Signal Recovery from Randomly Quantized Data Using Neural Network Approach

Ali Al-Shaikhi 1,2

1 Department of Electrical Engineering, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia; shaikhi@kfupm.edu.sa
2 Center for Communication Systems and Sensing, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia

Abstract: We present an efficient scheme based on a long short-term memory (LSTM) autoencoder for accurate seismic deconvolution in a multichannel setup. The technique is beneficial for compressing massive amounts of seismic data. The proposed robust estimation ensures the recovery of sparse reflectivity from acquired seismic data that have been under-quantized. By adjusting the quantization error, the technique considerably improves the robustness of data to the quantization error, thereby boosting the visual saliency of seismic data compared to the other existing algorithms. This framework has been validated using both field and synthetic seismic data sets, and the assessment is carried out by comparing it to the steepest decent and basis pursuit methods. The findings indicate that the proposed scheme outperforms the other algorithms significantly in the following ways: first, in the proposed estimation, fraudulently or overbearingly estimated impulses are significantly suppressed, and second, the proposed guesstimate is much more robust to the quantization interval changes. The tests on real and synthetic data sets reveal that the proposed LSTM autoencoder-based method yields the best results in terms of both quality and computational complexity when compared with existing methods. Finally, the relative reconstruction error (RRE), signal-to-reconstruction error ratio (SRER), and power spectral density (PSD) are used to evaluate the performance of the proposed algorithm.

Keywords: quantization; deconvolution; LSTM neural network

1. Introduction

Seismic deconvolution is the process of removing a wavelet from the recorded seismic traces, which also diminishes short-period multiples and repercussions. Seismic deconvolution is an important component of the seismic data processing pipeline because it produces a more interpretable seismic section. Deconvolution techniques are widely adopted in seismic applications and beyond [1]. Blind deconvolution is a common challenge in seismic deconvolution because the blurring kernel, which is the source seismic wavelet, is unidentified [2].

In a seismic survey, the wave field originated from a seismic source and received by an array of sensors ultimately produces a seismic data section. Usually, in exploration geophysics, we deal with a multichannel scenario [3–5], in which seismic traces are commonly represented as convolutions of the same source waveform with multiple channels stemming from various reflectivity models. In the multichannel blind deconvolution application, conventional approaches do not produce satisfactory results for a number of reasons. The primary reason is the high similarity among neighboring reflectivity series, which either makes the issue numerically sensitive or, in the worst-case scenario, hard to solve and ill-posed [6].

In order to address this problem, ref. [7] proposed a Bayesian approach in which a regularization term based on a hybrid $\ell_1/\ell_2$-norm is adopted for sparsity promotion. This
method, called the steepest decent (SD) approach, performs admirably with both field and synthetic data sets. However, its high computational complexity prevents it from being adapted to industrial-scale data sets. The other solution is a series of deconvolution filtering concept variations, such as the commonly used deconvolution prediction filter [8] and the fast sparse method for blind multichannel deconvolution [6]. These techniques yield a deconvolution filter based on an established criterion—for example, the $\ell_1/\ell_2$-norm or the least-squares of the deconvoluted signal. Their computational cost is typically far lower compared to the other methods. However, the limited filter length becomes a bottleneck in their performance. For instance, the Wiener filter will increase the noise in the process of deconvolution, and this method cannot produce a wide band spiky deconvolution result.

This work considers a realistic application in which the recorded seismic time-series traces are quantized element-by-element in an attempt to lessen storage and communication costs. The quantization is performed on the randomly selected samples in a trace. According to lossy compression methods, for example, [9–12], the higher the compression gain, the lower the reconstruction quality. Consequently, depending on the quantization interval, the accuracy of predicting the reflectivity series from quantized traces is jeopardized. To alleviate this annoyance, we propose a robust optimization scheme that improves prediction robustness and accuracy.

