the remaining part are to be determined by the boundary conditions that we have chosen. To achieve this, we write

$$\widetilde{H}_2 H_2 \mathbf{f}^{(n)} = \widetilde{H}_2 \Lambda H_2 \mathbf{f}^{(n)} + \frac{1}{16} A^T A \mathbf{f}^{(n)} \tag{19}$$

where

$$\Lambda = \mathrm{diag}(\underbrace{1, \cdots, 1}_{K}, \underbrace{0, \cdots, 0}_{N}, \underbrace{1, \cdots, 1}_{K}).$$

Replacing $A\mathbf{f}^{(n)}$ by $\mathbf{g}$ (i.e. $\widetilde{H}_2 H_2 \mathbf{f}^{(n)}$ by $\widetilde{H}_2 \Lambda H_2 \mathbf{f}^{(n)} + \frac{1}{16}A^T\mathbf{g}$), and using (18), Step (ii) of Algorithm 1 becomes

$$\mathbf{f}^{(n+1)} = \widetilde{H}_0 H_0 \mathbf{f}^{(n)} + \widetilde{H}_1 H_1 \mathbf{f}^{(n)} + (\widetilde{H}_2 \Lambda H_2 \mathbf{f}^{(n)} + \frac{1}{16} A^T \mathbf{g}).$$

Hence we have the following matrix version of Algorithm 1:

Algorithm 2 (Matrix form of Algorithm 1).

(i) Set an initial approximation $\mathbf{f}^{(0)}$;

(ii) Iterate on n *until convergence:*

$$\mathbf{f}^{(n+1)} = \widetilde{H}_0 H_0 \mathbf{f}^{(n)} + \widetilde{H}_1 H_1 \mathbf{f}^{(n)} + (\widetilde{H}_2 \Lambda H_2 \mathbf{f}^{(n)} + \frac{1}{16} A^T \mathbf{g}).$$

By (18), $\widetilde{H}_0 H_0 \mathbf{f}^{(n)} + \widetilde{H}_1 H_1 \mathbf{f}^{(n)} = \mathbf{f}^{(n)} - \widetilde{H}_2 H_2 \mathbf{f}^{(n)}$. Hence by (19), Step (ii) is reduced to

$$\mathbf{f}^{(n+1)} = \mathbf{f}^{(n)} + \frac{1}{16}(A^T\mathbf{g} - A^T A \mathbf{f}^{(n)}).$$

Comparing it with (5), Algorithm 2 is just the Landweber method with $\nu = 1/16$, and can be considered as a reformulation of the method based on the tight frame wavelet theory.

This new formulation puts the whole restoration process into the multiresolution analysis framework and gives analytical representations of the chopped and nodded image and the true image in terms of the tight frame wavelet systems. The analysis tells us that the algorithms (ours as well as the Landweber method with $\nu = 1/16$) improves the low resolution (coarse level) approximation of the true image, which contains most of the low frequency information folded by the chop and nod process, by combining the corresponding part of the previous iterate with the given chopped and nodded image. This new wavelet viewpoint allows us to incorporate wavelet thresholding denoising schemes into our algorithm. This will be done in the next section.

5 Noise and Artifacts Removal

Signals contains noise. A successful algorithm has to be able to reduce noise. So far Algorithm 2 only restores the low frequency information of the image. Here in the first two subsections, we add to Algorithm 2 two extra steps to reduce the noise and also the artifacts (jumps and ghosts). Both steps are based on Dohono's thresholding scheme which is easy to implement and is non-linear in nature. In contrast, the Landweber method is a linear process for denoising. For denoising problems, the Donoho's scheme has been shown to be more efficient than linear methods in general (see e.g. [7, 9]). Our complete algorithm will be given in the last subsection.

5.1 Noise Reduction

In Algorithm 2, we have chosen the tight frame wavelet system $\{H_i, \widetilde{H}_i\}_{i=0}^2$ mainly because we have to include the high pass filter $\sin^2(K\frac{\omega}{2})$ from the chop and nod process as one of our wavelet masks. However, for noise and artifacts removal, we are free to choose any tight frame wavelet systems. To get a more efficient algorithm, we choose the spline tight frame wavelet system given in [11]. Its masks are: $\alpha = \{\frac{1}{4}, \frac{1}{2}, \frac{1}{4}\}$, $\beta_1 = \{-\frac{\sqrt{2}}{4}, 0, \frac{\sqrt{2}}{4}\}$, and $\beta_2 = \{-\frac{1}{4}, \frac{1}{2}, -\frac{1}{4}\}$. We note that this is the tight frame wavelet system obtained in Section 3 if we set $K = 1$. Hence it has the same shape as that given in Figure 1. The corresponding decomposition matrices and reconstruction matrices are given by $G_0 = S_+(\alpha)$, $G_1 = S_+(\beta_1)$, $G_2 = S_+(\beta_2)$, and $\widetilde{G}_0 = S_+(\alpha)$, $\widetilde{G}_1 = S_-(-\beta_1)$, $\widetilde{G}_2 = S_+(\beta_2)$.

Let $\mathbf{f}$ be the current iterate. The wavelet decomposition of $\mathbf{f}$ gives three sets of data: $G_0\mathbf{f}$, $G_1\mathbf{f}$ and $G_2\mathbf{f}$, where $G_0\mathbf{f}$ gives the coefficients of the coarser level approximation of $\mathbf{f}$ and $G_1\mathbf{f}$ and $G_2\mathbf{f}$ give the wavelet coefficients. By the perfect reconstruction formula (cf (18)), we have

$$\mathbf{f} = \widetilde{G}_0 G_0 \mathbf{f} + \widetilde{G}_1 G_1 \mathbf{f} + \widetilde{G}_2 G_2 \mathbf{f}.$$

One can further decompose $\mathbf{f}$ down to any level by applying the decomposition procedure to $G_0\mathbf{f}$, $G_0^2\mathbf{f}$, $G_0^3\mathbf{f}, \ldots$, whenever it is necessary to analyze $\mathbf{f}$. We further remark that since the down sample operator is not applied in the decomposition, it is a dyadic translation invariant wavelet decomposition and reconstruction algorithm.

