Convolutional Sparse Coding for Noise Attenuation of Seismic Data

Zhaolun Liu, Kai Lu and Xiaodan Ge, King Abdullah University of Science and Technology.*

SUMMARY

Convolutional sparse coding (CSC) is proposed to attenuate noise for seismic data. CSC gives a data-driven set of basis functions whose coefficients form a sparse distribution. The noise attenuation method by CSC can be divided into the training phase and the denoising phase. The seismic data with a relatively high signal-to-noise ratio are chosen for training to get the learned basis functions. Then we use all (or a subset) of the basis functions to attenuate the random or coherent noise in the seismic data. Numerical experiments on synthetic and field data indicate that the proposed method achieves good performance for denoising random and coherent noise and separating the ground roll.

INTRODUCTION

Seismic data may suffer from different sources of noise during field acquisition, which can degenerate the subsurface imaging quality of migration. Thus, noise attenuation is one of the key components of seismic data processing.

Traditional seismic denoise methods exploit the different characteristics of the noise and the seismic data in a transform domain, in which the signal and noise have different characteristics. Chanerley and Alexander (2002) used the stationary wavelet transform as an alternative to band-pass filtering for denoising. The curvelet transform is used by Hennenfent and Herrmann (2006) to attenuate both random and coherent noise in seismic data. Ibrahim and Sacchi (2014) use the hyperbolic Radon transform for deblending seismic data obtained from a survey with blended sources. The seislet transform is used for seismic denoising by Chen (2016).

The analytic transform mentioned above is a model-driven process based on a formulated mathematical model of the data. The performance may not be satisfactory due to the inability to adapt to changing data structures (Zhu et al., 2015). Alternatively, the sparse dictionary learning method is a data-driven process, which learns its dictionary so that complicated data can be represented by a sparse set of weighted basis functions. Kaplan et al. (2009) used the sparse-coding algorithm to attenuate both coherent and incoherent noise. Chen et al. (2016) 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.

Most of the sparse coding (SC) denoising methods partition the seismic data into overlapped patches, and process each patch separately. These methods, however, ignore the consistency of structures in overlapped patches, for which the learned features often contain shifted versions of the same features so that latent structures of the underlying signal may be lost when dividing it into small patches (Bristow et al., 2013; Heide et al., 2015). In this report, we propose a convolutional sparse coding (CSC) denoising method to address the consistency issue. As opposed to SC, CSC operates on whole seismic images, thereby seamlessly capturing the correlation between local neighborhoods.

We first introduce the theory of CSC and the workflow of noise attenuation by CSC. Then we apply CSC denoising method to synthetic data and field records. Numerical results indicate that the proposed method achieves good denoising performance. Finally, we give the summary.

THEORY

The convolutional sparse coding problem can be defined as finding the optimal $\mathbf{d}$ and $\mathbf{z}$ that minimize the following objective function (Heide et al., 2015):

$$\arg\min_{\mathbf{d},\mathbf{z}} \frac{1}{2}\|\mathbf{x}-\mathbf{M}\sum_{k=1}^{K}\mathbf{d}_k*\mathbf{z}_k\|_2^2+\beta\sum_{k=1}^{K}\|\mathbf{z}_k\|_1+\sum_{k=1}^{K}\mathrm{ind}_C(\mathbf{d}_k), \tag{1}$$

where $\mathbf{x}$ is an $m\times n$ image in vector form, $\mathbf{d}_k$ refers to the k-th $d\times d$ filter in vector form, $\mathbf{z}_k$ is vector of sparse coefficients with size $(m+d-1)\times(n+d-1)$, β controls the l_1 penalty, and $*$ denotes the 2D convolution operator. $\mathbf{M}$ is a binary diagonal matrix that masks out the boundaries of the padded estimation $\sum_{k=1}^{K}\mathbf{d}_k*\mathbf{z}_k$. The term $\mathrm{ind}_C(\cdot)$ is an indicator function:

