Data-driven tight frame construction and image denoising

Jian-Feng Cai[a], Hui Ji[*,b], Zuowei Shen[b], Gui-Bo Ye[a]

[a] *Department of Mathematics, University of Iowa, Iowa City, Iowa 52242*
[b] *Department of Mathematics, National University of Singapore, Singapore 117543*

Abstract

Sparsity based regularization methods for image restoration assume that the underlying image has a good sparse approximation under a certain system. Such a system can be a basis, a frame, or a general over-complete dictionary. One widely used class of such systems in image restoration are wavelet tight frames. There have been enduring efforts on seeking wavelet tight frames under which a certain class of functions or images can have a good sparse approximation. However, the structure of images varies greatly in practice and a system working well for one type of images may not work for another. This paper presents a method that derives a discrete tight frame system from the input image itself to provide a better sparse approximation to the input image. Such an adaptive tight frame construction scheme is applied to image denoising by constructing a tight frame tailored to the given noisy data. The experiments showed that the proposed approach performs better in image denoising than those wavelet tight frames designed for a class of images. Moreover, by ensuring that the system derived from our approach is always a tight frame, our approach also runs much faster than other over-complete dictionary based approaches with comparable performance on denoising.

Key words: tight frame, image de-noising, wavelet thresholding, sparse approximation

1. Introduction

In the past decades, sparse approximation played a fundamental role in many signal processing areas, such as compression, data analysis, and signal restoration. Sparse approximation is about keeping most information of the given data with a linear combination of a small number of atoms of some system. Among many different systems, orthonormal wavelet bases [1] have been very

*Corresponding author
Email addresses: jianfeng-cai@uiowa.edu (Jian-Feng Cai), matjh@nus.edu.sg (Hui Ji), matzuows@nus.edu.sg (Zuowei Shen), guibo-ye@uiowa.edu (Gui-Bo Ye)

successful in signal processing as they can approximate piecewise smooth 1D signals very efficiently with only few non-zero wavelet coefficients. In recent years, over-complete systems have become more and more recognized and used in signal processing. Over-complete systems have several advantages over orthonormal bases for sparsely approximating signals, as signals are more likely to have a good sparse approximation under a redundant system. Also, they offer more flexibility and convenience in the filter design. One representative class of over-complete systems are the so-called *wavelet tight frames* derived from extension principles of [2, 3, 4], which are now widespread in many signal processing tasks. Wavelet tight frames sacrifice the orthonormality and linear independence of orthonormal bases while still enjoying the same efficient decomposition and reconstruction schemes as orthonormal wavelet bases.

Most wavelets used in image processing are separable wavelet bases defined from tensor products of 1D wavelet bases. Despite their successes in 1D signal processing, tensor wavelets are much less efficient when approximating images since tensor wavelets focus mostly on horizontal and vertical discontinuities of images. When discontinuities of the target image have complex geometric properties, the sparsity of a good approximation under tensor wavelets is not satisfying. In recent years, many tight frames have been proposed to more efficiently represent natural images, including *ridgelets* [5], *curvelets* [6, 7], *bandlets* [8], *shearlets* [9], and many others. However, the efficiency of these redundant systems heavily relies on certain functional assumptions of natural images, e.g. isolated objects with C^2 singularities assumed by curvelets. Such an assumption is applicable to cartoon-type images but not to textured images. Natural images vary greatly in terms of geometrical structure, and often they contain a significant percentage of irregular textures with fractal structures. A tight frame designed for efficiently representing one type of continuum may not provide a good sparse approximation to the input image. A better approach is to develop a tight frame system that is specifically optimized for the given image. In other words, the design of a tight frame system should be driven by the input data in order to achieve better performance in terms of sparse approximation.

The concept of "adaptivity" has been explored in recent years by the so-called learning approaches (e.g., [10, 11, 12, 13, 14, 15]). Learning approaches learn an over-complete dictionary from the input image itself to achieve better sparsity of the input image over the learned dictionary. The basic idea of most existing approaches is to first partition images into small image patches and then to find a set of atoms of the dictionary such that each image patch can be approximated by a sparse linear combination of atoms in the dictionary. The adaptively learned over-complete dictionaries derived by these approaches are very effective on sparsely approximating natural images with rich textures. As a result, these adaptive over-complete dictionary based approaches tend to outperform the sparsity-based wavelet thresholding methods in image denoising. Despite the success of these adaptively learning methods, the resulting over-complete dictionaries lack several properties desired for image restoration. One is the so-called *perfect reconstruction property* which ensures that the given signal can be perfectly represented by its canonical expansion in a manner sim-

ilar to orthonormal bases. Also, finding an optimal over-complete system often leads to a severely under-constrained ill-posed problem owing to the redundancy of the over-complete system. It remains a challenging task to develop fast and stable numerical methods for estimating an optimal over-complete system.

In this paper, we aim at developing a new approach to construct discrete tight frames that are adaptive to input images. The adaptively learned tight frame in our proposed approach is more likely to give a highly sparse approximation of the input image than existing wavelet tight frames. Contrast to general over-complete dictionaries, tight frames satisfy the perfect reconstruction property, which is appealing to many image restoration tasks. Also, the sparsity of the canonical frame coefficients of an image is closely related the regularity of the image, which is assumed by many image restoration approaches to obtain the results with less artifacts. Moreover, as we will show later, the minimization problems arising in the construction of tight frames are better conditioned than those of generic over-complete dictionaries, owing to the Parseval identity for tight frames. Thus, by considering a class of tight frames with certain special properties, a fast numerical method is available to construct data-driven tight frames.

To illustrate the benefits of adaptively constructed tight frames from the data itself, we derive an adaptive tight frame denoising method based on a data-driven tight frame construction scheme. The experiments show that our adaptive tight frame denoising technique significantly outperforms standard wavelet thresholding approaches on images of rich textures. Also, it is much faster than some over-complete dictionary based approaches (e.g. the K-SVD method [12]) with comparable performance. The rest of the paper is organized as follows. In Section 2, we first give a brief introduction to wavelet tight frames. Then, in Section 3, we introduce the proposed minimization model and the corresponding numerical method. Section 4 is devoted to the experimental evaluation of the proposed method and discussions.

2. Preliminaries and previous work

2.1. Wavelet and tight frame

In this section, we give a brief introduction to tight frames and wavelet tight frames for image processing. Interested readers are referred to [2, 4, 16, 17] for more details. Let $\mathcal{H}$ be a Hilbert space. A sequence $\{x_n\} \subset \mathcal{H}$ is a *tight frame* for $\mathcal{H}$ if

$$\|x\|^2 = \sum_n |\langle x, x_n\rangle|^2, \quad \text{for any } x \in \mathcal{H}.$$

There are two associated operators. One is the analysis operator W defined by

$$W : x \in \mathcal{H} \longrightarrow \{\langle x, x_n\rangle\} \in \ell_2(\mathbb{N})$$

and the other is its adjoint operator $W^\top$ called the synthesis operator:

$$W^\top : \{a_n\} \in \ell_2(\mathbb{N}) \longrightarrow \sum_n a_n x_n \in \mathcal{H}.$$

Then, the sequence $\{x_n\} \subset \mathcal{H}$ is a tight frame if and only if $W^\top W = I$, where $I : \mathcal{H} \longrightarrow \mathcal{H}$ is the identical operator. In other words, given a tight frame $\{x_n\}$, we have the following canonical expansion:

$$x = \sum_n \langle x, x_n \rangle x_n, \quad \text{for any } x \in \mathcal{H}.$$

The sequence $Wx := \{\langle x, x_n \rangle\}$ is called the *canonical* tight frame coefficient sequence. Thus, tight frames are often viewed as generalizations of orthonormal bases. In fact, a tight frame $\{x_n\}$ is an orthonormal basis for $\mathcal{H}$ if and only if $\|x_n\| = 1$ for all x_n.

One widely used class of tight frames in image processing is the *discrete wavelet tight frame* generated by a set of filters $\{a_i\}_{i=1}^m$. In this section, we will only discuss the un-decimal wavelet tight frames, as they are known to perform better than the un-decimal systems in most image restoration tasks (see e.g. [18, 16]). Given a filter $a \in \ell_2(\mathbb{Z})$, define the linear convolution operator $\mathcal{S}_a : \ell_2(\mathbb{Z}) \to \ell_2(\mathbb{Z})$ by

$$[\mathcal{S}_a v](n) := [a * v](n) = \sum_{k \in \mathbb{Z}} a(n-k)v(k), \quad \forall\, v \in \ell_2(\mathbb{Z}). \tag{1}$$

For a set of filters $\{a_i\}_{i=1}^m \subset \ell_2(\mathbb{Z})$, we define its analysis operator W by

$$W = [\mathcal{S}^\top_{a_1(-\cdot)}, \mathcal{S}^\top_{a_2(-\cdot)}, \ldots, \mathcal{S}^\top_{a_m(-\cdot)}]^\top. \tag{2}$$

Its synthesis operator is defined as the transpose of W:

$$W^\top = [\mathcal{S}_{a_1}, \mathcal{S}_{a_2}, \ldots, \mathcal{S}_{a_m}]. \tag{3}$$

The rows of W forms a tight frame for $\ell_2(\mathbb{Z})$ if and only if $W^\top W = I$. The *Unitary Extension Principle* (UEP) proposed in [2] can be used to construct such tight frames. The unitary extension principle in [2] says that as long as the filters satisfy the full UEP conditions, the wavelet system generated by the framelets corresponding to these filters form a wavelet tight frame in function space $L_2(\mathbb{R})$. Interested readers are also referred to [20, 21, 22, 23] for related works. In the setting of this paper, one of the full UEP conditions ([2, 19]):

$$\sum_{i=1}^m \sum_{n \in \mathbb{Z}^2} a_i(k+n)a_i(n) = \delta_k, \quad \text{for any } k \in \mathbb{Z}^2, \tag{4}$$

is equivalent to $W^\top W = I$ as shown in [19]. For example, the linear B-spline wavelet tight frame used in many image restoration methods (e.g. [24, 25, 26]) is constructed via the UEP. Its associated three filters are listed as follows,

$$a_1 = \frac{1}{4}(1,2,1)^\top; \quad a_2 = \frac{\sqrt{2}}{4}(1,0,-1)^\top; \quad a_3 = \frac{1}{4}(-1,2,-1)^\top. \tag{5}$$

It can be seen that the three filters given above satisfy (4).

