Fast dictionary learning for noise attenuation of multidimensional seismic data[a]

Yangkang Chen

ABSTRACT

The K-SVD algorithm has been successfully utilized for adaptively learning the sparse dictionary in 2D seismic denoising. Because of the high computational cost of many SVDs in the K-SVD algorithm, it is not applicable in practical situations, especially in 3D or 5D problems. In this paper, I extend the dictionary learning based denoising approach from 2D to 3D. To address the computational efficiency problem in K-SVD, I propose a fast dictionary learning approach based on the sequential generalized K-means (SGK) algorithm for denoising multidimensional seismic data. The SGK algorithm updates each dictionary atom by taking an arithmetic average of several training signals instead of calculating a SVD as used in K-SVD algorithm. I summarize the sparse dictionary learning algorithm using K-SVD, and introduce SGK algorithm together with its detailed mathematical implications. 3D synthetic, 2D and 3D field data examples are used to demonstrate the performance of both K-SVD and SGK algorithms. It has been shown that SGK algorithm can significantly increase the computational efficiency while only slightly degrading the denoising performance.

INTRODUCTION

Seismic data is inevitably corrupted by random noise in field acquisition, with important consequences for oil and gas exploration. Thus, random noise attenuation plays a fundamental role in seismic data processing and interpretation (Gulunay, 2000; Zhuang et al., 2015; Qu et al., 2015; Gan et al., 2016d; Li et al., 2016b,a). Over the past few decades, a large number of denoising methods for random noise have been developed. Prediction based methods utilize the predictable property of useful signals to construct prediction filters for enhancing signals and rejecting noise, for example, t-x predictive filtering (Abma and Claerbout, 1995), f-x deconvolution (Canales, 1984), the forward-backward prediction approach (Wang, 1999), the polinomial fitting based approach (Liu et al., 2011), non-stationary predictive filtering (Liu et al., 2012; Liu and Chen, 2013). Mean and median filters utilize the statistical difference between signal and noise to reject the Gaussian white noise or impulsive noise (Liu et al., 2009b; Liu, 2013; Gan et al., 2016c). Decomposition based approaches decompose the noisy seismic data into different components and then select the principal components to represent the useful signals. Empirical mode decomposition and

its variations (Huang et al., 1998), singular value decomposition based approaches (Bekara and van der Baan, 2007; Chen and Ma, 2014; Gan et al., 2015a), regularized non-stationary decomposition based approaches (Fomel, 2013) are usually used to extract the useful components in multidimensional seismic data. Rank-reduction based approaches assume the seismic data to be low-rank after some data rearrangement steps, such methods include the Cadzow filtering (Trickett, 2008), principal component analysis (Huang et al., 2016b), singular spectrum analysis (Oropeza and Sacchi, 2011; Huang et al., 2017), damped singular spectrum analysis (Huang et al., 2016a; Zhang et al., 2016; Chen et al., 2016b,c).

The sparse transform based random noise attenuation is one of the most widely used approaches (Zhang et al., 2015; Chen, 2016). Not only in seismic data processing, but also in all image processing fields, transformed domain thresholding approach has achieved very successful performance (Protter and Elad, 2009; Cai et al., 2013). The denoising step can be implemented by simply applying a thresholding operator in the transformed sparse domain, followed by an inverse sparse transform. Sparse transform can be divided into two types: analytical transform, which has an exact basis, and learning-based dictionary, which iteratively updates the basis by learning (Chen et al., 2016a). I will use *transform* and *dictionary* to refer to these two types of sparse transform, respectively, in this paper.

A lot of transforms have been used in denoising seismic data. Gao et al. (2006) used the wavelet transform to denoise prestack seismic data. Wang et al. (2008) used the second-generation wavelet transform, which is based on the lifting scheme, to denoise seismic data with a percentile thresholding strategy. Hennenfent and Herrmann (2006) and Neelamani et al. (2008) applied the curvelet transform to attenuate both random and coherent noise in seismic data. Zu et al. (2016) applied the curvelet transform to separate simultaneous sources based on the iterative soft-thresholding algorithm. Fomel and Liu (2010) designed a sparse transform that is tailored specifically for seismic data, which is called seislet transform, for sparse representation based processing of seismic data, including seismic denoising (Chen, 2016; Wu et al., 2016), seismic deblending (Chen et al., 2014; Gan et al., 2016b), and data restoration (Gan et al., 2015b, 2016a; Liu et al., 2016). Chen and Fomel (2015a) used the adaptive separation properties of empirical mode decomposition (EMD) (Huang et al., 1998) for preparing the stable input for the non-stationary 1D seislet transform and proposed a new EMD-seislet transform to denoise seismic data with strong spatial heterogeneity. Recently, Kong and Peng (2015) applied the shearlet transform to seismic random noise attenuation.

The learning-based dictionaries are becoming more and more popular for seismic data processing in recent years since their superior performances in adaptively learning the basis that can sparsely represent the complicated seismic data (Sahoo and Makur, 2013). Kaplan et al. (2009) used a data-driven sparse-coding algorithm to adaptively learn basis functions in order to sparsely represent seismic data and then perform denoising in the transformed domain. Based on a variational sparse-representation model, Beckouche and Ma (2014) proposed a denoising approach by

adaptively learning dictionaries from noisy seismic data. Chen et al. (2016a) combined the learning based dictionaries and the fixed-basis transforms and proposed a double-sparsity dictionary to better handle the special features of seismic data, which can can separate signals and noise more precisely.

K-SVD is one of the most effective dictionary learning algorithms (Aharon et al., 2006). However, the computational cost which requires thousands of SVD hinders its wide application in seismic data processing, especially in practical 3D or 5D problems. In this paper, I propose to apply a fast dictionary learning algorithm, which is called sequential generalized K-means (SGK) algorithm (Sahoo and Makur, 2013), to denoise multidimensional seismic data. Since sparse code is relatively new to the seismic community, I introduce the basic formulation of a sparse representation problem and mathematically analyze the principle of K-SVD algorithm and clarify its computational bottleneck. Then, I also introduce the SGK algorithm in detail and apply both K-SVD and SGK algorithms to denoise multidimensional seismic data. Three examples show that the SGK algorithm can significantly accelerate the dictionary learning process and cause no observably worse denoising performance.

METHOD

Problem formulation

Sparse representation via learning based dictionary consists of two main steps.

- Sparse coding. Given the observed data $\mathbf{d}$, sparse coding aims at solving the optimization problem:

$$\mathbf{m}^n = \arg\min_{\mathbf{m}} \parallel \mathbf{d} - \mathbf{F}^n\mathbf{m} \parallel_2^2, s.t. \parallel \mathbf{m} \parallel_0 \leq T, \tag{1}$$

where $\parallel \cdot \parallel_2$ and $\parallel \cdot \parallel_0$ denote the L_2 and L_0 norms of an input vector, respectively. T is the number of non-zero coefficients. $\mathbf{F}$ is the learned dictionary and $\mathbf{m}$ is the sparse representation of $\mathbf{d}$.

- Dictionary updating. For the obtained $\mathbf{m}^n$, update $\mathbf{F}^n$ such that

$$\mathbf{F}^{n+1} = \arg\min_{\mathbf{F}} \parallel \mathbf{d} - \mathbf{F}\mathbf{m}^n \parallel_2^2 . \tag{2}$$

Equations 1 and 2 are iterated *Niter* times to learn the optimal dictionary and the sparest representation.

The multidimensional seismic data is first reformulated into patch form $\mathbf{D}$. Each column vector in $\mathbf{D}$ is extracted from the multidimensional seismic data matrix. An example is given in Yu et al. (2015) and Chen et al. (2016a). Equations 1 and 2 then

become

$$\forall_i \mathbf{m}_i^n = \arg\min_{\mathbf{m}_i} \parallel \mathbf{D} - \mathbf{F}^n \mathbf{M} \parallel_F^2, s.t. \forall_i \parallel \mathbf{m}_i \parallel_0 \leq T, \tag{3}$$

$$\mathbf{F}^{n+1} = \arg\min_{\mathbf{F}} \parallel \mathbf{D} - \mathbf{F}\mathbf{M}^n \parallel_F^2, \tag{4}$$

where $\parallel \cdot \parallel_F$ denotes the Frobenius norm of an input matrix.

Problem 3 is a NP-hard problem, and directly finding the truly optimal $\mathbf{M}$ is impossible and is usually solved by an approximation pursuit method, such as the orthogonal matching pursuit (OMP) algorithm. To solve problem 4 for the adaptive dictionary $\mathbf{F}$, there are several different algorithms.

