IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 27, 2019, accepted December 6, 2019, date of publication December 13, 2019,
date of current version December 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2959238

Unsupervised Seismic Random Noise Attenuation Based on Deep Convolutional Neural Network

MI ZHANG[1,3], YANG LIU[1,2], AND YANGKANG CHEN[4]
[1]State Key Laboratory of Petroleum Resources and Prospecting, China University of Petroleum-Beijing, Beijing 102200, China
[2]School of Petroleum, China University of Petroleum-Beijing, Karamay 834000, China
[3]School of Earth and Atmospheric Sciences, Georgia Institute of Technology, Atlanta, GA 30332, USA
[4]School of Earth Sciences, Zhejiang University, Hangzhou 310027, China

Corresponding author: Yang Liu (wliuyang@vip.sina.com)

This work was supported by the National Science and Technology Major Project of China under Contract 2017ZX05018-005.

ABSTRACT Random noise attenuation is one of the most essential steps in seismic signal processing. We propose a novel approach to attenuate seismic random noise based on deep convolutional neural network (CNN) in an unsupervised learning manner. First, normalization and patch sampling are required to build training dataset and test dataset from raw noisy data. Instead of using synthetic noise-free data or denoised results via conventional methods as training labels, we adopt only the training set constructed from the raw noisy data as the input and design a robust deep CNN that just relies on the noisy input to learn the hidden features. The cross-entropy is chosen as the error criterion for establishing the cost function, which is minimized by the back-propagation algorithm to obtain the optimized parameters of the network. Then, we can reconstruct all patches of the test dataset via the optimized CNN. After patching processing and inverse normalization, the final denoised result can be obtained from reconstructed patches. Experimental tests on synthetic and real data demonstrate the effectiveness and superiority of the proposed method compared with state-of-the-art denoising methods.

INDEX TERMS Seismic data, noise attenuation, deep convolutional neural network, unsupervised learning.

I. INTRODUCTION

Random noise attenuation has always been a key step in seismic data processing. Unlike coherent noise, random noise in the seismic section does not have a fixed dominant frequency and apparent velocity, and it usually mixes with signals throughout all parts of data, increasing the difficulty of signal identification. Therefore, suppressing random noise can effectively improve the signal-to-noise ratio of seismic data, which is beneficial to imaging quality [1]–[3].

Effective suppression of seismic random noise is a challenging and attractive topic [4]–[7]. After decades of development, a number of random noise suppression methods have been proposed. According to the differences between assumptions and characteristics, the main methods can be roughly divided into five categories. The first type of methods is based on the stack of seismic data along the offset direction [8]–[10], nevertheless, such methods are not suitable to denoise the pre-stack seismic records. The second is based on prediction filtering [11]–[15]. Since the signals behave as continuous reflection events, the predictability of these events can be used to construct filters to achieve the separation of signals and noise. The third is based on mathematical transformations, which can convert raw seismic data into sparse domains to better separate signals and noise, such as Fourier transform [16], wavelet transform [17], [18], curvelet transform [19], dreamlet transform [20] and seislet transform [21]. The fourth category is based on matrix rank-reduction, which has developed rapidly in recent years [22]–[25]. Assuming that the ideal noise-free seismic data can be constructed to a low-rank matrix, additional random noise will increase the rank of matrix. Therefore, the removal of random noise can be regarded as a problem of low-rank matrix approximation. As the last branch, dictionary learning has achieved fruitful results in random noise attenuation [26]–[30]. It can realize the separation of signals and noise by constructing an over-complete dictionary with adaptive learning ability to sparsely decompose the noisy data and solve the optimal sparse expression [31]. The fidelity of the signals has always been the focus of the above methods, that is, it is hoped that the noise is completely suppressed while the damage of the signals is small. It is necessary to further explore new methods

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wang.

to generate the denoised results with high fidelity and high signal-to-noise ratio.

In recent years, deep learning has injected new vitality into the field of geophysics and has made breakthroughs in seismic data processing, inversion and interpretation [32]–[40]. It can construct abstract high-level representation by combining low-level features to discover the hidden features of complex data [41]–[43]. For seismic random noise attenuation algorithms based on deep learning [44]–[49], the well-known convolutional neural network (CNN) is adopted to learn the hidden features of the training dataset. The CNN is a multi-layered learning algorithm that can mimic the mechanism of the human brain and construct a neural network to analyze complex data, for instance, seismic dataset, which has its own unique characteristics and can be seen as a collection of many time series containing a wealth of waveform information. The tremendous potential of deep learning based on CNN makes it possible to address the problem of seismic random noise suppression. Such research has just begun, but it is of great value to the development of geophysics.

Taking the same strategy as [45] and [46], [47] employed synthetic noise-free records as labels for training a residual learning network [50] to attenuate seismic random noise, but the applications of this method on real data are not good enough. Additionally, [44], [48] and [49] selected denoised data with high SNR by conventional denoising method as labels to train CNN. Note that these existing research results based on CNN are all in a supervised manner, that is, using high-signal-to-noise ratio data (denoised results of existing methods or clean synthetic records) as training labels to construct the neural network. It is very critical to select the training labels for supervised learning, because it is related to whether the learned features are reliable. Although such a supervised strategy can effectively suppress the random noise of the synthetic data when the synthetic noise-free data are given, there is a problem in processing real data. The real seismic data do not have the corresponding noise-free data, and thus we cannot obtain the real training labels. On the one hand, if only the features obtained by training synthetic data are used to denoise the real data, the data from the training set and the test set do not match, and therefore the features cannot be perfectly inherited by the test set. In addition, blindly and extensively selecting synthetic seismic data as training samples will increase the redundancy of training work. On the other hand, if the denoised results of the conventional method are used as labels to train network, the denoised results reconstructed by the network may be difficult to greatly exceed that of the original labels.

In order to solve above problems, we adopt the manner of unsupervised learning [51], [52] to construct deep CNN that can effectively attenuate random noise. Instead of selecting denoised results of conventional methods or synthetic noise-free data as additional training labels, we directly utilize the training set built with raw noisy data as the input of network to build the cost function, and design a robust CNN that just relies on the noisy input to attenuate seismic random noise. The advantage of this idea is that we can always quickly build an available input dataset from raw noisy data to train the neural network. Therefore, we do not have the trouble of seeking relatively clean data as training labels. In addition, we design several special preprocessing steps to construct training dataset and test dataset using raw data. The back-propagation algorithm is utilized to optimize the cost function. The optimized parameters of network can be obtained after a stable optimization. Then, we can reconstruct the denoised patches via the optimized CNN. After patching processing and inverse normalization, the denoised patches turn into the final denoised result. We evaluate the proposed method on synthetic and real seismic data and compare it with state-of-the-art denoising algorithms.

II. METHODOLOGY

From the most basic point of view, the noisy data can be regarded as the sum of the signal and the noise, expressed as:

$$y = s + n, \tag{1}$$

where s denotes the unknown signal, n denotes the additional noise term and y denotes the noisy data recorded from the field. Note that the signal s is not correlated with the noise n. Since this paper focuses on the random noise attenuation, we assume that the noise term n is Gaussian noise, and each example $n_i \in \mathcal{N}(0, \sigma)$ is drawn from a zero-mean normal distribution with variance σ.

Our seismic random denoising method is based on a deep CNN to reconstruct the unknown s from the given y. Before building the deep CNN model, we first describe the pre-processing steps designed for constructing the training dataset and the test dataset.

A. PREPARATION OF TRAINING SET AND TEST SET

Given the raw seismic dataset y with noise, we first perform normalization to make the dataset better adapt to the neural network via

$$y^* = (y - min) / (max - min), \tag{2}$$

where y^* means the normalized dataset, the max and min values denote the maximum and minimum values of the raw data, respectively,.