The key step for denoising is to apply Donoho's thresholding scheme [7] to the wavelet coefficients of $\mathbf{f}$ at each level. The denoised $\mathbf{f}$ is given by:

$$\mathcal{D}(\mathbf{f}) = (\widetilde{G}_0)^L (G_0)^L \mathbf{f} + \sum_{\ell=0}^{L-1} (\widetilde{G}_0)^\ell \left[\widetilde{G}_1 \mathcal{T}_\lambda (G_1 G_0^\ell \mathbf{f}) + \widetilde{G}_2 \mathcal{T}_\lambda (G_2 G_0^\ell \mathbf{f}) \right]. \tag{20}$$

where

$$\mathcal{T}_\lambda((x_1, \ldots, x_l, \ldots)^T) = (t_\lambda(x_1), \ldots, t_\lambda(x_l), \ldots)^T,$$

with $t_\lambda(x) = x$ if $|x| > \lambda$ and 0 otherwise. Here L is the number of levels used in the decomposition and the thresholding parameter λ is chosen to be

$$\lambda = \kappa\sqrt{2\log(N+2K)} \tag{21}$$

where κ is the variance of $\mathbf{f}$ estimated numerically by the method given in [7].

5.2 Artifacts Reduction

The denoising scheme in Section 5.1 reduces the noise in each iteration. However, we still need to reduce the artifacts (i.e. the ghosts and the jumps) created by the chop and nod process. Fortunately, the astronomical infrared images that we are considering have peaks with high intensity while most of the artifacts are of low intensity. Furthermore, they are isolated. Since our main interest is to keep the intensity of the major peaks which are of high intensity and are isolated, the wavelet decomposition of the computed $\mathbf{f}$ at major peak areas will have large wavelet coefficients. On the other hand, most of the artifacts are of low intensity occuring in areas with average level of intensity, the wavelet coefficients of $\mathbf{f}$ in these area will be of average magnitude. Although the areas where small features occur also have wavelet coefficients with average magnitude, the average intensity of these areas is smaller than those near the artifacts. We will make use of these observations to distinguish between stars, features and artifacts.

The main idea is to threshold all wavelet coefficients except those that are from areas near the major stars or features. To identify those coefficients automatically, we first compute the L-level wavelet decomposition of $\mathbf{f}$ using the masks given in Section 5.1 (see Step 2 below). Then, we identify the positions of those wavelet coefficients that are contributed by the major stars. This is done by finding those positions where the intensity values of $\mathbf{f}$ are close to the maximum of $\mathbf{f}$. Theoretically, those positions should be mapped to the positions of corresponding wavelet coefficients. Since the dyadic translation invariant wavelet transform is used here, and that both the wavelet and the refinement masks (the high and the low pass filters) used here are of very small support, the positions of the wavelet coefficients contributed by the high intensity values of $\mathbf{f}$ can be identified by the positions where $\mathbf{f}$ has intensity values close to the maximum of $\mathbf{f}$. This approximation simplifies our scheme (see Step 3 below). Step 4 below is to find small features which occur at points where both intensity values and wavelet coefficients are of average magnitude. Then we identify the areas that do not contain large intensity values of $\mathbf{f}$, large wavelet coefficients, or small features (Step 5) and apply a simple thresholding scheme to reduce artifacts in these areas (Steps 6–7).

Artifacts Reduction:

1. Fix the parameters: $0 < s_d < s_u < s < 1$, and r and L, two positive integers. Let $\mathbf{f}$ be the current iterate with entries $\mathbf{f}(i)$, $1 \le i \le N+2K$.

2. Compute the L-level wavelet decomposition of $\mathbf{f}$ as in Section 5.1. This leads to one scaling coefficient $(G_0)^L\mathbf{f}$ and $2L$ wavelet coefficients $(G_p)(G_0)^\ell\mathbf{f}$ where $0 \le \ell < L$, $p = 1, 2$.

3. Define a binary position vector U_m where its ith entry $U_m(i)$ is given by

$$U_m(i) = \begin{cases} 1 & \text{if } \mathbf{f}(i) \ge s \cdot \{\max\limits_{1 \le j \le N+2K} \mathbf{f}(j)\}, \\ 0 & \text{otherwise.} \end{cases}$$

4. Let $F_p = \max_{1 \le j \le N+2K} |[G_p\mathbf{f}](j)|$ for $p = 1, 2$. Let $B_r(i) = \{j : |j-i| \le r\}$. Define $V(i) = \frac{1}{2}\left\{\min_{j \in B_r(i)} \mathbf{f}(j) + \max_{j \in B_r(i)} \mathbf{f}(j)\right\}$, and $\bar{V}(i)$ the average of $\mathbf{f}(j)$ for $j \in B_r(i)$. Define a

binary position vector U_w where its ith entry $U_w(i)$ is 1 if (i) $s_d F_p \le |[G_p \mathbf{f}](i)| \le s_u F_p$ for $p = 1$ or 2, and (ii) $s_d V(i) \le \bar{V}(i) \le s_u V(i)$. Set $U_w(i) = 0$ otherwise.

5. Generate a binary position vector U from U_m and U_w: $U(j) = 1$ for all j in $B_r(i)$ if either $U_m(i) = 1$ or $U_w(i) = 1$.