Dictionary learning by K-SVD

The K-SVD method (Aharon et al., 2006) is one approach that solves equation 4 with good performance. The dictionaries in $\mathbf{F}$ are not obtained at a time. Instead, K columns in $\mathbf{F}$ are updated one by one while fixing $\mathbf{M}$. In order to update the kth column, one can first write the objective function in equation 4 as

$$\begin{aligned}
\parallel \mathbf{D} - \mathbf{F}\mathbf{M} \parallel_F^2 &= \parallel \mathbf{D} - \sum_{j=1}^{K} \mathbf{f}_j \mathbf{m}_T^j \parallel_F^2, \\
&= \parallel \mathbf{D} - \sum_{j \neq k} \mathbf{f}_j \mathbf{m}_T^j - \mathbf{f}_k \mathbf{m}_T^k \parallel_F^2, \\
&= \parallel \mathbf{E}_k - \mathbf{f}_k \mathbf{m}_T^k \parallel_F^2,
\end{aligned} \tag{5}$$

where $\mathbf{f}_j$ is the jth column vector in $\mathbf{F}$, $\mathbf{m}_T^j$ is the jth row vector in $\mathbf{M}$. Here, $[\cdot]_T$ simply indicates a row vector. For simplicity, in equation 5 and the following derivations, I omit the superscript n shown in equation 4. $\mathbf{E}_k$ is the fitting error using all column vectors other than the kth dictionary and their corresponding coefficients row vectors. Note that in equation 5, $\mathbf{D}$ and $\mathbf{E}_k$ are of size $M \times N$, $\mathbf{F}$ is of size $M \times K$, and $\mathbf{M}$ is of size $K \times N$. Here, M is the length of each training signal, N is the number of training signals, and K is the number of atoms in the dictionary.

It is now obvious that the kth dictionary in $\mathbf{F}$ is updated by minimizing the misfit between the rank-1 approximation of $\mathbf{f}_k \mathbf{m}_T^k$ and the $\mathbf{E}_k$ term. The rank-1 approximation is then solved using the singular value decomposition (SVD).

A problem in the direct use of SVD for rank-1 approximation of $\mathbf{E}_k$ is the loss of sparsity in $\mathbf{m}_T^k$. After SVD on $\mathbf{E}_k$, $\mathbf{m}_T^k$ is likely to be filled. In order to solve such problem, K-SVD restricts minimization of equation 5 to a small set of training signals $\mathbf{D}_k = \{\mathbf{d}_i : \mathbf{m}_T^k(i) \neq 0\}$. To achieve this goal, one can define a transformation matrix $\mathbf{R}_k$ to shrink $\mathbf{E}_k$ and $\mathbf{m}_T^k$ by rejecting the zero columns in $\mathbf{E}_k$ and zero entries in $\mathbf{m}_T^k$. First one can define a set $r_k = \{i | 1 \leq i \leq N, \mathbf{m}_T^k(i) \neq 0\}$, which selects the entries in $\mathbf{m}_T^k$ that are non-zero. One then constructs $\mathbf{R}_k$ as a matrix of size $N \times N_r^k$ with ones on the $(r_k(i), i)$ entries and zeros otherwise.

Applying $\mathbf{R}_k$ to both $\mathbf{E}_k$ and $\mathbf{m}_T^k$, then the objective function in equation 5 becomes

$$\| \mathbf{E}_k\mathbf{R}_k - \mathbf{f}_k\mathbf{m}_T^k\mathbf{R}_k \|_F^2 = \| \mathbf{E}_k^R - \mathbf{f}_k\mathbf{m}_R^k \|_F^2 \tag{6}$$

and can be minimized by directly using SVD:

$$\mathbf{E}_k^R = \mathbf{U}\Sigma\mathbf{V}^T. \tag{7}$$

$\mathbf{f}_k$ is then set as the first column in $\mathbf{U}$, the coefficient vector $\mathbf{m}_R^k$ as the first column of $\mathbf{V}$ multiplied by first diagonal entry σ_1. After K columns are all updated, one turns to solve equation 3 for $\mathbf{M}$.

Fast dictionary learning by sequential generalized K-means

Although K-SVD can obtain very successful performance in a number of sparse representation based approaches, since there involves many SVD operations in the K-SVD algorithm, it is very computationally expensive. Especially when utilized in multidimensional seismic data processing (e.g. 3D or 5D processing), the computational cost is not tolerable. The sequential generalized K-means algorithm (SGK) was proposed to increase the computational efficiency (Sahoo and Makur, 2013). SGK tries to solve slightly different iterative optimization problem in sparse coding as equation 3:

$$\forall_i \mathbf{m}_i^n = \arg\min_{\mathbf{m}_i} \| \mathbf{D} - \mathbf{F}^n\mathbf{M} \|_F^2, s.t. \forall_i \mathbf{m}_i = \mathbf{e}_t. \tag{8}$$

t indicates that $\mathbf{m}_i$ has all 0s except 1 in the tth position. The dictionary updating in SGK algorithm is also different. In SGK, equation 6 also holds. Instead of using SVD to minimize the objective function, which is computationally expensive, SGK turns to use least-squares method to solve the minimization problem. Taking the derivative of $J = \| \mathbf{E}_k^R - \mathbf{f}_k\mathbf{m}_R^k \|_F^2$ with respect to $\mathbf{f}_k$ and setting the result to 0 gives the following equation:

$$\frac{\partial J}{\partial \mathbf{f}_k} = -2(\mathbf{E}_k^R - \mathbf{f}_k\mathbf{m}_R^k)(\mathbf{m}_R^k)^T = 0 \tag{9}$$

solving equation 9 leads to

$$\mathbf{f}_k = \mathbf{E}_k^R(\mathbf{m}_R^k)^T \left(\mathbf{m}_R^k(\mathbf{m}_R^k)^T \right)^{-1}. \tag{10}$$

It can be derived further that

$$\begin{aligned}
\mathbf{E}_k^R(\mathbf{m}_R^k)^T &= \left(\mathbf{D}_R - \sum_{j \neq k} \mathbf{f}_j\mathbf{m}_R^j \right) (\mathbf{m}_R^k)^T \\
&= \mathbf{D}_R(\mathbf{m}_R^k)^T + \sum_{j \neq k} \mathbf{f}_j\mathbf{m}_R^j(\mathbf{m}_R^k).^T
\end{aligned} \tag{11}$$

Here, $\mathbf{D}_R$ has the same meaning as $\mathbf{D}$ shown in equation 5 except for a smaller size due to the selection set r_k that selects the entries in $\mathbf{m}_T^k$ that are non-zero.

Since $\forall_i, \| \mathbf{m}_i \|_0 = 1$, as constrained in equation 8 then

$$\forall_{j \neq k} \mathbf{m}_R^j (\mathbf{m}_R^k)^T = 0. \tag{12}$$

Since $\mathbf{m}_R^k$ is a smaller version of row vector $\mathbf{m}_T^k$ and all its entries are all equal to 1, $\mathbf{D}_R(\mathbf{m}_R^k)^T$ is simply a summation over all the column vectors in $\mathbf{D}_R$. Considering that $\mathbf{D}_R = \{ \mathbf{d}_i : i \in r_k \}$,

$$\mathbf{D}_R(\mathbf{m}_R^k)^T = \sum_{i \in r_k} \mathbf{d}_i \tag{13}$$

Following equation 13, equation 11 becomes

$$\mathbf{E}_k^R(\mathbf{m}_R^k)^T = \sum_{i \in r_k} \mathbf{d}_i. \tag{14}$$

It is simple to derive that $\mathbf{m}_R^k(\mathbf{m}_R^k)^T = N_r^k$, where N_r^k denotes the number of elements in the set r_k, or the number of training signals associated with the atom $\mathbf{f}_k$. The kth atom in $\mathbf{F}$ is

$$\mathbf{f}_k = \frac{\sum_{i \in r_k} \mathbf{d}_i}{N_r^k}. \tag{15}$$

Thus, in SGK, one can avoid the use of SVD. Instead the trained dictionary can be simply expressed as an average of several training signals. In this way, SGK can obtain significantly higher efficiency than K-SVD. In the next section, I will use several examples to show that the overall denoising performance does not degrade when one can obtain a much faster implementation.

EXAMPLES

I will use three different examples to show the performance of SGK in denoising multidimensional seismic data. Please note that when using equations 3 (or 8) and 4 for dictionary learning, the multidimensional seismic data is first mapped from the original form to a 2D matrix according to some patching criteria. Some details about the patching method can be found in Yu et al. (2015) or Chen et al. (2016a). After iteratively solving equations 3 (or 8) and 4 several times, the denoised data is expressed as

$$\hat{\mathbf{D}} = \mathbf{F}^{Niter} \mathbf{M}^{Niter}. \tag{16}$$

An inverse mapping is then applied to $\hat{\mathbf{D}}$ to output the finally denoised data.

For measuring the denoising performance of synthetic data examples, where one knows the clean data, I use the signal-to-noise ratio (SNR) (Liu et al., 2009a; Huang et al., 2016a) measurement and the formula is expressed as follows:

$$SNR = 10 \log_{10} \frac{\| \mathbf{D}_{true} \|_F^2}{\| \mathbf{D}_{true} - \hat{\mathbf{D}} \|_F^2}, \tag{17}$$

where $\mathbf{D}_{true}$ denotes the clean data and $\hat{\mathbf{D}}$ denotes the denoised data.

In addition to the commonly used SNR measurement, one can also use local similarity (Fomel, 2007; Chen and Fomel, 2015b) as a convenient tool to evaluate denoising performance. The abnormal area in the local similarity map with high similarity indicates the area that contains significant signal leakage in the removed noise. A local similarity map with values that are close to zero, as well as the observed significant amount of removed noise, provides a valid support of a successful denoising performance. In addition, the local similarity measurement can be used in many cases since the input data of local similarity calculation are just the denoised data and removed noise. It provides an alternative in the case of field data processing, where the clean data is unknown and the SNR based evaluation becomes unavailable. Besides, local similarity is a local measurement of denoising performance, whereas the SNR is a global measurement, which cannot guide us to pick out the areas with poor denoising performances.