After normalization, the training set $x_{training}$ and the test set x_{test} can then be built by introducing the randomly patch sampling operator $\mathcal{P}_1$ and the regularly patch sampling operator $\mathcal{P}_2$. Therefore, we have

$$x_{training} = \mathcal{P}_1 y^*, \tag{3}$$

$$x_{test} = \mathcal{P}_2 y^*, \tag{4}$$

where the operator $\mathcal{P}_1$ randomly divides the input into patches of size $N_1 \times N_2$ and the operator $\mathcal{P}_2$ regularly divides the input into patches of size $N_1 \times N_2$. Training set and test set do not necessarily have the same number of patches, but each patch

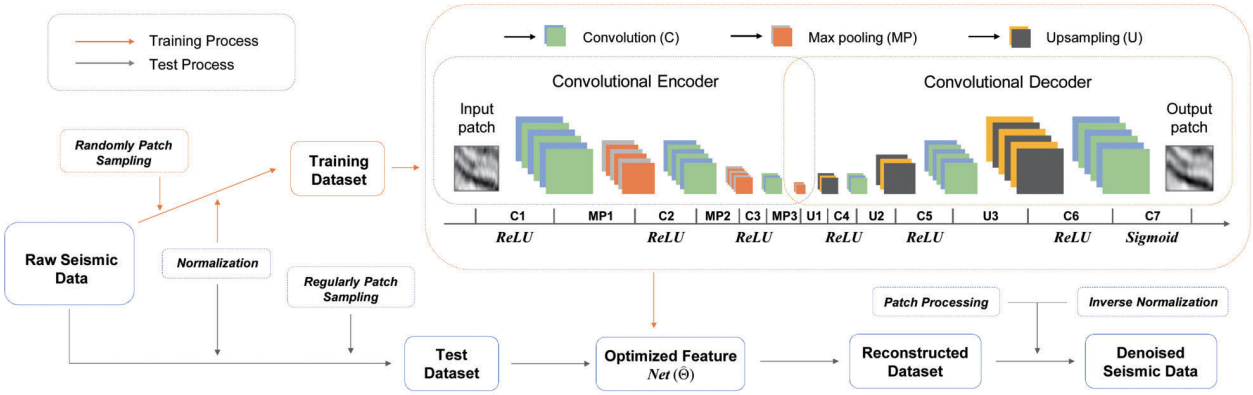


FIGURE 1. Architecture of the deep convolutional neural network we build for seismic denoising.

in the training set has the same dimensions as that in the test set. Additionally, the operator $\mathcal{P}_2$ ensures that all points of raw data are involved in the test step.

B. OUR CNN ARCHITECTURE FOR SEISMIC DENOISING

The training step of our CNN model is based on the encoder-decoder manner [53], which means that the input patch is first compressed to a lower-dimensional space to extract the hidden features and then expanded to reconstruct the original patch. A basic autoencoder (AE) [54], a three-layer neural network, first maps an input $x \in \mathcal{R}^d$ to the latent representation $h = \mathcal{F}(Wx+b)$, and then reconstruct the initial data by the inverse mapping $y = \mathcal{F}(\widehat{W}h + \widehat{b})$, where W and $\widehat{W}$ denote weights, b and $\widehat{b}$ denote bias, $\mathcal{F}$ is a nonlinear activation function. The parameters can be optimized by minimizing the cost function over the training set. Note that reconstructing data using low-dimensional features learned by AE is an unsupervised learning process.

However, the AE ignores the 2D structure of the input because the training set is extended to a 1D tensor as an input. This also introduces parameters redundancy, forcing weights are shared. Therefore, it is difficult for the conventional AE to learn meaningful features of seismic data with spatial continuity [55], [56]. To achieve better performance, convolution can be introduced into the basic AE to form a convolutional AE, which can effectively extract meaningful features for representing the noise-free data. The convolutional AE architecture is intuitively similar to the AE except that the parameters in hidden layer are global.

The latent representation of the k-th feature map with convolution layers can be expressed as [56]

$$h_k = \begin{cases} \mathcal{F}(x \otimes W_k + b_k), & k = 1 \\ \mathcal{F}(h_{k-1} \otimes W_k + b_k), & k \in (2, \ldots, \mathcal{M}) \end{cases} \tag{5}$$

where $\otimes$ denotes the convolution operator, $\mathcal{M}$ is the number of convolutional layers, W_k are convolutional filters, b_k are the biases. In the convolution process, the input feature map expands the dimension by zero-padding to ensure that the size of the output feature map is the same as that of the original input feature map. The rectified linear units (ReLu) $\mathcal{F}(s) = max(0, s)$ and the Sigmoid $\mathcal{F}(s) = (1 + e^{-s})^{-1}$ are chosen as the activation function to increase the nonlinearity of the neural network.

For CNN, max-pooling layers are often introduced to obtain translation-invariant representations and dimensionality reduction, with the advantages of speeding up the calculation and preventing over-fitting. Therefore, we add a max-pooling layer after each convolutional layer in the encoder step. The max-pooling layer reduces the size of the hidden feature map

$$h_k^{N_1 \times N_1} = max\left(h_k^{M_1 \times M_1} \mathcal{U}(l_1, l_2)\right) \tag{6}$$

from $M_1 \times M_1$ to $N_1 \times N_1$ by applying a window function $\mathcal{U}(l_1, l_2)$ to the input feature map (output of the previous convolutional layer) and saving the maximum in the neighborhood, where l_1 and l_2 denote the window length and the stride. When the stride is smaller than the window length, the windows will overlap.

Conversely, the dimensions of the hidden features are expanded in the decoder step to reconstruct the result with the same size as the original input. An effective way to achieve this is using upsampling processing based on the nearest-neighbor interpolation, which is similar to the inverse of the max-pooling. The upsampling layer expands the size of the hidden feature map

$$h_k^{N_2 \times N_2} = upsampling\left(h_k^{M_2 \times M_2} \mathcal{U}(l_1, l_2)\right) \tag{7}$$

from $M_2 \times M_2$ to $N_2 \times N_2$ by applying a window function $\mathcal{U}(l_1, l_2)$ to the input feature map (output of the previous layer) and extending the value to neighborhood.

After introducing the calculation principles of each kind of network layer, we clarify the overall network architecture shown as Fig. 1. In the convolutional encoder step, three pairs of alternately performed convolutional layers and max-pooling layers are involved. After the last max-pooling process (MP3), the latent feature map with the smallest

dimension of the entire network is generated and set as input to the convolutional decoder step. The convolutional decoder consists of three upsampling layers and four convolutional layers. Based on CNN, the corresponding reconstruction can be expressed as

$$y = Net\,(x;\,\Theta)\,, \tag{8}$$

where *Net* denotes our CNN architecture consisting of seven convolutional layers, three max-pooling layers and three upsampling layers. $\Theta = \{W, b\}$ denote the network parameters including the weight W and bias b. Then, we utilize cross-entropy [57] as the error criteria to construct cost function:

$$C(\Theta) = -\frac{1}{n}\sum_{i=1}^{n}\left[x_i ln y_i + (1-x_i) ln(1-y_i)\right], \tag{9}$$

where n stands for the sample number of x and y. By minimizing the cost function over the training set, parameters Θ are optimized to reconstruct the test set.

C. OPTIMIZATION AND RECONSTRUCTION

We employ the adaptive moment estimation approach [58] as a back propagation algorithm to minimize the cost function $C(\Theta)$, and briefly introduce some principles. At time step t of the back propagation, the gradients g_t^{Θ} of parameters Θ should be first updated as

$$g_t^{\Theta} = \nabla_{\Theta} C(\Theta)_{t-1}. \tag{10}$$

Then, the biased first moment estimate m^{Θ} and the biased second raw moment estimate v^{Θ} of parameters Θ are updated as

$$m_t^{\Theta} = \beta_1 \cdot m_{t-1}^{\Theta} + (1-\beta_1)\cdot g_t^{\Theta}, \tag{11}$$

$$v_t^{\Theta} = \beta_2 \cdot v_{t-1}^{\Theta} + (1-\beta_2)\cdot \left(g_t^{\Theta}\right)^2, \tag{12}$$

where, β_1 and β_2 are constants. For parameters Θ, bias-corrected first moment estimate $\widehat{m}_t^{\Theta}$ and second raw moment estimate $\widehat{v}_t^{\Theta}$ can be computed as

$$\widehat{m}_t^{\Theta} = m_t^{\Theta}/(1-\beta_1^t), \tag{13}$$

$$\widehat{v}_t^{\Theta} = v_t^{\Theta}/(1-\beta_2^t). \tag{14}$$

At last, parameters Θ are updated as [58]

$$\Theta_t = \Theta_{t-1} - \alpha \cdot \widehat{m}_t^{\Theta} / \left(\sqrt{\widehat{v}_t^{\Theta}} + \epsilon\right), \tag{15}$$

where α is the learning rate. Normally, the default parameters are $m_0^{\Theta} = 0$, $v_0^{\Theta} = 0$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. The completion of the parameter optimization means that the training process of the network is over.

Then we introduce the test process. With the learned parameters $\widehat{\Theta} = \left\{\widehat{W}, \widehat{b}\right\}$, we can reconstruct the test set using the optimized feature expression as

$$\widehat{x}_{test} = Net\left(x_{test};\,\widehat{\Theta}\right), \tag{16}$$

where $\widehat{x}_{test}$ means the denoised test set. Note that the form of $\widehat{x}_{test}$ is a set of patches, and thus the denoised patches need to be rebuilt to the data with the same dimensions as the original data y. We define an un-patching operator $\mathcal{P}_3$ to rearrange the patches via

$$y^*_{denoised} = \mathcal{P}_3 x_{test}, \tag{17}$$

where the operator $\mathcal{P}_3$ averages the overlapping regions of adjacent patches. Finally, we revert the amplitude back to the original order of magnitude by

$$y_{denoised} = y^*_{denoised} * (max - min) + min, \tag{18}$$

where $y_{denoised}$ denotes the final denoised result. To sum up, we utilize latent features learned by the deep CNN in an unsupervised manner to reconstruct noisy input so as to achieve noise attenuation.