Once the 1D framelet filter set $\{a_i\}_{i=1}^m$ for generating a tight frame for $\ell_2(\mathbb{Z})$ is constructed, one way to generate a 2D tight frame for $\ell_2(\mathbb{Z}^2)$ is to use the 2D framelet filters $\{a_i \otimes a_j\}_{i,j=1}^m$ constructed via the tensor product of the 1D framelet filters. A digital image can be viewed as a vector in $\mathbb{R}^N$ by placing the second column of the digital image below the first, the third below the second, and so on. For a given filter with finite support, the operator $\mathcal{S}_a \in \mathbb{R}^{N\times N}$ becomes a block-wise Toeplitz matrix under Neumann boundary conditions. Given a set of filters $\{a_i\}_{i=1}^m \subset \mathbb{R}^N$ satisfying the UEP condition (4), similarly, we define the analysis operator $W \in \mathbb{R}^{mN\times N}$ by (2) and the synthesis operator $W^\top \in \mathbb{R}^{N\times mN}$ by (3). The rows of W form a tight frame for $\mathbb{R}^N$, i.e. $W^\top W = I_N$. Interesting readers are referred to [25, 16] for more details.

2.2. Previous work on adaptively learning over-complete dictionaries

Several approaches have been proposed to learn an over-complete dictionary from a signal/image such that the sparsity of the target signal/image under the learned over-complete dictionary is optimized, e.g., [27, 10, 11, 12, 13, 14]. Earlier works are based on probabilistic reasoning. For example, the maximum likelihood method is used in [27, 10] and the maximum A-posteriori probability approach is used in [11] to construct over-complete dictionaries. In recent years, there has been steady progress on the development of deterministic approaches of learning an over-complete dictionary from an image (e.g. [12, 13, 14]). Representative along this direction is the so-called K-SVD method [12], which presents a minimization model to learn a dictionary from the input image and use the learned dictionary to denoise images. Since our work also belongs to this category, we only give a brief review on the K-SVD method.

Let f, g denote the vector forms of a noisy image $\mathbf{f}$ and its noise-free version $\mathbf{g}$. Let $D \in \mathbb{R}^{m,p}$ denote the dictionary whose column vectors denote the dictionary atoms. Partitioning the image into ℓ image patches of m pixels with overlaps. Let P_j denote the projection operator that maps the image to its j-th patch for $j = 1, 2, \ldots, \ell$. Then the K-SVD method is to denoise the image by solving the following minimization:

$$\min_{g,c,D} \frac{1}{2}\|f - g\|_2^2 + \sum_j \mu_j \|c_j\|_0 + \lambda \sum_j \|Dc_j - P_j g\|_2^2, \tag{6}$$

where $c = (c_1, c_2, \ldots, c_\ell) \in \mathbb{R}^{p,\ell}$ is the matrix whose j-th column vector c_j denotes the expansion coefficient vector of j-th image patch over the dictionary D. Here and throughout the paper, we denote $\|v\|_0$ the so-called ℓ_0 norm of v, i.e., $\|v\|_0$ stands for the cardinality of the set $\{k : v[k] \neq 0\}$. The K-SVD method showed better performance than the standard wavelet thresholding method in the application of image denoising. Such an improvement comes from the fact that the repeating texture elements are likely to be captured as atoms in the system learned by the method.

Despite the impressive performance of the K-SVD method in image denoising, how to efficiently solve the minimization (6) with satisfactory stability remains a challenging task, as the minimization (6) is a very challenging ill-posed

minimization problem. An alternating iterative method implemented in [12] is to alternatively update the estimations of c, g and D during each iteration. Under such an scheme, there are two challenging sub-problems to solve during each iteration. One is how to estimate the dictionary D given the current estimations on c and g, which is a severely under-constrained problem owing to the redundancy of D ($p \gg m$). A heuristic method is proposed in [12] to update the atoms of the dictionary one by one in a greedy manner, which lacks the rigorous treatment on the stability and optimality. This step is time-consuming as the updating of each atom needs one SVD. Another one is how to find a sparse coefficient vector c given an over-complete dictionary D could be a computationally expensive process. The orthogonal matching pursuit is used in [12] to find a sparse coefficient vector c during each iteration, which is also a computationally expensive process.

3. Minimization model and numerical method

In this section, we present a new minimization model for constructing an adaptive discrete tight frame for a given image. An efficient numerical solver for the proposed minimization model is also provided.

3.1. Basic idea

It is known that in many image restoration tasks, an un-decimal system is more effective in reducing artifacts than a decimal system (see e.g. [18, 16]). Thus, the tight frame constructed in our approach will be an un-decimal system. In other words, the tight frame constructed in our approach are the rows of a matrix W as in (2). Let $W(a_1, a_2, \ldots, a_m)$ denote the analysis operator associated with the tight frame generated by the filters $a_1, a_2, \ldots, a_m$. We will not make a distinction between W and $W(a_1, a_2, \ldots, a_m)$ if there is no ambiguity.

Let g denote an input image. Notice that constructing a wavelet tight frame is equivalent to finding filters $\{a_i\}_{i=1}^m$, which satisfies $W^\top W = I$, under suitable constraints. We propose to construct a wavelet tight frame with filters $(a_1, \ldots, a_m)$ by solving the following minimization problem:

$$\min_{v,\{a_i\}_{i=1}^m} \|v - W(a_1, a_2, \ldots, a_m)g\|_2^2 + \lambda^2\|v\|_0, \quad \text{subject to} \quad W^\top W = I. \quad (7)$$

There are two unknowns in (7). One is the coefficient vector v which sparsely approximates the canonical tight frame coefficients Wg, and the other is the set of filters $\{a_i\}_{i=1}^m$ that generates a tight frame as in (2) and (3). The first term $\|v - Wg\|_2^2$ of (7) is to make sure that v is close to the canonical coefficients of g under the analysis of W; the second term $\|v\|_0$ is to enforce the coefficient is sparse; the constraint $W^T W = I$ is to guarantee W is a tight frame. We take an iterative scheme to alternatively update the estimation of the coefficient vector v and the estimation of $\{a_i\}_{i=1}^m$. More specifically, let $\{a_i^{(0)}\}_{i=1}^m$ be the set of the initial filters to start with, e.g. linear spline framelet filters. Then for $k = 0, 1, \ldots, K-1$:

1. Given the frame filters $\{a_i^{(k)}\}_{i=1}^m$, we solve (7) with respect to v only to get the sparse coefficient vector $v^{(k)}$. Define $W^{(k)} = W(a_1^{(k)}, \ldots, a_m^{(k)})$ as in (2). Then, we have

$$v^{(k)} := \text{argmin}_v \|v - W^{(k)} g\|_2^2 + \lambda^2 \|v\|_0. \tag{8}$$

2. Given the sparse frame coefficient vector $v^{(k)}$, update the frame filters $\{a_i^{(k+1)}\}_{i=1}^m$ by solving (7) with respect to $\{a_i\}_{i=1}^m$ only, i.e.,

$$\{a_i^{(k+1)}\}_{i=1}^m := \text{argmin}_{\{a_i\}_{i=1}^m} \|v^{(k)} - W(a_1, a_2, \ldots, a_m) g\|_2^2 \tag{9}$$

   subject to $W^\top W = I_N$, where $W = W(a_1, \ldots, a_m)$ as in (2).

After K iterations, the tight frame adaptive to the image g is defined as

$$W^\top(a_1^{(K)}, a_2^{(K)}, \ldots, a_m^{(K)}).$$

There are two optimization sub-problems need to be solved during each iteration. The solution to the first minimization problem (8) is known to have a unique solution via hard thresholding and the second minimization problem (9) in general is a complex minimization problem. By considering a special class of wavelet tight frames, we will show in Section 3.3 that a modified version of (9) also has an analytic solution via singular value decomposition. As each minimization problem in the alternating iteration above is solved exactly, the cost functional (7) is monotonically decreasing with respect to the sequence $\{\{a_i^{(k)}\}_{i=1}^m\}_k$. In the next, we will give a detailed discussion on solving (8) and (9).

3.2. Minimization (8) *and its implications*

Let $T_\lambda : \mathbb{R}^{Nm} \to \mathbb{R}^{Nm}$ denote the *hard thresholding* operator defined by

$$[T_\lambda v](n) = \begin{cases} v(n) & \text{if } |v(n)| > \lambda; \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

Then it is well-known that the minimization (8) has a unique solution $\bar{v}$ given by applying a hard thresholding operator on the canonical frame coefficient vector $W^{(k)} g$:

$$\bar{v} := T_\lambda(W^{(k)} g). \tag{11}$$

For the completeness, we still provide its proof in the following proposition.

