Interpolation and Denoising of High-Dimensional Seismic Data by Learning Tight Frame

Shiwei Yu[1] *and Jianwei Ma*[2] *and Xiaoqun Zhang*[3] *and Mauricio Sacchi*[4]

ABSTRACT

We present an extension of data driven tight frame (DDTF) method for three-dimensional and five-dimensional seismic data simultaneous denoising and interpolation. With tight frame assumption, DDTF significantly reduces the computation time for training dictionary, which makes it available for high dimension data processing. Raw data is first divided into small blocks to form training sets. Then we use DDTF to obtain an optimized sparse tight frame representation for raw data. We use a thresholding strategy for data denoising and iteration shrinkage/thresholding strategy for data simultaneous denoising and interpolation. The computational time and redundancy is controlled by patch overlap. Numerical results show that the proposed methodology is capable of recovering prestack seismic data under different SNR scenarios. Subtle features tend to be better preserved in the DDTF method in comparison to approaches based on Fourier, Wavelet and Curvelet representations or Block Matching method.

INTRODUCTION

Seismic data denoising and interpolation are critical preconditioning processes that are often needed prior to seismic migration and inversion. Reconstruction of seismic data has attracted much attention in recent years and it has become a standard tool for industrial

[1]Department of Mathematics, Harbin Institute of Technology, Harbin, China

[2]Department of Mathematics, Harbin Institute of Technology, Harbin, China

[3]Department of Mathematics, MOE-LSC and Institute of Natural Sciences, Shanghai Jiao Tong University, Shanghai, China

[4]Department of Physics, University of Alberta, Canada

seismic data processing flows. For example, it is often important to reconstruct seismic data (Sacchi and Liu, 2005; Hunt et al., 2010) prior to critical processes such Amplitude versus Offset and Amplitude versus Azimuth analysis for estimation of petrophysical parameters, fracture characterization. Noise and missing traces will introduce artifacts in subsequent process and hamper our ability to obtain reliable high resolution images of the subsurface. A large number of algorithms have been proposed for seismic data denoising, interpolation, and simultaneous denoising and interpolation. Most denoising methods fall into transform domain methods, including single-scale methods (Hunt and Kubler, 1984; Canales, 1984), multi-scale methods (Chanerley and Alexander, 2002; Duval and Tran, 2001), and multi-scale and multi-direction methods (Candes et al., 2006; Herrmann and Hennenfent, 2008). The basic idea is that if one finds a sparse representation for seismic data in a transform domain, it should be easy to distinguish data and noise by a simple thresholding strategy.

Seismic data interpolation methods can mainly be classified into three classes: wave equation methods based on the modeling of seismic wave propagating, prediction filtering methods based on the linearity assumption of seismic events, and transform domain methods based on sparsity of seismic data. Among the thirdclass, Fourier transform is the only method currently used in industrial applications (Zwartjes and Sacchi, 2007; Liu and Sacchi, 2004; Duijndam et al., 1999; Trad, 2009). Other Fourier methods, like the generalized methods such as FGFT (fast generalized Fourier transform) (Naghizadeh and Innanen, 2011) and ALFT (anti-leakage Fourier transform)(Xu et al., 2005) were also used for interpolation problem. Multi-scale and multi-direction transform such as curvelet transform (Naghizadeh and Sacchi, 2010) and shearlet transform (Hauser and Ma, 2012) are applied to seismic data interpolation because they are suitable sparse representation for seismic data. However, computation efficiency and fixed bases model restrict these methods from becoming common workhorses for industrial applications. Taking the slopes of seismic events into consideration, seislet transform (Fomel and Liu, 2010) is proposed for seismic data interpolation. Physical wavelet (Zhang and Ulrych, 2003) is a specially designed wavelet for seismic data by considering hyperbolic features of seismic events. Some seismic data denoising and interpolation methods borrow ideas from image processing community. For instance, assuming data redundancy, non-local means algorithm (David and Sacchi, 2012) denoises each sample within an image by utilizing other similar samples. Rank-reduced methods (Trickett et al., 2010; Kreimer and Sacchi, 2012; Gao et al., 2013; Ma, 2013; Yang et al., 2013) assume that the seismic data itself or constant-frequency slices of the fully sampled data can be reformulated to a low-rank matrix for reconstruction purpose.

Usually seismic data processing methods deal with the data as a whole volume. This is not an optimal strategy when seismic data profile exhibits repeating local texture feature. A patching strategy , i.e. splitting the volume into blocks, is proposed for seismic data denoising or interpolation. Block matching with 4D transform (BM4D) (Maggioni et al., 2013) is a typical volume patching method for 3D data denoising and reconstruction. It finds similar cubes among nonlocal area and a 4D transform is applied on this group simultaneously to suppress the noise. Patching strategy with adaptive dictionary learning methods, such as K-SVD (Aharon et al., 2006) are proposed for signal processing applications (Hu et al, 2012; Xing et al., 2012) and seismic data denoising (Bechouche and Ma, 2014). Such methods train bases from a large training set consist of data patches from a database or from the raw data itself, then use the trained bases to represent the original data. The basic idea why these methods may achieve better results than fixed basis transform is that a priori information is exploited. When training the dictionary, self-similarities information of the data are excavated. However, the K-SVD is time-consuming in dictionary training, as there exists large mount of patches for training and the components of the dictionary are updating individually. A more efficient method is needed for large scale data processing.

Recently, a new and more efficient dictionary learning method, named data driven tight frame (DDTF method), has been proposed for image denoising (Cai et al., 2014). "Data driven" means that the raw data are used for training the filter bank (i.e. basis or dictionary). Tight frame is a frame with a perfect reconstruction property, which is different from the K-SVD method. Also different from K-SVD method, DDTF method updates the whole filter bank by one SVD decomposition, leading to its efficiency. The 2D DDTF method application in seismic data interpolation is given in Liang et al., 2014, and state-of-the-art results for real seismic data are obtained, where DDTF shows its potential in seismic data processing. In image processing community, the DDTF method was recently extended to non-local version by training nonlocal samplers (Quan et al., 2013), which exploits the global self-recursive prior of image structures over the image. In (Hu et al, 2012), DDTF method was used for reconstruction of 3-D brain tissue from $50nm$ spacing section images, which are obtained by electron microscopy. They use $10nm$ spacing section images to train the 3-D filter bank, rather than the $50nm$ ones.

High-dimensional seismic exploration is necessary for complex underground structure. In this paper, an extension to 3D and 5D seismic data simultaneous denoising and interpolation by DDTF method is studied. The 5D data accounts for two spatial dimensions for receivers, two spatial dimensions for shots and another one dimension for time sampling. We will present comparison in 3D cases with methods such as Fourier transform, wavelet transform, curvelet transform and BM4D methods. In terms of the 5D case, we provide the DDTF method reconstructions and their differences to original data. The rest of this paper is arranged as follows: The second part introduces the theory of data driven tight frame for denoising and interpolation. The third part gives comparison of five methods for 3D seismic data denoising and interpolation. Discussion about the DDTF method algorithm and extension of the DDTF method for 5D seismic data comes in part four. A conclusion is made in the final part.

THEORY

Definition of Tight Frame

We first introduce basic definitions that will be utilized throughout this paper. A sequence $\{\phi_n\}_{n\in Z^+}$ in Hilbert space H is called a frame if

$$a\|f\|_2^2 \leq \sum_{n\in Z^+} |\langle \phi_n, f\rangle|^2 \leq b\|f\|_2^2, \forall f \in H \tag{1}$$

where $\langle\cdot,\cdot\rangle$ and $\|\cdot\|_2$ denote the usual inner product and 2-norm in the Hilbert space H. The parameters a and b are two positive constants. The linear operator $\{\phi_n\}_{n\in Z^+}$ is called a tight frame in H when $a = b = 1$. We define W as the operator that transforms f to $\{\langle f, \phi_n\rangle\}$, and its adjoint operator W^T that transforms $\{\langle f, \phi_n\rangle\}$ to f. The definition of a tight frame is equivalent to $W^TW = I$ (Quan et al., 2013), where I denotes the identity operator. For a tight frame, it is possible to obtain a perfect reconstruction property:

$$f = \sum_{n\in Z^+} \langle \phi_n, f\rangle \phi_n, \forall f \in H \tag{2}$$

If the operator W simultaneously satisfies $W^TW = I$ and $WW^T = I$, then the tight frame becomes an orthogonal basis. If the system is generated by shifts and dilations of a wavelet basis, then the tight frame is called wavelet tight frame. Considering the multi-resolution analysis ability of wavelet and the unitary extension principle, it is practical and convenient to construct a tight frame of wavelets. Construction method for 1D wavelet tight frame is illustrated in Cai et al., 2014 and Liang et al., 2014.

Higher-dimensional tight frame can be constructed by the tensor product of 1D tight frame. In Figure 1, we give an illustration for constructed 1D, 2D and 3D tight frames by Haar filters. Figure 1 (a) arranges the four 1D Haar filters as columns of a matrix. Figure 1 (b) is produced by 2D tensor product of 1D filter. One small block stands for one 2D filter. By 3D tensor product of 1D filter we get 3D filter shown in Figure 1 (c). We put the small filter cubes on a plane for clear viewing.

Data-Driven Tight Frame

We refer to fixed basis transform as an implicit dictionary as they are described by a linear transform rather than by an explicit matrix. The DDTF method is typically represented as an explicit matrix. A sparsity-promoting constrained optimization algorithm is applied to adapt the elements of the matrix to observations.