Synthetic example

The first example is a 3D synthetic example, as shown in Figure 1. Figure 1a is the clean data and Figure 1d is the noisy data. Figures 1b and 1c show the denoised results using K-SVD and SGK, respectively. Figures 1e and 1f show the removed noise cubes of two approaches. It seems that the denoised results using both methods are very successful while the denoised result using SGK algorithm shows a little bit more residual noise, which is however negligible. The size of this example is $64 \times 16 \times 16$. I use a 3D patch of size $4 \times 4 \times 4$ and the overlap between neighbor patches is 3 points in all time, inline, xline directions. In this example, the sample signals $\mathbf{D}$ is of size 64×10309. The K-SVD takes 192.60s while SGK takes only 9.27s. The SNRs of noisy data, K-SVD result and SGK result are 0.68 dB, 9.61dB and 9.32dB respectively. While the SNRs using the K-SVD and SGK are very similar, the SGK method obtains about 5 times acceleration. To further demonstrate the denoising performance and compare the two methods regarding the tiny differences, I plot the local similarity between denoised data and removed noise in Figure 2. Figures 2a and 2b show the local similarity cubes without amplification that correspond to K-SVD and SGK methods, respectively. It can be seen that both methods cause negligible local similarity, or correlation, between denoised data and removed noise, confirming the extremely successful performance of the two methods. Figures 2c and 2d show the amplified local similarity cubes (similarity $\times$ 2) corresponding to K-SVD and SGK methods, respectively. It can be observed that there are some non-zero amplified similarity values around the events, indicating these tiny damages to the signal caused by both methods. It can also be observed that the amplified local similarity of SGK method is slightly higher than K-SVD method. Considering the slightly low SNR using SGK method, I conclude that SGK method might cause slightly worse performance than the K-SVD method while obtaining a huge improvement on computational efficiency. I also show some learned atoms of this example in Figure 3.

Figures 3a, 3b, and 3c show the atoms from initial dictionary, K-SVD learned dictionary, and SGK learned dictionary, respectively. I set up the initial dictionary using the discrete cosine transform. It can be seen from Figure 3a that the shapes of the atoms in the initial dictionary are rigid, which are not optimal to represent the highly-nonstationary seismic data. After dictionary learning, the updated dictionaries as shown in Figures 3b and 3c contain atoms with much varied shapes. These various atoms are more likely to capture the non-stationary features in the seismic data and thus can potentially improve the sparse representation of the observed noisy seismic data. Please note that each atom has been reshaped into a 3D cube and only 16 atoms are shown here.

In order to test the denoising performances of the two methods in different noise levels, I calculate the SNRs of two methods in four different cases with increasing noise levels (or decreasing input SNRs). The denoising performances of different input SNRs are compared in table 1. The SNR of input noisy data drops from 3.18 dB to -5.33 dB. Correspondingly, the SNRs of the best results using two methods drop from above 10 dB to around 1 dB. Besides, the K-SVD method consistently obtains a slightly higher SNR than SGK method.

In this paper, I compare the speed of SGK with the classic version of K-SVD algorithm (exact SVD calculation). Recently, there are a lot of new algorithms proposed to approximate SVD instead of exactly computing it, e.g., Rubinstein et al. (2008), Foster et al. (2012), and Menon and Elkan (2011). These methods can hopefully improve the efficiency of K-SVD algorithm, but at the expense of slightly degrading the performance. Both K-SVD and SGK use the fast OMP algorithm for sparse coding in the whole dictionary learning process. In order to compare the computing time fairly, I repeat the same calculation three times for each model size and calculate the average time as the final measured computing time.

Input SNR (dB)	K-SVD (dB)	SGK (dB)
3.18	10.60	10.60
0.68	9.61	9.32
-2.84	2.17	2.08
-5.33	1.27	1.11

Table 1: Comparison of denoising performances with different input SNRs between K-SVD and SGK methods.

Field data example

I first use a 2D field data example to compare the difference between K-SVD and SGK methods, as shown in Figure 4. The data size of this example is 512×512 . In this example, I choose a patch size of 8×8. The overlap between different patches is 7 points in both vertical and horizontal directions. Thus the atom size $M = 8 \times 8 = 64$,
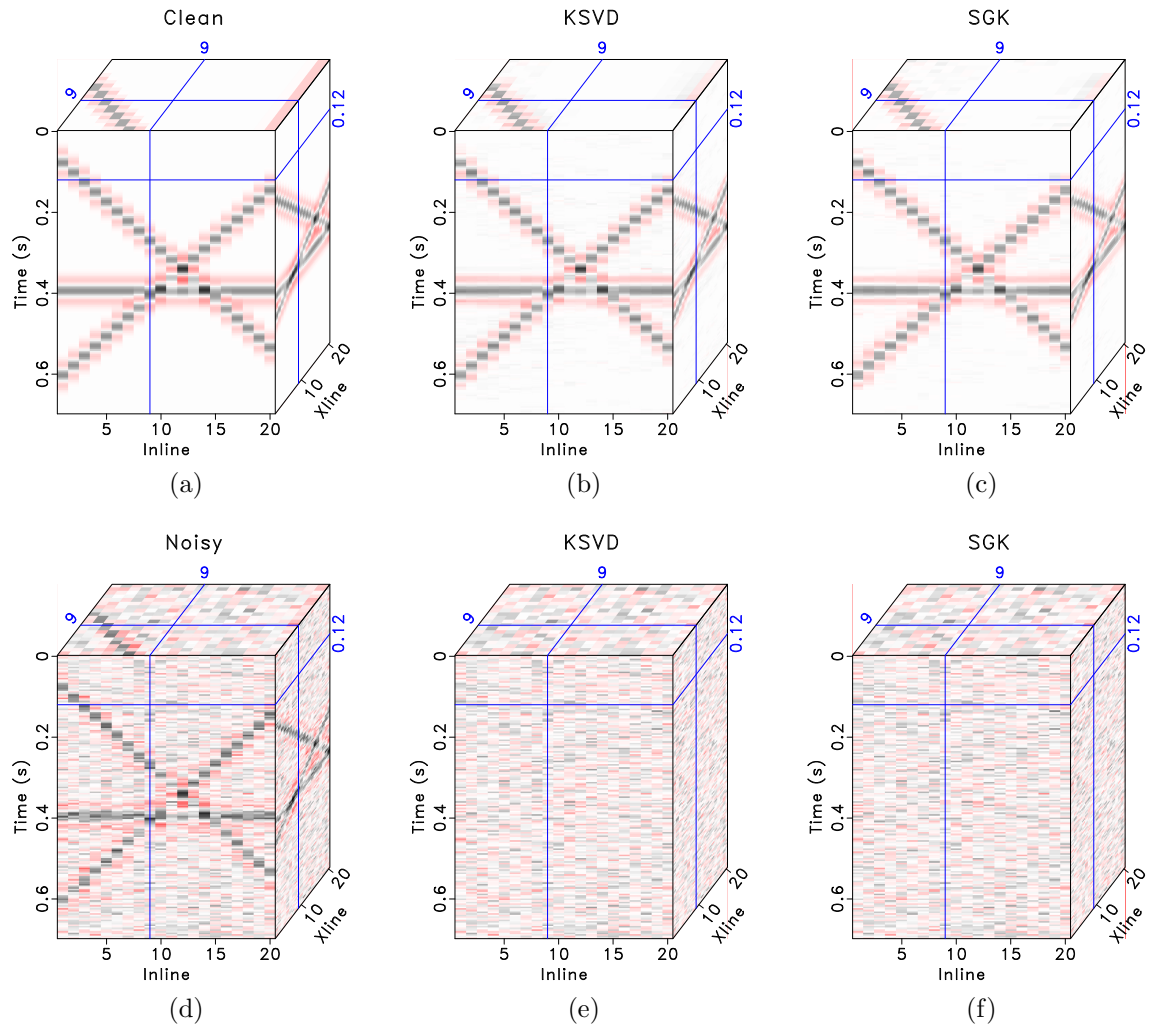
Figure 1: 3D synthetic example. (a) Clean data. (b) Denoised data using K-SVD. (c) Denoised data using SGK. (d) Noisy data. (e) Noise using K-SVD. (f) Noise using SGK.
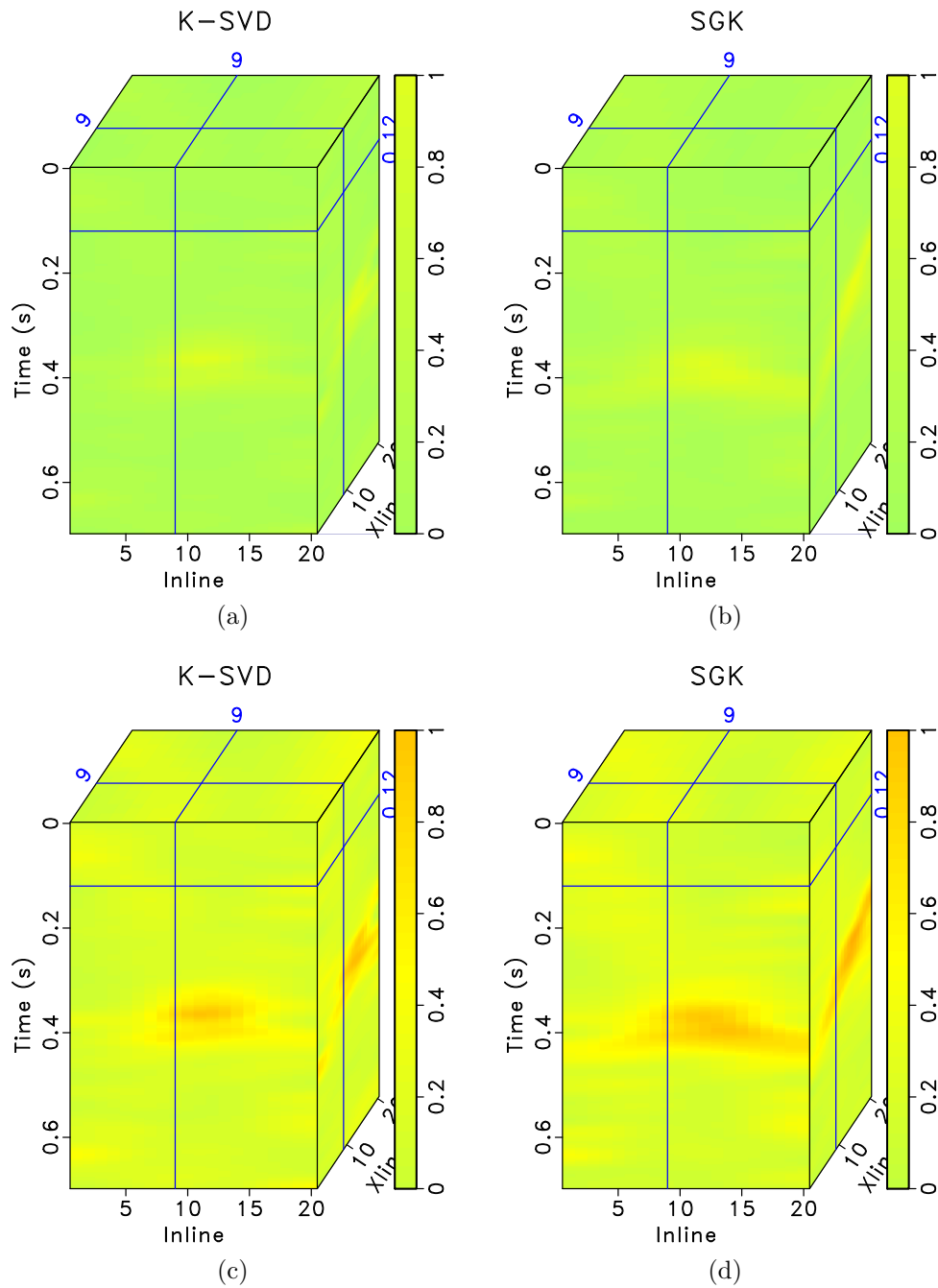
Figure 2: (a) & (b) Local similarity between denoised data and removed noise using K-SVD and SGK. (c) & (d) Amplified local similarity (×2) between denoised data and removed noise using K-SVD and SGK.

