Zero-offset data estimation using CNN for applying 1D full waveform inversion

Dawoon Lee1, Sungryul Shin1, Woohyun Son2 and Wookeen Chung1,*

1 Department of Ocean Energy and Resources Engineering, Korea Maritime and Ocean University, Busan, Korea
2 Petroleum and Marine Division, Korea Institute of Geoscience and Mineral Resources, Daejeon, Korea

*Corresponding author: Wookeen Chung. E-mail: wkchung@kmou.ac.kr

Received 20 August 2021, revised 2 November 2021
Accepted for publication 22 November 2021

Abstract
Full waveform inversion (FWI) in the time domain has limitations due to large computing time and memory requirements. Some studies have addressed this problem by using machine learning techniques. Most FWI studies using machine learning directly estimate the subsurface velocity structure by training the seismic data generated through various synthetic models to obtain the subsurface velocity structure. In this study, we propose a method to convert the common midpoint (CMP) gather to zero-offset data at a CMP location using a convolutional neural network (CNN) to increase the computing efficiency for the FWI. As the training data, we use synthetic data generated by the seismic exploration geometry and the source signature of field data. Since the proposed method performs FWI using the converted zero-offset data, it can be performed more efficiently than FWI using the existing multichannel data. However, it is difficult to apply a seismic exploration geometry and a source signature that has not been used for training. To verify the proposed method, it is applied to a synthetic model not used for learning as well as focused field data. It is confirmed that a proper subsurface velocity structure was obtained.

Keywords: full waveform inversion, zero-offset data, convolutional neural network, deep learning

1. Introduction
Full waveform inversion (FWI) is a technique used to obtain a high-resolution image of the subsurface using observed data from a seismic survey (Lailly & Bendnar 1983; Tarantola 1984; Pratt et al. 1998; Shin & Min 2006; Ray et al. 2016). However, generally, the FWI often faces unfavorable issues such as a high computing cost, nonlinearity, etc., due to the optimization method being based on a gradient and simulating the full-wave equation (Wang & Rao 2006; Virieux & Operto 2009; Fichtner 2010; Gholami & Siahkoohi 2010; Zhang & Lin 2020).

To improve the nonlinearity between the seismic data and the inverted parameters, Bunks et al. (1995) have proposed a multiscale seismic waveform inversion (Wang & Rao 2020), Sun & Alkhalifah (2020) have proposed a machine learning-descent optimization algorithm. Despite recent advances in computing technology, FWI still requires a high computing cost (Sun & Fu 2013). To reduce this high computing cost, Gholami et al. (2019) have used half the Nyquist rate to reduce the memory required for the time axis, and Kim et al. (2021) have used a discrete cosine transform (DCT)-based FWI method to reduce the memory for the spatial axis. In general, seismic data lack the low-frequency components to deal with media with a strong velocity contrast (Fei et al. 2012). Therefore, Shin & Cha (2008, 2009) have performed the waveform inversion in the Laplace domain to restore the lack of
low-frequency components, and Bozdag et al. (2011) have proposed the envelope inversion method using seismic data with ultra-low-frequency information to recover long wavelength components.

With the development of computing technology, research focusing on the application of machine learning to FWI has been performed (Röth & Tarantola 1994; Araya-Polo et al. 2018; Moseley et al. 2018; Wu & Lin 2018). Recently, machine learning-based FWI methods have been used to derive a model parameter directly from seismic data. For example, Li et al. (2019) and Liu et al. (2021) have directly derived the velocity model from seismic data using deep-learning inversion. Likewise, Wu & McMechan (2019) have proposed FWI with convolutional neural networks (CNNs) to derive the velocity model directly from seismic data. Nevertheless, the derivation of a velocity model from seismic data still presents many challenges. Alternatively, FWI with machine learning has been proposed to perform the inversion by preprocessing and reconstruction of seismic data or model parameters. For example, Sun & Demanet (2020) have performed FWI using extrapolated low-frequency information in seismic data with CNNs, and He & Wang (2021) have carried out reparameterized FWI using deep neural networks. However, estimating the velocity model using machine learning has a problem in that it is difficult to consider various seismic exploration environments such as the exploration geometry and source signature (Fabien-Ouellet & Sarkar 2020; He & Wang 2021). Although various methods for deriving a velocity model using machine learning have been proposed, it is still difficult to overcome the issue of computational efficiency in FWI using large-scale data.