We introduce the main principle of DDTF method in the following lines. Details pertaining the method are provided in the Appendix. We first define the objective function

$$\operatorname*{argmin}_{v,W} \|v - Wg\|_2^2 + \lambda^2 \|v\|_0 \text{ s.t. } W^T W = I \tag{3}$$

we designate with the variable v the transform coefficients. In addition, we denote g the training data that we will use to estimate the frame. The first term in the cost function is the misfit that expresses our desire to find a basis that when operating on the data produces coefficient, that in lieu of the second term, are sparse. The constant λ adjusts the weight of the sparse constraint.

Last expression (3) can be solved via an iterative method that includes two stages. In the first stage we use sparsity promoting optimization to optimize for the coefficients v and, in the second stage, an update is performed to estimate the operator W.

The first stage, often referred as the sparse coding stage ,fixes the operator W and solves for sparse coefficients v via the classical expression

$$v^{(k)} := \operatorname*{argmin}_{v} \|v - W^{(k)} g\|_2^2 + \lambda^2 \|v\|_0$$

This optimization problem can be solved by a hard thresholding method (Cai et al., 2014).

The operator W updating stage assumes fixed coefficients v and solves the following problem

$$W^{(k+1)} := \operatorname*{argmin}_{W} \|v^{(k)} - Wg\|_2^2 \text{ s.t. } W^T W = I$$

It is shown in (Cai et al., 2014) that the SVD decomposition method can be used to solve for W. We stress, however, that unlike K-SVD method, DDTF method updates all columns of $W^{(k+1)}$ at once by one SVD decomposition.

It is easy to generalize the DDTF method cost function for considering a large mount of training data

$$\operatorname*{argmin}_{V,W} \|V - WG\|_F^2 + \lambda^2 \|V\|_0 \text{ s.t. } W^T W = I$$

where G and V are the matrix combinations of individual data and coefficient (details are presented in appendix). $\|\cdot\|_F^2$ means Frobenius norm, $\|\cdot\|_0$ here means the number of non-zeros in the matrix. The solution for $W^{(k+1)}$ with fixed V follows the following closed-form expression

$$W^{(k+1)} = XU^T$$

where r is the number of columns of matrix W. I, where U and X are the SVD decomposition of VG^T, i.e. $VG^T = UDX^T$.

We remark that a patching strategy is used to obtain the training set G (Ma, 2013) . As shown in Figure 2, we extract small 3D patches from the training data volume, the patches can be overlapped to generate sufficient training data. The small 3D patch is reshaped into a vector that becomes a column of G.

Different from Cai in (Cai et al., 2014), we set the tight frame exactly the same as the filter bank. Or we say the filters act as the columns of our tight frame transform matrix. Leaving out the shift versions of original filters, we get a purely orthogonal transform. This strategy gives us an accelerated algorithm and nearly no affect on the reconstruction result, because we use the overlapped patches of the original data.

The training data G and the initial filterbank are used as the input of our algorithm. In the iteration progress, the sparse coding step and the dictionary updating step are carried out seperately. As the iteration goes on, the trained dictionary will become the most optimized sparse tight frame for representing the original data.

A training example of a 3D seismic data volume is given in Figure 3. The data size is $256 \times 256 \times 16$ and the patch size is $7 \times 7 \times 7$. We use the tensor linear spline framelet (Daubechies et al., 2003) as initial wavelet, as shown in Figure 3 (b) (20 out of 343 filters are shown). Corresponding trained filter bank is shown in 3 (c), small structure characteristic is presented on the surface of the small blocks. The structures are similar with the structures of the original data patches. And the training results contain structures from low frequency to high frequency corresponding to the initial filter bank. We select one patch from original data and expand it on the initial filter bank and the trained filter bank separately, the coefficients distribution are displayed in Figure 3 (d). As we can see, the coefficients are much sparser distributed on trained filter bank than on initial filter bank, which means that the trained filter bank is a better choice for denoising or interpolation work.

Denoising and Interpolation

Having derived operator W from our two-stage iteration, the denoising and interpolation procedures are carried out by a sparsity-promoting method (Cai et al., 2014). We now minimizes the following cost function

$$\hat{g} = \underset{g}{\operatorname{argmin}} \|g_0 - Ag\|_2^2 + \|Wg\|_1 \tag{4}$$

where g stands for the data, g_0 stands for the raw/sampled data. If $A = I$, it stands for a denoising problem. If A is a sampling matrix, the above model stands for an interpolation problem. The model can be solved by iterative shrinkage/thresholding (IST):

$$g' = D_\lambda(g^{(k)})$$
$$g^{(k+1)} = \alpha_k(g^0 - Mg') + g'$$

where the $D_\lambda(g)$ is defined by

$$D_\lambda(g) = W^T T_\lambda W(g) \tag{5}$$

T_λ is a soft shrinkage operator, and λ is a thresholding parameter. The parameter α_k controls the feedback. If the sampled data is out of noise, we can set α_k as constant 1; If not, α_k should decrease from 1 to 0 along iteration. In this paper, since we mainly focus on how the sparse transform can gain the performance of seismic data processing, we just apply one of the simplest algorithms, the IST, to solve the model (4). Many advanced optimization methods such as accelerated first-order algorithms (Beck and Teboulle, 2009; Zhang et at., 2011) can be easily applied in the proposed framework.

NUMERICAL RESULTS

In this part, we use the trained filter bank for 3D seismic data denoising and interpolation. As comparison, we present denoising or interpolation results with Fourier transform, wavelet transform, curvelet transform and BM4D methods. Deonising is based on thresholding strategy. IST method is used for data reconstruction. We use patch size $r = 8$ for BM4D and DDTF method, all other parameters are individually optimized.

We use SNR value to judge the restoration result, whose definition is:

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\|X^*\|_F^2}{\|X^* - X\|_F^2}\right)$$

Figure 4(a)(b)(c) show original data, noised data and sub-sampled data respectively with noise of a synthetic 3D seismic data volume. The data size is $128 \times 128 \times 128$. Figure 5 shows the denoising results of Figure 4(b). Figure 5(a) shows denoising results with Fourier transform. Non-zeros appear at original zero value position as Fourier transform causes Gibbs phenonmenon when using thresholding method. Figure 5(b) presents results with Daubechies' Db5 wavelet transform(Daubenchies, 1992). Apparent block characteristic can be seen because wavelet transform basis stands for dot-like shape. Figure 5(c) stands for results with curvelet transform which achieves higher SNR value than the former two methods. Curvelet is a global multi-scale and multi-direction transform, so continuous curve characteristic can be preserved but pseudo-Gibbs phenomenon causes oscillation where it should be zeros originally. Figure 5(d) is from BM4D method and Figure 5(e) is obtained by DDTF method, they are patching based methods and provide us good visual results. DDTF method gets a higher SNR value than BM4D and other methods.

In order to make the denoising result more easily comparable, we extract one trace from reconstruction profile (red solid lines) and compare it with the corresponding trace from clean data (blue dot-dashed lines) in Figure 6. Figure 6(a-e) show the trace comparison of Fourier method, wavelet method, curvelet method, BM4D method, and DDTF method separately. It is clear that our method achieves higher denoising quality both on zeros and non-zeros parts of original data than the others.

We use a $8 \times 8 \times 8$ Haar wavelet tight frame as initial filter bank, and 16 out of total 512 filters are shown in Figure 7(a). The corresponding trained filter bank is shown in 7(b). We mention here that unlike the result of Figure 3, the training data we use here is the original noised data, not the clean data. Figure 7(c) shows the coefficient distribution comparison of the same original data cube expanding on initial filter bank and trained filter bank. The latter is sparser but the difference is not as obvious as in Figure 3(d), as the expansion is on the noised data rather than the clean data.

In Figure 8 we present an interpolation experiment on the model in Figure 4(c). Total 50% traces are missing on the offset plane and additional noise is added. Trained filter bank interpolation gets much higher SNR value than Fourier or wavelet method. Curvelet interpolation method gives a better looking result for the continuous characteristics but the SNR value is low than DDTF method as non-zeros emerge where it is zero originally. BM4D method fails in SNR compared to DDTF method because less repeat patterns exist when there is missing data. And also a single trace comparison result is presented in Figure 9.

Denoising results for model in Figure 3(a) is presented in Figure 10. With trained filter bank, we can get much better denoising result than with Fourier or wavelet method in visual. Also, we get a higher SNR than curvelet shrinkage or BM4D with proposed method, which shows that our method is also available for denoising seismic data with complex structure.

Also a reconstruction experiment for model in Figure 3(a) is carried out in Figure11. Same sampling and reconstruction methods are used as in Figure 8. With DDTF method we get the highest SNR, which means our method is suitable for complex seismic data interpolation. Part (time sample: 1-50, trace sample: 51-82) waveform display of one slice (center slice along y axis) is shown in Figure 12. The left of figure 12(a) comes from original data, and the right comes from the corrupted data. We want to see the reconstruction quality for weak energy part. As we can see, the available trace is almost submerged in noise. Fortunately, the data is a volume, so information from adjacent slices may be utilized. Figure 12(b)-(f) shows reconstruction results (left) and errors (right) by Fourier, wavelet, curvelet, BM4D and DDTF methods, respectively. Among the reconstruction algorithms, wavelet method can hardly reconstruct the data, Fourier method leaves too much error.