Proposition 1. *For any given $W \in \mathbb{R}^{mN \times N}$ and $g \in \mathbb{R}^N$, the minimization problem given by*

$$\min_{v \in \mathbb{R}^{mN}} \|v - Wg\|_2^2 + \lambda^2 \|v\|_0 \tag{12}$$

has a unique solution $\bar{v} := T_\lambda(Wg)$. Moreover, for any $v \in \mathbb{R}^{mN}$ satisfying $\|v\|_0 \le \|\bar{v}\|_0$,

$$\|v - Wg\|_2 \ge \|\bar{v} - Wg\|_2. \tag{13}$$

Proof. Let $u = Wg$. Re-writing the objective functional in (8) as

$$\|v-u\|_2^2 + \lambda^2\|v\|_0 = \sum_{n=1}^{mN}[(v(n)-u(n))^2 + \lambda^2 \cdot |v(n)|_0],$$

where $|x|_0 = 0$ if $x = 0$ and $|x|_0 = 1$ otherwise. We have

$$\min_{v\in\mathbb{R}^{mN}} \|v-Wg\|_2^2 + \lambda^2\|v\|_0 = \sum_{n=1}^{mN} \min_{v(n)\in\mathbb{R}}[(v(n)-u(n))^2 + \lambda^2|v(n)|_0].$$

For any $u(n)$, we have

$$E(v(n)) = (v(n)-u(n))^2 + \lambda^2|v(n)|_0 = \begin{cases} u(n)^2, & \text{if } v(n) = 0, \\ (v(n)-u(n))^2 + \lambda^2, & \text{if } v(n) \neq 0. \end{cases}$$

Thus,

$$\min_{v(n)} E(v(n)) = \min\{(u(n))^2, \lambda^2\} = \begin{cases} u(n)^2 & \text{if } |u(n)| \leq \lambda, \\ \lambda^2 & \text{if } |u(n)| \geq \lambda. \end{cases}$$

The corresponding minimizer is $\bar{v}(n) = 0$ if $|u(n)| \leq \lambda$ and $\bar{v}(n) = u(n)$ otherwise. In other words, $T_\lambda(u)$ is the unique solution of (12).

For any $v \in \mathbb{R}^{Nm}$, by the definition of $\bar{v}$, we have

$$\|v-Wg\|_2^2 + \lambda^2\|v\|_0 \geq \|\bar{v}-Wg\|_2^2 + \lambda^2\|\bar{v}\|_0.$$

Thus

$$\|v-Wg\|_2^2 - \|\bar{v}-Wg\|_2^2 = \lambda^2(\|\bar{v}\|_0 - \|v\|_0) \geq 0.$$

□

Proposition 1 states that $\bar{v} = T_\lambda(Wg)$ is the best approximation to the canonical frame coefficient vector Wg among all sparse vectors whose cardinality no greater than the cardinality of $\bar{v}$. Using the coefficient vector $\bar{v}$, we can reconstruct a signal $\bar{g}$ defined by

$$\bar{g} = W^\top \bar{v} = W^\top(T_\lambda(Wg)).$$

When the rows of W form a redundant tight frame, the approximation error between reconstruction $\bar{g}$ and g is bounded by $\|\bar{v}-Wg\|_2$. However, $\bar{g}$ is not necessarily the best approximation to the signal g. The minimization (8) can be considered as a balanced approach. For this, we introduce the following result:

Proposition 2. *Let W denote the tight frame satisfying $W^\top W = I$. Then*

$$\|g - W^\top v\|_2^2 + \|(I-WW^\top)v\|_2^2 = \|v-Wg\|_2^2. \tag{14}$$

Proof. By the fact that $W^\top W = I$, we have

$$\begin{aligned}
&\|g - W^\top v\|_2^2 + \|(I - WW^\top)v\|_2^2 \\
&= g^\top g - 2g^\top W^\top v + v^\top WW^\top v + v^\top (I - WW^\top)(I - WW^\top)v \\
&= g^\top g - 2g^\top W^\top v + v^\top WW^\top v + v^\top (I - WW^\top)v \\
&= g^\top g - 2g^\top W^\top v + v^\top v \\
&= g^\top W^\top Wg - 2g^\top W^\top v + v^\top v \\
&= \|v - Wg\|_2^2.
\end{aligned} \tag{15}$$

□

Proposition 2 implies that the reconstruction $\bar{g}$ from the coefficients $\bar{v}$ obtained via (8) indeed minimizes the following objective function:

$$\|g - W^\top v\|_2^2 + \|(I - WW^\top)v\|_2^2 + \lambda^2 \|v\|_0. \tag{16}$$

The minimization (16) is actually the so-called *balanced* approach for sparsity-based regularizations (see more details in [16]). Next, we give a brief discussion on such a balanced approach. Given a signal f and a tight frame system W, there exist several regularizations to find a sparse approximation to f, which may have different outcomes. Suppose we are using the ℓ_1 norm as a convex replacement of the ℓ_0 norm to prompt sparsity. Most existing regularization methods of finding a sparse approximation $\bar{g}$ to the signal g are done via solving the following minimization:

$$\bar{g} := W^\top \bar{v}; \quad \bar{v} := \text{argmin}_v \|g - W^\top v\|_2^2 + \tau \|(I - WW^\top)v\|_2^2 + 2\lambda \|v\|_1, \tag{17}$$

Based on different values of τ, the minimization (17) can be classified into three categories. When $\tau = 0$, the model (17) is called the *synthesis* based approach (e.g. [28, 26]). When $\tau = \infty$, the minimization (17) can be rewritten as

$$\bar{g} := \text{argmin}_f \|g - f\|_2^2 + 2\lambda \|Wg\|_1 \tag{18}$$

since $v \in \text{range}(W)$. This is called the *analysis* based approach (e.g. [29, 30]). When $0 < \tau < \infty$, the model (17) is called a *balanced* approach (e.g. [31, 24, 25]).

The main difference among the three methods lies in how much the second term $\|(I - WW^\top)v\|_2^2$ contributes to the objective function. By rewriting $\|(I - WW^\top)v\|_2^2$ as $\|v - W(W^\top v)\|_2^2$, we see that it measures the distance between the coefficient vector v and the canonical coefficients of its corresponding reconstructed signal $W^\top v$. Since the magnitude of the canonical coefficients reflects the regularity of the reconstructed signal under some mild conditions on the tight frame W (see [32] fore more details), this distance is closely related to the regularity of the reconstructed signal. The smaller $\|(I - WW^\top)v\|_2^2$ is , the more accurately the corresponding coefficient vector v reflects the regularity of the underlying signal. Thus, the synthesis based approach emphasizes the sparsity of the coefficient vector $\bar{v}$, but the decay of $\bar{v}$ does not reflect the regularity of the resulting reconstruction $W^\top \bar{v}$. In contrast, the analysis based approach emphasizes the sparsity of the canonical coefficient vector $W\bar{g}$ which leads to

a more regular approximation. Indeed, it is shown in [33] that, by choosing parameters properly, the analysis based approach can be seen as sophisticated discretization of minimizations involving the widely used total variation penalty [34] and its variations (e.g. [35, 36, 37]). Notice that the coefficient vector $\bar{v}$ obtained from the synthesis based approach will be much sparser than the canonical coefficient vector $W\bar{g}$ obtained by the analysis-based approach. The balanced approach yields a result which balances the sparsity of the coefficient vector and the regularity of the reconstructed signal.

Each approach has its advantages and disadvantages. The choice of the approach depends on the nature of the targeted application. It is empirically observed that for image restoration, the reconstructed image obtained by the synthesis-based approach tends to have some visually unpleasant artifacts. In contrast, the results from the analysis-based approach and the balanced approach usually have less artifacts as they ensure certain regularities along image edges. For our purpose, choose the balanced approach and the reason is two-fold. First, our goal is to construct a tight frame that works better for image restoration than existing ones. Thus, we are not seeking for the tight frame that maximizes the sparsity of the coefficients. Second, the minimization model (18) resulting from the analysis-based approach requires an iterative solver, which is too expensive for our purpose as the minimization (18) will be called for many times in our approach. Therefore, we take an balanced approach where the result $W^\top(T_\lambda(Wg)$ seeks the balance between the regularity of the result and the sparsity of its associated tight frame coefficients.

3.3. The modification of model (9) *and its numerical solver*

The minimization model (9) is a constrained minimization with the quadratic constraints $W^\top W = I$. To ensure $W^\top W = I$, the set of the 2D filters $\{a_i\}_{i=1}^{m^2}$ to satisfy the following quadratic constraints:

$$\sum_{i=1}^{m^2} \sum_{n\in\mathbb{Z}^2} a_i(k+n)a_i(n) = \delta_k, \quad k \in \mathbb{Z}^2. \tag{19}$$

In general, the model (9) is rather complex and solving it could be computationally demanding. In this section, we propose to construct a tight frame with certain special structure such that the corresponding quadratic constraints (19) are greatly simplified. Indeed, the minimization problem (9) has an explicit solution using the wavelet tight frame filters specified in the following proposition.