III. NUMERICAL RESULTS

We test the denoising performance of the proposed method on synthetic and real seismic data, and three baseline methods are used for comparative experiments, i.e., f-x deconvolution (FXDECON) [12], [59], multichannel singular spectrum analysis (MSSA) [22], and deep learning based on autoencoder (AE) [52]. In order to qualitatively measure the quality of the denoised results, the signal-to-noise ratio is chosen as a criterion, which is expressed as

$$SNR = 10 log_{10}\frac{||D||_2^2}{||D-d||_2^2}, \tag{19}$$

where D and d stand for the clean and denoised data, respectively. The unit of SNR is decibel (dB). The greater the SNR value, the better the denoised result. However, for real seismic data without noise-free data and pure noise, SNR-based evaluation does not work. Therefore, we calculate the local similarity [60] that can roughly evaluate the signal leakage of denoised data as a second choice for evaluating denoising performance.

A. EXPERIMENTS WITH SYNTHETIC DATA

We first evaluate the denoising performance of three methods on synthetic data. The clean seismic data with 120 traces are part of a single-shot record obtained from the forward modeling, and each trace has 500 time sampling points. There are both strong amplitude signals and weak signals in the data, and the distribution of events is complicated, as illustrated in Fig. 2a. We add incoherent noise to the raw data and obtain the noisy data shown in Fig. 2b with a SNR of 1.90 dB. Many of events are blurred under noise pollution, and weak signals are almost indistinguishable. We first utilize both FXDECON and MSSA methods to denoise the noisy data, and obtain the denoised results displayed in Figs. 3a and 3b, respectively. It can be observed that a large amount of random noise is suppressed. Most of strong signals are clear, but some weak signals are still blurred. Figs. 3d and 3e demonstrate the corresponding noise removed by the two methods, where some coherent signals can be seen faintly. Then we test the denoising performance

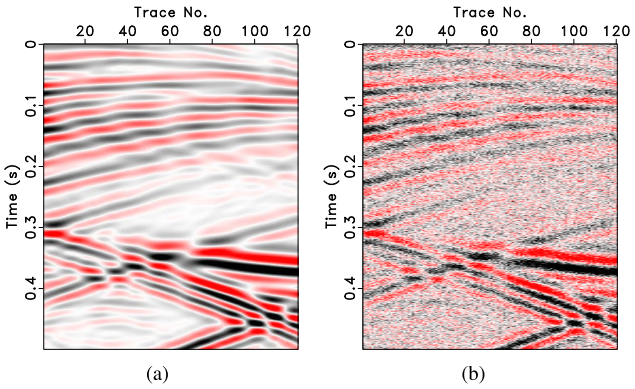


FIGURE 2. Synthetic example. (a) Clean dataset. (b) Noisy dataset (SNR = 1.90 dB).

of the proposed method. Due to the complexity of the network structure, the proposed method has more controllable parameters than the previous two algorithms. Therefore, before determining the final denoised result, we analyze the impact of some important relevant parameters in the constructed network to maximize the denoising performance of the network.

We mainly test the effect of the patch size, the size of the convolution filter, and the number of filters in each convolutional layer. Firstly, for the size of the patch, we test the denoising performances of 20×20, 40×40, 60×60, 80×80 four scales, where the two dimensions are the number of traces and the number of time sampling points. The SNRs of the denoised results are 20.85, 20.91, 20.88, 20.85 dB, respectively. It can be seen that for our deep CNN model, the size of the input patch has little effect on the denoised result. However, for a fixed-size seismic profile, the larger the size of patch, the larger the range covered by randomly extracting the same number of patches, and the more adequate the training. Based on the evaluation of the test results, we set the size of the patch to 40×40. Then, for the convolution filter, we test the effects of 2×2, 3×3, 4×4, 5×5, 6×6 five scales, and the SNRs of the denoised results are 17.43, 21.41, 21.63, 21.51, 19.83 dB, respectively, as illustrated in Table 1. We can see that the size of the convolution filter has a significant influence on the denoising ability of the network, and the ability to extract features of the network corresponding to the size is too small or too large is not optimal. Based on the test results, we select the best performing 4×4 as the filter size. Finally, we test the effect of the number of convolution filters. The designed network has symmetry in the encoding dimension and the de-encoding dimension, that is, C1 and C6, C2 and C5, C3 and C4 all have the same dimension and number of feature maps, and the number of additional C7 is 1 so that the output patch has the same size as the input patch. We test 10 different configurations and calculate the SNR of the corresponding denoised results, as shown in Table 2. Through test experiments, we can observe that the denoising ability of the proposed framework

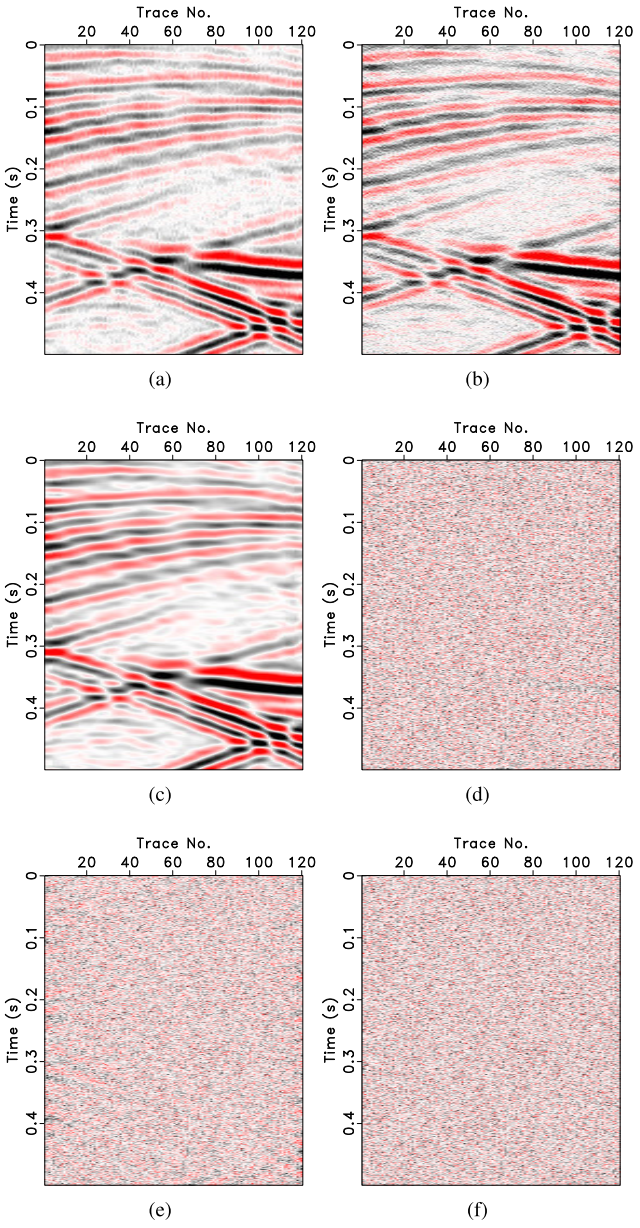


FIGURE 3. Denoising comparison of synthetic seismic data. Denoised result using (a) FXDECON (SNR = 14.20 dB), (b) MSSA (SNR = 9.74 dB), and (c) the proposed method (SNR = 16.59 dB). Removed noise using (d) FXDECON, (e) MSSA, and (f) the proposed method.

TABLE 1. Influence of the filter size of convolutional layers on denoising performance.

Size of convolution filter	SNR (dB)
2×2	17.43
3×3	21.41
4×4	21.63
5×5	21.51
6×6	19.83

increases with the number of feature maps within a certain range, and then decreases after reaching an ideal parameter.

TABLE 2. Influence of the number of filters in convolutional layers on denoising performance.

Filters in C1/C6	Filters in C2/C5	Filters in C3/C4	SNR (dB)
8	8	4	17.43
16	8	4	17.69
16	16	4	17.65
16	16	8	19.69
32	16	8	19.73
32	32	16	21.10
48	32	16	21.63
64	32	16	21.34
64	48	32	20.71
64	64	48	19.56

Therefore, we choose a convolutional layer configuration of 48-32-16-16-32-48-1 to get a satisfactory denoised result.