In the conventional processing of seismic reflection data, an image of the subsurface is obtained by stacking data generated by using velocity analysis and normal moveout to the sorted data for the common midpoint (CMP) (Baker 1999; Roberts & Bally 2012). Here, the stacked data are assumed to be the zero-offset data at the normal-incident reflection for line velocity at the CMP (Yilmaz 2001). In general, to increase the accuracy of the inversion results, FWI performs the wave equation simulating the seismic exploration environment for a real field and uses the common shot gathers. If the stacked data are used when performing the FWI, the computational efficiency can be increased by assuming one-dimensional (1D) data instead of using the inversion data (Wang 2016). When performing 1D FWI using stacked data, the accuracy is lower than that of the FWI using multichannel data because the effect on the long offset is not considered, and its application for subsurface structures that vary strongly in the horizontal direction is limited. However, the application of 1D FWI using stacked data can improve the computational efficiency and has the advantage that it can be applied to large-scale multichannel data without a massive parallel computer system.

To successfully apply the stacked data to 1D FWI, two problems must be overcome. First, velocity analysis is usually required to generate stacked data. Since velocity analysis is time-consuming and human-intensive work, the technique for converting the CMP gather to the stacked data to minimize the human-intensive work is required (Wang et al. 2021). Second, 1D FWI is performed by solving the 1D wave equation with different geometric spreading effects from the observed data; therefore, to perform 1D FWI using stacked data, a geometric spreading correction is needed for the stacked data.

In this paper, we propose a method to convert multichannel data to zero-offset data at the CMP using a CNN by successfully applying the multichannel data to 1D FWI. The CNN architecture was designed to act as a feature extractor to convert the CMP gather to zero-offset data at the CMP. To improve the applicability of the field data for the proposed method, we trained the CNN using synthetic data generated by considering the seismic exploration geometry and estimated the source signature for the field data. We predicted the zero-offset data at the CMP by applying the CMP gather to the trained network and performed 1D FWI using the predicted data.

This paper is organized as follows. First, we describe in detail how we performed the proposed method. Next, we present our application of the trained network to two-dimensional (2D) synthetic data and a field data set. Then we confirm that the proposed method can effectively estimate the velocity model from the 2D data set. Finally, we demonstrate that the trained network can be applied to field data obtained on an exploration performed in the Ulleung Basin (offshore of eastern Korea) by the Korea Institute of Geoscience and Mineral Resources (KIGAM).

2. Methodology

We propose an algorithm for converting the CMP gather to zero-offset data in order to perform the 1D FWI. Here, a CNN was used to convert the CMP gather to zero-offset data. In addition, we generated training data using the seismic exploration geometry and the source signature obtained through source estimation to reflect the characteristics of the field data. Label data were generated through 1D modeling using the line-velocity model for the CMP position.

To generate the training data, the Marmousi2 model, SEG/EAGE Salt dome model and the overthrust model were used (Aminzadeh et al. 1994; Martin et al. 2002). Figure 1 depicts the workflow for the proposed method. The process of the proposed method consisted of preprocessing, source estimation, generating the CMP gathers, DCT compression, network training and FWI. First, preprocessing was performed to increase the signal-to-noise ratio of the seismic data. Second, to generate the training data, source
estimation was performed. Third, the training data were compressed using DCT to increase the computational efficiency when learning the network. Fourth, the designed network architecture was trained using compressed training data and label data. Finally, 1D FWI was performed using the predicted zero-offset data obtained by applying the trained network to the field data.