BM4D method obtains an over smoothed result, curvelet method joins the events which are separated originally. DDTF method obtains the best visual quality and SNR value.

We mention that for reconstruction algorithms with adaptive filter bank method, we use a zero-order interpolation data as the initial training data for DDTF method. It is reasonable to think that if the interpolation result is used as a new training set to train the filter bank again, a higher SNR value may be obtained. We try this procedure as an iteration, and it turns out that the SNR value gets no benefit after four iterations, so we can use the result after four iterations as the final result.

DISCUSSION AND EXTENSION

We should discuss the efficiency of 3D DDTF method. Computation and storage costs mainly come from the huge amount of training volumes. The first way to accelerate DDTF method is to parallelize the algorithm, which is natural for dictionary learning like method. Another way, one can decrease the overlap between the neighbor patches. We study how the amount of overlap between patches affects the final result and the processing time. The test is done on model in figure 4(c). In Figure 13, we can see that the elapsed time increases 'exponentially' over the amount of overlap or SNR, but the SNR value changes slowly as the number of overlapped pixels increases from 4 to 7. So we could select 5 or 6 as the overlapping degree for considerations of reconstruction quality and computational efficiency.

The interest of current industry lies in reconstructing the data in five dimensions. More dimensions make the modeling of the data more accurate and the interpolation more effective especially in the areas of complex structure. Because DDTF deals with data blocks as vectors, extension of the DDTF method to 5D is straightforward. Figure 14 shows an example on DDTF method interpolation of a synthetic 5D data. The data is modeled with the Matlab toolbox 'SeismicLab' by SAIG. The data is in 'middle point-offset' observation system with $dmx = 5m, dmy = 5m, dhx = 10m, dhy = 10m$. The Ricker wavelet central frequency is $40Hz$. Time sampling interval is $2ms$. The data size is $16 \times 16 \times 16 \times 16 \times 64$.

Patching size is 4, overlap = 3. We perform 5 training-interpolation cycles. Figure 14(a) shows the original data. (b) shows a corrupted data with 1/3 downsampling ratio (i.e., 2/3 trace missing), (c) shows the DDTF method interpolation and (d) shows difference between the interpolated results and the original data. Further, in Figure 15, we provide an example for simultaneous interpolation and denoising of the 5D data. These preliminary results demonstrate the promising performance of DDTF method up to five dimensions.

CONCLUSION

We extended the DDTF method for simultaneous denoising and interpolating high-dimensional seismic data. By taking advantage of the learning capabilities of the adaptive filter bank and the perfect reconstruction property of tight frame, we achieve higher SNR value with DDTF method compared with 3D Fourier, wavelet, curvelet and BM4D methods. We also extended the application of DDTF method for 5D seismic data. The preliminary results show promising performance of the 5D DDTF method. Future work would focus on how to incorporate more seismic structure features into the DDTF framework, i.e., how to utilize information in full-sampled dimension to reconstruct sub-sampled dimension.

ACKNOWLEDGEMENTS

This work is supported by NSFC (grant number: 91330108,41374121,61327013, 11101277, 91330102), the Fundamental Research Funds for the Central Universities (grant number: HIT.BRETIV.201314), and the Program for New Century Excellent Talents in University (grant number: NCET-11-0804). The authors thank Jingwei Liang for the discussion of the DDTF model and the coding.

APPENDIX

The symbol notations are:

$l, m, n, p, q, r, s, T_0 \in Z^+$: length of one dimension. Z^+ means positive integer.

$G^0 \in R^{l\times m\times n}$: the raw data of three dimension.

$g \in R^{p\times 1}(p \in Z^+)$: vector form of the raw training data. No matter how many dimensions the data has, we treat it as a vector which is formed by the column vectors or original data. For example, $f \in R^{r\times r\times r}$ is a small patch from G^0 (we suppose that the length of the three dimensions are the same), then we join the columns of f to form the vector $g \in R^{r^3\times 1}$.

$G \in R^{p\times q}$: combine the training data as a matrix, i.e. $G = [g_1, g_2, \cdots, g_q]$.

$W \in R^{p\times p}, \text{rank}(W) = p$: the filter bank (dictionary), its vectors is also named as atoms for a transform.

$v \in R^{p\times 1}$: the transform coefficient for g under W, i.e. $v = W^T g$.

$V \in R^{p\times q}$: combine the coefficients as a matrix, i.e. $V = [v_1, v_2, \cdots, v_q]$.

Now we give the detail how to update the filter bank. The optimization problem is as following:

$$W^{(k+1)} := \underset{W}{\text{argmin}} \|V - WG\|_2^2 \text{ s.t. } W^T W = I \tag{6}$$

The objective function can be reformulated as:

$$\begin{aligned}
\|V - WG\|_2^2 &= \sum_{n=1}^{q} \|v_n - A^T g_n\|_2^2 \\
&= \sum_{n=1}^{q} v_n^T v_n + g_n^T A A^T g_n - 2v_n^T A^T g_n \\
&= \sum_{n=1}^{q} v_n^T v_n + \frac{1}{r^2} g_n^T g_n - 2(Av_n)^T g_n \\
&= \text{Tr}(V^T V) + \frac{1}{r^2}\text{Tr}(G^T G) - 2\text{Tr}(AVG^T)
\end{aligned}$$

where $\text{Tr}(\cdot)$ stands for the trace of a matrix, and $A = W^T$. We remove the constant item and get an equivalent maximization problem as (6):

$$\underset{A}{\operatorname{argmax}} \operatorname{Tr}(AVG^T), \text{ s.t. } A^T A = I \tag{7}$$

The theorem in Zou et al. (2006) helps solve (7):

Theorem 0.1. *Let α and β be $m \times k$ matrices and β has rank k. Consider the constrained maximization problem:*

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmax}}\ Tr(\alpha^T \beta), \text{ s.t. } \alpha^T \alpha = I_k$$

Suppose the SVD of β is $\beta = UDV^T$, then $\hat{\alpha} = UV^T$.

By the introduced theorem, we can get the solution of (7):

$$A_* = XU^T$$

where we take the SVD decomposition of VG^T to get U and X, such that:

$$VG^T = UDX^T$$

REFERENCES

Aharon, M., M. Elad, and A. Bruckstein, 2006, K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation: IEEE Transcations on Signal Processing, **54**, 4311–4322.

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

Beck, A. and M. Teboulle, 2009, A fast iterative shrinkage-thresholding algorithm for linear inverse problems: SIAM Journal of Imaging Sciences, **2**, 183–202.

Cai, J., H. Ji, Z. Shen, and G. Ye, 2014, Data-driven tight frame construction and image denoising: Applied and Computational Harmonic Analysis, **37**, 89–105.

Canales, L., 1984, Random noise reduction: Presented at the 54th SEG Mtg, 525–572.

Candés, E., L. Demanet, D. Donoho, L. Ying, 2006, Fast discrete curvelet transforms: Multiscale Modeling and Simulation, **5**, 861–899.

Chanerley, A., and N. Alexander, 2002, An approach to seismic correction which includes wavelet de-noising: Proceedings of the Sixth Conference on Computational Structures Technology, 107–108.

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

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

David, B., and M. Sacchi, 2012, Denoising seismic data using the nonlocal means algorithm: Geophysics, **77**, 5–8.

Duijndam, A., M. Schonewille, and C. Hindriks, 1999, Reconstruction of band-limited signals, irregularly sampled along one spatial direction: Geophysics, **64**, 524–538.

Duval, L., and V. Tran, 2001, Compression denoising: using seismic compression for uncoherent noise removal: EAGE 63rd Conference and Technical ExhibitionAmsterdam, The Netherlands, 11–15.

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

Gao J., M. Sacchi, and X. Chen, 2013, Fast reduced-rank interpolation method for prestack seismic volumes that depend on four spatial dimensions: Geophysics, **78**, 21–30.

Hauser, S., and J. Ma, 2012, Seismic data reconstruction via shearlet-regularized directional inpainting: Preprint, T. U. Kaiserslautern, http://www.mathematik.unikl.de/uploads/tx sibibtex/seismic.pdf, accessed 15 May 2012.

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

Herrmann, F., and G. Hennenfent, 2008, Non-parametric seismic data recovery with curvelet frames: Geophysical Journal International, **173**, 233–248.

Hu, T., J. Nunez-Iglesias, S. Vitaladevuni, L. Scheffer, S. Xu, M. Bolorizadeh, H. Hess, R. Fetter, and D. Chklovskii, 2012, Super-resolution using sparse representations over learned dictionaries: Reconstruction of brain structure using electron microscopy.

Hunt, B., and O. Kubler, 1984, KarhunenLoéve multispectral image restorationPart 1: theory: IEEE Transactions on Acoustics, Speech, and Signal Processing **32**, 592–599.

Hunt, L., J. Downton, S. Reynolds, S. Hadley, D. Trad, and M. Hadley, 2010, The effect of interpolation on imaging and AVO: A Viking case study: Geophysics, **75**, WB265–WB274.

Kreimer, N., and M. Sacchi, 2012, A tensor higher-order singular value decomposition for prestack seismic data noise reduction and interpolation: Gephysics, **77**, V113–V122.

Liang, J., J. Ma, and X. Zhang, 2014, Seismic data restroration via data-driven tight frame: Geophysics, **79**, 65–74.

Liu, B., and M. Sacchi, 2004, Minimum weighted norm interpolation of seismic records: Geophysics, **69**, 1560–1568.