(a)

(b)
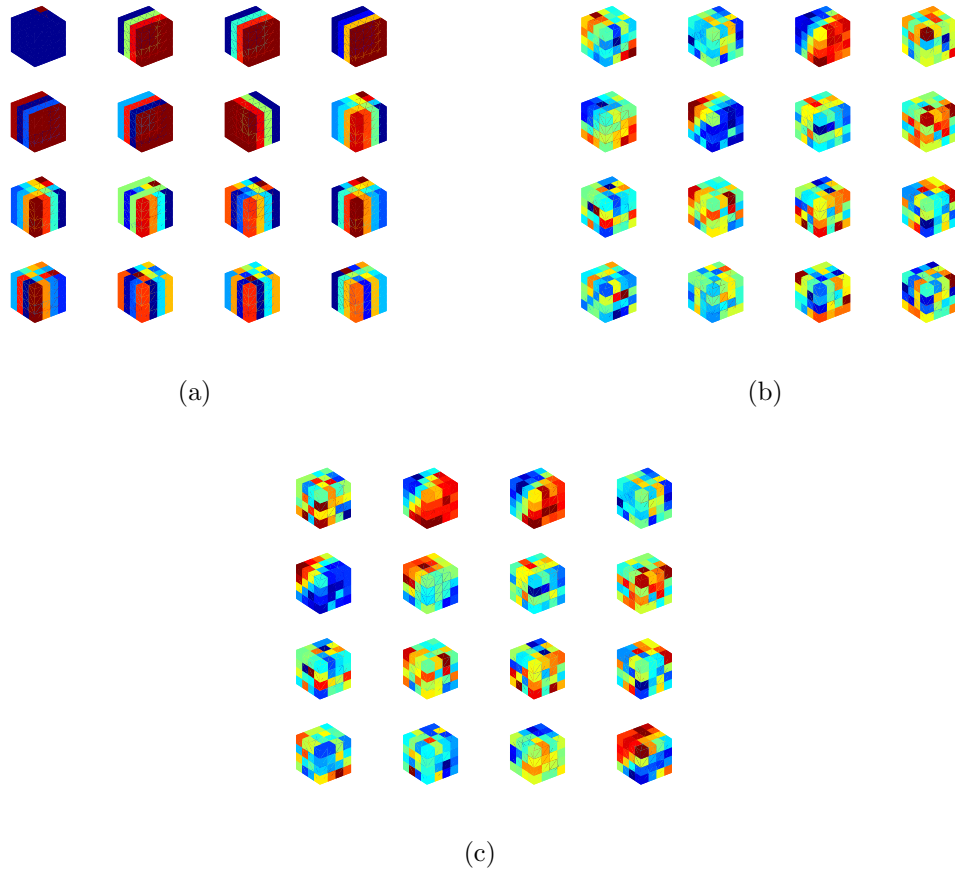
(c)

Figure 3: (a) Initial overcomplete DCT dictionary. (b) Learned dictionary using K-SVD. (c) Learned dictionary using SGK. Only the first 16 atoms in the dictionary are displayed and each dictionary has been reshaped into a 3D cube. It can see from (a) that the shapes of the atoms in the initial dictionary (discrete cosine transform) are rigid, which are not optimal to represent the highly-nonstationary seismic data. After dictionary learning, the updated dictionaries as shown in (b) and (c) contain atoms with much varied shapes. These various atoms are more likely to capture the non-stationary features hidden in the seismic data.

and the number of sample signals is thus $N = (512 - 7) \times (512 - 7) = 255025$. The size of $\mathbf{D}$ is 64×255025. For K-SVD, the dictionary updating process takes about 1590.2s, while for SGK, the dictionary updating process takes only 87.23s, which shows a great speedup. Figure 5a shows the initial input dictionary. Figures 5b and 5c show the learned dictionaries using K-SVD and SGK, respectively. The two learned dictionaries show some similarities but are not exactly the same. As can be seen in either Figure 5b or 5c that there are some atoms in the middle part of the dictionary map containing linear patterns, indicating a better representation of the locally linear events. In this example, I also compare the K-SVD and SGK methods with two other widely known methods, i.e. the DDTF method (Cai et al., 2013) and the seislet transform method (Fomel and Liu, 2010). The denoised results using four methods are shown in Figure 6. The corresponding noise sections are shown in Figure 7. Comparing the results in both Figures 6 and 7, I can roughly get some conclusions that K-SVD, SGK, and DDTF methods all seem to obtain successful denoised results while the result from seislet transform is a bit over-smoothed, which causes some observable low-frequency coherent energy in the noise section (Figure 7d). A better evaluation of denoising performance can be obtained using the local similarity measurement and is shown in Figure 8. The local similarity confirms my observation in that the local similarity corresponding to seislet method is very high, which is followed by the DDTF method. The DDTF method obtains a successful performance in most part of the data but causes some damages to the highly curved signals around the 2s near the left boundary, as indicated from the local similarity map (Figure 8c). The K-SVD and SGK methods obtain very close results but SGK results in a slightly higher local similarity in right part of the data.

I next use a 3D field data example to demonstrate the performance, as shown in Figure 9. Figures 9a, 9b, and 9c show noisy data, K-SVD denoised data and SGK denoised data, respectively. Figures 9d and 9e show the noise sections of two approaches. It is clear that both approaches obtain approximate performance. It is computationally expensive to use K-SVD to learn the dictionary for this example. While it takes about half an hour to learn the dictionary using SGK algorithm, it takes more than half a day to learn the dictionary using the K-SVD algorithm. The local similarity cubes between denoised data cubes and removed noise cubes using two methods are shown in Figure 10, which confirms the successful and comparable performance of both methods in that most part of the data is close to zero.