In this study, the applicability of the proposed method was analyzed by focusing on the exploration performed in the Ulleung Basin (offshore of eastern Korea) by KIGAM. In addition, the method was verified by applying the proposed method to the synthetic data generated using the same exploration geometry and source signature.

In the seismic exploration geometry, 1350 sources were excited in the horizontal direction at an interval of 25 m. The recording time was 3 s, and the sampling interval was 1 ms. The multichannel seismic data had 96 channels with a group interval of 12.5 m. The source depth and streamer depth are 5.0 and 7.0 m. The offset for horizontal displacement between source and receiver is 231.25–1425 m.

Figure 2 shows the fold values used for the field data. In this study, we converted the CMP gather with 24 folds into zero-offset data. The 1D FWI was performed using the zero-offset data obtained by applying the field data to the trained network. The source signature is generated by averaging the source estimated using the deconvolution method for 1350 shot gathers on the field (Kim et al. 2013). Also, the training data for network training is generated using the same estimated source signatures and the exploration geometry as field data.

### 2.1. DCT compression

Due to the size of the input data during network training, we used DCT compression to reduce the computational burden. The DCT is one of the methods used to apply image compression. We applied the 2D DCT and inverse DCT as follows (Wang 1984):

**DCT:**

$$x_{tr} = \sum_{i=0}^{Nt-1} \sum_{r=0}^{Nrec-1} y_{ij} \cos \left(\frac{\pi (2t + 1) i}{2Nt}\right) \times \cos \left(\frac{\pi (2r + 1) j}{2Nrec}\right),$$

$$x = 0, 1, \ldots, Nt - 1, \quad z = 0, 1, \ldots, Nrec - 1$$

**IDCT:**

$$y_{ij} = \sum_{i=0}^{Nt-1} \sum_{r=0}^{Nrec-1} \alpha_i \alpha_r x_{tr} \cos \left(\frac{\pi (2t + 1) i}{2Nt}\right) \times \cos \left(\frac{\pi (2r + 1) j}{2Nrec}\right),$$

$$i = 0, 1, \ldots, Nt - 1, \quad j = 0, 1, \ldots, Nrec - 1$$
Figure 3. (a) Original CMP gather, (b) data after DCT, (c) reconstructed CMP gather back to the original size after compressing the data into 500 of 3000, which is the time axis, in a $24 \times 3001$ original CMP gather, and (d) the difference between each type of data.

$$
\begin{align*}
\alpha_t &= \begin{cases} 
\frac{1}{\sqrt{Nt}} & t = 0 \\
\sqrt{\frac{2}{Nt}} & 1 \leq t \leq Nt - 1
\end{cases}, \\
\alpha_r &= \begin{cases} 
\frac{1}{\sqrt{Nrec}} & r = 0 \\
\sqrt{\frac{2}{Nrec}} & 1 \leq r \leq Nrec - 1
\end{cases},
\end{align*}
$$

where $\alpha_t$ and $\alpha_r$ are the input data and transformed data, respectively, and $Nt$ and $Nrec$ are the time index size and the number of receivers, respectively. As shown in equation (1), the DCT coefficients were obtained by calculating the sum of the cosine functions oscillating at different frequencies. To compress the data using DCT, we used only a portion of the coefficients from the transformed DCT domain. Figure 3 shows the reconstructed data back to its original size after compressing the data into 500 units from 3001 units, which is the time axis, in a $24 \times 3001$ CMP gather. We compared the traces for the receiver position at 681.25 m to see the difference in the detailed results (figure 4). The reconstructed data and the original data were confirmed to have almost the same amplitude and waveform. In this study, the CMP gathers with 24 folds and 3001 time indexes were used as the training data containing 24 folds and 500 compressed time indexes.