6. For all i such that $U(i) = 0$, we threshold the wavelet coefficients by replacing $[G_p(G_0)^\ell \mathbf{f}](i)$ by

$$\operatorname{sgn}([G_p(G_0)^\ell \mathbf{f}](i)) \max(|[G_p(G_0)^\ell \mathbf{f}](i)| - \lambda, 0) \quad \text{for } 0 \le \ell < L, p = 1, 2,$$

where λ is given in (21). The wavelet coefficients are unchanged if $U(i) = 1$.

7. Using the reconstruction formula, reconstruct the image from the modified wavelet coefficients together with the coarse level scaling coefficient $(G_0)^L \mathbf{f}$.

5.3 The Complete Algorithm

For 1D signals, we have the following complete algorithm.

Algorithm 3.

(i) Set an initial approximation $\mathbf{f}^{(0)}$*;*

(ii) Iterate on n until convergence:

$$\mathbf{f}^{(n+1)} = \mathcal{P}_+ \left[\widetilde{H}_0 \mathcal{D}(H_0 \mathbf{f}^{(n)}) + \widetilde{H}_1 \mathcal{D}(H_1 \mathbf{f}^{(n)}) + (\widetilde{H}_2 \Lambda H_2 \mathbf{f}^{(n)} + \frac{1}{16} A^T \mathbf{g}), \right].$$

where $\mathcal{D}$ *is given in (20) and* $\mathcal{P}_+$ *is the projection operator onto the set of nonnegative vectors.*

(iii) Perform the artifacts reduction steps as given in Section 5.2.

We note here that while denoising scheme is built in each iteration, the artifacts reducing scheme is only implemented at the last iteration. This is because without the denoising scheme, the random noise blows up with the iterations, especially those noise whose spectrum is close to the singularity of $\sin^2\left(K\frac{\omega}{2}\right)$, the convolution kernel of the chop and nod procedure. On the other hand, the artifacts generated by the chop and nod procedure are similar to the small features of the image that they usually do not blow up with the iterations. Hence, we do not need the artifact removal scheme at each iteration.

For computational complexity, Algorithm 3 needs $O(L(N + 2K))$ operations per iteration, since for each level, the cost of one wavelet transform is linear with respect to the number of unknowns. In contrast, the projected Landweber method requires $O(N + 2K)$ operations per iteration.

For 2D images, we first recall that the chop and nod process is a 1D process. In fact, in (3), we restore the image columnwise for each fixed j. Thus there is no need to change the restoration part of Algorithm 3. More precisely, we can still use the 1D tight frame wavelet system $\{H_i, \tilde{H}_i\}_{i=0}^2$ for 2D images. However, to better capture the noise and artifacts in between the columns, we use 2D noise and artifacts reduction schemes. Since both are wavelet schemes, the extension from 1D to 2D is easy by using tensor products.

For example, for 2D image $\mathbf{f}$ expressed as a matrix, the wavelet decomposition of $\mathbf{f}$ gives nine sets of data: $\{G_p\mathbf{f}G_q^T\}_{p,q=0}^2$ where G_i are given in Section 5.1. The denoising scheme in 2D is

$$\begin{aligned}\mathcal{D}(\mathbf{f}) &= (\widetilde{G}_0)^L(G_0)^L\mathbf{f}(G_0^T)^L(\widetilde{G}_0^T)^L \\ &\quad +\sum_{\ell=0}^{L-1}\sum_{\substack{0\le p,q\le 2\\(p,q)\neq(0,0)}}(\widetilde{G}_0)^\ell\widetilde{G}_p\mathcal{T}_\lambda\left(G_p(G_0)^\ell\mathbf{f}(G_0^T)^\ell G_q^T\right)\widetilde{G}_q^T(\widetilde{G}_0^T)^\ell. \end{aligned} \tag{22}$$

Similarly for the 2D artifacts reduction schemes, we can simply replace the 1-D wavelets by 2D wavelets. For example, in Step 4 of Section 5.2, F_p will be replaced by

$$F_{p,q}=\max_{j_1,j_2}|[G_p\mathbf{f}_nG_q^T](j_1,j_2)|,\quad 0\le p,q\le 2,(p,q)\neq(0,0),$$

and $B_r(i_1,i_2)$ will be a square of width $(2r+1)$ centered at pixel (i_1,i_2).

6 Numerical Results

In this section, we test Algorithm 3 against the projected Landweber method for one- and two-dimensional examples.

We first give three 1D examples, the first two of which are given in [2]. In all three examples, $N=128$ and $K=37$. The true object $\mathbf{f}^*$ has 202 points and the observation region (i.e. the support of $\mathbf{g}$) lies between point 38 and point 165. For parameters, we set $L=4$, $s=0.8$, $s_u=0.4$, $s_d=0.1$, $r=5$. The initial guess is the zero vector. White Gaussian noise with standard deviation 0.02 is added to all examples. We stop the iteration when

$$\varrho_n:=\frac{\|\mathbf{f}^{(n+1)}-\mathbf{f}^{(n)}\|_2}{\|\mathbf{f}^{(n)}\|_2}<0.0002. \tag{23}$$

Example 1. The true object consists of two narrow Gaussian functions (simulating two bright stars over a black background), one inside and one outside the observation region. The results are in Figure 2, with the numbers of iterations required for convergence listed in the caption.
Example 2. The true object consists of one narrow Gaussian function (simulating a bright star) over a smooth background. The results are in Figure 3.
Example 3. The true object consists of two narrow Gaussian functions (simulating two bright stars with different intensities) over a smooth background. The results are in Figure 4.
In all examples, we can see the improvement of our method over the projected Landweber method.