CONCLUSION

In this paper, I proposed a fast dictionary-learning based seismic denoising approach using the sequential generalized K-means (SGK) algorithm. In the SGK algorithm, each atom in the dictionary is updated by an average of several sample signals while K-SVD uses computationally expensive SVD to update each atom. Thus, the SGK algorithm can be much faster than K-SVD algorithm for adaptively learning the dictionary. I applied both K-SVD and SGK to dictionary learning of seismic data
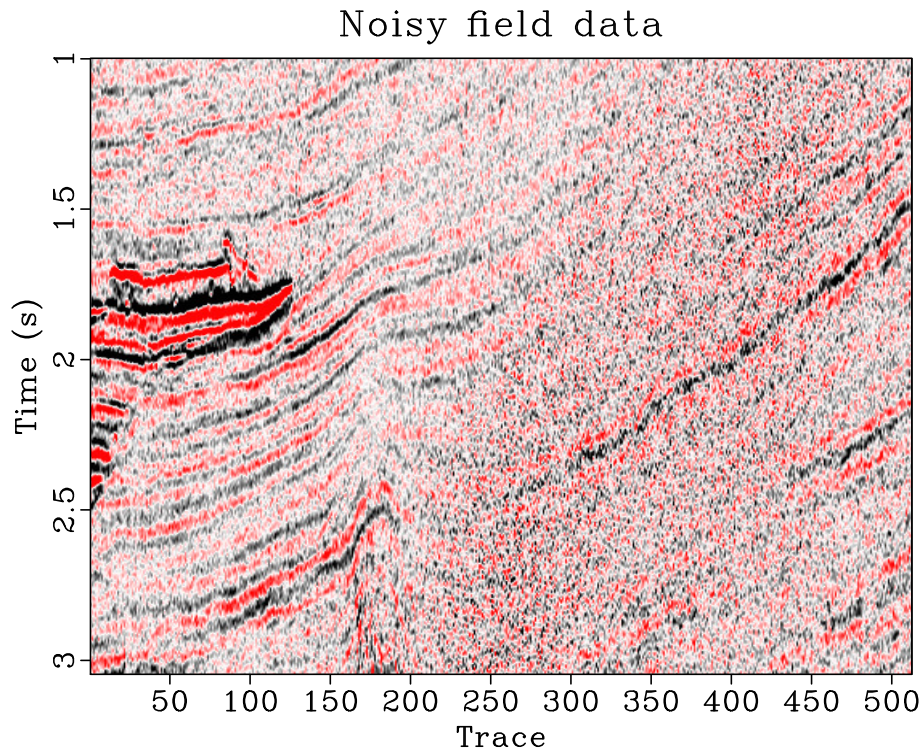
Figure 4: Noisy 2D field data.

for random noise attenuation. The results from three different examples show that SGK is much faster than K-SVD without sacrificing much denoising performance. I suggest substituting the K-SVD with SGK in any applications that require sparse coding. Future research direction may include applying the SGK based dictionary learning for multidimensional seismic data reconstruction.

ACKNOWLEDGMENTS

APPENDIX A: LOCAL SIMILARITY

Let $\mathbf{x}_1$ and $\mathbf{x}_2$ denote the two signal vectors that are reshaped from a 2D matrix or 3D tensor. In the case of evaluating denoising performance, $\mathbf{x}_1$ and $\mathbf{x}_2$ simply means signal and noise. The simplest way to measure the similarity between two signals is
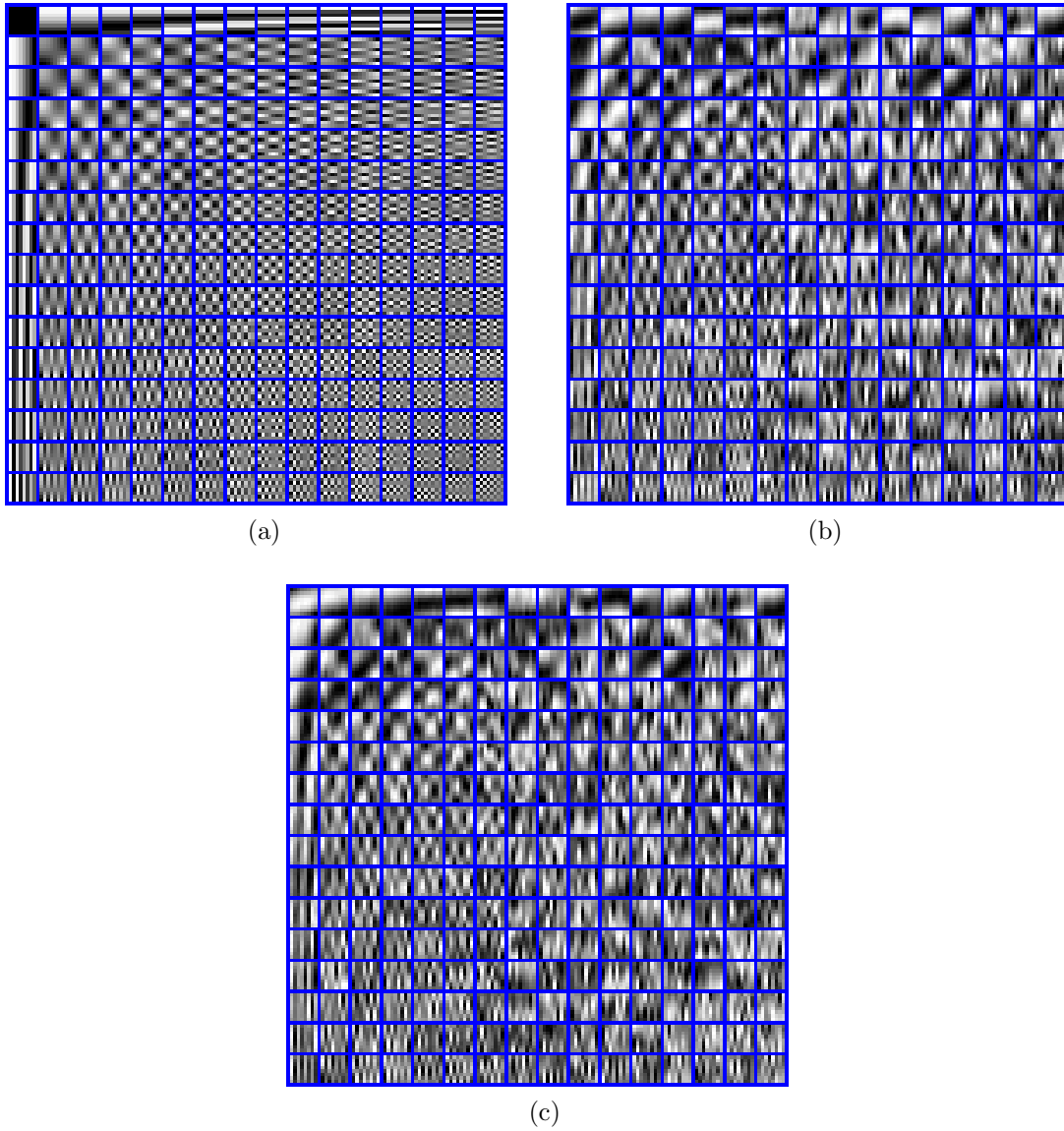
Figure 5: (a) Initial overcomplete DCT dictionary. (b) Learned dictionaries using K-SVD. (c) Learned dictionaries using SGK. Each atom in the dictionary has been reshaped into a 2D matrix. As can be seen in either (b) or (c) that there are some atoms in the middle part of the dictionary map containing linear patterns, indicating a better representation of the locally linear events.
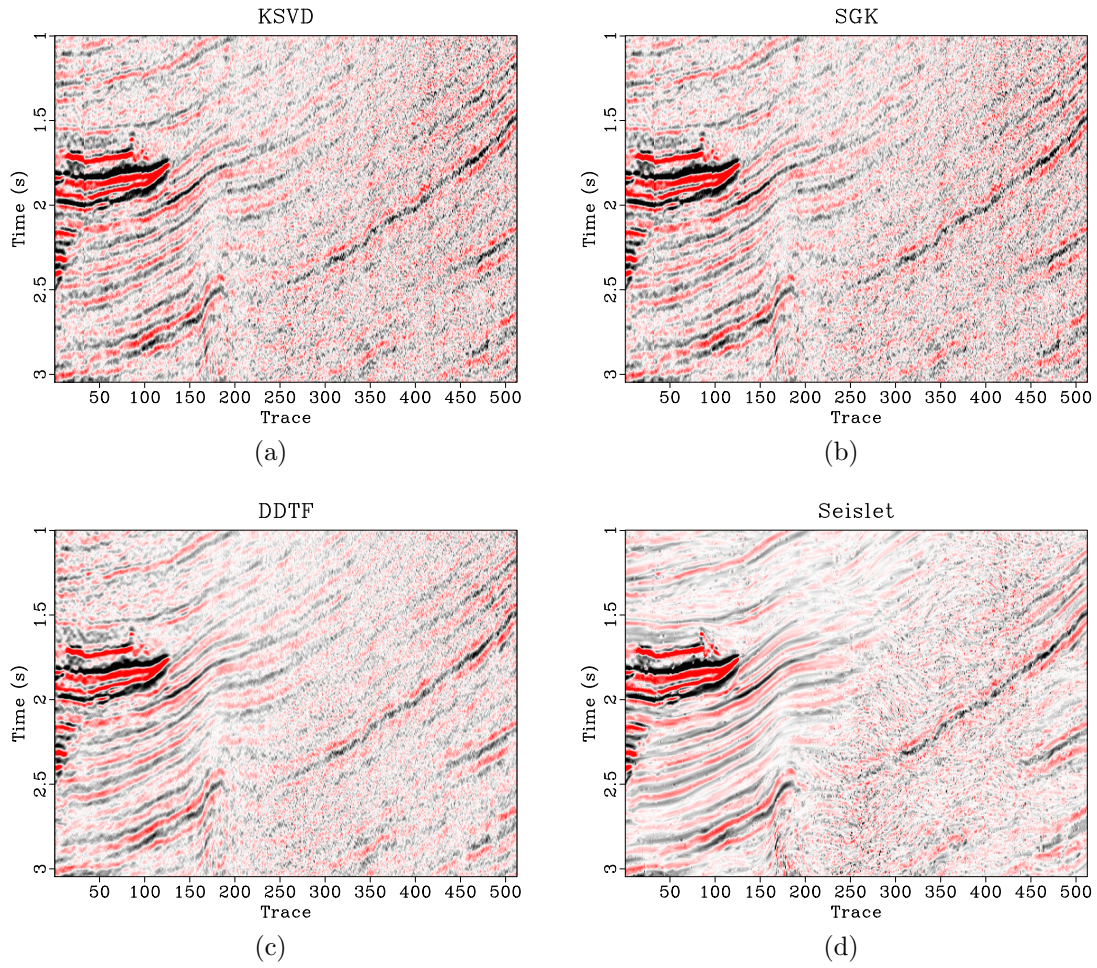
KSVD