Proposition 3. *Let $\{a_i\}_{i=1}^{r^2}$ be r^2 real-valued filters with support on $\mathbb{Z}^2 \cap [1, r]^2$ for some positive integer r. Then the filters $\{a_i\}_{i=1}^{r^2}$ satisfy* (19) *as long as they satisfy the following orthogonality constraints:*

$$\langle a_i, a_j\rangle = \sum_{k\in[1,r]^2\cap\mathbb{Z}^2} a_i(k)a_j(k) = \frac{1}{r^2}\delta_{i-j}, \quad 1 \le i, j \le r^2.$$

Proof. For each a_i, let $\vec{a}_i$ denote its column vector form by concatenating all its columns. Define the matrix $A \in \mathbb{R}^{r^2 \times r^2}$ by

$$A = [\vec{a}_1, \vec{a}_2, \ldots, \vec{a}_{r^2}]. \tag{20}$$

Then

$$(A^\top A)(i,j) = \vec{a}_i^\top \vec{a}_j = \frac{1}{r^2}\delta_{i-j}, \quad 1 \le i,j \le r^2.$$

Thus we have $A^\top A = \frac{1}{r^2} I$ which implies $AA^\top = \frac{1}{r^2} I$. Notice that

$$(AA^\top)(k,\ell) = \sum_{i=1}^{r^2} \vec{a}_i(k)\vec{a}_i(\ell) = \frac{1}{r^2}\delta_{k-\ell}, \quad 1 \le k,\ell \le r^2,$$

which gives $\sum_{i=1}^{r^2} a_i(m)a_i(n) = \frac{1}{r^2}\delta_{m-n}, m,n \in \mathbb{Z}^2$. Then we have

$$\begin{aligned} \sum_{i=1}^{r^2}\sum_{n\in\mathbb{Z}^2} a_i(k+n)a_i(n) &= \sum_{n\in\mathbb{Z}^2}\sum_{i=1}^{r^2} a_i(k+n)a_i(n) \\ &= \begin{cases} \frac{r^2}{r^2} = 1, & k = \mathbf{0}; \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

□

By considering the frame filters $\{a_i\}_{i=1}^{r^2}$ with a special structure proposed in Proposition 3, we greatly simplified the quadratic constraints on $\{a_i\}_{i=1}^{r^2}$ to ensure that $W^\top W = I$. The minimization (9) is now simplified to

$$\{a_i^{(k+1)}\}_{i=1}^{r^2} := \operatorname{argmin}_{\{a_i\}_{i=1}^{r^2}} \|v^{(k)} - W(a_1, a_2, \ldots, a_{r^2})g\|_2 \tag{21}$$

subject to $\langle a_i, a_j\rangle = \frac{1}{r^2}\delta_{i-j}$, $1 \le i,j \le r^2$, where W is the analysis operator associated with filters $\{a_i\}_{i=1}^{r^2}$. In the next, we derive the explicit analytic solution to the constrained minimization problem (21). We sequentially partition the coefficient vector $v^{(k)}$ into r^2 vectors, denoted by $v^{(k),i} \in \mathbb{R}^{N\times 1}, i = 1, 2 \ldots, r^2$. Then we can rewrite the objective function of (21) as follows,

$$\begin{aligned} \|v^{(k)} - Wg\|_2^2 &= \sum_{i=1}^{r^2} \|v^{(k),i} - \mathcal{S}_{a_i(-\cdot)}g\|_2^2 \\ &= \sum_{i=1}^{r^2}\sum_{n=1}^{N} \|v^{(k),i}(n) - [\mathcal{S}_{a_i(-\cdot)}g](n)\|_2^2 \\ &= \sum_{n=1}^{N}\sum_{i=1}^{r^2} \|v^{(k),i}(n) - [\mathcal{S}_{a_i(-\cdot)}g](n)\|_2^2. \end{aligned}$$

In the following, we write an image patch or a 2D filter as a column vector by placing the second column below the first, the third below the second and so on.

We denote this operation by placing the vector sign over the symbol denoting the matrix. For example, we denote $\vec{a}_i$ the vector form of a_i by concatenating all its columns. Since the convolution is commutative, for $i = 1, 2, \ldots, r^2$, we have

$$[\mathcal{S}_{a_i(-\cdot)}g](n) = [\mathcal{S}_{g(-\cdot)}\vec{a}_i](n) = \vec{g}_n^\top \vec{a}_i = \vec{a}_i^\top \vec{g}_n, \quad 1 \le n \le N,$$

where $\vec{g}_n$ denotes the transpose of the n-th row of $\mathcal{S}_{g(-\cdot)}$. Here g_n, $n = 1, \ldots, N$, are all $r \times r$ patches from the input image g. Let

$$\vec{v}_n = \left(v^{(k),1}(n), v^{(k),2}(n), \ldots, v^{(k),r^2}(n)\right)^\top, 1 \le n \le N,$$

and define

$$\begin{cases} V = (\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_N) \in \mathbb{R}^{r^2 \times N} \\ G = (\vec{g}_1, \vec{g}_2, \ldots, \vec{g}_N) \in \mathbb{R}^{r^2 \times N} \\ A = (\vec{a}_1, \vec{a}_2, \ldots, \vec{a}_{r^2}) \in \mathbb{R}^{r^2 \times r^2}. \end{cases} \tag{22}$$

Then we have

$$\begin{aligned} \|v^{(k)} - W^\top g\|_2^2 &= \sum_{n=1}^{N} \|\vec{v}_n - A^\top \vec{g}_n\|_2^2 \\ &= \sum_{n=1}^{N} \vec{v}_n^\top \vec{v}_n + \vec{g}_n^\top A A^\top \vec{g}_n - 2\vec{v}_n^\top A^\top \vec{g}_n \\ &= \sum_{n=1}^{N} \vec{v}_n^\top \vec{v}_n + \frac{1}{r^2}\vec{g}_n^\top \vec{g}_n - 2(A\vec{v}_n)^\top \vec{g}_n \\ &= \mathrm{Tr}(V^\top V) + \frac{1}{r^2}\mathrm{Tr}(G^\top G) - 2\mathrm{Tr}(AVG^\top) \end{aligned}$$

where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix. Since the first two terms are constants, the minimization (21) can be rewritten as follows,

$$\max_A \mathrm{Tr}(AVG^\top) \quad \text{s.t.} \quad A^\top A = \frac{1}{r^2} I_{r^2}. \tag{23}$$

The following theorem gives an explicit solution to the above minimization (23).

Theorem 4. [38] *Let B and C be $m \times r$ matrices where B has rank r. Consider the constrained maximization problem:*

$$B_* = \mathrm{argmax}_B \mathrm{Tr}(B^\top C), \quad s.t.\ B^\top B = I_r,$$

Suppose that the single value decomposition (SVD) of C is $C = UDX^\top$. Then $B_* = UX^\top$.

By Theorem 4, we obtain the solution of (23):

$$A_* = \frac{1}{r}(UX^\top)^\top = \frac{1}{r}XU^\top, \tag{24}$$

where U and X are the SVD decomposition of $VG^\top$ such that

$$VG^\top = UDX^\top.$$

In other words, the vector form of the filters $a_i^{(k+1)}$ defined by the minimizer of (21) is exactly the i-th column vector of the matrix A_* given by (24).

In summary, the ultimate minimization model we proposed for constructing a tight frame adaptive to the input image is as follows,

$$\{a_i^*\}_{i=1}^{r^2} := \text{argmin}_{v,\{a_i\}_{i=1}^{r^2}} \|v - W(a_1, \ldots, a_{r^2})g\|_2^2 + \lambda^2\|v\|_0 \tag{25}$$

subject to $\langle a_i, a_j\rangle = \frac{1}{r^2}\delta_{i-j}$, $1 \le i, j \le r^2$, where W is the analysis operator defined by $W = [\mathcal{S}_{a_1(-\cdot)}^\top, \mathcal{S}_{a_2(-\cdot)}^\top, \ldots, \mathcal{S}_{a_r(-\cdot)}^\top]^\top$. The complete description of the numerical solver for solving (25) is given in Algorithm 1. Notice that there are two steps during each iteration and each involves solving one minimization problem. The first is simply done by applying a hard thresholding operator on tight frame coefficients $W(a_1^{(k)}, \ldots, a_{r^2}^{(k)})g$, and the second can be obtained by the single value decomposition of the matrix $VG^\top$. Thus, the computation cost for each iteration is very low.

Algorithm 1 Construction of an un-decimal tight frame adaptive to an image

Input: an image g (clean or noisy)

Output: an un-decimal discrete tight frame $W^\top$ defined by filters $\{a_i^{(K)}\}_{i=1}^{r^2}$

Main procedure:

(I) Initialize tight frame filters $\{a_i^{(0)}\}_{i=1}^{r^2}$ of size at most $r \times r$ using some existing tight frame.