2.2. Network architecture for training

To convert the CMP gather to zero-offset data, we designed a network architecture with 17 neural networks, excluding the input layer (figure 5). To obtain the feature of zero-offset data in the CMP gather, we constructed the CNN architecture. Table 1 shows the detailed architecture of the designed network. To extract the features for correction of geometric spreading effect and conversion to zero-offset data from CMP gather, we designed network architecture that includes a convolution 2D layer, average pooling layer and activation layer (ReLU). We include dropout layer and batch normalization layer for avoiding overfitting and improving the network performance. Also, we used root-mean-square error (RMSE) as a loss function. RMSE is defined as follows:

$$
RMSE = \sqrt{\frac{1}{N} \sum (y - \hat{y})^2},
$$

where $y$ is the corresponding label data, $\hat{y}$ is the predicted data, $\bar{y}$ is the mean of the corresponding label data and $N$ is compressed sample size. The time-series data generated by performing 1D modeling using the line-velocity model at the CMP were used as the label data, and the CMP gather was...
used as the input data. By using the label data generated using the same estimated source as the training data, we expected to be able to correct the geometry spreading effect.

Figure 6 presents some feature maps used to convert the CMP gather to zero-offset data using the designed network architecture for neural network layers 6, 9 and 12 (shown in Table 1).

**2.2.1. Synthetic training model set.** In this study, to train the designed network architecture, we generated training data for various subsurface structure models. The Marmousi2 model, SEG/EAGE salt dome model and overthrust model were used as the synthetic models to generate the training data. The Marmousi2 and the overthrust models were modified to generate the training data according to the subsurface shape. The salt dome model was used by modifying the 2D model for 12 lines extracted from the existing SEG/EAGE 3D salt dome model. The number of models and is 15 for the Marmousi2, eight for the overthrust model and 12 for salt dome model. The sizes of the Marmousi2, overthrust and salt dome models were $17.0 \times 3.04375$, $20.0 \times 2.5$ and $13.5 \times 2.8125$ km, respectively, with spacing of 6.25 m in the vertical and horizontal directions.

**2.2.2. Generation of trained data and label data.** In this study, we generated the training data using the exploration geometry and the source signature, the same as the field data obtained in the Ulleung Basin (offshore of eastern Korea).

Here, the input and label data of the training dataset were generated by solving the 2D and 1D acoustic wave equations.
Figure 6. The feature maps to convert the CMP gather to zero-offset data using the designed network architecture for neural network layers 6, 9 and 12 (shown in Table 1).

in the finite-difference time domain. Moreover, 1D modeling was performed using the same source signature and line-velocity model at the CMP to generate the label data.

The source signature was estimated through source estimation using the deconvolution method proposed by Kim et al. (2013). Figure 7 shows the time-series and Fourier spectra of the estimated source. We used 96 receivers at intervals of 25 with the same offset to the field data in each model to generate input data set. The number of sources used in each model is 593 for the Marmousi2, 718 for the overthrust model and 461 for the salt dome model. The number of CMP gather and the zero-offset data set used in each model is 8550 for the Marmousi2 model, 5560 for the overthrust model and 5256 for the salt dome model, and the total is 19366. The sampling interval is 0.5 ms and the recording time is 3 s. We resampled the generated CMP gathers and zero-offset data to sampling intervals, which is the same as the field data. We also compressed the input data from $24 \times 3001$ to $24 \times 500$ and the label data from 3001 to 500 using DCT to reduce the computational burden during training. By using the same estimated source to generate the input and label data, we expected to be able to correct the geometry spreading effect. Furthermore, when performing 1D FWI using the predicted zero-offset data, the same source could be used. Finally, the designed network was trained using the input data and label data with the direct wave removed.