The selected density of our test parameters is not very high, but it is enough to determine a relatively stable CNN. Before training the network, the noisy data in Fig. 2b are randomly divided into 3000 patches with the patch size of 40 × 40 to build the training data and are regularly cut into 2436 patches to construct the test set, which guarantees that all data points are covered by the test set. The remaining parameters are selected based on the above discussion results about network training to reconstruct the final denoising data. We totally use seven convolutional layers, and the size of convolution filter in each layer is 4 × 4. The number of convolution filters in these seven layers is 48, 32, 16, 16, 32, 48, 1, respectively. The three max-pooling layers compress the size of the input patch from 40 × 40 to 20 × 20 and then to 10 × 10. On the other hand, the three upsampling layers expand the size of the input patch from 10 × 10 to 20 × 20 and then to 40 × 40. The learning rate is a typical value 0.001 [58]. After 30 epochs (an epoch means completing one forward calculation and one back propagation for the entire training set), we obtain the learned network to reconstruct the test set, and then gain the final denoised result after patch processing and inverse normalization.

The denoised result of the proposed method is illustrated in Fig. 3c, where almost no interference of incoherent noise can be observed. In addition, there are no obvious coherent signals in the corresponding residual profile (Fig. 3f), which proves that the signal error is small. After denoising, the SNR by the proposed method is 16.59 dB, higher than 9.74 dB and 14.20 dB by MSSA and FXDECON. Although the network parameters we select are not necessarily optimal, the denoised results reach better results than other two denoising methods in the industry. Therefore, the comparison of test results confirms the effectiveness of the proposed method. Further, we demonstrate the denoising performance of three methods on data with ten different levels of random noise. Fig. 4 shows the SNR comparison of three approaches with respect to the noise level of data. The blue line represents the SNR of the noisy data, and the green, black, and red line represent

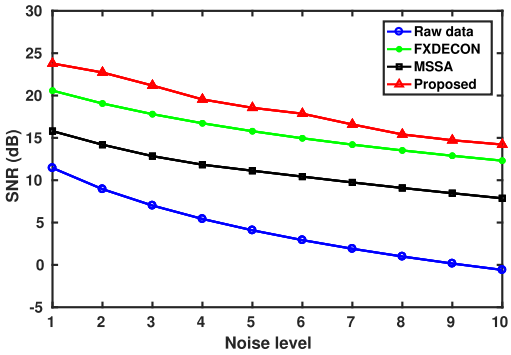


FIGURE 4. The SNR comparison of three methods with regard to the noise level.

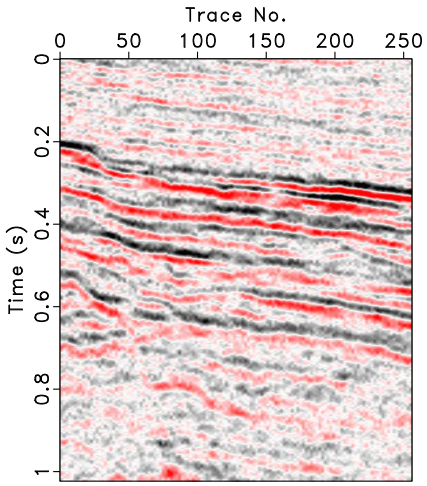


FIGURE 5. The first real seismic data.

the SNR values of denoised results via FXDECON, MSSA, and the proposed method, respectively. For each method, the greater the noise level of the data, the lower the signal-to-noise ratio of the denoised result. It can be seen that the denoised results of the proposed method at different noise levels have higher SNR than those of other two methods, which proves the remarkable denoising performance of the proposed method.

B. EXPERIMENTS WITH REAL DATA

In this section, we assess the practicality of the proposed method on real seismic data. The first real dataset displayed in Fig. 5 consists of 256 traces with 512 time samples. Before training the network, the seismic data are randomly divided into 12000 patches with the patch size of 40 × 40 to build the training set, and the seismic data are regularly cut into 10665 patches to construct the test set. There are seven convolution layers in the whole network, and the size of convolution filter in each layer is 3 × 3. The other network parameters are consistent with those of the synthetic example. After 30 epochs, we obtain the optimized neural network to reconstruct the test dataset, and then rebuild the final denoised result.

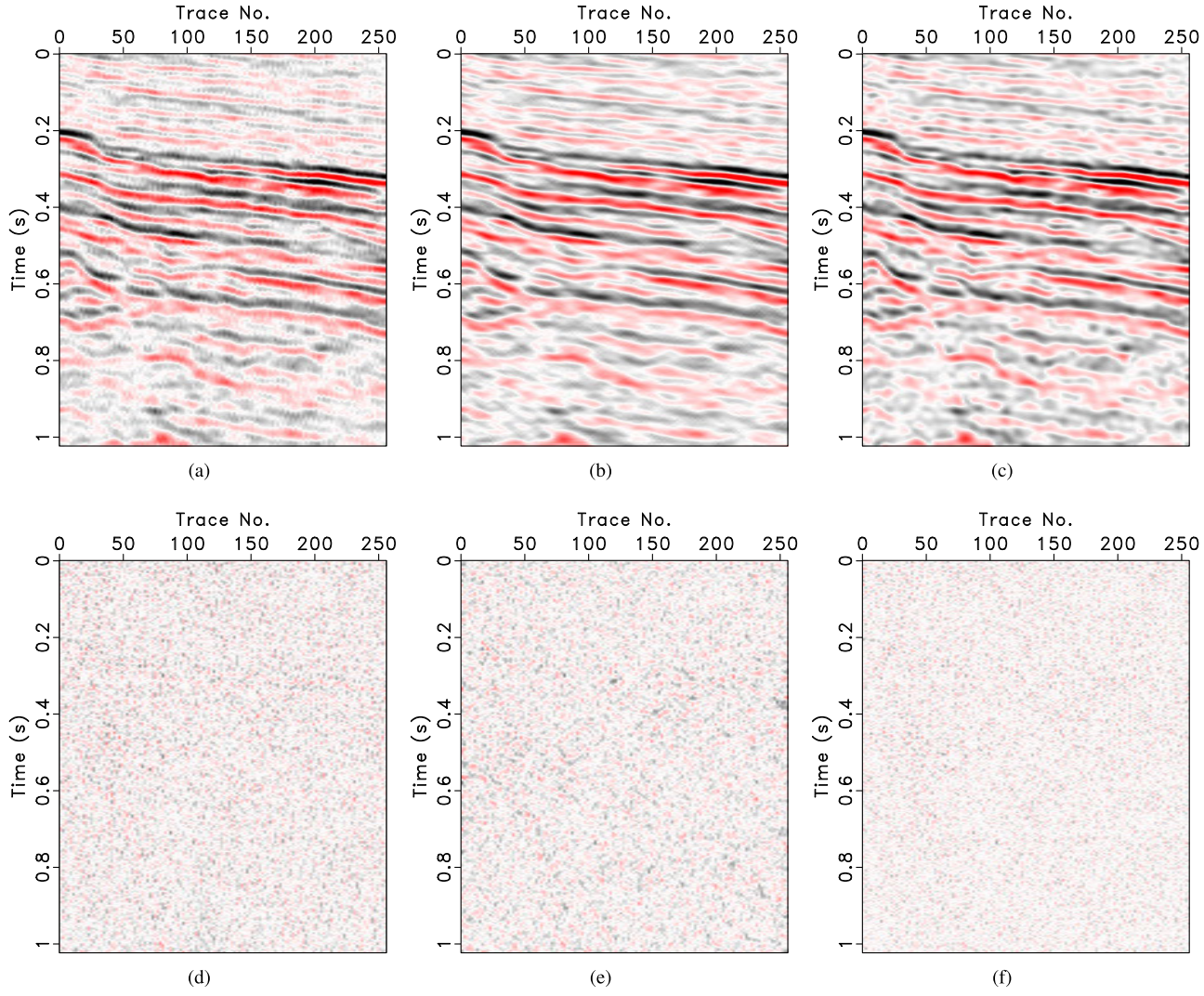


FIGURE 6. Denoising comparison of the first real seismic data. Denoised result using (a) FXDECON, (b) MSSA and (c) the proposed method. Removed noise using (d) FXDECON, (e) MSSA, and (f) the proposed method.

Figs. 6a, 6b and 6c display the denoised results of FXDECON, MSSA and the proposed method, respectively. There is more residual noise in the denoised result of FXDECON than that of other methods. Although there is no significant residual noise in the denoised result of MSSA, some details of the signals are missing, which means that the fidelity of the result is low. In the corresponding error sections shown in Figs. 6d, 6e and 6f, we can hardly find obvious continuous reflection signals except for some unusual amplitude points, which indicates that the signals are not seriously damaged during the denoising process of three methods. But from the zoomed graphs displayed in Fig. 7, some small differences between denoised results and original data can be observed. As can be seen from the marked position in the Fig. 7d, the proposed method does not cause serious damage to the fine structure when thorough noise attenuation is performed. Further, we can calculate the local similarity maps to assess the extent of signal leakage. Figs. 8a, 8b and 8c show the local similarity between the denoised result and removed noise using FXDECON, MSSA and the proposed method, respectively. Some high similarity anomalies indicate that the denoised results have signal leakage in the corresponding location. Note that the proposed method has less signal leakage than FXDECON and MSSA, indicating that the proposed method can adequately preserve the signals and effectively attenuate the noise.