Ma, J., 2013, Three-dimensional irregular seismic data reconstruction via low-rank matrix completion: Geophysics, **78**, 181–192.

Maggioni, M., V. Katkovnik, K. Egiazarian, and A. Foi, 2013, Nonlocal Transform-Domain Filter for Volumetric Data Denoising and Reconstruction: IEEE Transactions on Image Processing, **22**, 119–133.

Naghizadeh, M., and M. Sacchi, 2010, Beyond alias hierarchical scale curvelet interpolation of regularly and irregularly sampled seismic data: Geophysics, **75**, 189–202.

Naghizadeh, M., and K. Innanen, 2011, Seismic data interpolation using a fast generalized Fourier transform: Geophysics, **76**, V1–V10.

Quan, Y., H. Ji, and Z. Shen, 2013, Data-driven non-local wavelet frame construction and image restoration: UCLA Computational and Applied Mathematics Report.

Sacchi, M., and B. Liu, 2005, Minimum weighted norm wavefield reconstruction for AVA imaging: Geophysical Prospecting, **53**, 787–801.

Trad, D., 2009, Five-dimensioinal interpolation: Recovering from acquisition constraints: Geophysics, **74**, V123–V132.

Trickett, S., L. Burroughs, A. Milton, L. Walton, and R. Dack, 2010, Rank-reduction-based trace interpolation: GeoCanada.

Xing Z., M. Zhou, A. Castrodad, G. Sapiro, and L. Carin, 2012, Dictionary learning for noisy and incomplete hyperspectral images: SIAM J. Imaging Sci., **5**, 33–56.

Xu, S., Y. Zhang, D. Pham, and G. Lambaré, 2005, Antileakage Fourier transform for seismic data regularization: Geophysics, **70**, V87–V95.

Yang, Y., J. Ma, and S. Osher, 2013, Seismic data reconstruction via matrix completion: Inverse Problem and Imaging, 7, **4**, 1379–1392.

Zhang, X., M. Burger, and S. Osher, 2011, A unified primal-dual algorithm framework based on bregman iteration: Journal of Scientific Computing, **46**, 20–46.

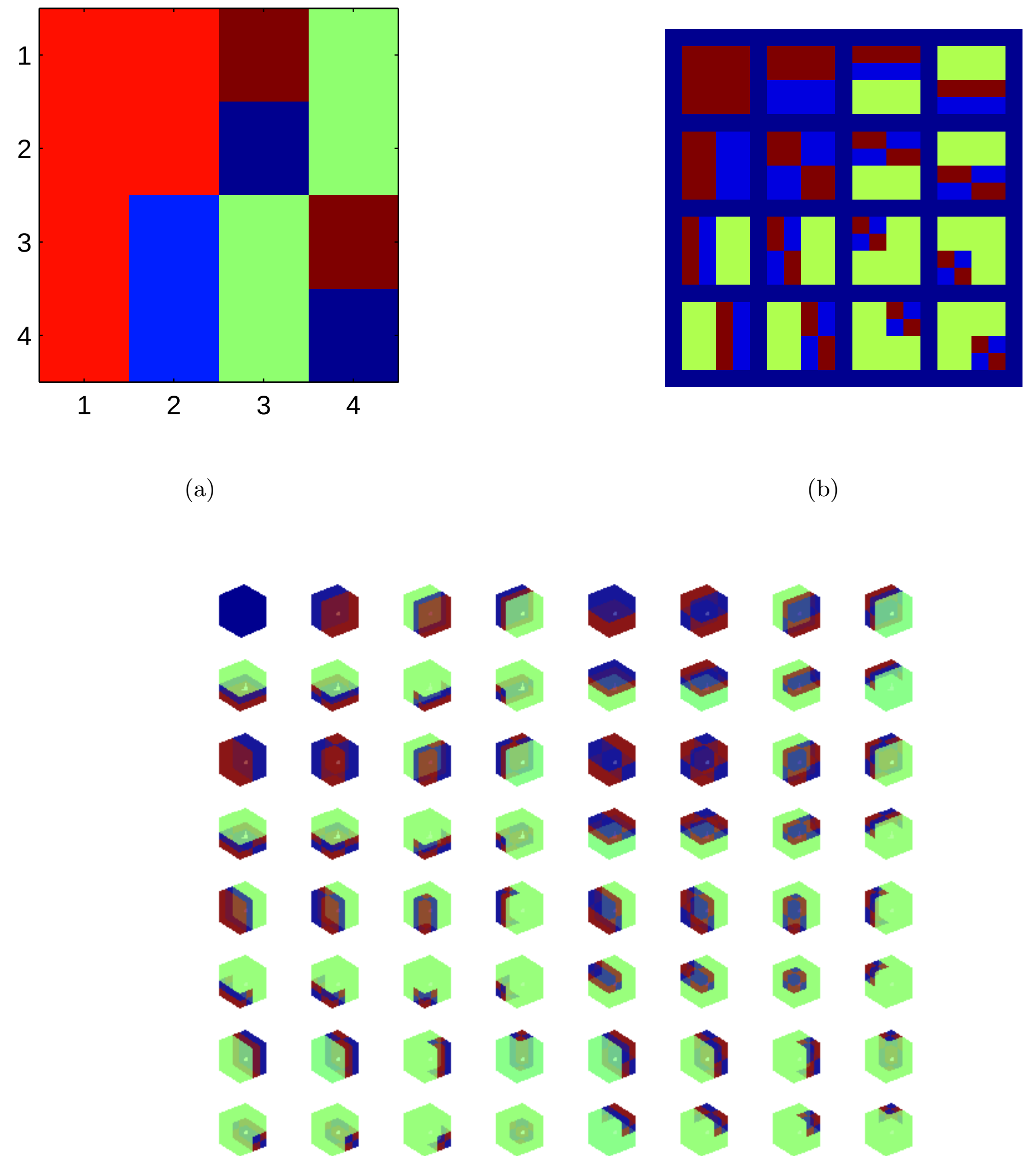
Zhang, R., and T. Ulrych, 2003, Physical wavelet frame denoisng: Geophysics, **68**, 225-231.

Zou, H., T. Hastie, and R. Tibshirani, 2006, Sparse principal component analysis: Journal of Computational and Graphical Statistics, **15**, 265–286.

Zwartjes, P., and M. Sacchi, 2007, Fourier reconstruction of nonuniformly sampled, aliased seismic data: Geophysics, **72**, V21–V32.

FIGURE/TABLE CAPTIONS

1. Figure 1: A demonstration for haar wavelet tight frame for 1D(a),2D(b) and 3D(c).
2. Figure 2: A demonstration for patching method of 3D data volume.
3. Figure 3: A demonstration for filter training initialized with $7 \times 7 \times 7$ sin wavelet.
4. Figure 4: 3D seismic data model.
5. Figure 5: Denoising results for 3D seismic data.
6. Figure 6: Single trace comparison and filter bank for denoising.
7. Figure 7: Filter bank for denoising.
8. Figure 8: Simultaneous interpolation and denoising for synthetic data.
9. Figure 9: Single trace comparison for interpolation.
10. Figure 10: Denoising results for real 3D seismic data.
11. Figure 11: Simultaneous interpolation and denoising for real data.
12. Figure 12: Partial waveform display of slices from figure 11.
13. Figure 13: SNR and elapsed time v.s. overlapped pixels.
14. Figure 14: 5D seismic data interpolation by the DDTF method.
15. Figure 15: Simultaneous interpolation and denoising for the 5D data by DDTF method.



(a) (b)

(c)

Figure 1: A demonstration for Haar wavelet tight frame for 1D(a),2D(b) and 3D(c).

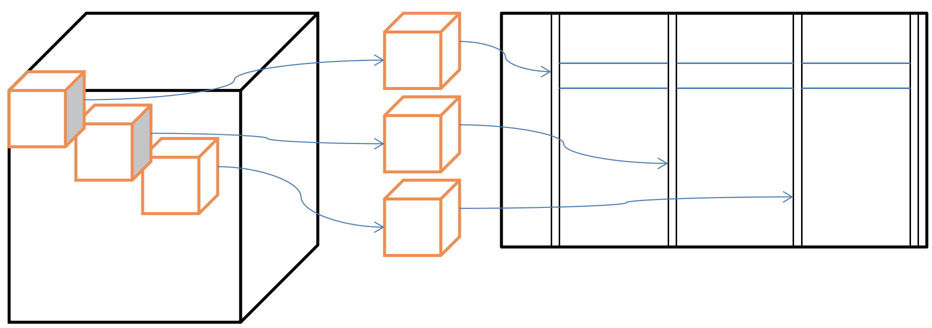
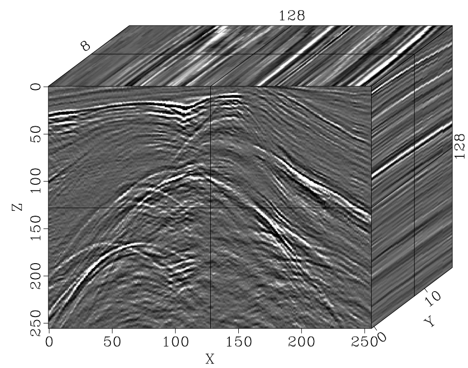
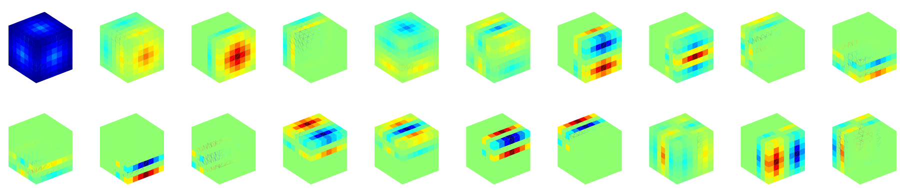


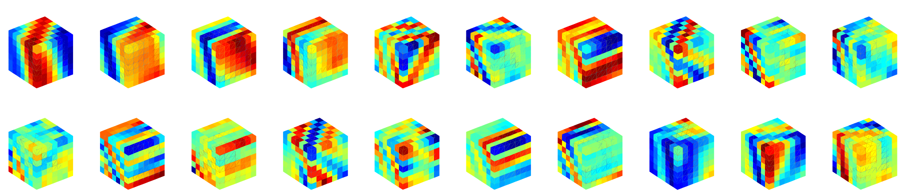
Figure 2: A demonstration for patching method of 3D data volume.