The proposed scheme is an efficient deconvolution method that utilizes long short-term memory (LSTM) autoencoder networks [13] by exploiting the multichannel structure of the seismic section and band-limited/sparse characteristics of the seismic source wavelet [14]. Furthermore, it is a type of recurrent neural network that can learn long-term dependencies in data. The seismic trace is well-suited for this kind of architecture due to the high correlation between the samples. In this work, the parameters (number of layers, weights, biases) in LSTM are selected based on the selectivity analysis. The LSTM itself has not been modified or improved. The quantized data are used to recover the seismic reflectivity. Simulations with both field and synthetic data sets illustrate that this novel approach gives fast and high-quality deconvolution results. In comparison with prevailing methods, we show examples where the proposed method outperforms the SD method and the basis pursuit (BP) method (discussed in Section 3) in both deconvolution results and computational efficiency.

**Notation:** Vectors/Matrices are represented by bold notations, whereas scalars are represented by lower-case letters. The rest of this paper is organized as follows: Sections 2 and 3 are devoted to the convolution model and the traditional deconvolution approaches, respectively. The proposed deconvolution approach is presented in Section 4, whereas the experimental results and discussion are reported in Section 5. Finally, Section 6 draws the conclusions.

### 2. Convolution Model

The recorded seismic trace can be modeled as a linear system [15] in which the Earth’s impulse response is convoluted with a seismic source wavelet. The $j$-th received seismic trace, in particular, is given as follows [16]:

$$x_j[n] = w[n] * r_j[n] + z_j[n] = \sum_k w[n-k]r_j[k] + z_j[n], \quad j = 1, \ldots, J,$$

where $*$ represents the convolution operation, $w[n]$ denotes the source wavelet and $r_j[n]$ denotes the reflectivity model with $J$ number of channels. Figure 1 gives the pictorial view of the convolutional model.
Before delving into the effect of noise $z_j[n]$, we concentrate on the noiseless case. The $z$-transform of $x_j[n]$ in the noiseless version of (1), i.e., $z_j[n] = 0$, gives [7]:

$$X_j(z) = W(z)R_j(z). \quad (2)$$

Similarly, the $z$-transform of $x_i[n]$ gives:

$$X_i(z) = W(z)R_i(z). \quad (3)$$

Multiplying (2) and (3) by $R_i(z)$ and $R_j(z)$, respectively, we can conclude that the succeeding system of linear equations can be acquired by contemplating (2) for a couple of traces:

$$X_i(z)R_j(z) - X_j(z)R_i(z) = 0, \quad \text{for } i \neq j, \quad (4)$$

which can be rewritten in matrix form as follows:

$$X_i r_j - X_j r_i = 0, \quad (5)$$

whereas $X_j$ represents the $j$-th channel convolution matrix, $r_i = [r_i[1], \ldots, r_i[N]]^T$ is the vectorized reflectivity and $N$ represents the samples in a trace. We can combine all instances of Equation (5) into the one equation as follows: [6]

$$A r = 0, \quad (6)$$
where

\[
A = \begin{bmatrix}
X_2 - X_1 \\
X_3 - X_1 \\
X_4 - X_1 \\
\vdots \\
X_3 - X_2 \\
X_4 - X_2 \\
X_5 - X_2 \\
\vdots \\
\end{bmatrix}
\]

and \(r = [r_1^T, r_2^T, \ldots, r_J^T]^T\). The modification for the additive noise scenario is [6]:

\[
A r = e,
\]

but the noise vector \(e\) is neither white nor Gaussian [7].

The direct benefit of using less quantization levels is that it compresses the seismic data volume. In this case, the compression gain is defined as the number of bits used by the original seismic trace divided by the number of bits used by the quantized seismic trace. It is calculated as follows: if \(N\) binary bits are used to store each sample of the original seismic trace, then the quantization interval is defined as \(q = \delta Y\), where \(0 < \delta \leq 1\) is the scaling factor and \(Y\) is the sample’s maximum absolute value. Hence, the codewords are \((2/\delta) + 1\) in size, and each quantized seismic sample is represented by \(\lceil \log_2[(2/\delta) + 1] \rceil\) number of bits. Eventually, \(C_r\), the compression gain is given by:

\[
C_r(q, N) = \frac{N}{\log_2[(2Y/q) + 1]} \approx \frac{N}{\log_2 2Y \log_2 q}
\]