We remark that in [2], the relative discrepancy error, defined by

$$\varepsilon_n:=\frac{\|A\mathbf{f}^{(n)}-\mathbf{g}\|_2}{\|\mathbf{g}\|_2}, \tag{24}$$

is used as the stopping criterion. Here we use (23) instead of (24) because we find that ε_n is not a good measure of the relative restoration error which is defined as:

$$\varsigma_n:=\frac{\|\left[\mathbf{f}^{(n)}+\mathrm{mean}(\mathbf{f}^*-\mathbf{f}^{(n)})\right]-\mathbf{f}^*\|_2}{\|\mathbf{f}^*\|_2},$$

see [2]. We illustrate this in Figure 5 where we measure ε_n and ς_n against n. We see that smaller ε_n does not imply necessarily smaller ς_n, and that as far as the relative restoration error ς_n is concerned, Algorithm 3 is more accurate than the Projected Landweber method.

In fact, ε_n measures the relative error between $A\mathbf{f}^{(n)}$ and $\mathbf{g}$. We note that $A\mathbf{f}^{(n)}$ are the wavelet coefficients of $\mathbf{f}^{(n)}$ with respect to the wavelet ψ_2 of (12) and $\mathbf{g}$ are the corresponding wavelet coefficients of $\mathbf{f}^*$ corrupted by noise. Since $\mathbf{f}^*$ is corrupted with noise, only few outliers of its wavelet coefficients are contributed by the image $\mathbf{f}^*$, and most of small wavelet coefficients are contributed by the noise. Thus if $A\mathbf{f}^{(n)}$ is too close to $\mathbf{g}$, it may indicate that the approximation $\mathbf{f}^{(n)}$ follows too close to the noise and this is not desirable.

Finally we consider a real 2D image obtained from United Kingdom Infra-Red Telescope [2]. For parameters, we have chosen $L = 2$, $s = 0.8$, $s_u = 0.4$, $s_d = 0.1$, $r = 3$. We stop the iteration when $\varrho_n < 0.001$. The results are given in Figure 6. It is clear from the figures that our results have much less noise and the artifacts are greatly reduced.

7 Extension and Conclusion

From (14), we see that if $\tau_1\widehat{v} = \widehat{b_1 * v}$ is available, then we can improve Algorithm 1 by replacing (14) by

$$\widehat{v}_{n+1} = \overline{\tau_0}\tau_0\widehat{v}_n + \overline{\tau_1}\widehat{b_1 * v} + \overline{\tau_2}\widehat{b_2 * v}.$$

In matrix terms, $\widehat{b_1 * v}$ is the image $\mathbf{b}$ with entries

$$\mathbf{b}(j,m) = -\frac{1}{2}f_{j,m-K} + \frac{1}{2}f_{j,m+K}$$

(cf. (2)). It can be obtained from the true image by doing a chop and chop process. We test the improvement of our Algorithm 3 for the three 1-D test examples in Section 6. The results are shown in Figure 7, and we can easily see the improvement.

In this paper, we present a method for denoising chopped and nodded images obtained from infrared astronomy. Our wavelet approach allows us not only to denoise the image, but in the same time, protect the intensity level at regions crucial to the observers (like stars), and remove artifacts that are intrinsic to the mathematical model.

As remarked in Remark 2 in Section 3.2, our idea here can be extended to high resolution image reconstruction applications where we are given a lowpass filter. The tight frame wavelet system built up on the given lowpass filter via the unitary extension principle of [11] can provide a nature approach to the problem and better numerical results.

Acknowledgment: We thank Prof. Bertero for introducing us to the topic and his many advices and helps in our research.

References

[1] M. Bertero and P. Boccacci. *Introduction to inverse problems in imaging.* Institute of Physics Pub., Bristol, 1998.

[2] M. Bertero, P. Boccacci, F. Benedetto, and M. Robberto. Restoration of chopped and nodded images in infrared astronomy. *Inverse Problem*, 15:345–372, 1999.

[3] M. Bertero, P. Boccacci, and M. Robberto. An inversion method for the restoration of chopped and nodded images. *Infrared Astronomical Instrumentation, Proc.. SPIE*, ed. A. Fowler, 3354:877–886, 1998.

[4] R. Chan, T. Chan, L. Shen, and Z. Shen. Wavelet algorithms for high-resolution image reconstruction. *SIAM Journal on Scientific Computing*, 24(4):1408–1432, 2003.

[5] I. Daubechies. Ten lectures on wavelets, *CBMS Conference Series in Applied Mathematics*, V. 61. SIAM, Philadelphia, 1992.

[6] C. de Boor, R. DeVore, and A. Ron. On the construction of multivariate (pre)-wavelets. *Constr. Approx.*, 9:123–166, 1993.

[7] D. Donoho and I. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81:425–455, 1994.

[8] R.Q. Jia and Z. Shen. Multiresolution and wavelets. *Proceedings of the Edinburgh Mathematical Society*, 37: 271-300, 1994.

[9] S. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 2nd edition.

[10] M. Ng, R. Chan, and W. Tang. A fast algorithm for deblurring models with Neumann boundary conditions. *SIAM Journal on Scientific Computing*, 21:851–866, 2000.

[11] A. Ron and Z. Shen. Affine system in $L_2(\mathbb{R}^d)$: the analysis of the analysis operator. *Journal Func. Anal.*, 148:408–447, 1997.

[12] A. Ron and Z. Shen. Compactly supported tight affine spline frames in $L_2(\mathbb{R}^d)$. *Mathematics of Computation*, 67:191–207, 1998.