We further explore the denoising performance on more complex real data for a more reliable evaluation. The second real dataset displayed in Fig. 9 consist of 400 CMPs, each of which has 800 time sampling points. Although some of the strongly reflected signals are continuous, the data are still contaminated by a large amount of incoherent background noise, causing some weak signals to be blurred.

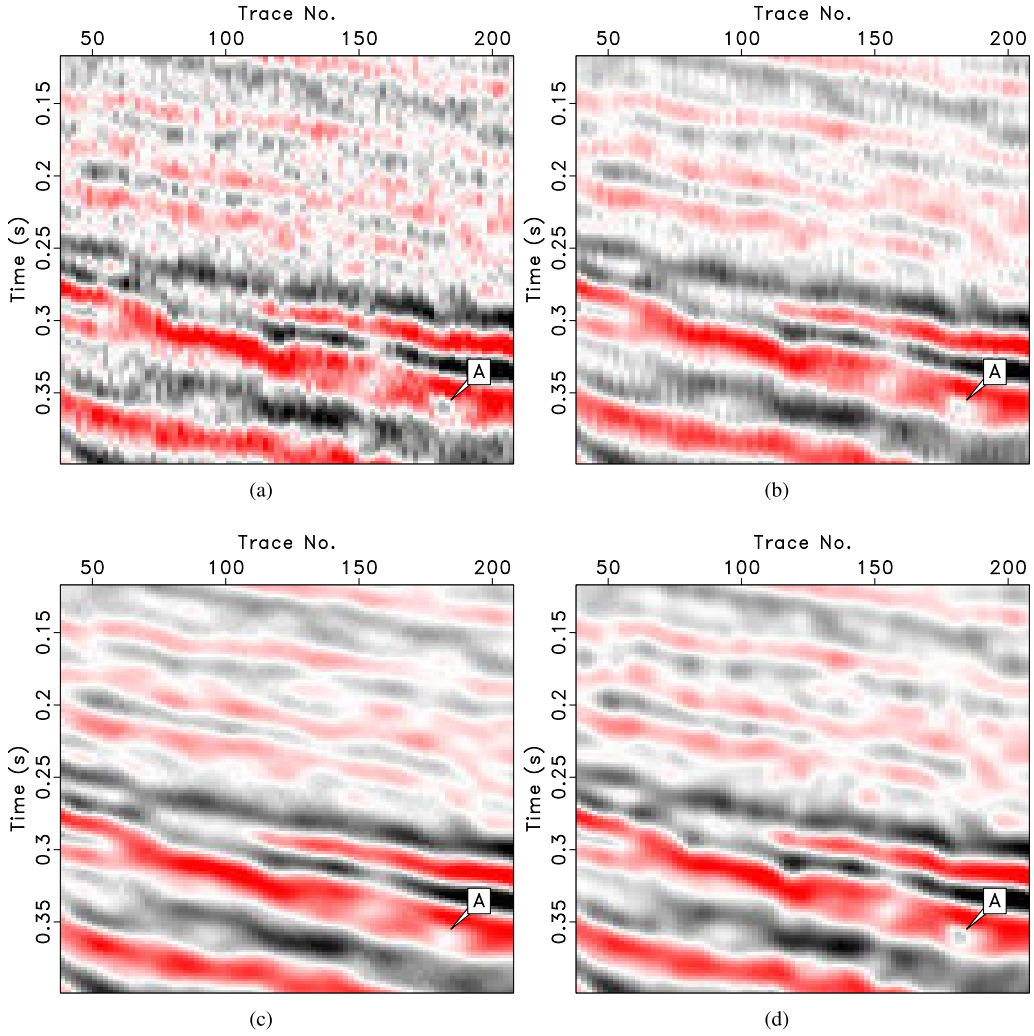


FIGURE 7. Zoomed denoised results tested on the first real data. (a) The first raw real data. (b) Denoised result using FXDECON. (c) Denoised result using MSSA. (d) Denoised result using the proposed method.

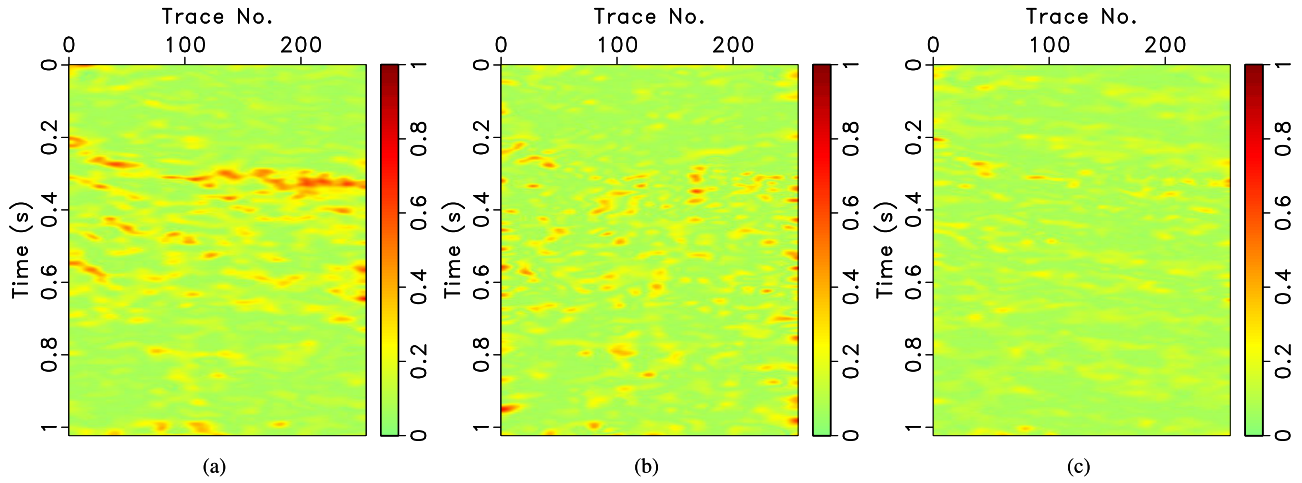


FIGURE 8. Local similarity comparison of three different methods for the first real data. Local similarity map of (a) FXDECON, (b) MSSA, and (c) the proposed method.

Before training the network, the second real data are randomly divided into 15000 patches with the patch size of 40 × 40 to build the training set and are regularly sampled into 8736 patches to construct the test set. All network

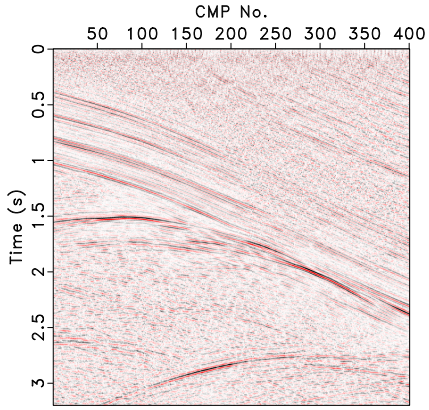


FIGURE 9. The second real seismic data.

parameters are consistent with those of the first real example. After 30 epochs, the optimized CNN is used to reconstruct the test set, and then the final denoised result is obtained. Figs. 10a, 10b, 10c and 10d display the denoised results of the second real example using FXDECON, MSSA, AE and the proposed method, respectively. We can observe that four methods can effectively suppress random noise. Although there is no significant residual noise in the denoised results, some details of the signals are missing, which means that the fidelity of denoised results is affected. In the corresponding noise sections shown in Figs. 10e, 10f, 10g and 10h, we can observe some obvious continuous reflection signals except for random noise, demonstrating that the signals are damaged during the denoising process of four methods. Compared to the proposed method, FXDECON and MSSA are more damaging to the effective signal. Additionally, both AE and the proposed method reconstruct the denoised result based on the hidden features obtained by deep learning. However, the proposed method keeps the spatial correlation features better due to the participation of convolution operations, thus making reconstruction of signals more accurate, as shown in Figs 10g and 10h. Therefore, compared with the two traditional methods and a deep-learning method, the proposed method can effectively attenuate the noise and fully protect the effective signals.