Despite the fact that compression gain increases exponentially with the \(q\) value, as can be seen from (9), the quantization error, which is defined as the difference between the quantized and original samples, i.e., \(e_i = x_i - x_i'\), has an upper-bound determined by the \(q\) value:

\[
|e_i| \leq \frac{1}{2} q
\]

and, similarly, the \(l_2\)-norm of the error quantization vector, \(e = x - x'\):

\[
\|e\|_2 \leq \frac{1}{2} q \sqrt{m}
\]

where \(m\) is the number of samples in a trace. The above Equation (11) pertains to a trade-off among the increased compression gain and adequate preservation of the signal quality, which applies to all lossy compression strategies. We link the two options of conundrum with that of the value of \(q\) under an application domain, namely, reflectivity estimation. This effectively calibrates the trade-off between accuracy and compression in this study. By integrating the quantization into the convolution model, the seismic quantized trace is characterized as a linear system with admixture bounded noise:

\[
x' = wr + e
\]

We are interested in locating a sparse linear combination of basic elementary signals which is impacted by the confined quantization error. To be precise, the work is understood as follows: given an observed signal \(x'\) and its linear model in (12), approximate the coefficient vector \(r\).
3. Deconvolution Methods

In this section, we look at some traditional techniques that will be used as benchmarks for our proposed approach. We divide these methods into two classes, i.e., SD and BP methods. Our aim in these methods is to optimize [17]:

$$\hat{r} = \arg\min_r \left\{ \frac{1}{2} \| Ar \|_2^2 + \lambda \sum_j R_\epsilon(r_j) \right\}, \text{ subject to } r^T r = 1,$$

(13)

where the constraint $r^T r = 1$ is added to avoid a trivial solution in (13). The regularization term is a differentiable hybrid $\ell_1/\ell_2$-norm, which makes the optimization easier, and $R_\epsilon(r_j) = \sum_n \left( \sqrt{r^2_j[n]} + \epsilon_k^2 - \epsilon_k \right)$, which approximates an $\ell_1$-norm and is added in order to promote the sparsity of the output $\hat{r}$. As illustrated in Figure 2, small values of $\epsilon$ generate a mixed norm that can be very close to the $\ell_1$ norm.

In order to honor the constraint $r^T r = 1$ in computation, [7] introduced a variant of iterative steepest descent which employs Rodrigues's rotation formula [18]. The resulting updated equation for the SD method is given as [17]:

$$\hat{r}^{(i+1)} = \cos(\theta^{(i)}) \hat{r}^{(i)} + \sin(\theta^{(i)}) \nabla L^{(i)} / |\nabla L^{(i)}|,$$

(14)

Figure 2. $\ell_1/\ell_2$-norm $R_\epsilon(x)$ for different $\epsilon$ values.
1. Initial peak detection for each channel. For the channel

\[ \mu(i) = \arg\min_{\mu} \sum_{j} \beta_{i,j} = \arg\min_{\mu} \sum_{j} \left( \frac{r_{j}^{2}(n)}{\sigma_{j}^{2} + \epsilon_{j}} + \frac{\epsilon_{j}}{\sigma_{j}^{2}} \right)^{-1/2} Q_{k}^{(i)}(j,n) \]

(18)

where

\[ \nabla L^{(i)} = \mathbf{A}^{T} \mathbf{A}^{(i)} + \lambda \left( \mathbf{r}^{(i)} \odot \mathbf{r}^{(i)} + 1 \epsilon^{2}_{k} \right) \odot \mathbf{r}^{(i)} \]

(15)

and \( \theta^{(i)} = -\alpha / (h^{T} \nabla L^{(i)} / |\nabla L^{(i)}|) \). The notation \( \odot \) represents the element-by-element multiplication of two vectors, and 1 is a vector of all ones with length equal to \( \mathbf{r}^{(i)} \).