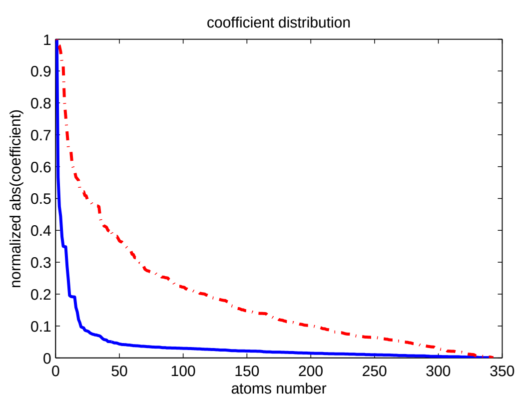
(a)



(b)

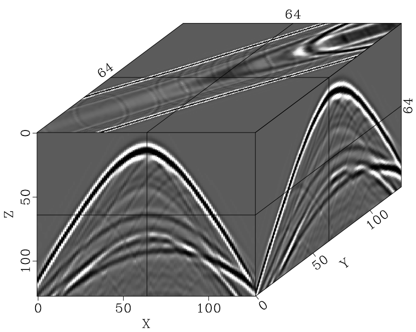


(c)

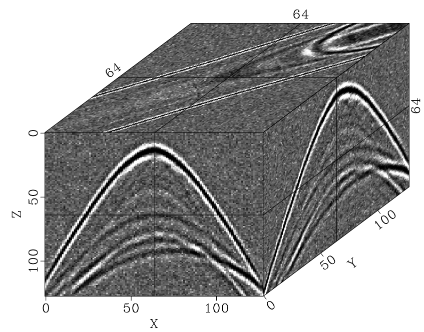


(d)

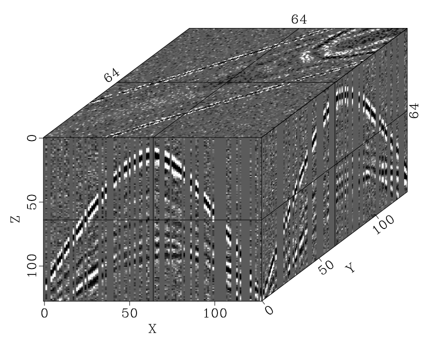
Figure 3: A demonstration for filter training initialized with $7 \times 7 \times 7$ tensor linear spline framelet. (a) 3D synthetic cube of seismic data. (b) Initial filter bank. (c) Trained filter bank. (d) Coefficient distribution of one block of (a) on trained filter bank (blue solid line) and initial filter bank (red dash-dot line).



(a)



(b)



(c)

Figure 4: 3D seismic data model. (a) Original data; (b) Data with noise; (c) Data with missing traces and noise.

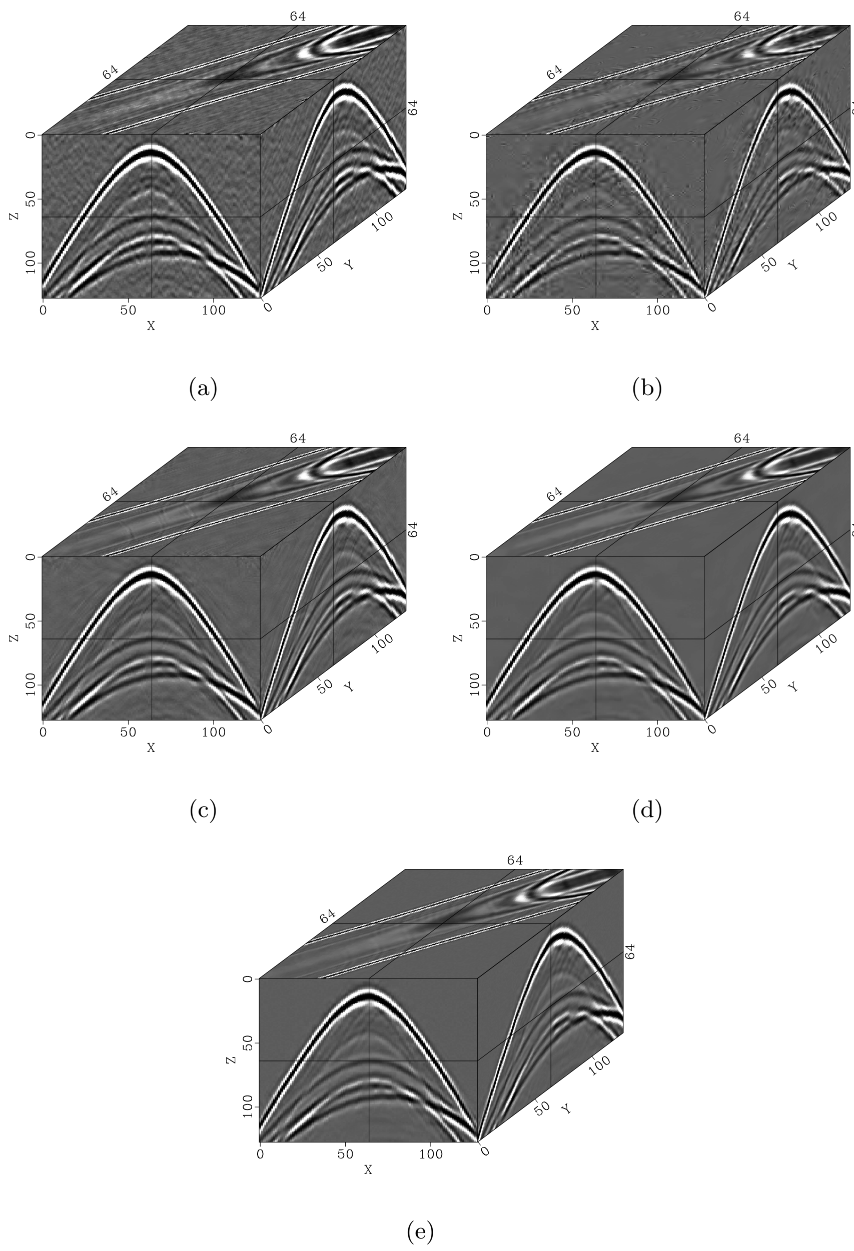


Figure 5: Denoising results for 3D seismic data. (a) Denoising result by Fourier transform, SNR=12.94. (b) By db5 wavelet, SNR=10.53. (c) By curvelet, SNR=16.91. (d) By BM4D, SNR=18.69 (e) By DDTF method, SNR=19.23.