IV. DISCUSSION

Although the denoising effect of the proposed method is good, the computational time is relatively long. We roughly test the efficiency of the proposed algorithm on a laptop with a 1.8 GHz Intel Core i7 processor and 8 GB of memory. The computational time of the proposed method is mainly spent on learning the training set. Therefore, the computational time depends mainly on the size of the training set and the structure of CNN. The larger the training set, the longer it takes to train the network. In addition, increasing the size or number of convolution kernels also increases the training time. Under the premise of ensuring that the denoising effect is not seriously affected, reducing the size of the training set can effectively reduce the training time. One approach is to build a small training set via the patch sampling of a portion of the target noisy data. Another approach is to randomly sample the entire data but reduce the number of patches to build a small training set. To this end, we discuss the effect of two different strategies for narrowing the training set on the denoised results, as shown in Fig. 11. By comparing Fig. 11a and Fig. 11d, we can see that the denoised result of the training set constructed with partial data has large signal errors, which also indicates that the CNN trained by using synthetic data or other data is unstable on the target real data. By learning the training set constructed by sampling the whole data, the learned hidden features are adequate and reliable, and thus the denoised result is good. When sampling the whole data, reducing the number of patches has less impact on the denoised result within a certain range. Therefore, it is a more reasonable choice to build a small training set by reducing the number of patches sampled on the entire data. In detail, we show the training time corresponding to the training set with different numbers of patches in Table 3. For each additional 3000 patches in the training set, the training time required for each epoch increases by about 11 seconds. Since we have a high tolerance for computational time, the training set with 15000 patches is chosen for the final denoising to ensure a good denoising effect. In fact, the user can make a trade-off between the amount of computational time and the denoising effect to determine the corresponding size of the training set. Compared with the other two methods that only take a few seconds, AE and the proposed method cost more processing time, which is a common problem of the current deep-learning-based method. Therefore, some scholars have improved the sampling strategy of patches [61], [62], which can increase the processing efficiency to some extent. How to shorten the computational time of the denoising method based on learning algorithm is worth further research and improvement.

For noise attenuation using a supervised learning algorithm, an accurate noise-free seismic data set is required as a label. Nevertheless, the actual data does not have corresponding noise-free data, resulting in the inability to obtain accurate labels. Therefore, we have not adopted a supervised approach. Although we have not yet implemented a viable supervised noise suppression method, we do not deny the feasibility of supervised learning algorithms. Currently, there are few seismic random noise suppression algorithms based on supervised and unsupervised deep learning, and the published researches are not deep enough. The denoised data obtained by the proposed method can indeed be employed as training labels for supervised learning methods and is more accurate than the other three methods. However, even if we set unsupervised denoised results as labels for supervised learning algorithm, we also doubt whether the final denoising effect will exceed the effect of the labels. After all, the labels are not the noise-free data that the real data

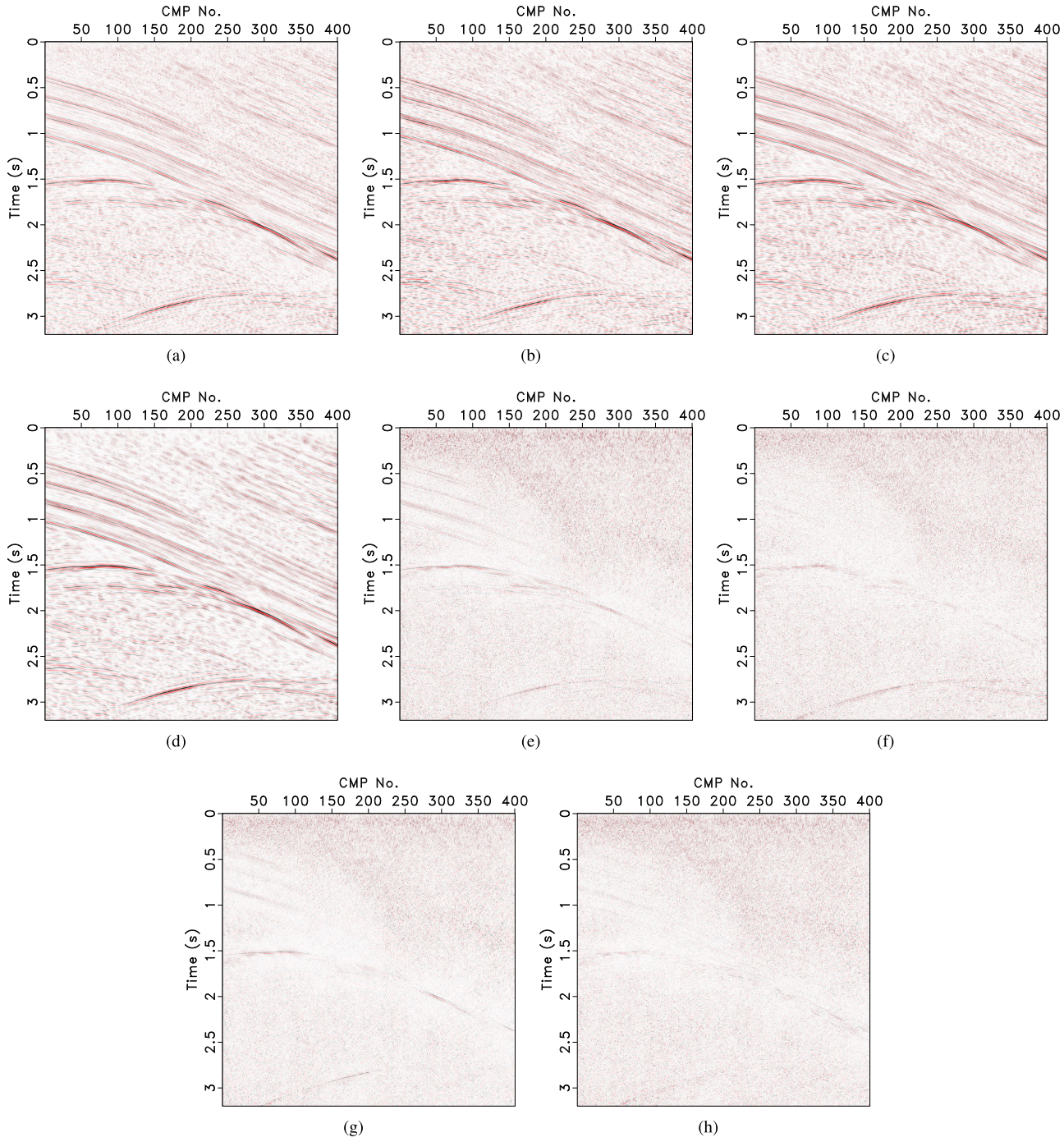


FIGURE 10. Denoising comparison of the second real seismic data. Denoised result using (a) FXDECON, (b) MSSA, (c) AE, and (d) proposed method. Removed noise using (e) FXDECON, (f) MSSA, (g) AE, and (h) the proposed method.

correspond to. In this paper, we implement an unsupervised learning method to suppress seismic random noise. The contribution and innovation of our work is to show how to solve the seismic denoising as an inverse problem based on CNN with only the target noisy seismic data. The unsupervised mode of the proposed method avoids preparing synthetic noise-free data or denoised results via conventional methods as training labels, and can directly construct a training set for feature learning by using known noisy data. The final denoising effect reaches our expectations, and we confirm that unsupervised learning is feasible to suppress seismic random noise. In the field of seismic noise suppression, supervised and unsupervised learning algorithms need to be further explored.