Another solution is to design a filter \( \mathbf{v}[n] \) for deconvolution that minimizes the \( \ell_{1}/\ell_{2}\)-norm of the deconvolved trace \( \hat{r}[n] = \sum_{k} \mathbf{v}[k]x[n - k] = x[n] \ast \mathbf{v}[n] \). Thus, we solve the following optimization problem [7]:

\[ \hat{\mathbf{v}} = \arg\min_{\mathbf{v}} \sum_{j} \mathcal{R}_{c}(\mathbf{r}_{j}), \text{ subject to } \mathbf{v}^{T} = 1, \]  

(16)

where the sparse-promoting norm \( \mathcal{R}_{c} \) is defined as \( \mathcal{R}_{c}(\mathbf{y}_{j}) = \sum_{n} \left( \frac{r_{j}^{2}(n)}{\sigma_{j}^{2} + \epsilon_{j} - \epsilon_{f}} \right) \) and \( \sigma_{j}^{2} = \sum_{n} r_{j}^{2}(n)/N \) is the predicted variance obtained from the estimated reflectivity series. The conditionality \( \mathbf{v}^{T} \mathbf{v} = 1 \) is levied to prevent the trivial solution.

The iterative SD algorithm provides an accurate solution to problem (16). The updated equation is:

\[ \hat{\mathbf{v}}^{(i+1)} = \hat{\mathbf{v}}^{(i)} - \mu \nabla L^{(i)}, \quad \hat{\mathbf{v}}^{(i+1)} = \frac{\hat{\mathbf{v}}^{(i+1)}}{||\hat{\mathbf{v}}^{(i+1)}||_{2}} \]  

(17)

where \( \mu \) is the step size of the gradient algorithm and

\[ \nabla L_{k}^{(i)} = \sum_{j} \sum_{n} \left( \frac{r_{j}^{2}(n)}{\sigma_{j}^{2} + \epsilon_{j}} + \epsilon_{j} \right)^{-1/2} Q_{k}^{(i)}(j,n) \]

\[ Q_{k}^{(i)}(j,n) = \sigma_{j}^{2} / \sigma_{j}^{2} \hat{r}_{j}^{(i)}(n)x_{j}(n - k) - \frac{1}{N} \sum_{n} \hat{r}_{j}^{(i)}(n)x_{j}(n - k). \]

(18)

Compared with the first approach, the number of parameters in the second approach (i.e., the length of the deconvolution filter) is much less; as a result, the second approach’s computational complexity is substantially lower as compared to the first one.

Another method, called the basis pursuit method [17], is implemented in the following three steps:

1. Initial peak detection for each channel. For the \( j \)-th trace, we find an initial reflectivity \( \mathbf{r}_{j}^{0} \) via:

\[ \mathbf{r}_{j}^{0} = \arg\min_{\mathbf{r}_{j}} (\mathbf{x}_{j}^{T} \mathbf{r}_{j} - 1)^{2} \text{ subject to } ||\mathbf{r}_{j}||_{1} = \tau. \]  

(19)

where \( \mathbf{x}_{j} \) is the \( j \)-th received trace normalized by its maximal absolute value.

2. Wavelet estimation.

The assumption that the wavelet is identical for all the channels is now exploited. A seismic source is usually band-limited in reality, so the sparsity of the wavelet in the frequency domain is a reasonable assumption. The Fourier transform of the seismic trace, \( X_{j}[\omega] \) in (1) would be sampled in the frequency domain with a DFT, represented by \( \mathcal{F} \{ \cdot \} \), in order to create length-\( N_{\text{DFT}} \) vectors \( W_{j}, R_{j} \) and \( X_{j} \). After replacing \( R_{j} \) with the DFT of the
initial estimate of reflectivity, $R_j^0 = \mathcal{F}\{r_j^0\}$, the following minimization problem can be solved for the estimated wavelet in each channel:

$$\tilde{W}_j = \arg\min \|W_j\|_1 \text{ subject to } \|X_j - \text{diag}(R_j^0)W_j\|_2 \leq \|Z_j\|_2,$$

where $\text{diag}(R_j^0)$ is a diagonal matrix with entries from the vector $R_j^0$. Once an estimate of the wavelet is obtained for each channel, stacking is used to improve the SNR of the common wavelet estimate [8]. The stacking process is carried out using $\tilde{W}_j$ over all channels to obtain a single estimated wavelet $\tilde{w}$ in the time domain:

$$\tilde{w} = \Re\left[\mathcal{F}^{-1}\left\{\frac{1}{J} \sum_{j=1}^{J} \tilde{W}_j\right\}\right],$$

where $\Re$ represents the real part.