$$\mathrm{ind}_C(\mathbf{d})=\begin{cases}0 & \mathbf{d}\in C\\ +\infty & \mathbf{d}\notin C,\end{cases} \tag{2}$$

which is defined on the convex set of the constraints $C=\{\mathbf{d}\mid\|\mathbf{d}\|_2^2\le 1\}$. Equation 1 can be expressed as the following sum of functions

$$\arg\min_{\mathbf{d},\mathbf{z}} f_1(\mathbf{D}\mathbf{z})+\sum_{k=1}^{K}(f_2(\mathbf{z}_k)+f_3(\mathbf{d}_k)), \tag{3}$$

where,

$$f_1(\mathbf{v})=\frac{1}{2}\|\mathbf{x}-\mathbf{M}\mathbf{v}\|_2^2,\ f_2(\mathbf{v})=\beta\|\mathbf{z}_k\|_1,\ f_3(\mathbf{v})=\mathrm{ind}_C(\mathbf{v}). \tag{4}$$

Here, $\mathbf{z}=[\mathbf{z}_1^T,\cdots,\mathbf{z}_K^T]^T$ is the coefficient matrix. $\mathbf{D}=[\mathbf{D}_1,\cdots,\mathbf{D}_K]$ is a concatenation of Toeplitz matrices, each one representing a convolution with respect to the filter $\mathbf{d}_k$. Equation 3 is a sum of functions f_i, which are simple to optimize individually. However, computing their sum is challenging. Following Heide et al. (2015), equation 3 is a bi-convex problem for $\mathbf{z}$ (or $\mathbf{d}$) when $\mathbf{d}$ (or $\mathbf{z}$) is fixed. So, we can use the alternating coordinate descent method to solve it. First, compute the filter update:

$$\arg\min_{\mathbf{d}} f_1(\mathbf{Z}\mathbf{d})+\sum_{k=1}^{K}f_3(\mathbf{d}_k). \tag{5}$$

Then do the coefficient update:

$$\arg\min_{\mathbf{z}} f_1(\mathbf{Dz}) + \sum_{k=1}^{K} f_2(\mathbf{z}_k). \tag{6}$$

Repeat the above two steps until there is no more progress in both directions.

Generalization of the Objective Function

The above two subproblems have the same format and so we can define their generalized form as

$$f(\mathbf{Ku}) = \sum_{i=1}^{I} f_i(\mathbf{K}_i\mathbf{u}), \tag{7}$$

where $\mathbf{K} = [\mathbf{K}_1, \mathbf{K}_2, \cdots, \mathbf{K}_I]^T$. $\mathbf{K}_i$ are arbitrary matrices, and I is the number of functions. Here, $\mathbf{u}$ can be $\mathbf{d}$ or $\mathbf{z}$. For example, if $\mathbf{K}_1 = \mathbf{D}$, $\mathbf{K}_2\mathbf{u} = \mathbf{z}_1 \cdots$, the problem for the coefficient update can be written as:

$$\arg\min_{\mathbf{u}} f(\mathbf{Ku}) = \arg\min_{\mathbf{z}} f_1(\mathbf{Dz}) + \sum_{k=1}^{K} f_2(\mathbf{z}_k). \tag{8}$$

Heide et al. (2015) use the alternating direction method of multipliers (ADMM) (Boyd et al., 2011) to solve $\mathbf{u}$ in equation 8. We will see that the resulting minimization by ADMM becomes separable for all the f_i. For example, the following problem is the same as the problem in equation 8:

$$\arg\min_{\mathbf{u}} h(\mathbf{u}) + f(\mathbf{y}) \quad \text{subject to} \quad \mathbf{Ku} = \mathbf{y}, \tag{9}$$

where $h(\mathbf{u}) = 0$. Its augmented Lagrangian can be written as:

$$L_\rho(\mathbf{u}, \mathbf{y}, \alpha) = h(\mathbf{u}) + f(\mathbf{y}) + \alpha^T(\mathbf{Ku} - \mathbf{y}) + (\rho/2)\|\mathbf{Ku} - \mathbf{y}\|_2^2.$$

The scaled form of the augmented Lagrangian is:

$$L_\rho(\mathbf{u}, \mathbf{y}, \alpha) = h(\mathbf{u}) + f(\mathbf{y}) + (\rho/2)\|\mathbf{Ku} - \mathbf{y} + \lambda\|_\mathbf{2}^\mathbf{2}, \tag{10}$$

where $\lambda = (\mathbf{1}/\rho)\alpha$. The ADMM of equation 10 is described by the following three steps.