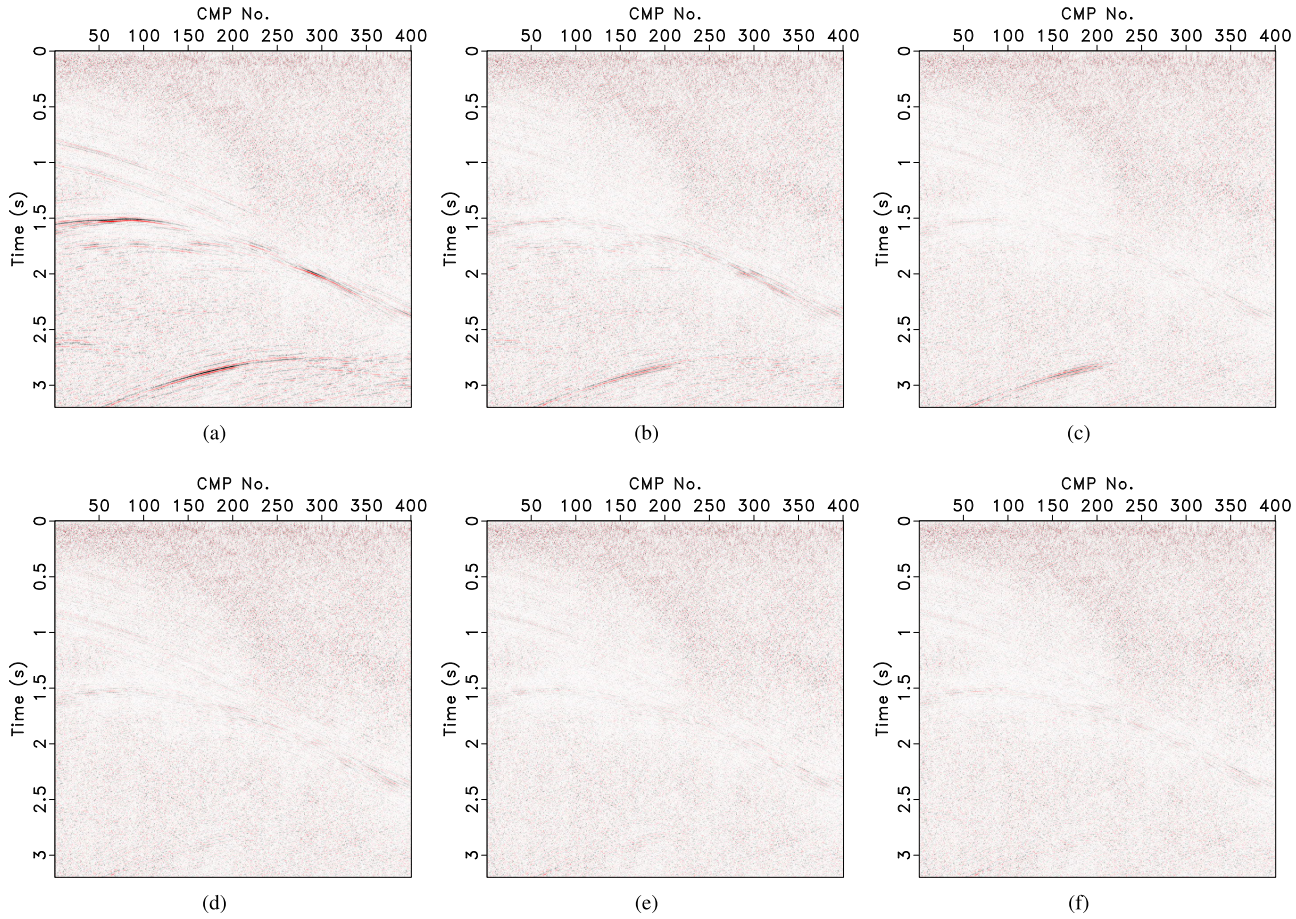


FIGURE 11. Further discussion on different strategies of constructing a small training set to narrow down the computational time. Removed noise via the proposed method by learning the training set with (a) 6000 patches sampled on a quarter of the raw data, (b) 9000 patches sampled on half of the raw data, (c) 12000 patches sampled on three-quarters of the raw data, (d) 6000 patches sampled on the whole raw data, (e) 9000 patches sampled on the whole raw data, (f) 12000 patches sampled on the whole raw data. The final denoising in Fig. 10 uses the training set with 15000 patches sampled on the whole raw data.

TABLE 3. The relationship between the training time per epoch and the number of patches in the training set.

Number of patches	Training time per epoch (second)
6000	21.74
9000	32.86
12000	43.75
15000	54.62

V. CONCLUSION

We have proposed a novel denoising framework for seismic data based on deep convolutional neural network in an unsupervised manner. The proposed method is a kind of data-driven method by constructing training set only based on the raw target noisy data, which makes it reasonable to use CNN for seismic noise attenuation in practical applications. The strategy for building training dataset and test dataset is convenient and efficient to achieve our goals. The dimensions of the network in the unsupervised manner are gradually compressed into the middle to extract the hidden features of seismic data. Denoised patches can be reconstructed using the weights and biases of the optimized network. After patching processing and inverse normalization, the denoised patches turn into the final denoised result. The designed network with convolutional layers, max-pooling layers and upsampling layers can effectively attenuate the random noise in seismic data. Experiments with synthetic and real seismic data confirm that the proposed method performs superior to other state-of-the-art denoising algorithms. The proposed method has the least damage to the signals while effectively suppressing random noise. Our proposed method provides a new insight for the development of unsupervised deep learning in seismic signal processing, and its greater potential deserves further exploration.

ACKNOWLEDGMENT

The authors would like to thank the editors and anonymous reviewers for their help in improving the quality of the paper. M. Zhang would like to thank the China Scholarship Council for providing financial support for his study at Georgia Institute of Technology.

REFERENCES

[1] J. C. Robinson, "Noise attenuation on common-depth-point seismic records by a semideterministic approach," *Geophysics*, vol. 33, no. 5, pp. 723–733, 1968.

[2] S. Yu, S. Osher, J. Ma, and Z. Shi, "Noise attenuation in a low-dimensional manifold," *Geophysics*, vol. 82, no. 5, pp. V321–V334, 2017.

[3] Q. Zhao, Q. Du, X. Gong, and Y. Chen, "Signal-preserving erratic noise attenuation via iterative robust sparsity-promoting filter," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3547–3560, Jun. 2018.

[4] X. Deng, D. Yang, J. Peng, G. Xin, and B. Yang, "Noise reduction and drift removal using least-squares support vector regression with the implicit bias term," *Geophysics*, vol. 75, no. 6, p. V119, 2010.

[5] Y. Liu, "Noise reduction by vector median filtering," *Geophysics*, vol. 78, no. 3, pp. V79–V87, 2013.

[6] K. Chen and M. D. Sacchi, "Robust reduced-rank filtering for erratic seismic noise attenuation," *Geophysics*, vol. 80, no. 1, pp. V1–V11, 2015.

[7] E. Ghaderpour, W. Liao, and M. P. Lamoureux, "Antileakage least-squares spectral analysis for seismic data regularization and random noise attenuation," *Geophysics*, vol. 83, no. 3, pp. V157–V170, 2018.

[8] G. Liu, S. Fomel, J. Long, and X. Chen, "Stacking seismic data using local correlation," *Geophysics*, vol. 74, no. 3, p. V43–48, 2009.

[9] W. H. Mayne, "Common reflection point horizontal stacking techniques," *Geophysics*, vol. 27, no. 6, pp. 927–938, 2012.

[10] W. Yang, R. Wang, J. Wu, Y. Chen, S. Gan, and W. Zhong, "An efficient and effective common reflection surface stacking approach using local similarity and plane-wave flattening," *J. Appl. Geophys.*, vol. 117, pp. 67–72, Jun. 2015.

[11] L. Canales, "Random noise reduction," *SEG Tech. Program Expanded Abstr.*, pp. 525–527, 1984, doi: 10.1190/1.1894168.

[12] N. Gulunay, "FXDECON and complex wiener prediction filter," *SEG Tech. Program Expanded Abstr.*, pp. 279–281, 1986, doi: 10.1190/1.1893128.

[13] R. Abma and J. Claerbout, "Lateral prediction for noise attenuation by t-x and f-x techniques," *Geophysics*, vol. 60, no. 6, pp. 1887–1896, 1995.

[14] M. Naghizadeh and M. Sacchi, "Multicomponent f-x seismic random noise attenuation via vector autoregressive operators," *Geophysics*, vol. 77, no. 2, pp. V91–V99, 2012.

[15] Y. Liu and B. Li, "Streaming orthogonal prediction filter in the t-x domain for random noise attenuation," *Geophysics*, vol. 83, no. 4, pp. F41–F48, 2018.

[16] D. Alsdorf, "Noise reduction in seismic data using Fourier correction coefficient filtering," *Geophysics*, vol. 62, no. 5, pp. 1617–1627, 1997.

[17] S. M. Mousavi and C. A. Langston, "Hybrid seismic denoising using higher-order statistics and improved wavelet block thresholding," *Bull. Seismol. Soc. Amer.*, vol. 106, no. 4, pp. 1380–1393, 2016.

[18] Z. Yu, R. Abma, J. Etgen, and C. Sullivan, "Attenuation of noise and simultaneous source interference using wavelet denoising," *Geophysics*, vol. 82, no. 3, pp. V179–V190, 2017.

[19] R. Neelamani, A. I. Baumstein, D. G. Gillard, M. T. Hadidi, and W. L. Soroka, "Coherent and random noise attenuation using the curvelet transform," *Lead. Edge*, vol. 27, no. 2, pp. 240–248, 2008.

[20] B. Wang, R.-S. Wu, X. Chen, and J. Li, "Simultaneous seismic data interpolation and denoising with a new adaptive method based on dreamlet transform," *Geophys. J. Int.*, vol. 201, no. 2, pp. 1180–1192, 2015.

[21] S. Fomel and Y. Liu, "Seislet transform and seislet frame," *Geophysics*, vol. 75, no. 3, pp. V25–V38, 2010.

[22] V. Oropeza and M. Sacchi, "Simultaneous seismic data denoising and reconstruction via multichannel singular spectrum analysis," *Geophysics*, vol. 76, no. 3, pp. V25–V32, 2011.

[23] W. Huang, R. Wang, Y. Chen, H. Li, and S. Gan, "Damped multichannel singular spectrum analysis for 3D random noise attenuation," *Geophysics*, vol. 81, no. 4, pp. V261–V270, 2016.