3. Refining the estimate of the reflectivity model. The following basis pursuit denoising problem is solved in order to estimate a sparse reflectivity series:

$$\hat{r}_j = \arg\min \|r_j\|_1 \text{ subject to } \|x_j - Cr_j\|_2 \leq \|z_j\|_2,$$

where $C$ represents the convolution matrix having entries related to $\tilde{w}$ and $x_j = [x_j[1], \ldots, x_j[N]]^T$.

4. Neural Network Architecture

The proposed method that makes use of the LSTM neural network-based autoencoder is to be explained in this section. The network architecture is shown in Figure 3. It consists of LSTM layers for performing the encoder/decoder functions.

![Figure 3. LSTM neural network for SNR enhancement with convolutional (Conv) and LeakyRelu layers.](image)

The LSTM layer in time series and sequence data learns protracted correlations among time steps. The layer state comprises the cell state $c_t$ and the hidden state (also known as the output state) $h_t$. The output of the LSTM layer for time step $t$ is stored in the hidden state at that time. Information from previous time steps is stored in the cell state. The layer updates the cell state at each time step by adding or removing the information. Finally, gates are used by the layer to control the updates.
The parameters of an LSTM layer are learnable and given as the bias $b$ (Bias), the recurrent weights $R$ (RecurrentWeights), and the input weights $w$ (InputWeights). The matrices $b$, $w$ and $R$ are concatenations of the bias, the recurrent weights, and the input weights of each component, respectively. The following equations illustrate how these matrices are concatenated:

$$
\begin{align*}
    b &= \begin{bmatrix} b_i & b_f & b_g & b_o \end{bmatrix}, \\
    R &= \begin{bmatrix} R_i & R_f & R_g & R_o \end{bmatrix}, \\
    w &= \begin{bmatrix} w_i & w_f & w_g & w_o \end{bmatrix},
\end{align*}
$$

where $o$, $g$, $f$ and $i$ denote the output gate, cell candidate, forget gate and input gate, respectively. At time step $t$, the cell state is updated as:

$$
c_t = f_t \odot c_{t-1} + i_t \odot g_t,
$$

where the symbol $\odot$ represents the element-wise multiplication of vectors, i.e., the Hadamard product. At time step $t$, the hidden state is updated as:

$$
h_t = o_t \odot \sigma_c(c_t),
$$

where $\sigma_c$ represents the state activation function. The LSTM layer function uses the softsign function in order to compute the state activation function, i.e.,

$$
\text{softsign}(x) = \frac{x}{1 + |x|}.
$$

The following formulas describe the components at time step $t$. The hidden and cell states of the layer are controlled by the following components:

- **Output gate ($o$)**: controls the cell state level that is added to the hidden state;
- **Cell candidate ($g$)**: adds information to the cell state;
- **Forget gate ($f$)**: controls the level of cell state reset (forget);
- **Input gate ($i$)**: controls the level of cell state update.