SGK

DDTF

Seislet

Figure 6: (a) Denoised data using K-SVD. (b) Denoised data using SGK. (c) Denoised data using DDTF. (d) Denoised data using seislet thresholding.
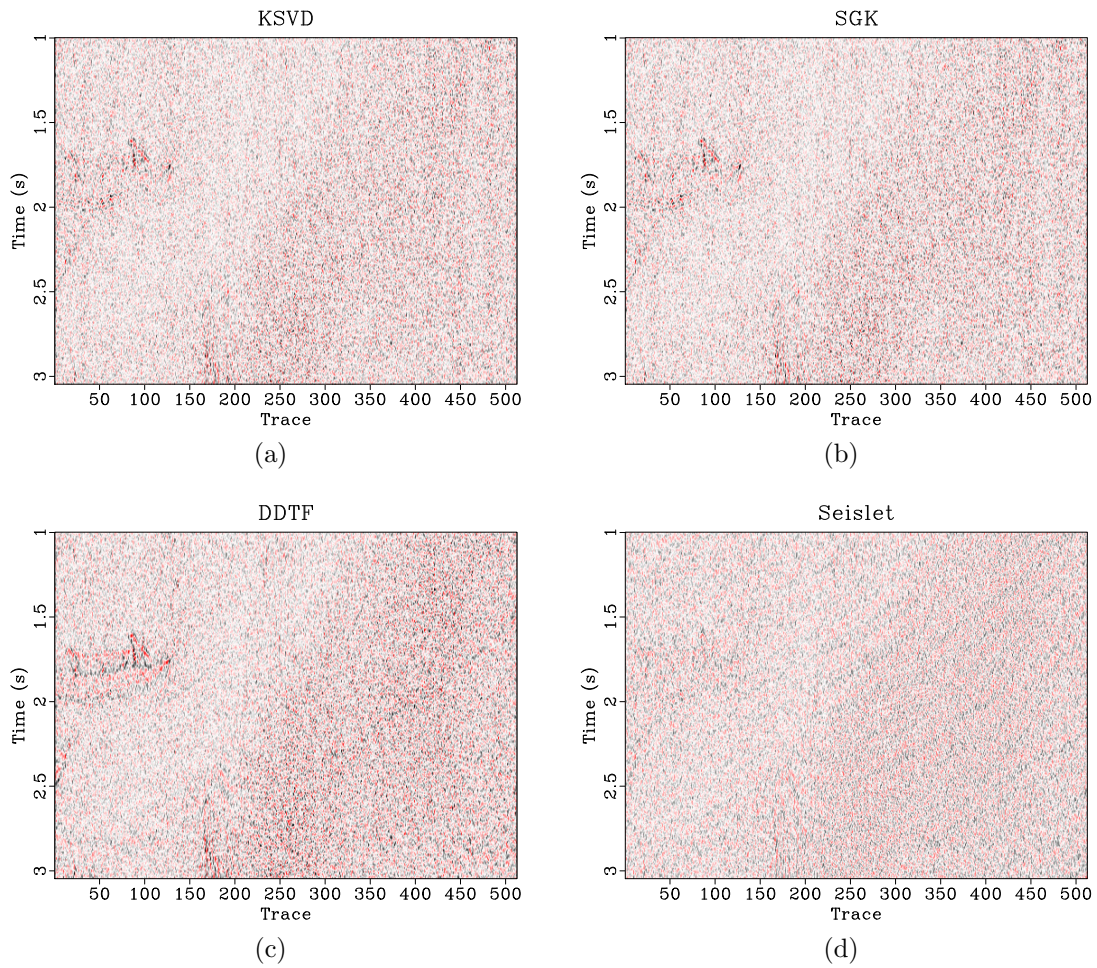
Figure 7: (a) Removed noise using K-SVD. (b) Removed noise using SGK. (c) Removed noise using DDTF. (d) Removed noise using seislet thresholding.
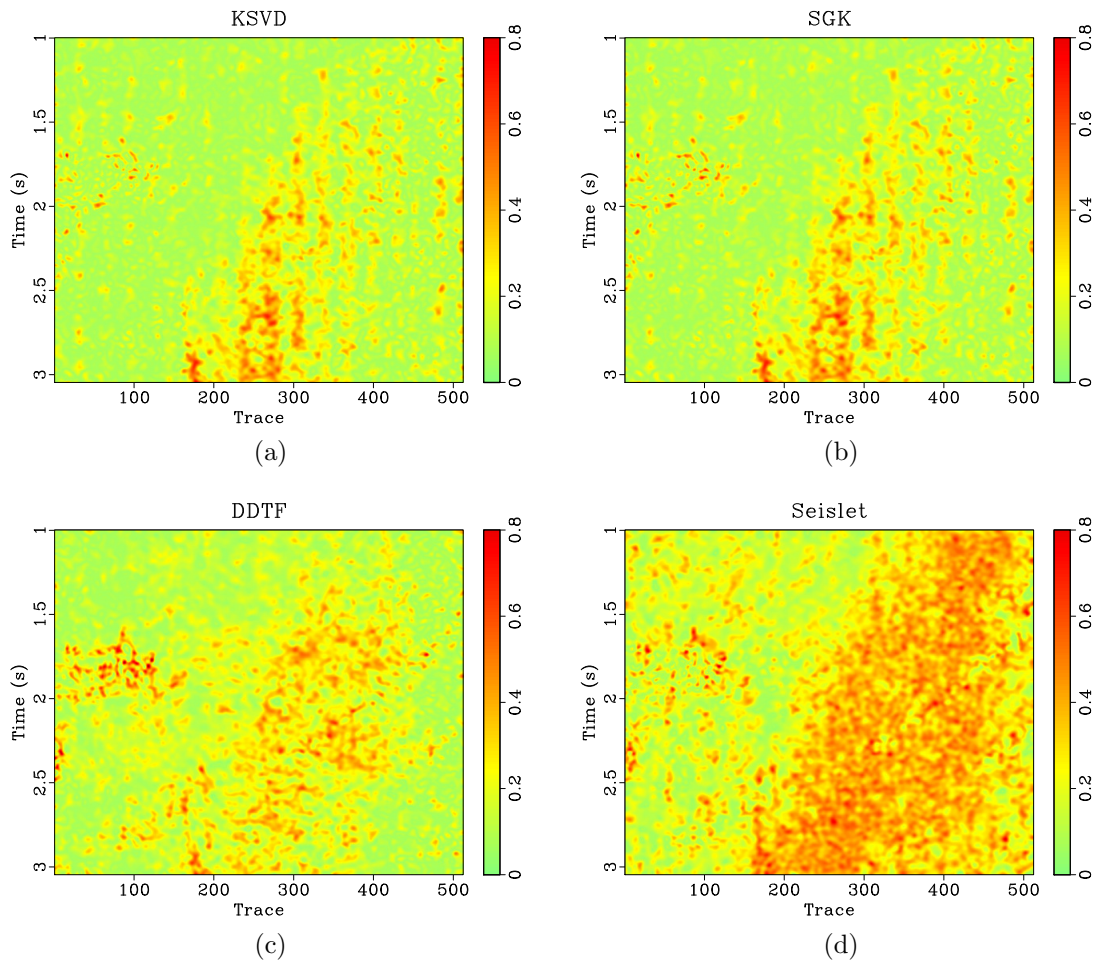
Figure 8: Local similarity between denoised data and removed noise using (a) K-SVD, (b) SGK, (c) DDTF and (d) seislet thresholding.
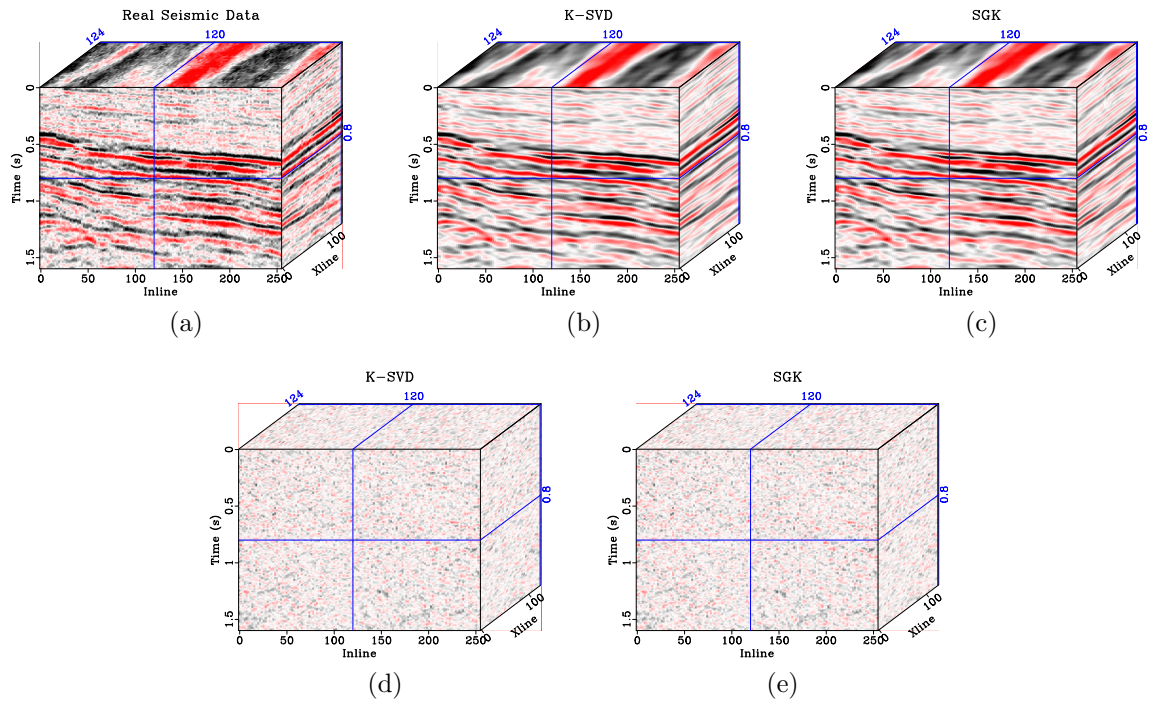
Figure 9: (a) Noisy 3D field data. (b) & (c) Denoised data using K-SVD and SGK. (d) & (e) Noise cubes using K-SVD and SGK.
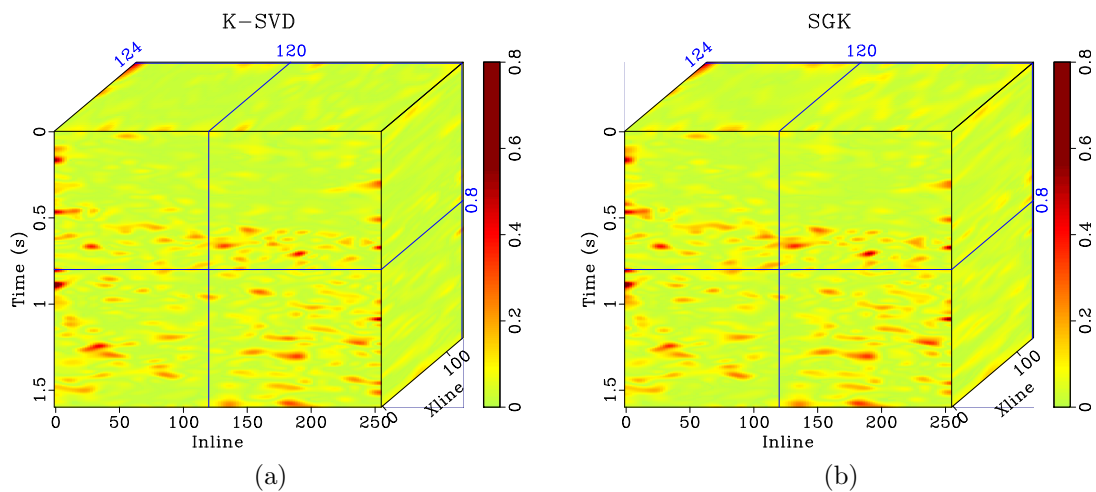