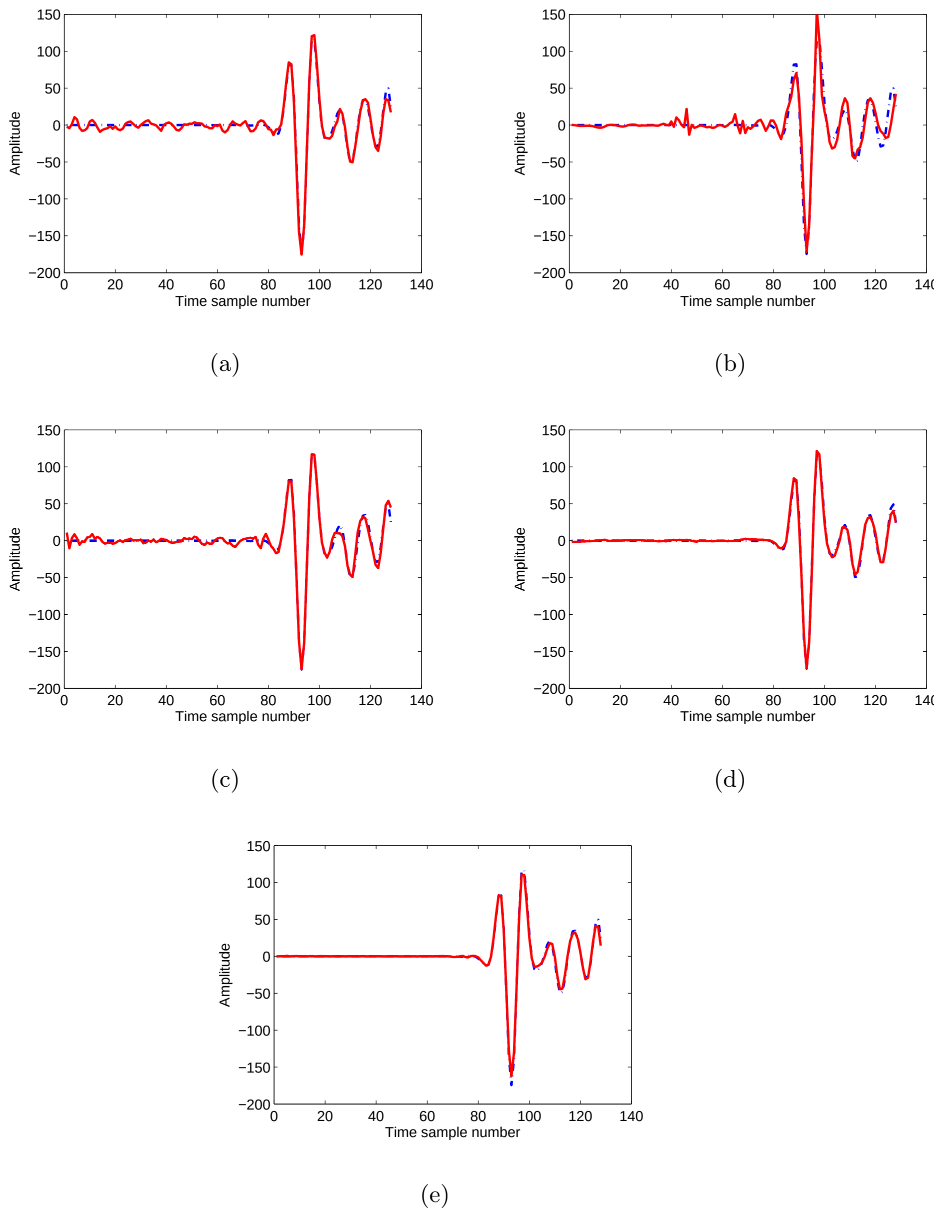
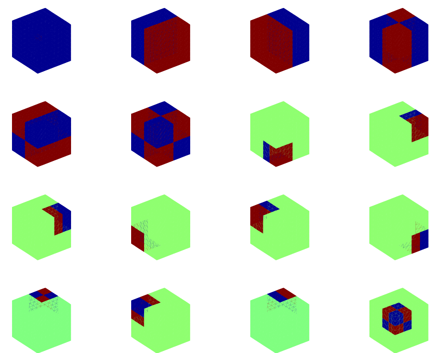
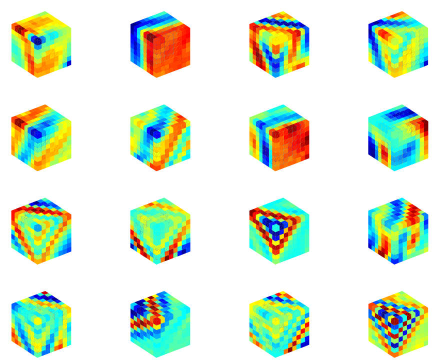


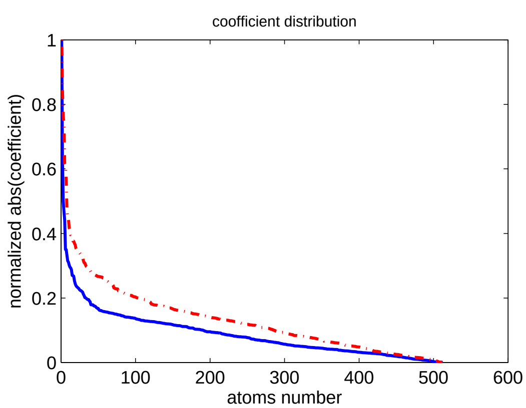
Figure 6: Single trace comparison for denoising. (a) Fourier transform; (b) wavelet transform; (c) curvelet transform; (d) BM4D; (e) DDTF method;



(a)

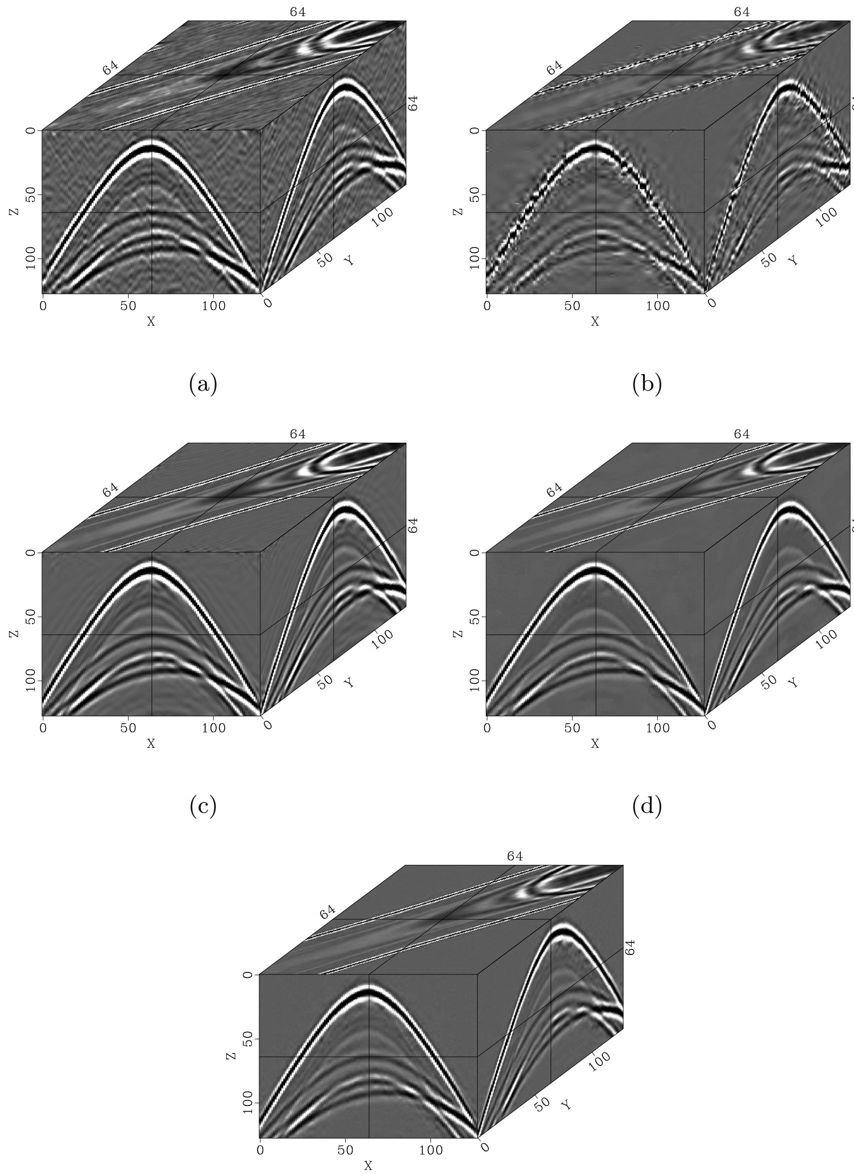


(b)



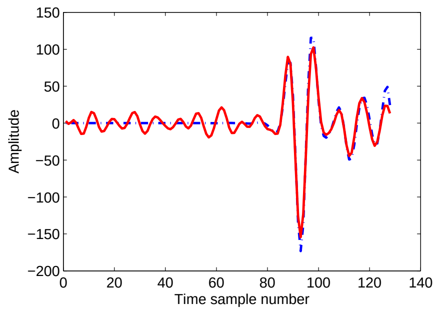
(c)

Figure 7: Filter bank for denoising. (a) Initial filter bank for DDTF method; (b) Trained filter bank for DDTF method; (c) Coefficients distribution comparison.

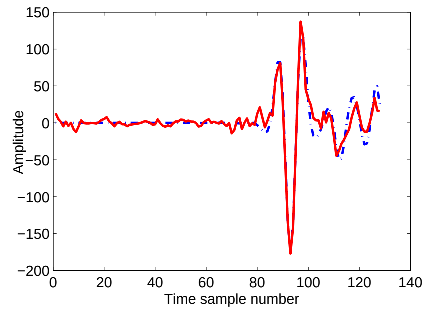


(a) (b) (c) (d) (e)

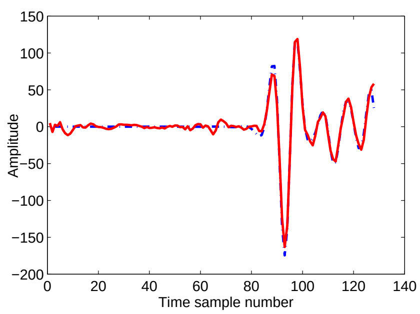
Figure 8: Simultaneous interpolation and denoising for the synthetic data. (a) Fourier transform. SNR = 9.42; (b) Db5 wavelet transform. SNR = 3.73; (c) curvelet transform. SNR = 13.88; (d) BM4D. SNR = 15.37. (e) DDTF method. SNR = 16.70.