- $\mathbf{u}$ update:

$$\begin{aligned} \mathbf{u}^{k+1} &= \arg\min_{\mathbf{u}} L_\rho(\mathbf{u}^k, \mathbf{y}^k, \lambda^k), \\ &= \arg\min_{\mathbf{u}} \|\mathbf{Ku}^k - \mathbf{y}^k + \lambda^k\|_2^2. \end{aligned} \tag{11}$$

- $\mathbf{y}$ update:

$$\begin{aligned} \mathbf{y}^{k+1} &= \arg\min_{\mathbf{y}} L_\rho(\mathbf{u}^{k+1}, \mathbf{y}^k, \lambda^k), \\ &= \arg\min_{\mathbf{y}} \rho\left\{ f(\mathbf{y}^k)/\rho + \frac{1}{2}\|\mathbf{Ku}^{k+1} - \mathbf{y}^k + \lambda^k\|_2^2 \right\}. \end{aligned} \tag{12}$$

- error update:

$$\begin{aligned} \lambda^{k+1} &= \arg\max_{\lambda} L_\rho(\mathbf{u}^{k+1}, \mathbf{y}^{k+1}, \lambda^k), \\ &= \lambda^k + (\mathbf{Ku}^{k+1} - \mathbf{y}^{k+1}). \end{aligned} \tag{13}$$

Because f is a sum of f_i, equation 12 can be separated as:

$$\mathbf{y}_i^{k+1} = \arg\min_{\mathbf{y}_i} f_i(\mathbf{y}_i^k)/\rho + \frac{1}{2}\|\mathbf{K}_i\mathbf{u}_i^{k+1} - \mathbf{y}_i^k + \lambda_i^k\|_2^2, \quad \text{for all } i \in \{1, \cdots, I\}. \tag{14}$$

Appendix A shows how to solve the problems defined in equations 11 and 14.

Workflow

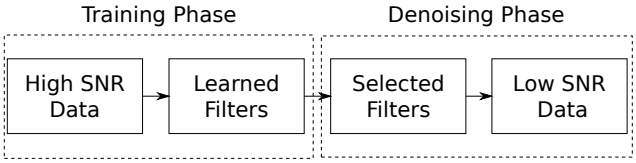

Figure 1: Workflow of CSC for noise attenuation.

The workflow of CSC for noise attenuation is shown in Figure 1. It includes the following two steps:

- Training phase: solve for the filter $\mathbf{d}_k$ and the coefficient vector $\mathbf{z}_k$. The training data are chosen with a relatively high signal-to-noise ratio (SNR). The filter size should be larger than one wavelength.
- Denoising phase: solve for the coefficient vector $\mathbf{z}_k$ with the knowledge of the learned vector $\mathbf{d}_k$. Change the value of the coefficient β for the l_1 penalty function in order to control the degree of noise removal. For coherent noise, remove some of the learned filters that are more indicative of noise than signal, and the remaining filters are used for denoising.

NUMERICAL EXAMPLES

The performance of CSC for denoising seismic data is demonstrated with two examples: (1) synthetic data from the Marmousi model and (2) field data from Saudi Arabia. The synthetic data example tests the performance of CSC in removing random noise in seismic data, and the field data example demonstrates the removal of coherent noise. The SNR used in our synthetic test is defined as follows:

$$\text{SNR} = 10\log_{10}\left(\frac{\|X_{signal}\|_2^2}{\|X_{noise}\|_2^2}\right) \tag{15}$$

where X_{signal} and X_{noise} denote the signal and noise, respectively.