[24] R. Anvari, M. A. N. Siahsar, S. Gholtashi, A. R. Kahoo, and M. Mohammadi, "Seismic random noise attenuation using synchrosqueezed wavelet transform and low-rank signal matrix approximation," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6574–6581, Nov. 2017.

[25] C. Wang, Z. Zhu, H. Gu, X. Wu, and S. Liu, "Hankel low-rank approximation for seismic noise attenuation," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 561–573, Jan. 2019.

[26] S. Yu, J. Ma, X. Zhang, and M. D. Sacchi, "Interpolation and denoising of high-dimensional seismic data by learning a tight frame," *Geophysics*, vol. 80, no. 5, pp. V119–V132, 2015.

[27] Y. Chen, J. Ma, and S. Fomel, "Double sparsity dictionary for seismic noise attenuation," *Geophysics*, vol. 81, no. 2, pp. V17–V30, 2016.

[28] M. A. N. Siahsar, S. Gholtashi, A. R. Kahoo, C. Wei, and Y. Chen, "Data-driven multi-task sparse dictionary learning for noise attenuation of 3D seismic data," *Geophysics*, vol. 82, no. 6, pp. V385–V396, 2017.

[29] P. Turquais, E. G. Asgedom, and W. Söllner, "A method of combining coherence-constrained sparse coding and dictionary learning for denoising," *Geophysics*, vol. 82, no. 3, pp. V137–V148, 2017.

[30] L. Liu, J. Ma, and G. Plonka, "Sparse graph-regularized dictionary learning for suppressing random seismic noise," *Geophysics*, vol. 83, no. 3, pp. V215–V231, 2018.

[31] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[32] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[33] M. Hall and B. Hall, "Distributed collaborative prediction: Results of the machine learning contest," *Lead. Edge*, vol. 36, no. 3, pp. 267–269, 2017.

[34] L. Huang, X. Dong, and T. E. Clee, "A scalable deep learning platform for identifying geologic features from seismic attributes," *Lead. Edge*, vol. 36, no. 3, pp. 249–256, 2017.

[35] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, Apr. 2017.

[36] G. Pilikos and A. C. Faul, "Bayesian feature learning for seismic compressive sensing and denoising," *Geophysics*, vol. 82, no. 6, pp. O91–O104, 2017.

[37] H. Li, W. Yang, and X. Yong, "Deep learning for ground-roll noise attenuation," *SEG Tech. Program Expanded Abstr.*, pp. 1981–1985, 2018, doi: 10.1190/segam2018-2981295.1.

[38] F. Qian, M. Yin, X.-Y. Liu, Y.-J. Wang, C. Lu, and G.-M. Hu, "Unsupervised seismic facies analysis via deep convolutional autoencoders," *Geophysics*, vol. 83, no. 3, pp. A39–A43, 2018.

[39] Z. Wang, H. Di, M. A. Shafiq, Y. Alaudah, and G. AlRegib, "Successful leveraging of image processing and machine learning in seismic structural interpretation: A review," *Lead. Edge*, vol. 37, no. 6, pp. 451–461, Jun. 2018.

[40] T. Wrona, I. Pan, R. L. Gawthorpe, and H. Fossen, "Seismic facies analysis using machine learning," *Geophysics*, vol. 83, no. 5, pp. O83–O95, 2018.

[41] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[42] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[43] A. Siahkoohi, M. Louboutin, R. Kumar, and F. J. Herrmann, "Deep-convolutional neural networks in prestack seismic: Two exploratory examples," *SEG Tech. Program Expanded Abstr.*, pp. 2196–2200, 2018, doi: 10.1190/segam2018-2998599.1.

[44] D. Liu, W. Wang, W. Chen, X. Wang, Y. Zhou, and Z. Shi, "Random-noise suppression in seismic data: What can deep learning do?" *SEG Tech. Program Expanded Abstr.*, pp. 2016–2020, 2018, doi: 10.1190/segam2018-2998114.1.

[45] X. Si and Y. Yuan, "Random noise attenuation based on residual learning of deep convolutional neural network," *SEG Tech. Program Expanded Abstr.*, pp. 1986–1990, 2018, doi: 10.1190/segam2018-2985176.1.

[46] D. Liu, W. Wang, W. Chen, X. Wang, Y. Zhou, and Z. Shi, "Seismic data denoising by deep-residual networks," *SEG Tech. Program Expanded Abstr.*, pp. 4593–4597, 2018, 10.1190/segam2018-2998619.1.

[47] Y. Zhang, H. Lin, and Y. Li, "Noise attenuation for seismic image using a deep-residual learning," *SEG Tech. Program Expanded Abstr.*, pp. 2176–2180, 2018, doi: 10.1190/segam2018-2997974.1.

[48] S. Mandelli, V. Lipari, P. Bestagini, and S. Tubaro, "Interpolation and denoising of seismic data using convolutional neural networks," 2019, *arXiv:1901.07927*. [Online]. Available: https://arxiv.org/abs/1901.07927

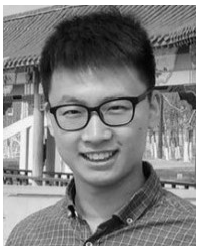
[49] Y. Wang, W. Lu, J. Liu, M. Zhang, and Y. Miao, "Random seismic noise attenuation based on data augmentation and CNN," *Chin. J. Geophys.-Chin. Ed.*, vol. 62, no. 1, pp. 421–433, 2019.

[50] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian Denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[51] A. M. Cheriyadat, "Unsupervised feature learning for aerial scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 439–451, Jan. 2014.

[52] L. Deng and D. Yu, *Deep Learning: Methods and Applications*. Hanover, MA, USA: Now Publishers, 2014.

[53] G. E. Hinton and R. S. Zemel, "Autoencoders, minimum description length and helmholtz free energy," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 6, 1994, pp. 3–10.
[54] C.-Y. Liou, W.-C. Cheng, J.-W. Liou, and D.-R. Liou, "Autoencoder for words," *Neurocomputing*, vol. 139, pp. 84–96, Sep. 2014.
[55] M. Zhang, Y. Liu, M. Bai, and Y. Chen, "Seismic noise attenuation using unsupervised sparse feature learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 9709–9723, Aug. 2019.
[56] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *Artificial Neural Networks and Machine Learning—ICANN*. Berlin, Germany: Springer, 2011, pp. 52–59.
[57] D. M. Kline and V. L. Berardi, "Revisiting squared-error and cross-entropy functions for training neural network classifiers," *Neural Comput. Appl.*, vol. 14, no. 4, pp. 310–318, Dec. 2005.
[58] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, pp. 1–15, Dec. 2014. [Online]. Available: https://arxiv.org/abs/1412.6980
[59] N. Gulunay, "Signal leakage in f-x deconvolution algorithms," *Geophysics*, vol. 82, no. 5, pp. W31–W45, 2017.
[60] S. Fomel, "Local seismic attributes," *Geophysics*, vol. 72, no. 3, pp. A29–A33, May 2007.
[61] S. Yu, J. Ma, and S. Osher, "Monte Carlo data-driven tight frame for seismic data recovery," *Geophysics*, vol. 81, no. 4, pp. V327–V340, 2016.
[62] S. Zu, H. Zhou, R. Wu, M. Jiang, and Y. Chen, "Dictionary learning based on dip patch selection training for random noise attenuation," *Geophysics*, vol. 84, no. 3, pp. V169–V183, 2019.



MI ZHANG received the bachelor's degree in exploration geophysics from the China University of Petroleum, Beijing, China, in 2016, where he is currently pursuing the Ph.D. degree with the School of Geophysics, under the supervision of Prof. Y. Liu.

He is currently a Visiting Scholar with the Georgia Institute of Technology, Atlanta, GA, USA. His research interests include seismic signal processing, seismic imaging, and machine learning.

YANG LIU received the Ph.D. degree in applied geophysics from the University of Petroleum, Beijing, in 1998.

He has been with the China University of Petroleum at Beijing (CUPB), since 1998, where he is currently a Professor. He was a Visiting Scholar with The University of Texas at Austin, from 2008 to 2009. His research interests include seismic modeling, seismic signal processing, seismic inversion, and seismic interpretation. He is also a member of the Society of Exploration Geophysicists (SEG) and the European Association of Geoscientists and Engineers (EAGE).

YANGKANG CHEN received the B.S. degree in exploration geophysics from the China University of Petroleum, Beijing, in 2012, and the Ph.D. degree in geophysics from The University of Texas at Austin, in 2015, under the supervision of Prof. S. Fomel.

He then joined the Oak Ridge National Laboratory as a Distinguished Postdoctoral Research Associate, where he conducted a research on global adjoint tomography. He is currently an Assistant Professor with Zhejiang University. His research interests include simultaneous-source data deblending and imaging, global adjoint tomography, machine learning, and high-performance computing.