(II) For $k = 0, 1, 2, \ldots, K-1$ do

(1) define $W^{(k)}$ from $\{a_i^{(k)}\}_{i=1}^{r^2}$ by (2);

(2) set $v^{(k)} = T_\lambda(W^{(k)}g)$, where T_λ is the hard thresholding operator defined in (11);

(3) construct matrices V, G by (22);

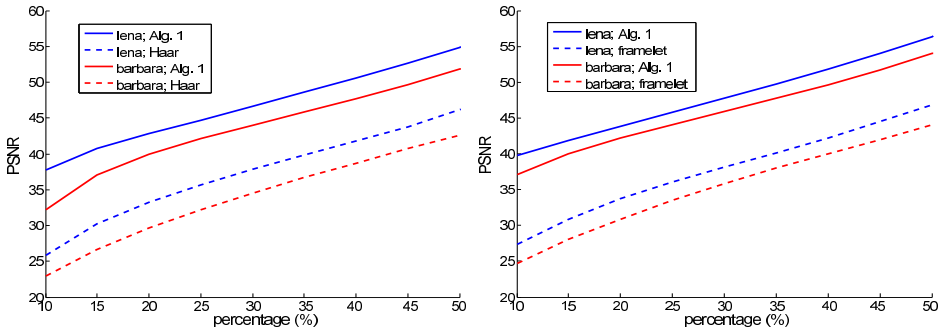
(4) run the SVD decomposition on $VG^\top$ s.t. $VG^\top = UDX^\top$;

(5) set $a_i^{(k+1)}$ to be the i-th column vector of the matrix $A^{(k+1)} = \frac{1}{r}XU^\top$ for $i = 1, \ldots, r^2$.

(III) Output $\{a_i^{(K)}\}_{i=1}^{r^2}$.

3.4. Experiments on real images

To illustrate the tight frame constructed from Algorithm 1, the proposed algorithm is tested on some real images which contain both cartoon-type regions



(a) (b)

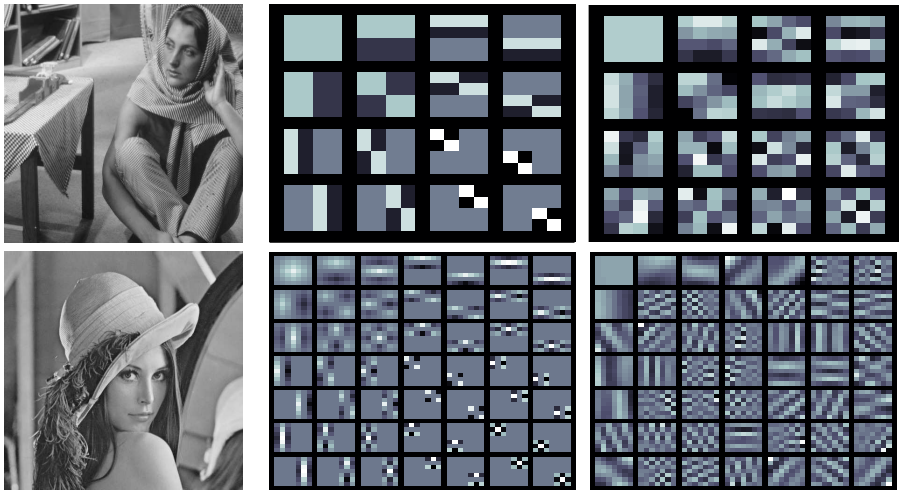
Figure 1: Comparison of the PSNR values of the reconstructions using only a portion of the tight frame coefficients under different tight frame systems. The x-axis denotes the percentage of the tight frame coefficients used in the reconstruction and the y-axis denotes the PSNR value of the reconstruction. (a) Comparison between the 5-level un-decimal Haar wavelet system and the tight frame systems obtained by Algorithm 1 taking as input the Haar wavelet filters. Both system have 16 filters in total. (b) Comparison between the 6-level un-decimal linear spline framelet system and the tight frame systems obtained by Algorithm 1 taking as input the linear framelet filters. Both systems have 49 filters in total.

and texture regions. As we discussed in Section 3.2, neither Algorithm 1 is seeking for the tight frame which maximizes the sparsity of the canonical tight frame coefficients, nor it is seeking for the tight frame in which the signal can be approximated by a optimally sparse coefficient vector. Instead, Algorithm 1 seeks for the tight frame system whose resulting approximation balances the sparsity of the tight frame coefficients and the regularity of the reconstruction. Nevertheless, the tight frame reconstructed by Algorithm 1 is still much more effective on sparsifying the canonical tight frame coefficients than the existing non-adaptive ones. In the experiments, we measure the effectiveness of such a sparsification as follows. Given a tight frame W and an image g, let $\Gamma_{\alpha\%}$ denote the hard thresholding operator which keeps $\alpha\%$ of the largest frame coefficients in absolute value and set all other coefficients to zero. Then we calculate the PSNR value of $W^\top(\Gamma_{\alpha\%}(Wg))$ to measure the quality of the reconstructed image by only using $\alpha\%$ of frame coefficients. When the same percentage of canonical frame coefficients is used, a larger PSNR value of the reconstruction implies that the tight frame sparsifies the canonical coefficients of the input image more effectively. For an image $\mathbf{x}$, the peak signal to noise ratio (PSNR) of its estimate $\widehat{\mathbf{x}}$ is defined as

$$\mathrm{PSNR}(\widehat{\mathbf{x}},\mathbf{x}) = 10\log_{10}\frac{255^2}{\frac{1}{LM}\sum_{i=1}^{L}\sum_{j=1}^{M}(\widehat{\mathbf{x}}(i,j)-\mathbf{x}(i,j))^2}$$

where L and M are the dimensions of the image $\mathbf{x}$, and $\mathbf{x}(i,j), \widehat{\mathbf{x}}(i,j)$ are the pixel values of the input and the estimate images at the pixel location (i,j).

In the experiments, Algorithm 1 is applied to two images "Barbara" and "Lena" shown in Fig. 2 (a) with two different initializations. The filters of one



(a) (b) (c)

Figure 2: Illustration of the data-driven tight frame filters constructed by Algorithm 1. (a) Two tested images "Barbara" and "Lena"; (b): filters associated with the multi-level Haar wavelets (on the top) and the filters associated with the multi-level linear spline framelet (on the bottom); (c) corresponding adaptive tight frame filters constructed by Algorithm 1 using the filters in (b) as the input. One small block of the images shown in (b) and (c) represents one filter.

initialization are the 2-level tensor Haar wavelet filters with totally 16 filters $a_{ij}^{(0)} = \frac{1}{4} a_i a_j^\top$ for $i, j = 1, \ldots, 4$, where

$$a_1 = \frac{1}{2}\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, a_2 = \frac{1}{2}\begin{pmatrix} 1 \\ 1 \\ -1 \\ -1 \end{pmatrix}, a_3 = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -1 \\ 0 \\ 0 \end{pmatrix}, a_4 = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 \\ 0 \\ 1 \\ -1 \end{pmatrix}.$$

The filters of the other initialization is the tensor linear spline framelets [4] with

totally 49 filters $a_{ij}^{(0)} = \frac{1}{7} a_i a_j^T$ for $i, j = 1, \ldots, 7$, where

$$a_1 = \frac{1}{16}\begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 3 \\ 2 \\ 1 \end{pmatrix}; a_2 = \frac{\sqrt{2}}{16}\begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ -1 \\ 0 \\ -1 \end{pmatrix}, a_3 = \frac{\sqrt{2}}{16}\begin{pmatrix} -1 \\ 2 \\ -3 \\ 4 \\ -3 \\ 2 \\ -1 \end{pmatrix}, a_4 = \frac{\sqrt{2}}{4}\begin{pmatrix} 1 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

$$a_5 = \frac{1}{4}\begin{pmatrix} -1 \\ 2 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, a_6 = \frac{\sqrt{2}}{4}\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ -1 \end{pmatrix}, a_7 = \frac{1}{4}\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 2 \\ -1 \end{pmatrix}.$$

In Figure 1, the PSNR values of the reconstruction using different percentages of tight frame coefficients are plotted with respect to different images and different tight frame systems. Clearly, the tight frames constructed by Algorithm 1 are more effective on sparsifying the canonical tight frame coefficients of the input images than the Haar wavelets and linear framelets, as the PSNR values of the reconstructed images are significantly higher using the same percentage of canonical frame coefficients. Such an effectiveness comes from the fact that the tight frame constructed by Algorithm 1 can efficiently capture the repeating complex texture patterns of two input images while the two existing frame systems can not. See Fig. 2 for an illustration of the filters of the tight frames constructed by Algorithm 1 with respect to two different initializations and two different images. It can be seen that the filters constructed by Algorithm 1 tend to fit the repeating patterns in the images.

4. Adaptive tight frame image denoising method

Algorithm 1 can easily be extended to an adaptive tight frame denoising method as follows. The proposed construction scheme of an adaptive tight frame is first applied to a noisy input and then the obtained tight frame is used for noise removal. Let $f = g + n$ denote some noisy observation of g where n denotes the i.i.d. Gaussian white noise. Then the first two steps in (II) of Algorithm 1 can be viewed as a thresholding denoising method under a tight frame. The intermediate denoising result during each iteration is

$$g^{(k)} = {W^{(k)}}^{\top}(T_\lambda(W^{(k)} f))$$

with $W^{(k)} = W(a_1^{(k)}, \ldots, a_{r^2}^{(k)})$, where λ is the threshold whose value depends on both the noise level and the desired sparsity degree of the image. After

Algorithm 1 is terminated, the resulting tight frame $W^{(K)} = W(a_1^{(K)}, \ldots, a_{r^2}^{(K)})$ can be used to denoise the image

$$g^{(K)} = {W^{(K)}}^{\top}(T_{\widetilde{\lambda}}(W^{(K)} f)), \tag{26}$$

where $\widetilde{\lambda}$ is the threshold only determined by the noise level. Thus, the value of λ should be set larger than the value of $\widetilde{\lambda}$. It can be empirically observed that $\lambda \approx 2\widetilde{\lambda}$ is a good choice. Notice that the tight frame constructed during each iteration is directly estimated from the noisy data f instead of the noise-free image g.