Synthetic Test

The first example is a 2-D synthetic dataset calculated from the Marmousi model. A part of the second common shot gather (CSG) is shown in Figure 2a, and its size is 181×100. Five CSGs with high SNR are selected for training. There are 50 learned filters (see Figure 3) and the filter size is 55×55. Assume that the second CSG is contaminated with some noise (see Figure 2b) and its SNR is -1.76. The denoised CSG is

shown in Figure 2c and its SNR increases to 14.5. Figure 2d show the residual of the denoised data. We can change the coefficient β of the l_1 penalty function to control the noise level. Figures 4a and 4b show the denoised results by setting β as 0.2 and 5, respectively. It is observed that more noise is removed with larger values of β. But it may also hurt some useful signals, which can be seen in the comparison of the residuals in Figures 4c and 4d.

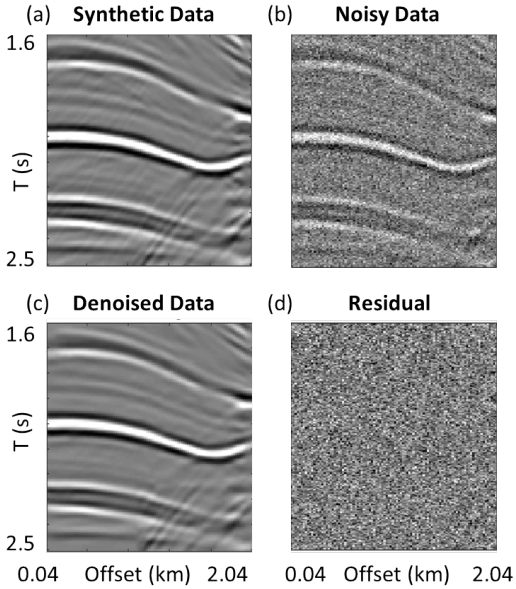

Figure 2: (a) Synthetic data; (b) noisy data; (c) denoised data; (d) residual between (b) and (c).

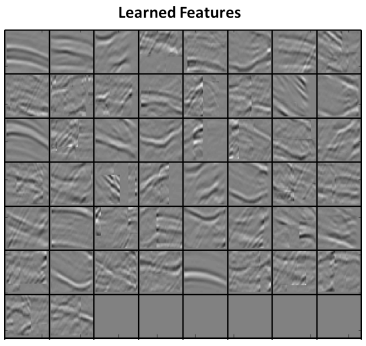

Figure 3: Learned filters

Field Data Test

Seismic data are recorded over the Qademah fault system, approximately 30 km north of the KAUST campus. The dataset is first filtered by a bandpass filter with a frequency range between 10 Hz to 60 Hz. Ten CSGs are chosen for training and one of CSGs is shown in Figure 5a. Its offset range is from 0.2 km to 1.3 km. The surface waves are muted out in the training set. There are 30 21 $\times$ 21 learned filters displayed in Figure 5b. Some features of the coherent noise are learned from the training data set as indicated by the red boxes in Figure 5b.

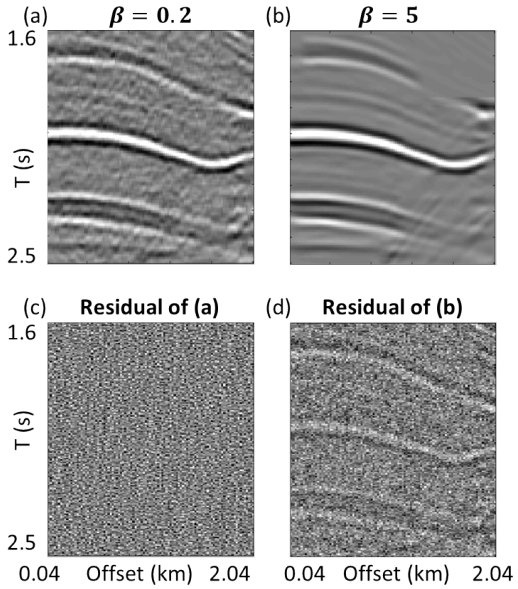

Figure 4: Denoised data by setting $\beta =$ (a) 0.2 and (b) 5 and their corresponding residuals (c) and (d).