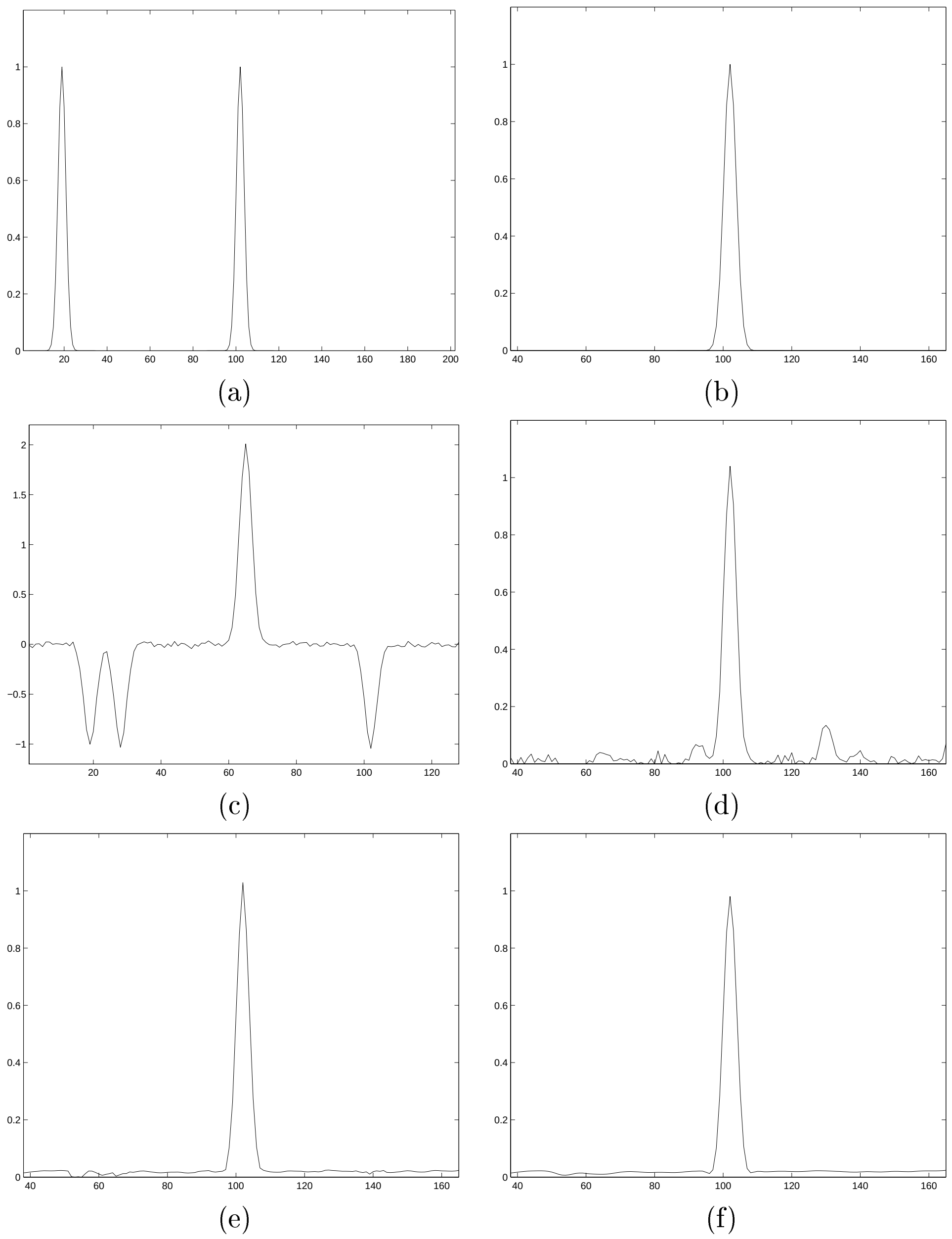


Figure 2: Example 1. (a) Original signal; (b) observed region; (c) chopped and nodded **g** with noise; (d) result by the projected Landweber method (84 iterations); (e) result by Algorithm 3 without artifacts reduction (264 iterations); (f) result by Algorithm 3 with artifacts reduction.