Figure 10: (a) Local similarity between denoised data using K-SVD and the corresponding noise cube. (b) Local similarity between denoised data using SGK and the corresponding noise cube.

to calculate the correlation coefficient,

$$c = \frac{\mathbf{x}_1^T \mathbf{x}_2}{\| \mathbf{x}_1 \|_2 \| \mathbf{x}_2 \|_2}, \tag{18}$$

where c is the correlation coefficient, $\mathbf{x}_1^T \mathbf{x}_2$ denotes the dot product between $\mathbf{x}_1$ and $\mathbf{x}_2$. $\| \cdot \|_2$ denotes the L_2 norm of the input vector. A locally calculated correlation coefficient can be used to measure the local similarity between two signals,

$$c(i) = \frac{\sum_{i_w=-N_w/2}^{N_w/2} x_1(i + i_w) x_2(i + i_w)}{\sqrt{\sum_{i_w=-N_w/2}^{N_w/2} x_1(i + i_w)^2} \sqrt{\sum_{i_w=-N_w/2}^{N_w/2} x_2(i + i_w)^2}}, \tag{19}$$

where $x_1(i)$ and $x_2(i)$ denote the $i$the entries of vectors $\mathbf{x}_1$ and $\mathbf{x}_2$, respectively. i_w denotes the index in a local window. $N_w + 1$ denotes the length of each local window. The windowing is sometime troublesome, since the measured similarity is largely dependent on the windowing length and the measured local similarity might be discontinuous because of the separate calculations in windows. To avoid the negative performance caused by local windowing calculations, Fomel (2007) proposed an elegant way for calculating smooth local similarity via solving two inverse problems. The local similarity I use to evaluate denoising performance in this paper is defined as

$$\mathbf{s} = \sqrt{\mathbf{s}_1 \circ \mathbf{s}_2}, \tag{20}$$

where $\mathbf{s}$ is the calculated local similarity, $\circ$ denotes Hadamard (or Schur) product, and $\mathbf{s}_1$ and $\mathbf{s}_2$ come from two least-squares inverse problem:

$$\mathbf{s}_1 = \arg \min_{\tilde{\mathbf{s}}_1} \|\mathbf{x}_1 - \mathbf{X}_2 \tilde{\mathbf{s}}_1\|_2^2, \tag{21}$$

$$\mathbf{s}_2 = \arg \min_{\tilde{\mathbf{s}}_2} \|\mathbf{x}_2 - \mathbf{X}_1 \tilde{\mathbf{s}}_2\|_2^2, \tag{22}$$

where $\mathbf{X}_1$ is a diagonal operator composed from the elements of $\mathbf{x}_1$: $\mathbf{X}_1 = diag(\mathbf{x}_1)$ and $\mathbf{X}_2$ is a diagonal operator composed from the elements of $\mathbf{x}_2$: $\mathbf{X}_2 = diag(\mathbf{x}_2)$. Equations 21 and 22 are solved via shaping regularization

$$\mathbf{s}_1 = [\lambda_1^2 \mathbf{I} + \mathcal{T}(\mathbf{X}_2^T \mathbf{X}_2 - \lambda_1^2 \mathbf{I})]^{-1} \mathcal{T} \mathbf{X}_2^T \mathbf{x}_1, \tag{23}$$

$$\mathbf{s}_2 = [\lambda_2^2 \mathbf{I} + \mathcal{T}(\mathbf{X}_1^T \mathbf{X}_1 - \lambda_2^2 \mathbf{I})]^{-1} \mathcal{T} \mathbf{X}_1^T \mathbf{x}_2, \tag{24}$$

where $\mathcal{T}$ is a smoothing operator, and λ_1 and λ_2 are two parameters controlling the physical dimensionality and enabling fast convergence when inversion is implemented iteratively. These two parameters can be chosen as $\lambda_1 = \|\mathbf{X}_2^T \mathbf{X}_2\|_2$ and $\lambda_2 = \|\mathbf{X}_1^T \mathbf{X}_1\|_2$ (Fomel, 2007).

REFERENCES

Abma, R., and J. Claerbout, 1995, Lateral prediction for noise attenuation by $t - x$ and $f - x$ techniques: Geophysics, **60**, 1887–1896.

Aharon, M., M. Elad, and A. M. Bruckstein, 2006, The K-SVD: An algorithm for designing of evercomplete dictionaries for sparse representation: IEEE Transactions on Signal Processing, **54**, 4311–4322.

Beckouche, S., and J. Ma, 2014, Simultaneous dictionary learning and denoising for seismic data: Geophysics, **79**, A27–A31.

Bekara, M., and M. van der Baan, 2007, Local singular value decomposition for signal enhancement of seismic data: Geophysics, **72**, V59–V65.

Cai, J., S. Huang, H. Ji, Z. Shen, and G. Ye, 2013, Data-driven tight frame construction and image denoising: Applied and Computational Harmonic Analysis, doi: 10.1016/j.acha.2013.10.001.

Canales, L., 1984, Random noise reduction: 54th Annual International Meeting, SEG, Expanded Abstracts, 525–527.

Chen, Y., 2016, Dip-separated structural filtering using seislet thresholding and adaptive empirical mode decomposition based dip filter: Geophysical Journal International, **206**, no. 1, 457–469.

Chen, Y., and S. Fomel, 2015a, EMD-seislet transform: 85th Annual International Meeting, SEG, Expanded Abstracts, 4775–4778.

——, 2015b, Random noise attenuation using local signal-and-noise orthogonalization: Geophysics, **80**, WD1–WD9.

Chen, Y., S. Fomel, and J. Hu, 2014, Iterative deblending of simultaneous-source seismic data using seislet-domain shaping regularization: Geophysics, **79**, no. 5, V179–V189.