The above adaptive tight frame denoising method is evaluated on several test images with different configurations. Through all experiments, all noisy images degraded by i.i.d. Gaussian white noise are synthesized as follows:

$$\mathbf{f} = \mathbf{g} + \epsilon(\sigma),$$

where $\epsilon(\sigma)$ is the i.i.d Gaussian noise with zero mean and standard deviation σ. Besides visual inspection, the PSNR measurement is used to quantitatively evaluate the quality of the de-noised results. There are only two parameters in our adaptive tight frame denoising method, namely λ in Algorithm 1 and $\widetilde{\lambda}$ in (26). Both of them are closely related to the noise variance σ and the redundancy degree of the used tight frame system. Through all experiments, we uniformly set $\lambda = 5.1\sigma$ and $\widetilde{\lambda} = 2.6\sigma$. Two weights 5.1 and 2.6 is empirically chosen to achieve the best average PSNR value of the results for many real images.

4.1. Performance of the proposed method with respect to different settings

In this experiment, we would like to see how the performance of the proposed method is influenced by certain settings, including the filter size, the initialization of tight frame and the number of iterations. Three different types of initialized tight frames: un-decimal local discrete cosine transform (DCT), un-decimal Haar wavelets [1] and un-decimal linear framelet [16], are tested on the image "Barbara" with different filter sizes. The filters for DCT are the tensor products of columns of the $r \times r$ DCT-IV transform matrix [39] defined by $D(k,n) = w(k)\cos\left(\frac{\pi(2n-1)(k-1)}{2r}\right)$, where $w(k) = \frac{1}{\sqrt{r}}$ for $k = 1$ and $w(k) = \sqrt{\frac{2}{r}}$ for $k = 2, \ldots, r$. The Haar wavelet filters of size 2×2, 4×4, 8×8 and 16×16 are corresponding to the filters associated with the $1, 2, 3, 4$-level un-decimal wavelet decompositions and reconstructions respectively. The linear tight frame filters of size 3×3, 7×7, 15×15 are corresponding to the filters associated with the $1, 2, 3$-level framelet decompositions and reconstructions respectively. Table 1 lists the PSNR values of the results from our approach after 25 iterations, using different initializations on tight frame of different filter sizes. It can be seen from Table 1 that the choice of tight frame used for initialization does not matter much. The PSNR values of the results for the three different types of tight frames are nearly the same. However, the performance of our method

initial tight frame	filter size	hard thresholding	Algorithm 1
local DCT	2×2	26.90	26.32
	4×4	28.60	28.96
	8×8	30.08	30.47
	9×9	30.21	30.64
	16×16	30.45	30.93
Haar Wavelet	2×2	26.82	26.34
	4×4	27.59	29.91
	8×8	27.99	30.46
	16×16	28.05	30.95
linear framelet	3×3	27.52	28.14
	7×7	28.70	30.22
	15×15	28.95	30.86

Table 1: Comparison of the PSNR values (dB) of the de-noised results by the wavelet thresholding denoising method and by Algorithm 1 with respect to different initializations on tight frames and different filter sizes.

varies significantly with respect to different filter sizes. It can be seen that the larger the filter size is, or equivalently the more filters are used, the higher the PSNR values of the results will be. It is not surprising as the larger the filter size is and the more filters are used, which makes the learned tight frame filters are more likely to capture the special structures of the input image. At last, the learned adaptive tight frame filters after 25 iterations are shown in Fig. 2 (a), in comparison with the initial inputed 4-level Haar wavelet filters shown in Fig. 2 (b).

Table 2 lists the PSNR values of the results for the image "Barbara" of size 512×512 by Algorithm 1 with different maximum iteration numbers. The tight frame is initialized by 64 3-level Harr wavelet filters. It can be seen from Table 2 that after 50 iterations, there is little improvement on the PSNR values of the results. In other words, 50 iterations seem to be adequate to yield a good discrete tight frame for image denoising. Therefore, through all experiments conducted in this section, the maximum iteration number of Algorithm 1 is set to 50.

4.2. Computational efficiency and denoising performance

The experiments are conducted on six images of different types, as shown in Fig. 3. The noisy inputs are synthesized by adding i.i.d. Gaussian white noise with different standard deviation σ to the original images. Two filter sizes are used in the experiments: 8×8 and 16×16. The experiments are run under MATALB R2011b (64 bit) on a PC workstation with a 6 core INTEL Xeon CPU (2.4 Ghz) and 48 GB memory. Under the same hardware and software environment, Algorithm 1 is compared against that of the most related method, the K-SVD method [12], in terms of computational efficiency and denoising performance.

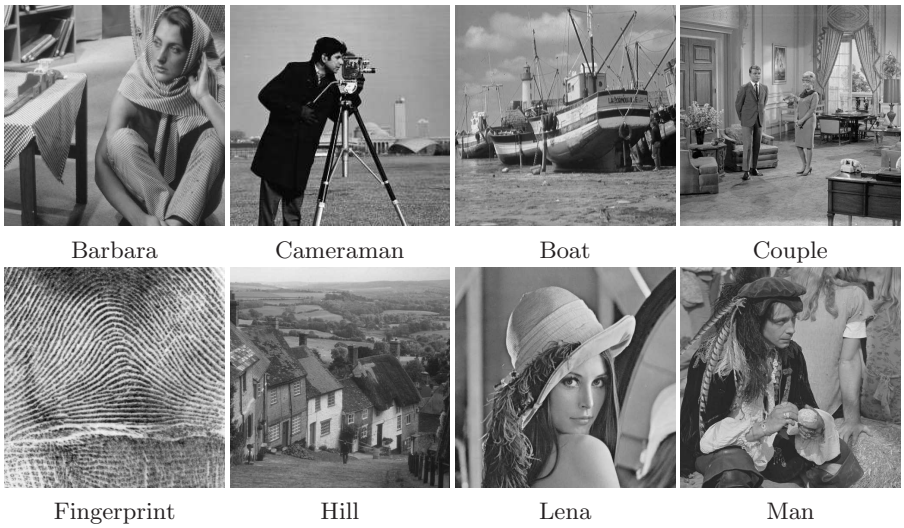


Figure 3: Visualization of eight tested images.

The MATLAB implementation of the K-SVD method used for comparison comes from its original authors[1] and is available online. There are two stages during each iteration of the K-SVD method: sparse coding and dictionary update. The sparse coding is to estimate a sparse coefficient vector under an over-complete dictionary, which is done via orthogonal matching pursuit (OMP) [40, 41]. The dictionary update in the K-SVD method is done via sequentially updating each atom in the dictionary via SVD. For an image of size 512×512, the dimension of the learned dictionary is 256 when using patch size 8×8. Thus, totally 256 SVD operations for matrices of size $256 \times N$ are needed when updating the dictionary, where N (about 6×10^4) is the number of patches. Usually 15 iterations are required for the K-SVD method to generate satisfactory results, which leads to more than 3000 SVD operations during the computation.

Similarly, there are also two stages during each iteration of Algorithm 1: hard thresholding and tight frame update. The hard thresholding is to set the tight frame coefficients to zero if they are below the given threshold and do nothing otherwise. In contrast to the K-SVD method, the tight frame update in Algorithm 1 can update all elements of tight frame in one SVD operation for a matrix of size 64×64 when using filter size 8×8. Overall 50 SVD operations are needed for 50 iterations of Algorithm 1 to generate good results. The most costly computation happens on the synthesis of the matrix $VG^\top$ for $V, G \in \mathbb{R}^{r^2,N}$ in Step (II)-(4), where N is the number of image pixels. One way to further speed up Algorithm 1 is to reduce the size of the matrices V and G by down-sampling the columns. Such a down-sampling operation on V and G can

[1] `http://www.cs.technion.ac.il/~elad/software/`

σ/K	input	0	5	15	25	50	100	200	K-SVD
5	34.14	36.47	37.84	38.16	38.23	38.32	38.35	38.37	38.14
10	28.13	31.71	33.34	34.49	34.63	34.69	34.73	34.73	34.43
15	24.59	28.96	31.00	31.89	32.35	32.51	32.55	32.57	32.42
20	22.11	27.11	28.07	30.69	30.87	30.95	30.99	31.00	30.93
25	20.18	25.68	27.38	29.62	29.76	29.80	29.81	29.82	29.76

Table 2: The PSNR values (dB) of the noisy image, denoised results by Algorithm 1 with maximum iteration number $K = 0, 5, 15, 25, 50, 100, 200$ and de-noised results by the K-SVD method with 15 iterations respectively, where σ denote the standard deviation of the image noise. The support of the filters in Algorithm 1 is set to 8×8. The patch size is set to 8×8 in the K-SVD method.

Method	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$
K-SVD; 8×8	2425.1	1091.4	691.7	529.6	441.8	389.2
Alg. 1; 8×8	13.1	12.9	13.0	13.1	13.0	12.9
K-SVD; 16×16	14144.3	3917.9	1767.9	1017.4	705.1	546.5
Alg. 1; 16×16	89.3	89.2	89.0	89.0	88.9	89.1

Table 3: Comparison of average running time (in second) between the K-SVD method and Algorithm 1 over all tested images. The PSNR values of all results are listed in Table 5.

be viewed as using a decimal wavelet system in image domain when learning a tight frame. For example, when down-sampling the columns of V and G by 4, the computation time is reduced to 1/7 of the original one while the loss on PSNR value is in general smaller than 0.2.