Output gate:

$$
o_t = \sigma_g(w_i x_t + R_i h_{t-1} + b_o)
$$

Cell candidate:

$$
g_t = \sigma_c(w_s x_t + R_s h_{t-1} + b_g)
$$

Forget gate:

$$
f_t = \sigma_g(w_f x_t + R_f h_{t-1} + b_f)
$$

Input gate:

$$
i_t = \sigma(g(w_i x_t + R_i h_{t-1} + b_i))
$$

The gate activation function is denoted by $\sigma_g$ in these calculations. The hard-sigmoid function used by the LSTM layer function is given by:

$$
\sigma_g(x) = \begin{cases} 
1 & \text{if } x > 2.5 \\
0 & \text{if } x < -2.5 \\
0.2x + 0.5 & \text{if } -2.5 \leq x \leq 2.5
\end{cases}
$$

5. Results and Discussion
5.1. Synthetic Data Results

In the synthetic simulation, we generate 20 traces using the reflectivity shown in Figure 4, where the sampling frequency is 500 Hz. The original traces shown in Figure 4 are the result of the convolution between the Ricker wavelet, having a center frequency of 40 Hz and a phase shift of 50 degrees, and the reflectivity series. The noise is modeled as the additive white Gaussian noise (AWGN) with an SNR of 10 dB, whereas the SNR is defined for the additive model $y = x + z$ as:
\[ \text{SNR} = 10 \log_{10} \left( \frac{\|x\|_2}{\|z\|_2} \right), \]

where \( x \) and \( z \) are the signal part and noise part, respectively.

*Figure 4.* Deconvolution result for \( Q = 0.8 \ Y \). Top to bottom: True reflectivity, received data with \( \text{SNR} = 10 \ \text{dB} \), BP method, SD method, proposed method (initial), proposed method (refined).

For the SD method, a deconvolution filter of length 51 is used. The filter is initialized with a solitary spike that is located approximately in the middle of the filter used for deconvolution. Other parameters used in the experiments are given as follows: \( \alpha = 0.2 \), \( \epsilon_f = 1 \), step size \( \mu = 0.02 \), the stopping criterion is marked as 1000 iterations, the hybrid norm parameter \( \epsilon_k \) is set to 0.005, and the regularization parameter \( \lambda \) is fixed to 2. The SD method’s iteration count is set to 800 to ensure that the algorithm reaches a steady
A number of efficient basis pursuit solvers are available to solve the embedded basis pursuit problem in the BP method. The SPGL1 package developed by [19] is utilized here to solve this $\ell_1$ minimization problem.

The estimated reflectivity series using different algorithms are depicted in Figure 4. We can see that the proposed method gives the spikiest result with the least noise, which is the most desirable for interpretation.

Power spectrum density (PSD) is used to evaluate the quality of the proposed method as well, presented in Figure 5. Among the three schemes, the proposed technique provides the flattest PSD, which is pretty close to the true PSD of the reflectivity series.

![Figure 5. Normalized PSD of the original synthetic traces, original reflectivity, and estimated reflectivity.](image)

The first iteration and the last iteration of the proposed method are shown in Figure 4. The intermediate iteration, with the recovery of the wavelet, is detailed in Figures 6 and 7. Note the exact match after the final iteration in the frequency domain. Upon obtaining the time-domain wavelet, a chunk of it (consisting of 51 samples and designated as a window) is used to refine the reflectivity in the BD algorithm’s final iteration.

To compare the performance of the proposed algorithm with the SD method and the BD method, two figures of merit are used: relative reconstruction error (RRE) and signal-to-reconstruction error ratio (SRER). They are defined by:

$$\text{RRE} = \frac{(r - \hat{r})^T(r - \hat{r})}{\hat{r}^T r}$$

and

$$\text{SRES} = 10 \log_{10} \frac{||r||_2^2}{||r - \hat{r}||_2^2}.$$  

Figures 8 and 9 show the parameters SRES and RRE for various SNR levels. As can be seen from these figures, the proposed algorithm outperforms the SD method and BD method algorithms in terms of SRES and RRE for all noise levels.
Figure 6. Frequency-domain views of the wavelet recovery process (Q = 0.8 Y). Top: frequency-domain first iteration; Bottom: frequency-domain final iteration.