Chen, Y., and J. Ma, 2014, Random noise attenuation by *f-x* empirical mode decomposition predictive filtering: Geophysics, **79**, V81–V91.

Chen, Y., J. Ma, and S. Fomel, 2016a, Double-sparsity dictionary for seismic noise attenuation: Geophysics, **81**, V17–V30.

Chen, Y., D. Zhang, W. Huang, and W. Chen, 2016b, An open-source matlab code package for improved rank-reduction 3D seismic data denoising and reconstruction: Computers & Geosciences, **95**, 59–66.

Chen, Y., D. Zhang, Z. Jin, X. Chen, S. Zu, W. Huang, and S. Gan, 2016c, Simultaneous denoising and reconstruction of 5D seismic data via damped rank-reduction method: Geophysical Journal International, **206**, no. 3, 1695–1717.

Fomel, S., 2007, Local seismic attributes: Geophysics, **72**, A29–A33.

——, 2013, Seismic data decomposition into spectral components using regularized nonstationary autoregression: Geophysics, **78**, O69–O76.

Fomel, S., and Y. Liu, 2010, Seislet transform and seislet frame: Geophysics, **75**, V25–V38.

Foster, B., S. Mahadevan, and R. Wang, 2012, A gpu-based approximate svd algorithm: Parallel Processing and Applied Mathematics, Book chapter, 569–578.

Gan, S., Y. Chen, S. Wang, X. Chen, W. Huang, and H. Chen, 2016a, Compressive sensing for seismic data reconstruction using a fast projection onto convex sets algorithm based on the seislet transform: Journal of Applied Geophysics, **130**, 194–208.

Gan, S., Y. Chen, S. Zu, S. Qu, and W. Zhong, 2015a, Structure-oriented singular value decomposition for random noise attenuation of seismic data: Journal of

Geophysics and Engineering, **12**, 262–272.

Gan, S., S. Wang, Y. Chen, and X. Chen, 2016b, Simultaneous-source separation using iterative seislet-frame thresholding: IEEE Geoscience and Remote Sensing Letters, **13**, no. 2, 197–201.

Gan, S., S. Wang, Y. Chen, X. Chen, and K. Xiang, 2016c, Separation of simultaneous sources using a structural-oriented median filter in the flattened dimension: Computers & Geosciences, **86**, 46–54.

Gan, S., S. Wang, Y. Chen, S. Qu, and S. Zu, 2016d, Velocity analysis of simultaneous-source data using high-resolution semblance-coping with the strong noise: Geophysical Journal International, **204**, 768–779.

Gan, S., S. Wang, Y. Chen, Y. Zhang, and Z. Jin, 2015b, Dealiased seismic data interpolation using seislet transform with low-frequency constraint: IEEE Geoscience and Remote Sensing Letters, **12**, 2150–2154.

Gao, J., J. Mao, W. Chen, and Q. Zheng, 2006, On the denoising method of prestack seismic data in wavelet domain: Chinese J. Geophys, **49**, 1155–1163.

Gulunay, N., 2000, Noncausal spatial prediction filtering for random noise reduction on 3-d poststack data: Geophysics, **65**, 1641–1653.

Hennenfent, G., and F. Herrmann, 2006, Seismic denoising with nonunformly sampled curvelets: Computing in Science & Engineering, **8**, 16–25.

Huang, N. E., Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, 1998, The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis: Proceeding of the Royal Society of London Series A, **454**, 903–995.

Huang, W., R. Wang, Y. Chen, H. Li, and S. Gan, 2016a, Damped multichannel singular spectrum analysis for 3D random noise attenuation: Geophysics, **81**, V261–V270.

Huang, W., R. Wang, Y. Yuan, S. Gan, and Y. Chen, 2017, Signal extraction using randomized-order multichannel singular spectrum analysis: Geophysics, **82**, no. 2, V59–V74.

Huang, W., R. Wang, Y. Zhou, Y. Chen, and R. Yang, 2016b, Improved principal component analysis for 3D seismic data simultaneous reconstruction and denoising: 86th Annual International Meeting, SEG, Expanded Abstracts, 4777–4781.

Kaplan, S. T., M. D. Sacchi, and T. J. Ulrych, 2009, Sparse coding for data-driven coherent and incoherent noise attenuation: 79th Annual International Meeting, SEG, Expanded Abstracts, 3327–3331.

Kong, D., and Z. Peng, 2015, Seismic random noise attenuation using shearlet and total generalized variation: Journal of Geophysics and Engineering, **12**, 1024–1035.

Li, H., R. Wang, S. Cao, Y. Chen, and W. Huang, 2016a, A method for low-frequency noise suppression based on mathematical morphology in microseismic monitoring: Geophysics, **81**, no. 3, V159–V167.

Li, H., R. Wang, S. Cao, Y. Chen, N. Tian, and X. Chen, 2016b, Weak signal detection using multiscale morphology in microseismic monitoring: Journal of Applied Geophysics, **133**, 39–49.

Liu, G., and X. Chen, 2013, Noncausal f-x-y regularized nonstationary prediction filtering for random noise attenuation on 3D seismic data: Journal of Applied

Geophysics, **93**, 60–66.

Liu, G., X. Chen, J. Du, and J. Song, 2011, Seismic noise attenuation using nonstationary polynomial fitting: Applied Geophysics, **8**, 18–26.

Liu, G., X. Chen, J. Du, and K. Wu, 2012, Random noise attenuation using *f-x* regularized nonstationary autoregression: Geophysics, **77**, V61–V69.

Liu, G., S. Fomel, L. Jin, and X. Chen, 2009a, Stacking seismic data using local correlation: Geophysics, **74**, V43–V48.

Liu, W., S. Cao, S. Gan, Y. Chen, S. Zu, and Z. Jin, 2016, One-step slope estimation for dealiased seismic data reconstruction via iterative seislet thresholding: IEEE Geoscience and Remote Sensing Letters, **13**, no. 10, 1462–1466.

Liu, Y., 2013, Noise reduction by vector median filtering: Geophysics, **78**, V79–V87.

Liu, Y., C. Liu, and D. Wang, 2009b, A 1D time-varying median filter for seismic random, spike-like noise elimination: Geophysics, **74**, V17–V24.

Menon, A. K., and C. Elkan, 2011, Fast algorithms for approximating the singular value decomposition: Journal ACM Transactions on Knowledge Discovery from Data (TKDD), **5**, 1–36.

Neelamani, R., A. Baumstein, D. Gillard, M. Hadidi, and W. Soroka, 2008, Coherent and random noise attenuation using the curvelet transform: The Leading Edge, **27**, 240–248.

Oropeza, V., and M. Sacchi, 2011, Simultaneous seismic data denoising and reconstruction via multichannel singular spectrum analysis: Geophysics, **76**, V25–V32.

Protter, M., and M. Elad, 2009, Image sequence denoising via sparse and redundant representations: IEEE Trans Image Process, **18**, 27–35.

Qu, S., H. Zhou, Y. Chen, S. Yu, H. Zhang, J. Yuan, Y. Yang, and M. Qin, 2015, An effective method for reducing harmonic distortion in correlated vibroseis data: Journal of Applied Geophysics, **115**, 120–128.

Rubinstein, R., M. Zibulevsky, and M. Elad, 2008, Efficient implementation of the k-svd algorithm using batch orthogonal matching pursuit: Tech. Rep.

Sahoo, S. K., and A. Makur, 2013, Dictionary training for sparse representation as generalization of k-means clustering: IEEE Signal Processing Letters, **20**, 587–590.

Trickett, S., 2008, F-xy cadzow noise suppression: CSPG CSEG CWLS Convention, 303–306.

Wang, D., C. Liu, Y. Liu, and G. Liu, 2008, Application of wavelet transform based on lifting scheme and percentiles soft-threshold to elimination of seismic random noise: Progress in Geophysics (in Chinese), **23**, 1124–1130.

Wang, Y., 1999, Random noise attenuation using forward-backward linear prediction: J. Seism. Explor., **8**, 133–142.

Wu, J., R. Wang, Y. Chen, Y. Zhang, S. Gan, and C. Zhou, 2016, Multiples attenuation using shaping regularization with seislet domain sparsity constraint: Journal of Seismic Exploration, **25**, no. 1, 1–9.

Yu, S., J. Ma, X. Zhang, and M. Sacchi, 2015, Denoising and interpolation of high-dimensional seismic data by learning tight frame: Geophysics, **80**, V119–V132.

Zhang, C., Y. Li, H. Lin, and B. Yang, 2015, Signal preserving and seismic random noise attenuation by hurst exponent based time-frequency peak filtering: Geophysical Journal International, **203**, 901–909.

Zhang, D., Y. Chen, W. Huang, and S. Gan, 2016, Multi-step damped multichannel singular spectrum analysis for simultaneous reconstruction and denoising of 3d seismic data: Journal of Geophysics and Engineering, **13**, 704–720.

Zhuang, G., Y. Li, Y. Liu, H. Lin, H. Ma, and N. Wu, 2015, Varying-window-length TFPF in high-resolution radon domain for seismic random noise attenuation: IEEE Geoscience and Remote Sensing Letters, **12**, 404–408.

Zu, S., H. Zhou, Y. Chen, , S. Qu, X. Zou, H. Chen, and R. Liu, 2016, A periodically varying code for improving deblending of simultaneous sources in marine acquisition: Geophysics, **81**, V213–V225.