2.3. Training strategy

In this study, we performed network training using the ‘Deep-Learning Toolbox’ in MATLAB (Kim 2017). In addition, Gaussian noise between 5 and 30 dB was randomly added to the training data using the ‘awgn’ command in MATLAB to improve the robustness of the network. Since the dropout layer and batch normalization layer were included in the designed network to reduce overfitting, a validation split to confirm overfitting in total data was applied at a relatively low 10%, and then a test split of 3% was applied. All of the hyperparameters of the network are shown in Table 2. In hyperparameters, we used an initial learning rate of 0.001 for stable training while avoiding overfitting. Since the input data size used for training in this study was very large compared
Table 2. Detailed hyperparameters of the network.

| CNN parameters       | Value          |
|---------------------|----------------|
| Number of CMP gathers | 19,366         |
| Trained data        | 16,850 (87%)   |
| Validation data     | 1936 (10%)     |
| Test data           | 580 (3%)       |
| Optimizer           | Adam           |
| Activation          | ReLU           |
| Loss function       | RMSE           |
| Dropout rate        | 0.5            |
| Initial learning rate | 0.001       |
| Epochs              | 30             |
| Batch size          | 20             |
| Shuffle             | Every-epoch    |
| L2 regularization   | 0.0005         |

To the size used for training of a general network, we used the benchmark size of 20 considering our computer performance and efficiency of training networks. To obtain the regularization results while avoiding the overfitting problem, a dropout layer ratio of 50% was used according to the proposal by Baldi & Sadowski (2013) and L2 regularization value of 0.0005. We used Adam as the optimizer of the designed network to optimize our RMSE loss function. Adam is a computationally efficient algorithm for first-order gradient-based optimization with few memory requirements (Kingma & Ba 2014). We generally used ReLU activation functions, which are known to facilitate the development of very deep neural networks due to their non-saturating properties (Hahnloser et al. 2000). Finally, we shuffled our dataset at every-epoch to reduce overfitting and variance. Figure 8 depicts the loss curves for the training data set and validation set with respect each epoch iteration.

3. Numerical examples

To verify our proposed method, a numerical test using a 2D synthetic dataset was first performed. Second, the field data test was conducted using the benchmarked field data acquired in the Ulleung Basin (offshore eastern Korea). We performed the 1D FWI using the zero-offset data in each example. Table 3 shows the hardware specifications of the computer on which the network training and the 1D FWI were executed. In the two cases, the conjugate gradient method (Polyak 1969) was used for local optimization, and the direct method (Pica et al. 1990) was used to calculate the step length.

3.1. Numerical test

The proposed method was applied to synthetic data in 2D SEG advanced modeling (SEAM) with a complex salt dome model. The velocity model used for the test was generated by resampling the original 2D SEAM model (figure 9). The size of the model was $560 \times 526$. The 1301 sources were evenly distributed in the horizontal direction at 25-m intervals. The recording time was 3.0 s, and the sampling interval was 0.5 ms. Each source had 96 channels of the streamer type at 12.5-m intervals.

First, CMP sorting was performed to apply the trained network to the synthetic data. We used 1278 of the CMP gathers out of all 5112 to consider the same geometry used in the network training. Figure 10 shows the near-offset data in the CMP gathers and the label data obtained using the line-velocity model at the CMP. Figure 11 depicts the predicted data obtained by applying the trained network to the CMP gather. When the predicted data and the label data were compared, it was confirmed that the amplitude phase including the first-arrival traveltime was well predicted for the first-arrival waveform. However, it was demonstrated that the quality of the prediction data was low for the salt dome and

![Figure 8. Loss curves for the training and validation data sets during the training process.](https://academic.oup.com/jge/article-lookup/doi/10.1093/jge/gkz052/6516508)
multiple signals. For a more detailed comparison, the time-series data at CMPs of 5, 15, and 30 km were extracted from the predicted data and the label data (figure 12). Using the predicted data, it was confirmed that the first-arrival signal was well predicted, but it was difficult to predict an accurate amplitude and waveform information in the subcomplex structure. To evaluate the accuracy of the predicted data obtained from the proposed method, we compared the RMSE and the coefficient of determination ($R^2$) between the predicted data and the label data (figure 13). $R^2$ for evaluation of predicted data is defined as follows:

$$R^2(y, \hat{y}) = 1 - \frac{\sum (y - \hat{y})^2}{\sum (y - \bar{y})^2}.$$  \hfill (5)

The findings reveal that the proposed method provided relatively stable results when applied to the CMP gather for a relatively simple structure; however, it had a low $R^2$ value in a complex structure.