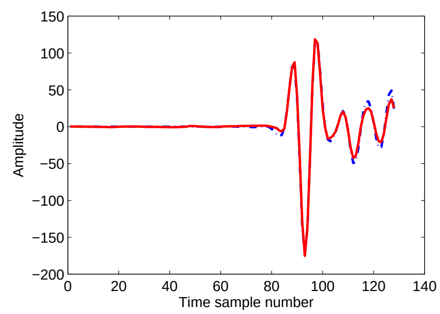
(a)



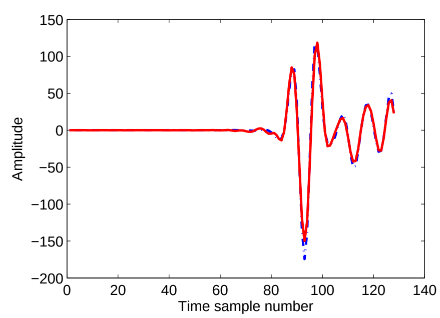
(b)



(c)

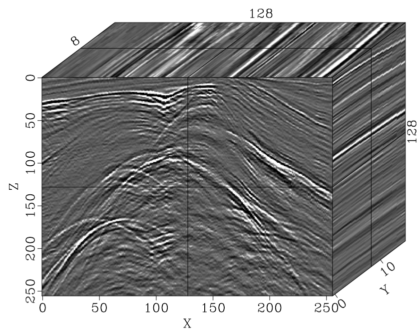


(d)

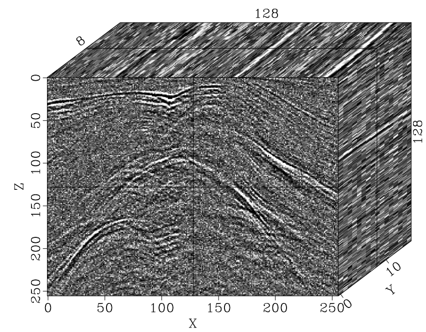


(e)

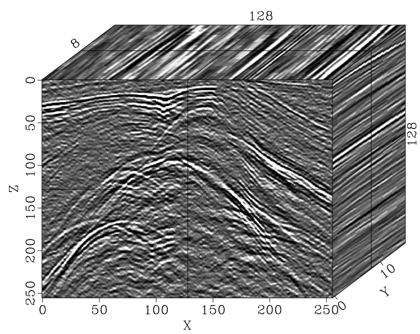
Figure 9: Single trace comparison for interpolation. (a) Fourier transform; (b) wavelet transform; (c) curvelet transform; (d) BM4D; (e) DDTF method;



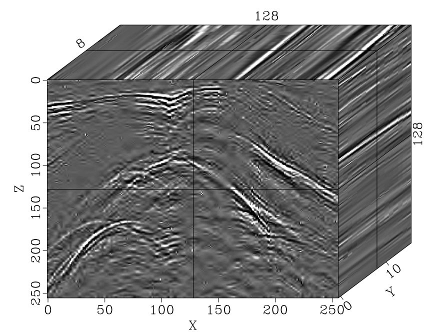
(a)



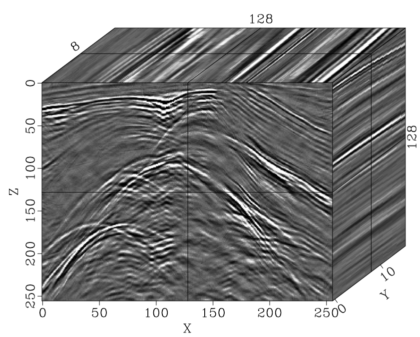
(b)



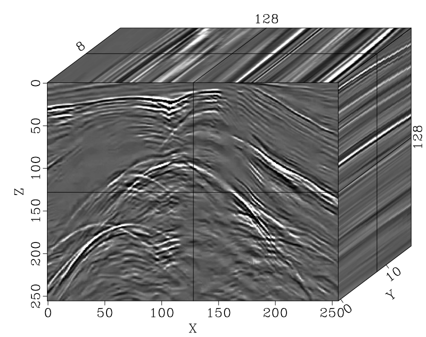
(c)



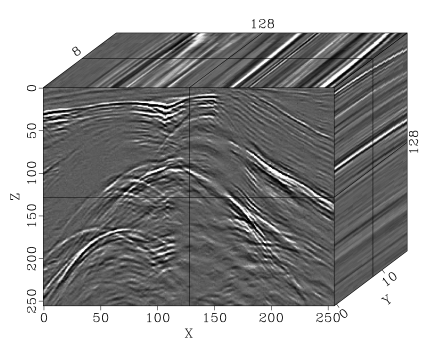
(d)



(e)

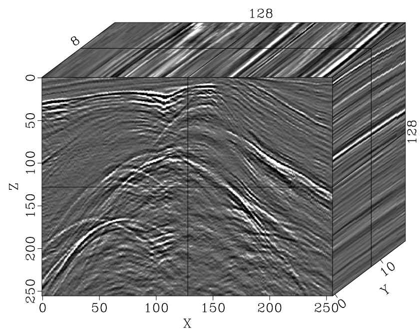


(f)

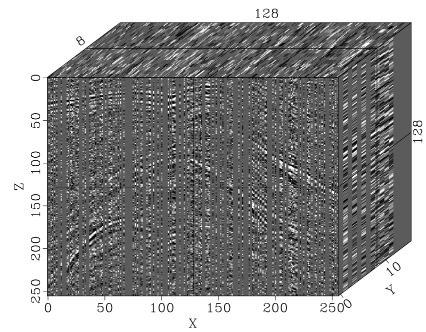


(g)

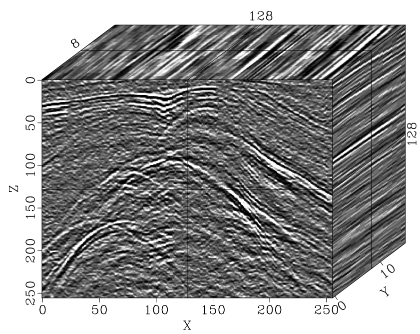
Figure 10: Denosing results for 3D seismic data. (a) Original data. (b) Noise data. (c) Denoising result by Fourier transform, SNR=4.99. (d) Db5 wavelet, SNR=4.20. (e) curvelet, SNR=7.17. (f) BM4D, SNR=8.79. (g) DDTF method, SNR=8.83.



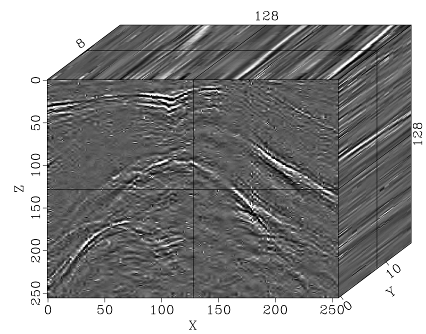
(a)



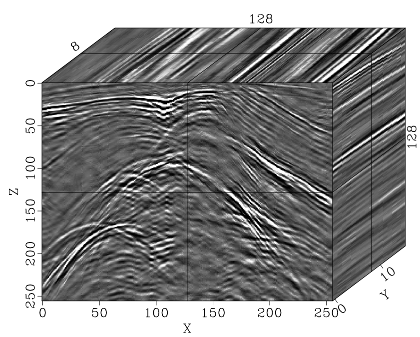
(b)



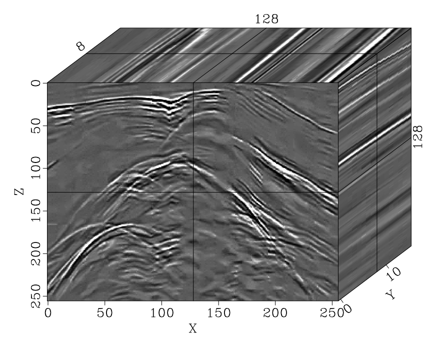
(c)



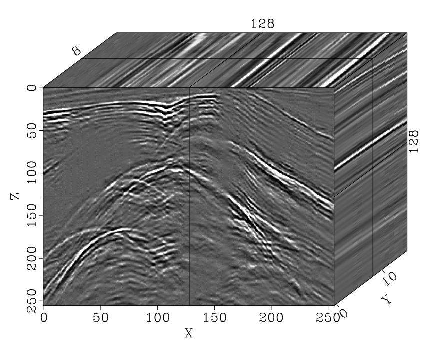
(d)



(e)