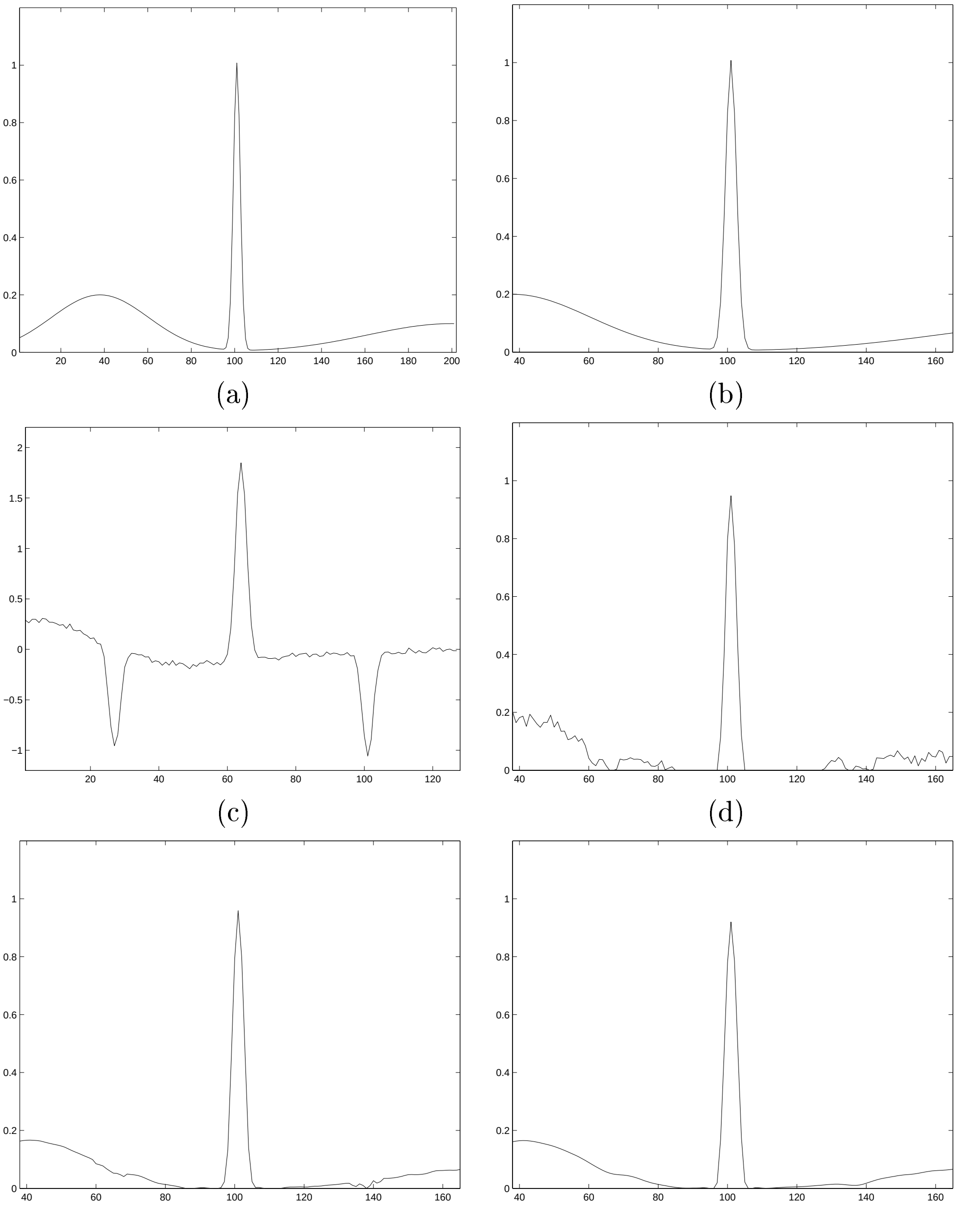


Figure 3: Example 2. (a) Original signal; (b) observed region; (c) chopped and nodded **g** with noise; (d) result by the projected Landweber method (152 iterations); (e) result by Algorithm 3 without artifacts reduction (280 iterations); (f) result by Algorithm 3 with artifacts reduction.

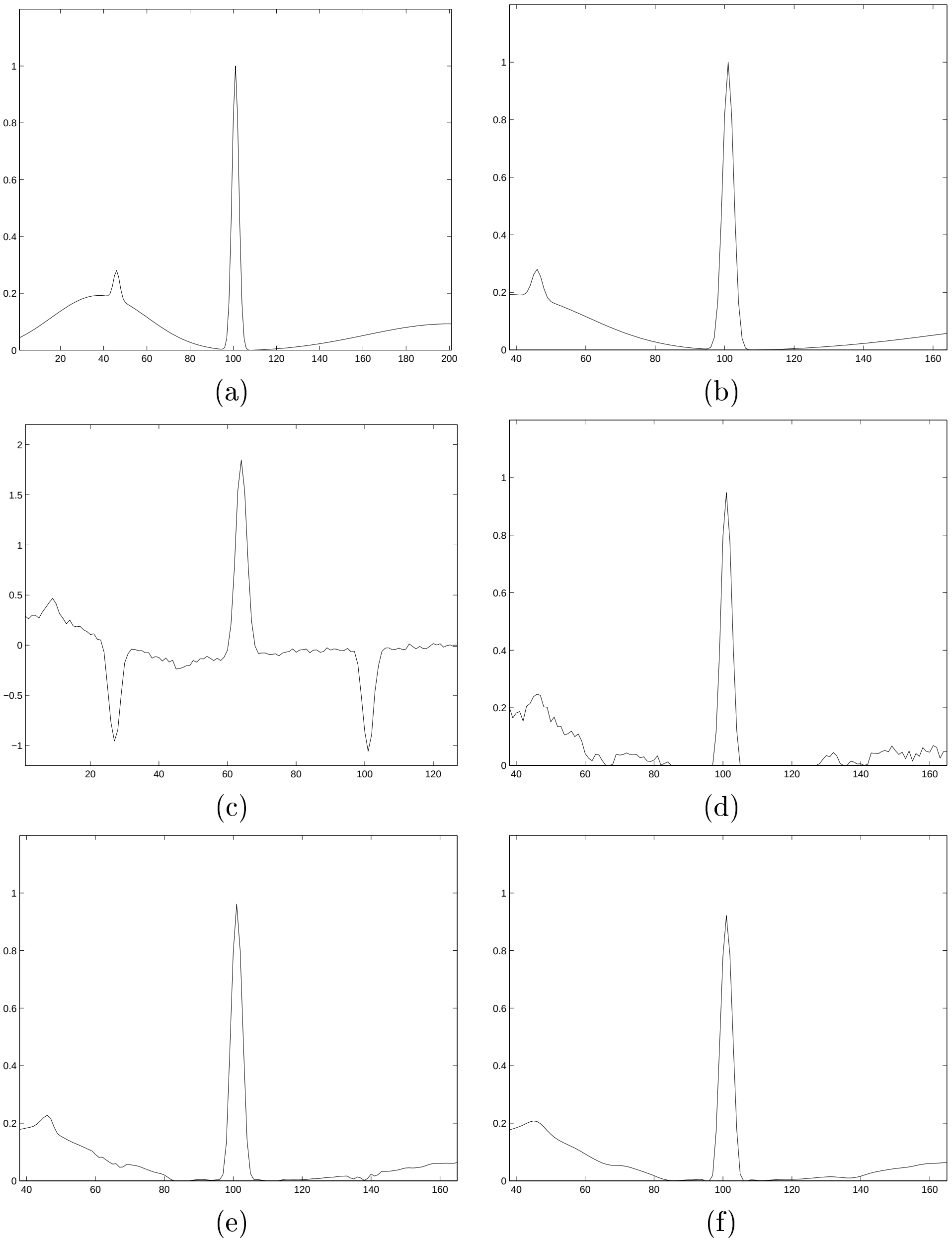


Figure 4: Example 3. (a) Original signal; (b) observed region; (c) chopped and nodded **g** with noise; (d) result by the projected Landweber method (151 iterations); (e) result by Algorithm 3 without artifacts reduction (335 iterations); (f) result by Algorithm 3 with artifacts reduction.