Figure 14 shows the initial velocity model obtained by applying a smoothing operator to the true model. We performed the 1D FWI using the predicted data shown in figure 11, and the result was obtained in 950 iterations. When
Figure 13. The RMSE curve (red line) and $R^2$ (blue line) values for the predicted data and label data.

Figure 14. The initial velocity model for FWI.

Figure 15. The inverted velocity model obtained through 1D FWI using (a) label data and (b) the predicted data.

performing the inversion calculation, 20 cores were used, and a computing time of 6439 s was required. Figure 15 depicts the inversion results using the predicted data and the label data. Moreover, figure 16 reveals the summed RMSE curves for all CMPs at each iteration. Additionally, the inverse results for data at CMPs of 5, 15 and 30 km were compared (figure 17). By comparing the 1D FWI results performed using label data and predicted data, we confirm that the trend of the inversion results performed using label data was similar to the predicted data.
3.2. Field data test

After verifying the proposed method with a numerical test, we applied the proposed method to 2D marine seismic data. We used CMP gathers with as much as 1327 out of all 5308 to consider the same geometry used in the network training. Figure 18 shows the near-offset data in the CMP gather and the predicted data obtained by applying the trained network to the CMP gather. The predicted data show continuity of seismic events, but abnormal signals in the form of a flat layer were confirmed to appear at a position of 15–27 km (red box). We performed the 1D FWI using the predicted data (figure 18b), and the result was obtained in 50 iterations. A velocity model that increases linearly in the depth direction was used as the initial velocity model for FWI. Figure 19 depicts the initial velocity model and inversion results using the predicted data. Furthermore, figure 20 shows the summed RMSE curves for all CMPs at each iteration.
When performing the inversion calculation, 20 cores were used and a computing time of 8186 s was required.

4. Discussion

The proposed method has some weaknesses. First, a lot of computing time was required to generate the training data. Second, it was difficult to predict the complex subsurface structures using the CMP gather. Third, in the case of the velocity model obtained through 1D FWI using the predicted data, there might have been discontinuities with the velocity model at the CMP axis since this model used 1D time-series data. Fourth, the training data were generated considering 2D geometry spreading, so training data considering 3D geometry spreading are required. The proposed method can be applied to the seismic exploration data acquired in the same exploration geometry and source signature, but if not, it is thought that the reliability in the application will be lower.

5. Conclusions

We propose a zero-offset data conversion method by applying the CMP gather to 1D FWI using a CNN. In this paper, a network architecture was designed by focusing on the field data acquired from the Ulleung Basin in the East Sea, and the designed network was verified through synthetic data testing. As a result of applying the synthetic data, it was confirmed that the prediction was difficult for the complex structure model, but the result with the simple structure model was similar to the label data. By comparing the two results of the FWI using the predicted and label data in the synthetic data test, we demonstrated that reasonable results could be obtained when the results found by applying the trained network to data not used for training were used.

In addition, we predicted the zero-offset data by applying the trained network to field data and performed 1D FWI using the predicted data. The predicted data showed the continuity of seismic events well, but it was confirmed that abnormal signals appeared in certain areas. Importantly, when performing 1D FWI using predicted data on field data, we obtained a velocity model over a wide area with a low computing cost.

In conclusion, when using the method proposed in this study, we generated a reasonable velocity model with a low computing cost through 1D FWI using predicted data. Also, we could expect the generated velocity model to be used as an initial velocity model for 2D FWI. In the case of a trained network, it is anticipated that it will be possible to provide higher quality results by improving the training data generated from various velocity models through transfer learning.
Acknowledgements

This research was supported by the Basic Research Project (grant no. 20-3313) of the Korea Institute of Geoscience and Mineral Resources (KIGAM) funded by the Ministry of Science, ICT and Future Planning of Korea, and the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (grant no. 2021R111A3045763).