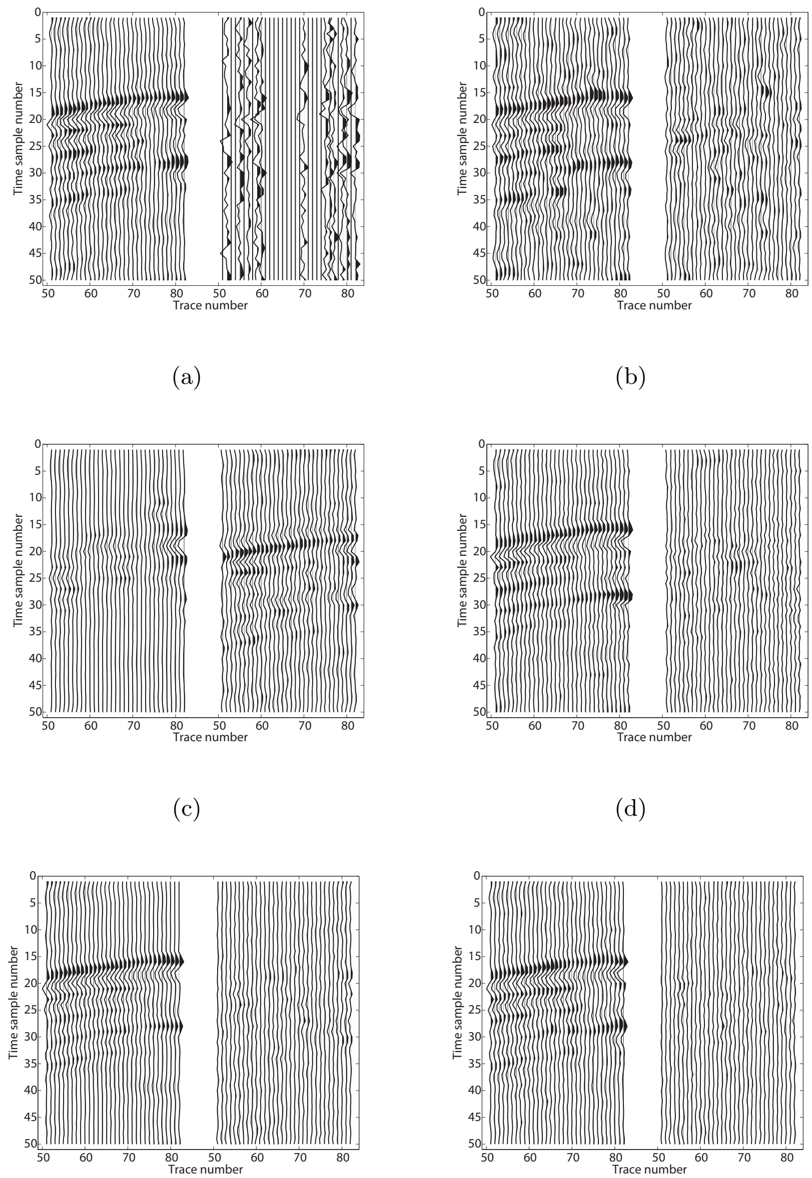


(f)



(g)

Figure 11: Simultaneous interpolation and denoising for the real data. (a) Original data. (b) Data with missing traces and noise. (c) Result by Fourier transform. SNR=3.35. (d) Db5 wavelet. SNR=2.04. (e) curvelet. SNR=6.11. (f) BM4D. SNR=6.81. (g) DDTF



(a) (b)

(c) (d)

(e) (f)

Figure 12: Partial waveform display of slices from figure 11. (a) Left part: original data, right part: corrupted data. (b)-(f) Reconstruction (left) and errors (right) by Fourier,wavelet,curvelet,BM4D and DDTF method, respectively.

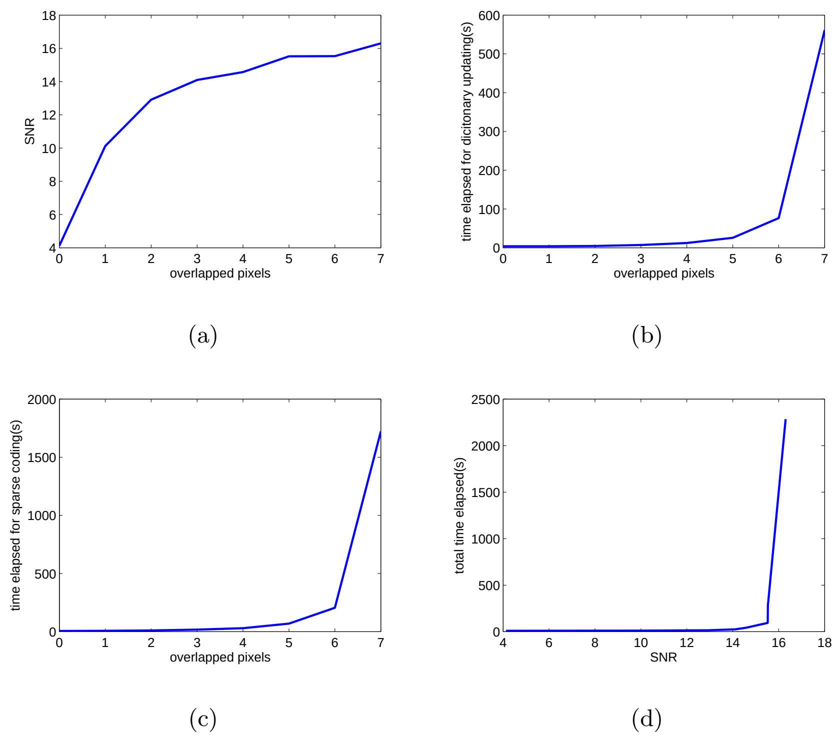


Figure 13: SNR and elapsed time v.s. overlapped pixels. (a) SNR v.s. overlapped pixels. (b) Elapsed time for sparse coding v.s. overlapped pixels. (c) Elapsed time for dictionary updating v.s. overlapped pixels. (d) SNR v.s. total elapsed time.

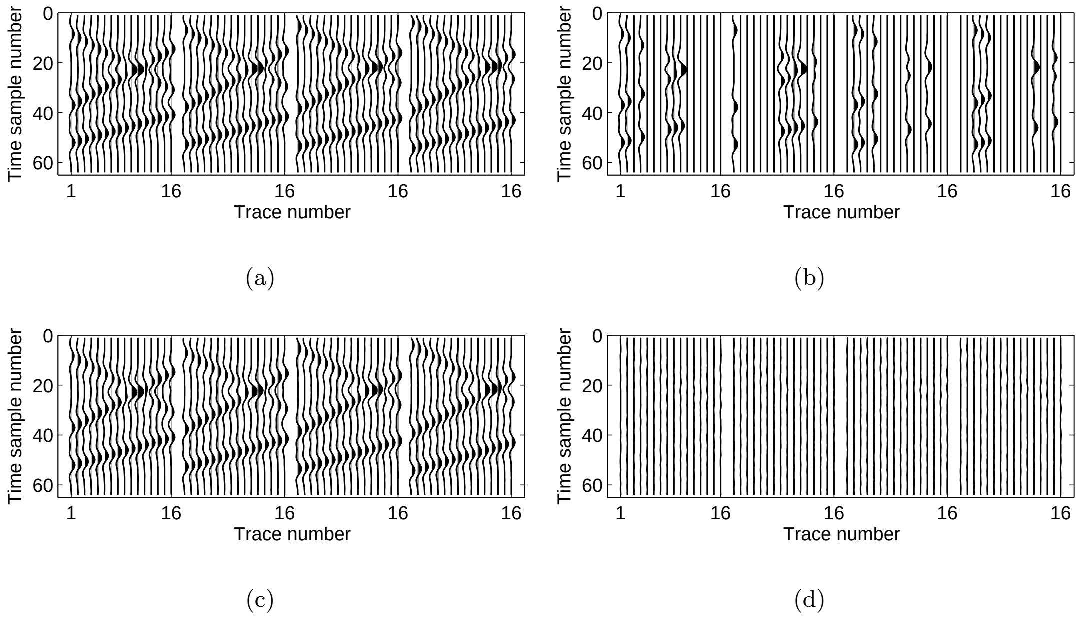


Figure 14: 5D seismic data interpolation by the DDTF method. (a) Original clean data. (b) Subsampled data with 1/2 sampled ratio. (c) Interpolation by DDTF method, SNR=22.96. (d) Difference between the interpolated result and original clean data.

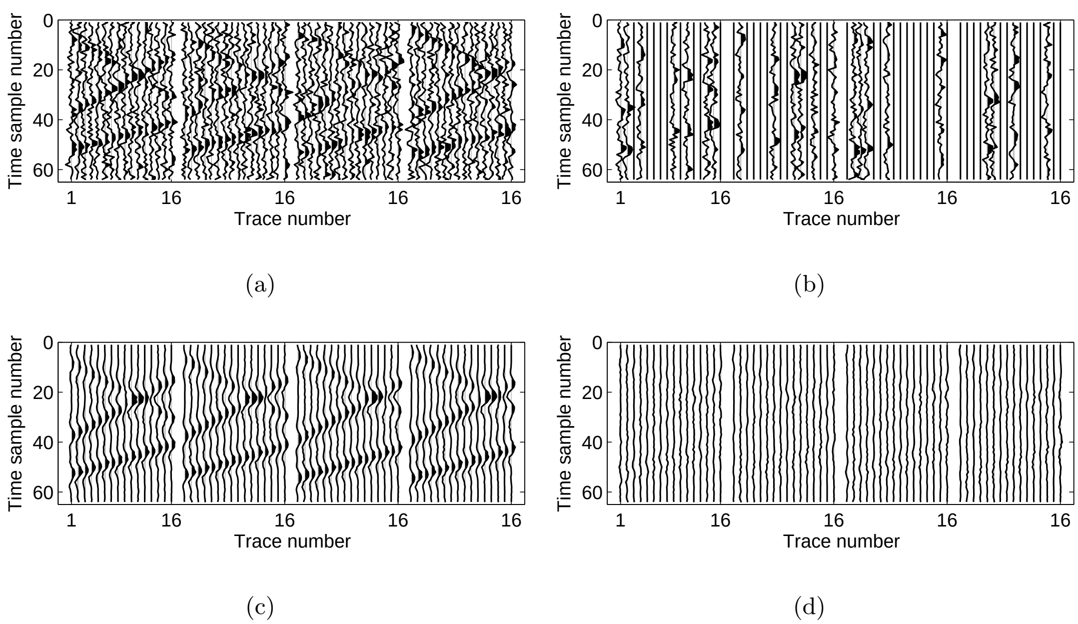


Figure 15: Simultaneous interpolation and denoising for the 5D data by DDTF method. (a) Noisy data. (b) Subsampled data with 1/3 sampling ratio for (a). (c) Interpolation by DDTF method, SNR=11.20. (d) Difference between the interpolated result and original clean data.