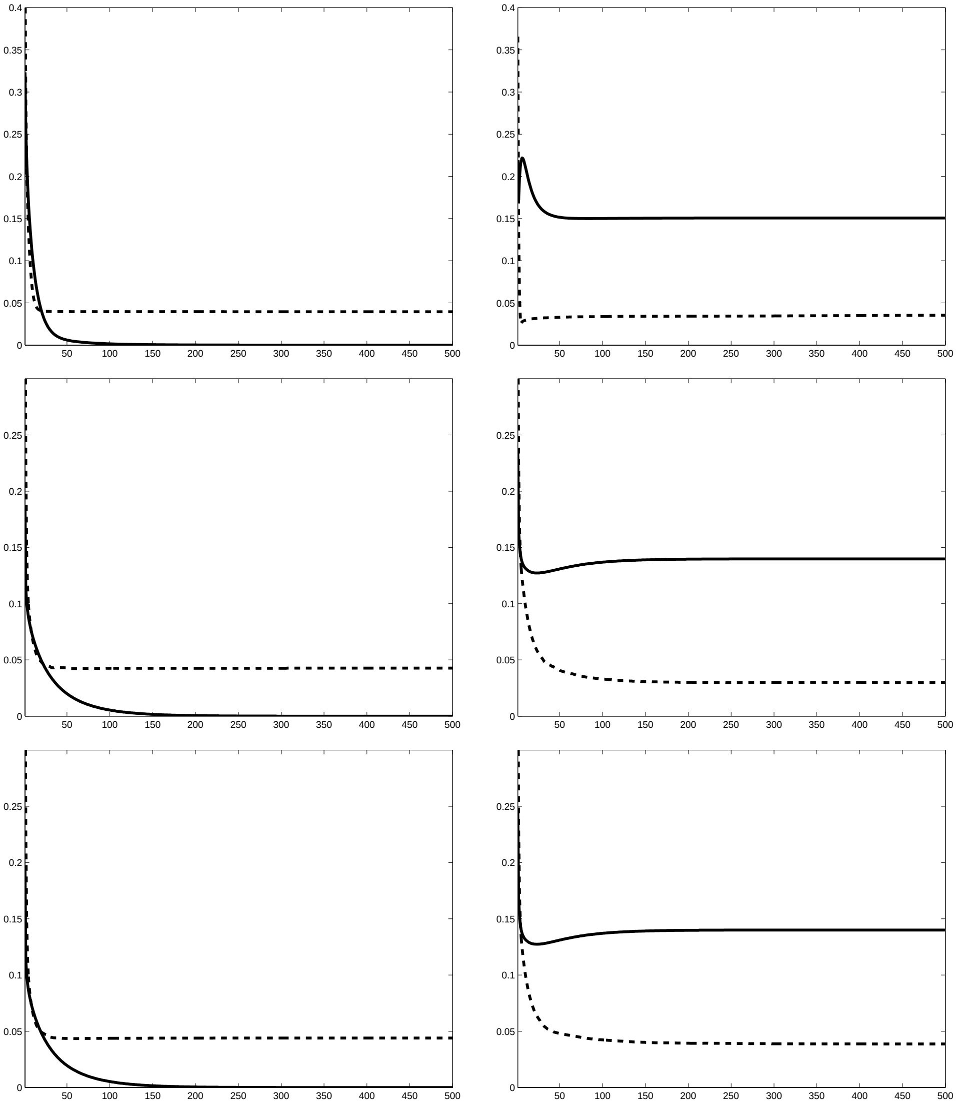


Figure 5: Relative discrepancy error ε_n versus n (left column) and relative restoration error ς_n versus n (right column) for the three 1D examples (from top to bottom). Solid line: Projected Landweber method; dashed line: Algorithm 3.