Table 3 lists the running time (in second) of the K-SVD method and Algorithm 1 averaged over all six tested images with respect to different noise levels. It can be seen that Algorithm 1 is considerably faster than the K-SVD method, while the PSNR values of the results from the two methods are comparable (see Table 5 for the details). Table 4 shows the detailed information about the running time of each module in both the K-SVD method and Algorithm 1 when processing the image "Barbara" of size 512×512 with noise of standard deviation $\sigma = 20$. It can be seen that the sparse coding by OMP in the K-SVD method accounts for most of the running time. In contrast, the hard thresholding in Algorithm 1 runs much faster. Also, the tight frame update in Algorithm 1 is much faster than the dictionary update in the K-SVD method, since far less SVD operations (50 vs. 3000) are called in Algorithm 1.

To evaluate the denoising performance of the proposed method, the results from Algorithm 1 are compared against the results from the wavelet thresholding method by 3-level un-decimal Haar wavelets, the K-SVD method with patch size 8×8 and the K-SVD method with patch size 16×16. Table 5 summarizes the PSNR values of the denoising results from these methods with different configurations.

As can be seen from Table 5, both the K-SVD method and our method per-

patch / filter size	stage	8×8	16×16
K-SVD	dictionary update	6.80	20.77
	sparse coding	54.44	145.30
Algorithm 1	tight frame update	0.07	0.89
	thresholding	0.14	1.05

Table 4: Running time (in second) breakdown for one iteration of the K-SVD method and of Algorithm 1 when processing the image "barbara" with noise level $\sigma = 20$.

formed much better than the Haar wavelet thresholding method on all images. In particular, there are significant improvements using the adaptive systems on the images "Barbara", "Fingerprint" and "Lena". The main reason is that these three images have some complex texture regions which the Haar wavelet transform cannot effectively sparsify. In general, the performances of the K-SVD method and our approaches are comparable. There are images on which the K-SVD method performed better and there are some on which our approaches performed better. Overall, the performances of our proposed method and the K-SVD method are comparable in terms of PSNR value, and so is the visual quality. See Fig. 4 and Fig. 5 for visual inspection of the results for the image "Barbara" by different methods.

5. Conclusion and future work

Finding a sparse approximation of a given image plays an important role in many image restoration tasks. Wavelet tight frames have been successfully used to restore the image of interest by utilizing its sparsity under the wavelet tight frame, e.g, the framelets or un-decimal wavelets. However, due to the significant variations of image structure, a pre-defined redundant system is not efficient when representing images of complex structures. In this paper, we developed an iterative numerical scheme to construct a discrete tight frame that is adaptive to the given image. Different from most existing learning based approaches which learn an over-complete dictionary, the dictionary constructed in our approach is always a tight frame during each iteration. Based on the proposed construction scheme, the derived adaptive tight frame denoising method shows its advantage over the traditional wavelet thresholding method. Also, our tight frame based approach is much faster than the over-complete dictionary based approaches, e.g., the K-SVD method.

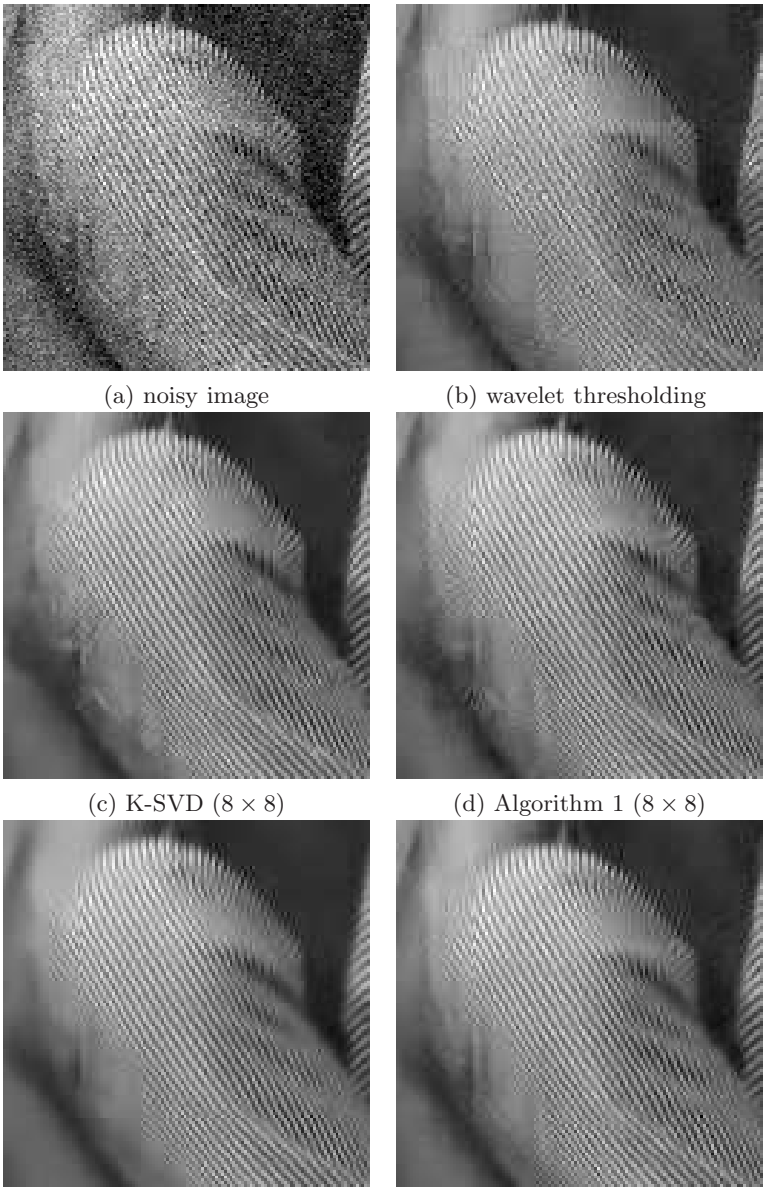
The data-driven tight frame proposed in this paper can not only be used for image denoising, but also for solving other image restoration tasks such as image deconvolution. In near future, we would like to extend the concept of data-driven tight frame construction to other image restoration applications. Moreover, the sparsity prior explored in the existing wavelet frames and the proposed data-driven tight frame only refers to the sparse nature of local variations of the image intensity. In other words, such a sparsity prior is a local image

(a) noisy image; 20.09 dB (b) wavelet thresholding; 27.98 dB

(c) K-SVD (8×8); 30.93 dB (d) Algorithm 1 (8×8); 30.42 dB

(e) K-SVD (16×16); 30.16 dB (f) Algorithm 1 (16×16); 31.01 dB

Figure 4: Visual comparison of denoising results. (a) Noisy image; (b) result by the hard-thresholding method under 3-level un-decimal Haar wavelet system; (c) and (e): results by he K-SVD method with patch size 8×8 and 16×16 respectively; (d) and (f): results by Algorithm 1 with filter size 8×8 and 16×16 respectively.



(a) noisy image (b) wavelet thresholding

(c) K-SVD (8×8) (d) Algorithm 1 (8×8)

(e) K-SVD (16×16) (f) Algorithm 1 (16×16)

Figure 5: Demonstration of one local region of the results shown in Fig. 4. (a) Noisy image; (b) result by the hard-thresholding method under 3-level un-decimal Haar wavelet system; (c) and (e): results by the K-SVD method with patch size 8×8 and 16×16 respectively; (d) and (f): results by Algorithm 1 with filter size 8×8 and 16×16 respectively.

Image	noise σ	Framelet thresholding	K-SVD 8×8	K-SVD 16×16	Alg. 1 8×8	Alg. 1 16×16
Barbara	10	32.08	34.52	34.11	34.36	**34.63**
	20	27.98	30.88	30.26	30.60	**31.07**
	30	25.76	28.58	27.82	28.42	**29.07**
	40	24.25	26.96	26.16	26.88	**27.60**
	50	23.18	25.43	24.68	25.67	**26.48**
	60	22.32	24.32	23.55	24.72	**25.64**
Cameraman	10	32.93	**33.83**	32.84	33.62	33.29
	20	28.98	**29.97**	29.06	29.80	29.67
	30	26.69	**28.10**	27.22	27.66	27.71
	40	25.11	**26.94**	25.89	26.26	26.34
	50	23.57	**25.71**	25.05	25.04	25.33
	60	22.53	**24.74**	24.05	23.96	24.44
Boat	10	32.80	**33.63**	33.03	33.62	33.59
	20	29.36	30.39	29.51	30.38	**30.41**
	30	27.25	28.44	27.38	28.39	**28.45**
	40	25.74	27.09	25.90	27.06	**27.18**
	50	24.48	25.96	24.86	25.99	**26.08**
	60	23.51	24.92	23.99	25.02	**25.37**
Couple	10	33.06	33.54	32.88	**33.63**	33.55
	20	29.42	30.06	29.15	30.09	**30.19**
	30	27.24	27.95	26.96	28.16	**28.27**
	40	25.60	26.42	25.17	26.72	**26.95**
	50	24.39	25.37	24.13	25.68	**25.87**
	60	23.36	24.49	23.26	24.80	**25.04**
Fingerprint	10	30.44	**32.37**	31.88	32.23	32.25
	20	26.49	**28.50**	27.68	28.32	28.40
	30	24.26	26.30	25.23	26.18	**26.34**
	40	22.70	24.71	23.59	24.67	**24.95**
	50	21.45	23.31	22.32	23.52	**23.88**
	60	20.49	21.87	21.20	22.43	**23.07**
Hill	10	32.69	**33.38**	32.83	33.28	33.28
	20	29.46	30.20	29.41	30.22	**30.30**
	30	27.58	28.40	27.39	28.56	**28.61**
	40	26.12	27.13	26.00	27.36	**27.52**
	50	24.96	26.29	24.91	26.48	**26.63**
	60	23.98	25.61	24.22	25.62	**25.92**
Lena	10	34.22	35.52	34.99	35.52	**35.65**
	20	30.69	32.48	31.74	32.25	**32.56**
	30	28.52	30.38	29.61	30.22	**30.58**
	40	26.83	29.04	28.19	28.80	**29.16**
	50	25.50	27.91	26.93	27.60	**28.14**
	60	24.36	26.81	25.77	26.73	**27.26**
Man	10	32.76	**33.59**	32.76	33.57	33.51
	20	29.17	**30.18**	29.15	30.07	30.01
	30	27.14	**28.31**	27.16	28.20	28.24
	40	25.79	**27.07**	25.90	27.00	26.99
	50	24.63	26.08	24.79	26.11	**26.13**
	60	23.67	25.29	24.05	25.25	**25.38**