Figure 7. Recovered and true wavelets.
For complexity analysis, the simulation time taken by the estimation methods is charted against the count of the traces. Figures 8–10 are the results of a Monte Carlo simulation performed with 20 different realizations of noise and seismic reflectivity. The BP method takes much less time, as depicted in Figure 10. We can see that when we simulate 200 traces, the BP method’s computation takes approximately 12% of the time needed for the SD method. In addition, for a simulation with this many traces, the SD method requires a prohibitively large amount of memory, so it fails to run on the same laptop machine. The computation time of the proposed approach is a bit high due to the training phase. However, in the proposed method, the number of parameters to tweak is very limited.
5.2. Field Data Results

The results of the SD, BD, and proposed methods are applied on a field seismic data set, and the results are presented in this section. The field seismic data set was obtained from the Line ID 51-101, Legacy Data Archive by USGS (1976), National Petroleum Reserve, Alaska (NPRA).

For the field data set, we run the SD method in windows with a duration of 0.35 s in time with the total number of traces set as 150. For the SD method, the deconvolution filter size is fixed to 21 samples, and a spike is marked at the filter’s center for the purpose of initialization. Furthermore, the step size for the SD method algorithm is set to $\mu = 0.1$. For the BP method, the window is the same as in the SD method. The deconvolution filter used for the SD method is estimated for the entire field data set. All three techniques’ remaining parameters are identical to those used in the synthetic data test.

Figure 11 shows the deconvolution results for the three algorithms together with the recorded data. Figure 12 depicts the relevant information of a zoomed-in field data section prior to and after the processing, i.e., deconvolution. The proposed technique has a more spiky deconvolution outcome than the other two algorithms, as shown by these results. The respective normalized power spectral densities (PSD) are shown in Figure 13. In terms of the power densities, the SD method provides significant gain in the 40–70 Hz band, whereas the BD method has a relatively flat normalized PSD. Figures 11 and 13 are plotted after applying RMS automatic gain control with a window length of 0.1 s to compensate for the lost amplitude after deconvolution.
Finally, as shown in Figure 14, another field data set consisting of a two-dimensional (2D) land line from east Texas, USA [20], is used for a performance comparison. The graph demonstrates the superior performance of the proposed method in comparison to the other methods.

In the BP method, several parameters require attention. First, $\tau$ in Equation (19) is the parameter that controls the number of peaks (i.e., sparsity) in the initial estimate of the per channel reflectivity. If we pick $\tau = 1$, the solution from (19) is a trivial solution that pinpoints only one peak at the highest value of $x_j[n]$. On the other hand, if $\tau$ is too large, the per channel estimate of the reflectivity is not sparse. Empirically, a value of 1.6 works well over a wide range of SNR for both synthetic and field data.
Figure 12. Deconvolution results. Top to bottom: Subset of received data for $2 \leq t \leq 3 \text{s}$ and 500 traces, BP method, SP method, Proposed (first iteration) and Proposed (last iteration).

Figure 13. Normalized PSD of the original field traces, original reflectivity, and estimated reflectivity.
Figure 14. Deconvolution results. Top to bottom: 100 traces of raw field data set, BP method, SP method, Proposed (first iteration), and Proposed (last iteration).

6. Conclusions

This study addresses the issue of recovering sparse reflectivity sequences from randomly quantized seismic data and thus proposes a robust reconstruction approach based on a neural network. The technique is beneficial for compressing massive amounts of seismic data. In experiments with both synthetic and field data sets, the proposed scheme significantly improves the solution’s robustness to changes in quantization error and thus enhances the visual saliency of properly salvaged reflectivity impulses compared with existing methods. During the training phase, only a small number of fundamental signals are chosen in order to reduce the time. In the experiments with the synthetic data, since the true reflectivity is known, the quality of the recovery is easy to evaluate. The proposed method is effective and robust against the quantization noise, i.e., for a wide range of SNRs, it outperforms the BD and SD methods in terms of relative reconstruction error and signal-to-reconstruction error ratio. All this is achieved at the expense of the computation time, which is a bit high due to the training phase. Finally, in the proposed method, the
number of parameters to tweak is very limited, making the scheme simple to implement in real-world applications.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data are available by contacting the corresponding author.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**