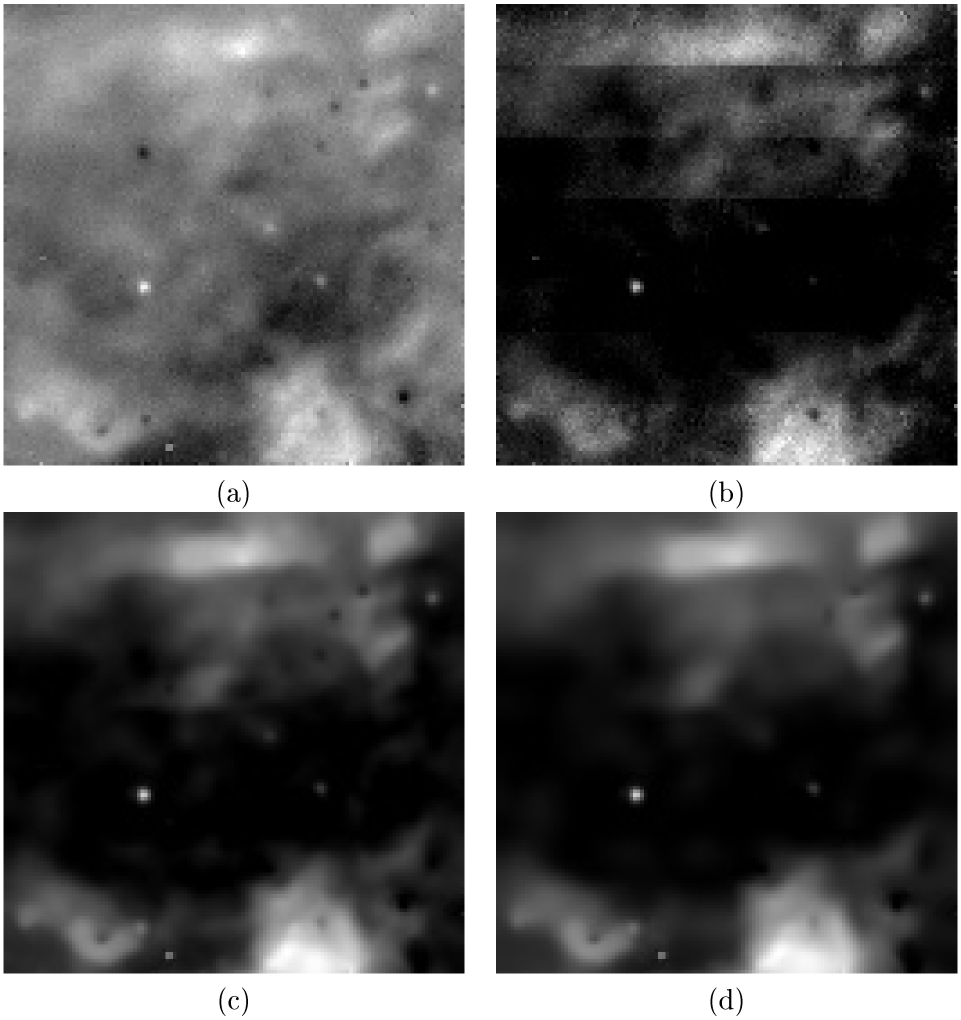
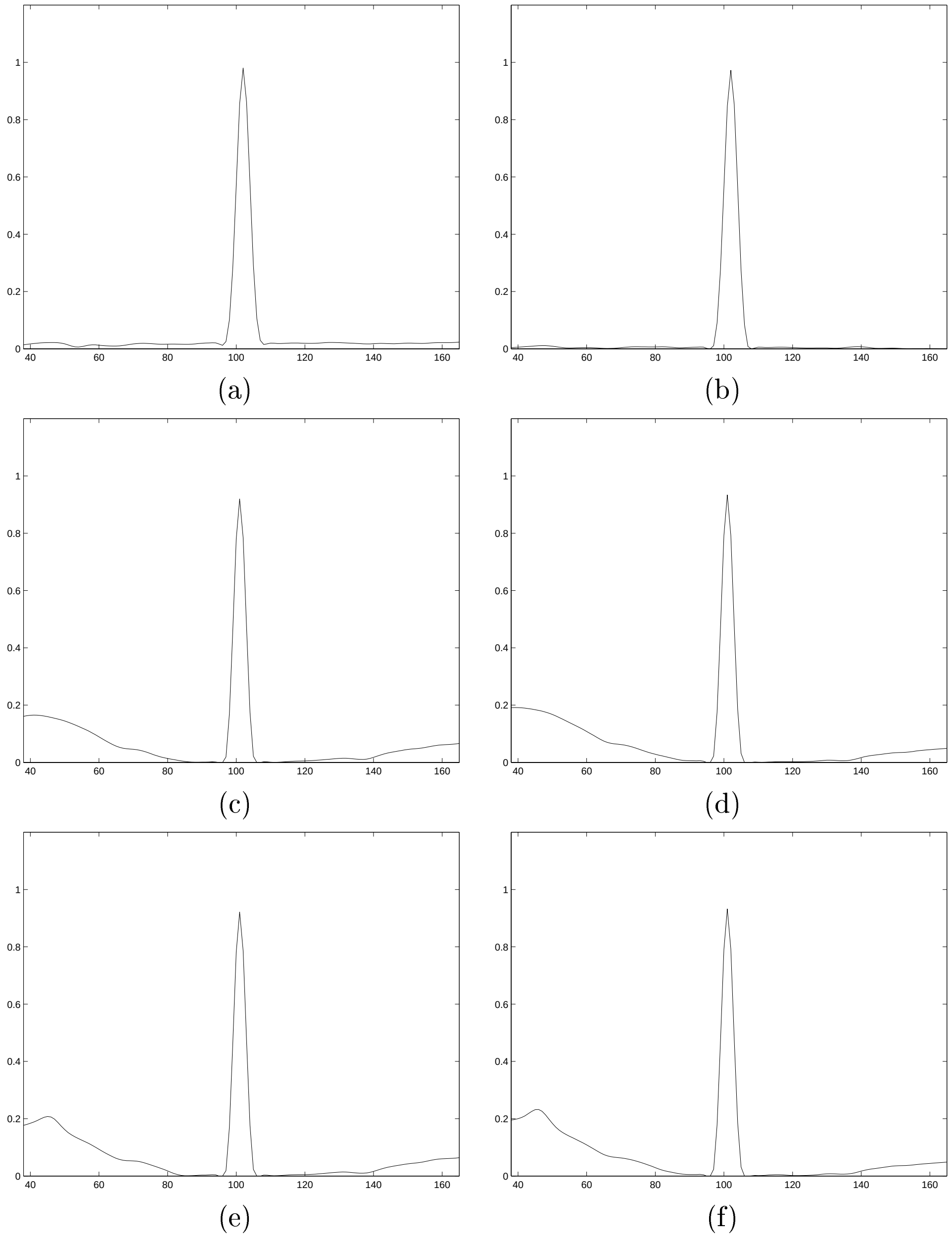


Figure 6: $K = 37$. (a) The chopped and nodded image $\mathbf{g}$; (b) result from the projected Landweber method (118 iterations); (c) result from Algorithm 3 without artifacts reduction (201 iterations); (d) result from Algorithm 3 with artifacts reduction.



(a) (b)

(c) (d)

(e) (f)

Figure 7: The results on the left column are done with only **g** available and those on the right have both **g** and **b** available. (a)–(b) Example 1 (264 and 41 iterations), (c)–(d) Example 2 (280 and 87 iterations), (e)–(f) Example 3 (335 and 92 iterations).