We select the first CSG (see Figure 6a) for denoising. Its offset range is from 0.8 km to 1.9 km. β is set to 0.8 during the denoising phase. Using all the learned filters in Figure 5b for denoising gives the denoised result displayed in Figure 6b. Its corresponding residual is shown in Figure 7a. There is still coherent noise in the area indicated by the red box in Figure 6b. The reason is that the learned filters include the features with coherent noise.

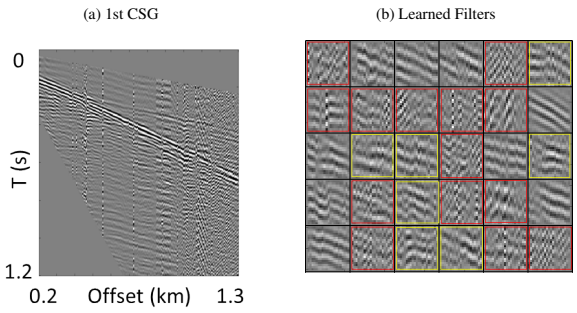

Figure 5: (a) A training CSG; (b) learned filters.

Next we exclude the filters with noise features indicated by the red boxes in Figure 5b. Then we apply the remaining filters for denoising. The denoised result and its residual are shown in Figure 6c and Figure 7b, respectively. We can see that the noise level within the red box of Figure 6c is reduced by only using the selected filters. We continue to exclude more filters as indicated by the yellow boxes in Figure 5b during the denoising phase. The denoised result is shown in Figure 6d and its residual is shown in Figure 7c. From the residuals in Figure 7, we can see that more coherent noise is removed.

We choose the near-offset traces (see Figure 8a) contaminated with surface waves to perform the test. All the filters shown in Figure 5b are used for denoising. The denoised data and the residual are shown in Figures 8b and 8c, respectively. The learned filters don't include the features from the surface waves, so the surface waves are removed.

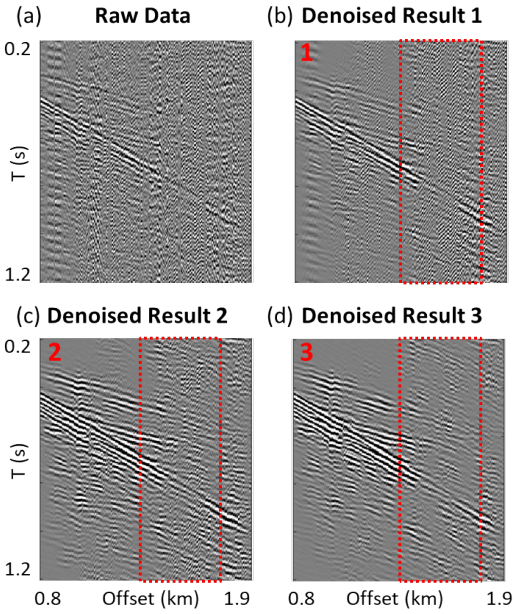

Figure 6: (a) Raw data; denoised data by (a) all filters, and selected filters excluding those indicated by (b) red and (c) red and yellow boxes in Figure 5b.

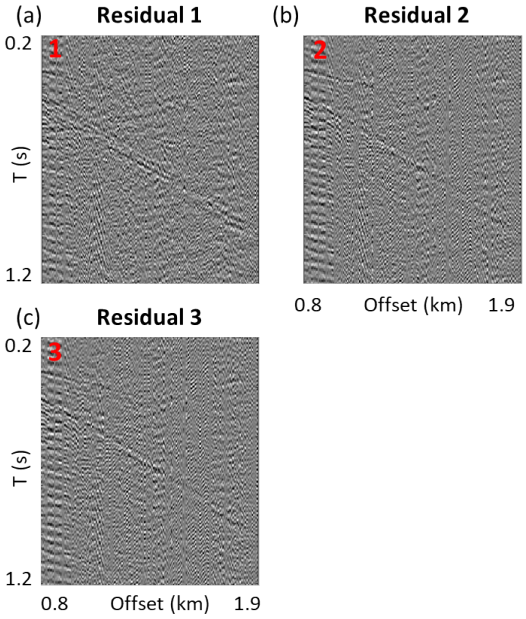

Figure 7: Residual of denoised data.