Conflict of interest statement. None declared.

References

Aminzadeh, F., Burkhard, N., Nicoletis, L., Rocca, F. & Wyatt, K., 1994. SEG/EAGE 3-D modeling project: 2nd update, *The Leading Edge*, 13, 949–952.

Araya-Polo, M., Jennings, J., Adler, A. & Dahlke, T., 2018. Deep-learning tomography, *The Leading Edge*, 37, 58–66.

Baker, G.S., 1999. *Processing Near-Surface Seismic-Reflection Data: A Primer*. SEG Books.

Baldi, P. & Sadowski, P., 2013. Understanding dropout, *Advances in neural information processing systems*, 26, 2814–2822.

Bozdağ, E., Trampert, J. & Tromp, J., 2011. Misfit functions for full waveform inversion based on instantaneous phase and envelope measurements, *Geophysical Journal International*, 185, 845–870.

Bunks, C., Saleck, F.M., Zaleski, S. & Chavent, G., 1995. Multiscale seismic waveform inversion, *Geophysics*, 60, 1457–1473.

Fei, T.W., Luo, Y., Qin, F. & Kelamis, P.G., 2012. Full waveform inversion without low frequencies: a synthetic study. In 2012 SEG Annual Meeting. OnePetro.

Fichtner, A., 2010. *Full Seismic Waveform Modelling and Inversion*. Springer Science and Business Media.

Fabien-Oulette, G. & Sarkar, R., 2020. Seismic velocity estimation: a deep recurrent neural-network approach, *Geophysics*, 85, U21–U29.

Gholami, A. & Shahkoohi, H.R., 2010. Regularization of linear and nonlinear geophysical ill-posed problems with joint sparsity constraints, *Geophysical Journal International*, 180, 871–882.

Gholami, A., Richardson, A., Zand, T. & Malcolm, A., 2019. Efficient wavefield reconstruction at half the Nyquist rate, *Geophysics*, 84, A43–A46.

Hahnloser, R.H., Sarpeshkar, R., Mahowald, M.A., Douglas, R.J. & Seung, H.S., 2000. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit, *Nature*, 405, 947–951.

He, Q. & Wang, Y., 2021. Reparameterized full-waveform inversion using deep neural networks, *Geophysics*, 86, V1–V13.

Kingma, D. & Ba, J., 2014. Adam: a method for stochastic optimization. In ICLR.

Kim, P., 2017. *MATLAB Deep Learning: With machine learning, neural networks and artificial intelligence*, 130 (21).

Kim, S., Chung, W.K. & Lee, J.H., 2021. Acoustic full waveform inversion using discrete cosine transform (DCT), *Journal of Seismic Exploration*, 30, 365–380.

Kim, Y., Cho, Y. & Shin, C., 2013. Estimated source wavelet-incorporated reverse-time migration with a virtual source imaging condition, *Geophysical Prospecting*, 61, 317–333.

Lailly, P. & Bednar, J., 1983. The seismic inverse problem as a sequence of before stack migrations. In: *Conference on Inverse Scattering, Theory and Application*, Society for Industrial and Applied Mathematics, Philadelphia, 206–220.

Li, S., Liu, B., Ren, Y., Chen, Y., Yang, S., Wang, Y. & Jiang, P., 2019. Deep-learning inversion of seismic data, *IEEE Transactions on Geoscience and Remote Sensing*, 58, 2135–2149.

Liu, B., Yang, S., Ren, Y., Xu, X., Jiang, P. & Chen, Y., 2021. Deep-learning seismic full-waveform inversion for realistic structural models, *Geophysics*, 86, R31–R44.

Martin, G.S., Marfurt, K.J. & Larsen, S., 2002. Marmousi-2: an updated model for the investigation of AVO in structurally complex areas 21. SEG Technical Program, Expanded Abstracts, 1979–1982.