Table 5: Comparison of the PSNR values (dB) of the results from Algorithm 1 and the K-SVD method.

prior. Another promising image restoration approach emerging in recent years is the non-local approach that is based on another often seen image prior: the global self-recursive prior of small image structures. Examples are the *non-local mean* introduced in [42] for image de-noising and the patch based *BM3D* image denoising method [43]. In the future, we also would like to investigate how to construct adaptive tight frame that not only utilize the local sparsity-based image prior but also exploit the global self-recursion prior of image structures.

Acknowledgement

The authors would especially like to thank Sibin Huang, Yuhui Quan, Chenglong Bao, and Andreas Heinecke for their help on conducting numerical experiments and improving the presentation of this paper. Also, the authors would like to thank the reviewers for their helpful comments and suggestions. This work was partially supported by Singapore MOE Research Grant R-146-000-165-112 and MOE2011-T2-1-116.

References

[1] I. Daubechies, Ten lectures on wavelets, 1st Edition, CBMS-NSF Lecture Notes, SIAM, 1992.

[2] A. Ron, Z. Shen, Affine system in $L_2(R^d)$: the analysis of the analysis operator, J. of Func. Anal. 148.

[3] C. K. Chui, W. He, J. Stöckler, Compactly supported tight and sibling frames with maximum vanishing moments, Appl. Comput. Harmon. Anal. 13 (3) (2002) 224–262.

[4] I. Daubechies, B. Han, A. Ron, Z. Shen, Framelets: MRA-based constructions of wavelet frames, Appl. Comput. Harmon. Anal. 14 (2003) 1–46.

[5] E. Candes, Ridgelets: estimating with ridge functions, Ann. Statist. 31 (1999) 1561–1599.

[6] E. Candes, D. L. Donoho, New tight frames of curvelets and optimal representations of objects with piecewise-C^2 singularities, Comm. Pure Appl. Math 57 (2002) 219–266.

[7] E. J. Candes, D. L. Donoho, Recovering edges in ill-posed inverse problems: Optimality of curvelet frames, Ann. Statist. 30 (3) (2002) 784–842.

[8] S. Mallat, E. Lepennec, Sparse geometric image representation with bandelets, IEEE Transactions on Image Processing 14 (2005) 423–438.

[9] G. Kutyniok, D. Labate, Construction of regular and irregular shearlet frames, J. Wavelet Theory and Appl 1 (2007) 1–10.

[10] M. Lewicki, J. Sejnowski, Learning overcomplete representation, Neural Computation 12 (2000) 227–365.

[11] K. Kreutz-Delgado, J. Murray, B. Rao, K. Engan, T. Lee, T. Sejnowski, Dictionary learning algorithms for sparse representation, Neural Computation 15 (2) (2003) 349–296.

[12] M. Elad, M. Ahron, Image denoising via sparse and redundant representations over learned distionaries, IEEE Trans. on Image Processing 54 (12) (2006) 3736–3745.

[13] J. Mairal, M. Elad, G. Sapiro, Sparse representation for color image restoration, IEEE Transactions on Image Processing 17 (1) (2008) 53–69.

[14] J. Mairal, F. Bach, J. Ponce, G. Sapiro, A. Zisserman, Non-local sparse models for image restoration, in: ICCV, 2009.

[15] W. Dong, X. Li, L. Zhang, G. Shi, Sparsity-based image denoising via dictionary learning and structural clustering, in: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2011, pp. 457–464.

[16] Z. Shen, Wavelet frames and image restorations, Proceedings of the International Congress of Mathematicians, Hyderabad, India.

[17] B. Dong, Z. Shen, MRA based wavelet frames and applications, IAS Lecture Notes Series, Summer Program on "The Mathematics of Image Processing", Park City Mathematics Institute.

[18] D. Donoho, De-noising by soft thresholding, IEEE Trans. Inf. Theory 41 (3) (1995) 613–627.

[19] B. Han, G. Kutyniko, Z. Shen, Adaptive multiresolution analysis structures and shearlet systems, SIAM Journal on Numerical Analysis 49 (5) (2011) 921–1946.

[20] W. M. Lawton, Necessary and sufficient conditions for constructing orthonormal wavelet bases, J. Math. Phys. 32 (1) (1991) 57–61.

[21] L. W. Baggett, P. E. T. Jorgensen, K. D. Merrill, J. A. Packer, Construction of Parseval wavelets from redundant filter systems, J. Math. Phys. 46 (8) (2005) 083502, 28.

[22] M. Paluszyński, H. Šikić, G. Weiss, S. Xiao, Generalized low pass filters and MRA frame wavelets, J. Geom. Anal. 11 (2) (2001) 311–342.

[23] T. Stavropoulos, The geometry of extension principles, Houston J. Math. 38 (3) (2012) 833–853.

[24] A. Chai, Z. Shen, Deconvolution: A wavelet frame approach, Numer. Math. 106 (2007) 529–587.

[25] J.-F. Cai, R. Chan, Z. Shen, A framelet-based image inpainting algorithm, Appl. Comput. Harmon. Anal. 24 (2008) 131–149.

[26] J.-F. Cai, H. Ji, C. Liu, Z. Shen, Blind motion deblurring from a single image using sparse approximation, in: CVPR, 2009.

[27] B. Olshausen, B. Field, Sparse coding with an overcomplete basis set: a strategy employed by v1, Vision Research 37 (1997) 3311–3325.

[28] J.-L. Starck, M. Elad, D. L. Donoho, Image decomposition via the combination of sparse representations and a variational approach, IEEE Trans. Image Proc. 14 (2005) 1570–1582.

[29] I. Daubechies, M. Defrise, C. D. Mol, An iterativethresholding algorithm for linear inverse problems with asparsity constraint, Comm. Pure Appl. Math 57 (11) (2004) 1413–1457.

[30] M. Fadili, J.-L. Starck, F. Murtagh, Inpainting and zooming using sparse representations, The Computer Journal 52 (2009) 64–79.

[31] R. H. Chan, T. F. Chan, L. Shen, Z. Shen, Wavelet algorithms for high-resolution image reconstruction, SIAM J. Sci. Comput. 24 (2003) 1408–1432.

[32] L. Borup, R. Gribonval, M. Nielsen, Bi-framelet systems with few vanishing moments characterize besov spaces, Appl. Comput. Harmon. Anal. 14 (1).

[33] J.-F. Cai, B. Dong, Z. Shen, S. Osher, Image restoration: total variation; wavelet frames; and beyond, J. Amer. Math. Soc. 25 (4) (2012) 1033–1089.

[34] L. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys. D 60 (1992) 259–268.

[35] A. Chambolle, P. Lions, Image recovery via total variation minimization and related problems, Numerische Mathematik 76 (2) (1997) 167–188.

[36] T. Chan, A. Marquina, P. Mulet, High-order total variation-based image restoration, SIAM Journal on Scientic Computing 22 (2) (2001) 503–516.

[37] R.-Q. Jia, H. Zhao, W. Zhao, Relaxation methods for image denoising based on difference schemes, Multiscale Model. Simul. 9 (1) (2011) 355–372.

[38] H. Zou, T. Hastie, R. Tibshirani, Sparse principle component analysis, Journal of Computational and Graphical Statistics 15 (2) (2006) 265–286.

[39] G. Strang, The discrete cosine transform, SIAM Rev. 41 (1) (1999) 135–147 (electronic).

[40] G. Davis, S. Mallat, Z. Zhang, Adaptive time-frequency decompositions, Opt. Eng. 33 (7) (1994) 2183–91.

[41] J. A. Tropp, Greed is good: Algorithmic results for sparse approximation, IEEE Trans. Inf. Theory 50 (2004) 2231–42.

[42] A. Buades, B. Coll, J. Morel, A non-local algorithm for image denoising, in: CVPR, 2005, pp. 60–65.

[43] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3-d transform-domain collaborative filtering, Image Processing, IEEE Transactions on 16 (8) (2007) 2080–2095.