SUMMARY

We present a review of convolutional sparse coding and its application to seismic noise attenuation. A high SNR data set is used to train a set of filters and then the learned filters are used for denoising. For random noise, the value of the coefficient of the l_1 penalty function is selected by a trial-and-error procedure for denoising. For the coherent noise attenuation, the learned basis functions must exclude coherent noise features, such as the ground roll. The latter may need an algorithm to label the basis function according to a physical attribute that differentiates signal from noise, which will be our next study.

ACKNOWLEDGMENTS

The research reported in this publication was supported by the King Abdullah University of Science and Technology (KAUST) in Thuwal, Saudi Arabia. We are grateful to the sponsors of the Center for Subsurface Imaging and Modeling Consortium for their financial support. For computer time, this research used the resources of the Supercomputing Laboratory at KAUST and the IT Research Computing Group. We thank them for providing the computational resources required for carrying out this work.

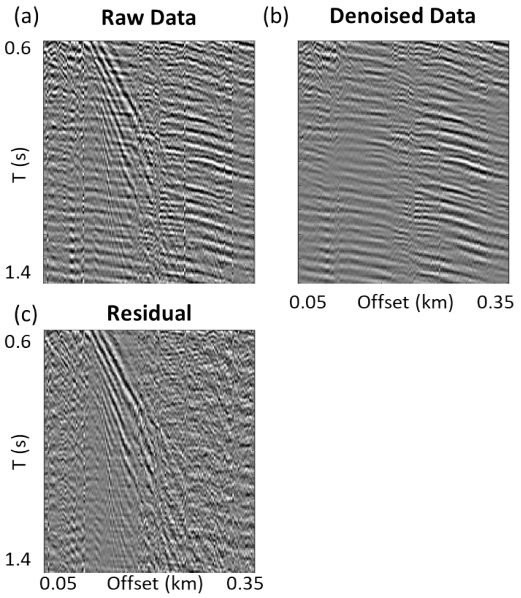

Figure 8: (a) Near-offset data with surface waves, (b) denoised data, and (c) its residual.

REFERENCES

Boyd, S., N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., 2011, Distributed optimization and statistical learning via the alternating direction method of multipliers: Foundations and Trends® in Machine learning, **3**, 1–122.

Bristow, H., A. Eriksson, and S. Lucey, 2013, Fast convolutional sparse coding: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, IEEE, 391–398.

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, Civil-Comp Press, 107–108.

Chen, Y., 2016, Dip-separated structural filtering using seislet transform and adaptive empirical mode decomposition based dip filter: Geophys. J. Int., **206**, 457–469.

Chen, Y., J. Ma, and S. Fomel, 2016, Double-sparsity dictionary for seismic noise attenuation: Geophysics, **81**, V103–V116.

Heide, F., W. Heidrich, and G. Wetzstein, 2015, Fast and flexible convolutional sparse coding: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5135–5143.

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

Ibrahim, A., and M. Sacchi, 2014, Eliminating blending noise using fast apex shifted hyperbolic radon transform: Presented at the 76th EAGE Conference and Exhibition 2014.

Kaplan, S. T., M. D. Sacchi, T. J. Ulrych, et al., 2009, Sparse coding for data-driven coherent and incoherent noise attenuation: Presented at the 2009 SEG Annual Meeting, Society of Exploration Geophysicists.

Zhu, L., E. Liu, and J. H. McClellan, 2015, Seismic data denoising through multiscale and sparsity-promoting dictionary learning: Geophysics, **80**, WD45–WD57.