Moseley, B., Markham, A. & Nissen-Meyer, T., 2018. Fast approximate simulation of seismic waves with deep learning, *arXiv: Geophysics*.

Pica, A., Diet, J.P. & Tarantola, A., 1990. Nonlinear inversion of seismic reflection data in a laterally invariant medium, *Geophysics*, 55, 284–292.

Polyak, B.T., 1969. The conjugate gradient method in extreme problems, *USSR Computational Mathematics and Mathematical Physics*, 9, 94–112.

Pratt, R.G., Shin, C.S. & Hicks, G.J., 1998. Gauss-Newton and full Newton methods in frequency-space seismic waveform inversion, *Geophysical Journal International*, 133, 341–362.

Ray, A., Sekar, A., Hoversten, G.M. & Albertin, U., 2016. Frequency domain full waveform elastic inversion of marine seismic data from the Alba field using a Bayesian trans-dimensional algorithm, *Geophysical Journal International*, 205, 915–937.

Roberts, D.G. & Bally, A.W., 2012. Regional geology and Tectonics: Principles of Geologic Analysis. Elsevier.

Roth, G. & Tarantola, A., 1994. Neural networks and inversion of seismic data, *Journal of Geophysical Research*, *Solid Earth*, 99, 6753–6768.

Shin, C. & Chay, Y.H., 2008. Waveform inversion in the Laplace domain, *Geophysical Journal International*, 173, 922–931.

Shin, C. & Chay, Y.H., 2009. Waveform inversion in the Laplace-Fourier domain, *Geophysical Journal International*, 177, 1067–1079.

Shin, C. & Min, D., 2006. Waveform inversion using a logarithm waveform, *Geophysics*, 71, R31–R42.

Sun, B. & Alkhalifah, T., 2020. ML-descent: an optimization algorithm for full-waveform inversion using machine learning, *Geophysics*, 85, 1ND–Z30.

Sun, H. & Demanet, L., 2020. Extrapolated full-waveform inversion with deep learning, *Geophysics*, 85, R275–R288.

Sun, W. & Fu, L.Y., 2013. Two effective approaches to reduce data storage in reverse time migration, *Computers and Geosciences*, 56, 69–75.

Tarantola, A., 1984. Inversion of seismic reflection data in the acoustic approximation, *Geophysics*, 49, 1259–1266.

Virieux, J. & Operto, S., 2009. An overview of full-waveform inversion in exploration geophysics, *Geophysics*, 74, WCC1–WCC26.

Wang, W., McMechan, G.A., Ma, J. & Xie, F., 2021. Automatic velocity picking from semblances with a new deep-learning regression strategy: Comparison with a classification approach, *Geophysics*, 86, U1–U13.

Wang, Y., 2016. *Seismic Inversion: Theory and Applications*, Wiley Blackwell, 187–201.

Wang, Y. & Rao, Y., 2006. Crosshole seismic waveform tomography - I. Strategy for real data application, *Geophysical Journal International*, 166, 1224–1236.

Wang, Y. & Rao, Y., 2020. Seismic waveform modeling and tomography, *Encyclopedia of Solid Earth Geophysics* (2nd edition), Springer, Cham.

Wang, Z., 1984. Fast algorithms for the discrete W transform and for the discrete Fourier transform, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 4, 803–816.

Wu, Y. & Lin, Y., 2018. InversionNet: a real-time and accurate full waveform inversion with CNNs and continuous CRFs, *arXiv: Signal Processing*.

Wu, Y. & McMechan, G.A., 2019. Parametric convolutional neural network-domain full-waveform inversion, *Geophysics*, 84, R881–R896.

Yilmaz, Ö., 2001. *Seismic data analysis*. Society of Exploration Geophysicists. SEG Investigations in Geophysics.

Zhang, Z. & Lin, Y., 2020. Data-driven seismic waveform inversion: a study on the robustness and generalization, *IEEE Transactions on Geoscience and Remote Sensing*, 58, 6900–6913.