1. Kundur, D.; Hatzinakos, D. Blind image deconvolution. *Signal Process. Mag.* **1996**, *13*, 43–64. [CrossRef]
2. Ulrych, T.; Velis, D.; Sacchi, M.D. Wavelet estimation revisited. *Lead. Edge* **1995**, *14*, 1139–1143. [CrossRef]
3. Xu, G.; Liu, H.; Tong, L.; Kailath, T. A least-squares approach to blind channel identification. *IEEE Trans. Signal Process.* **1995**, *43*, 2982–2993.
4. Kaaresen, K.F.; Taxt, T. Multichannel blind deconvolution of seismic signals. *Geophysics* **1998**, *63*, 2093–2107. [CrossRef]
5. Ram, I.; Cohen, I.; Raz, S. Multichannel deconvolution of seismic signals using statistical MCMC methods. *IEEE Trans. Signal Process.* **2010**, *58*, 2757–2770. [CrossRef]
6. Nose-Filho, K.; Takahata, A.K.; Lopes, R.; Romano, J.M. A fast algorithm for sparse multichannel blind deconvolution. *Geophysics* **2016**, *81*, V7–V16. [CrossRef]
7. Kazemi, N.; Sacchi, M. Sparse multichannel blind deconvolution. *Geophysics* **2014**, *79*, V143–V152. [CrossRef]
8. Yilmaz, O. *Seismic Data Analysis: Processing, Inversion and Interpretation of Seismic Data*, 2nd ed.; SEG: Houston, TX, USA, 2001.
9. Zheng, F.; Ling, Y.; Tang, Y.; Hui, S.; Yang, H. A fidelity-restricted distributed principal component analysis compression algorithm for non-cable seismographs. *J. Appl. Geophys.* **2019**, *169*, 29–36. [CrossRef]
10. Iqbal, N. 1-ADM-CNN: A Lightweight In-field Compression Method for Seismic Data. *IEEE Trans. Circuits Syst. II Express Briefs* **2022**, *1*. [CrossRef]
11. Iqbal, N.; Zerguine, A.; Khan, S. OFDMA-TDMA-Based Seismic Data Transmission Over TV White Space. *IEEE Commun. Lett.* **2021**, *25*, 1720–1724. [CrossRef]
12. Iqbal, N.; Lawal, A.; Zerguine, A. Mitigating Wireless Channel Impairments in Seismic Data Transmission Using Deep Neural Networks. *Sensors* **2021**, *21*, 6105. [CrossRef] [PubMed]
13. Vennerød, C.B.; Kjærran, A.; Bugge, E.S. Long Short-Term Memory RNN. *arXiv* **2021**, arXiv:2105.06756.
14. Iqbal, N. DeepSeg: Deep Segmental Denoising Neural Network for Seismic Data. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *1*–*8*. [CrossRef] [PubMed]
15. Robinson, E.A.; Treitel, S. *Geophysical Signal Analysis*; Prentice Hall, Inc.: Hoboken, NJ, USA, 1980.
16. Gao, Z.; Hu, S.; Li, C.; Chen, H.; Jiang, X.; Pan, Z.; Gao, J.; Xu, Z. A Deep-Learning-Based Generalized Convolutional Model For Seismic Data and Its Application in Seismic Deconvolution. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–17. [CrossRef]
17. Iqbal, N.; Liu, E.; Mcclellan, J.H.; Al-Shuhail, A.A. Sparse Multichannel Blind Deconvolution of Seismic Data via Spectral Projected-Gradient. *IEEE Access* **2019**, *7*, 23740–23751. [CrossRef]
18. Murray, R.M.; Li, Z.; Sastry, S.S. *A Mathematical Introduction to Robotic Manipulation*; CRC Press: Boca Raton, FL, USA, 1994.
19. van den Berg, E.; Friedlander, M.P. Probing the Pareto frontier for basis pursuit solutions. *SIAM J. Sci. Comput.* **2008**, *31*, 890–912. [CrossRef]
20. Mousa, W.A.; Al-Shuhail, A.A. Processing of Seismic Reflection Data Using MATLAB™. *Synth. Lect. Signal Process.* **2011**, *5*, 1–97. [CrossRef]