
A SELF-SUPERVISED LEARNING FRAMEWORK FOR SEISMIC LOW-FREQUENCY EXTRAPOLATION

Shijun Cheng, Randy Harsuko, Tariq Alkhalifah

King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia.
sjcheng.academic@gmail.com, {mohammad.randycaesario, tariq.alkhalifah}@kaust.edu.sa

Yi Wang, Qingchen Zhang

Innovation Academy for Precision Measurement Science and Technology,
Chinese Academy of Sciences, Wuhan 430077, China.
{wangyi, qczh}@apm.ac.cn

ABSTRACT

Full waveform inversion (FWI) is capable of generating high-resolution subsurface parameter models, but it is susceptible to cycle-skipping when the data lack low-frequency. Unfortunately, the low-frequency components (< 5.0 Hz) are often tainted by noise in real seismic exploration, which hinders the application of FWI. To address this issue, we develop a novel self-supervised low-frequency extrapolation method that does not require labeled data, enabling neural networks to be trained directly on real data. This paradigm effectively addresses the significant generalization gap often encountered by supervised learning techniques, which are typically trained on synthetic data. We validate the effectiveness of our method on both synthetic and field data. The results demonstrate that our method effectively extrapolates low-frequency components, aiding in circumventing the challenges of cycle-skipping in FWI. Meanwhile, by integrating a self-supervised denoiser, our method effectively performs simultaneously denoising and low-frequency extrapolation on noisy data. Furthermore, we showcase the potential application of our method in extending the ultra-low frequency components of the large-scale collected earthquake seismogram.

Plain Language Summary

Full waveform inversion (FWI) is a method used to provide detailed underground images for efforts in oil exploration or studying earthquakes. However, the method is prone to a problem known as "cycle-skipping", caused by the lack of low frequencies in the data, and as a result, the inversion converges to an inaccurate velocity model. We propose a new way to enhance the low-frequency components using a neural network-based self-supervised approach. This means our method learns directly from real data, rather than relying on artificially created data, which is a common limitation in the supervised paradigm. Our method not only helps to overcome the issue of skipping cycles but also demonstrates robustness against noisy data, enhancing its practical application potential. The tests on exploration data validate its ability to predict the low-frequency components, helping to avoid local minima and thus improving the accuracy of FWI. Also, we demonstrate how our method can be applied to invert earthquake data, where it can extend the ultra-low frequency information. This research could contribute to providing a better and a more reliable way to obtain images of the Earth's interior.

Keywords Self-supervised learning · Low-frequency extrapolation · Iterative data refinement

1 Introduction

Full waveform inversion (FWI) can provide high-resolution subsurface parameter model by minimizing the misfit between observed and modeled seismic waveforms. This approach has drawn a lot of attention [1, 2, 3, 4, 5, 6, 7] since

it was proposed by Tarantola [8, 9]. However, one common problem for FWI is the high nonlinearity of the problem, resulting in many local minima, which can prevent the algorithm from converging to the global minimum. To address this problem, various strategies and algorithms have been developed to help eliminate or escape local minima in FWI.

Bunks et al. [10] introduced a multi-scale strategy that initiates with lowest frequencies, and then progressively adds higher frequencies to the inversion. This method, applied widely, attempts to help FWI avoid local minima. It accomplishes this goal by smoothing the objective function in the early stages and subsequently allows higher resolution information as the inversion advances. Following this work, Boonyasirawat et al. [11] provided an optimal multi-scale frequency selection algorithm to achieve computational efficiency in the time domain. Besides, Gao et al. [12] and Ren and Liu [13] also attempted to refine the multi-scale algorithm to improve the inversion quality of FWI. Nevertheless, these methods do not address the issue of the lack of low frequencies in the data.

In order to build the long-wavelength components of the subsurface parameter model in the case of lacking low-frequencies, Ha and Shin [14] and Kim et al. [15] proposed the Laplace-domain waveform inversion. However, this method has the drawback that the penetration depth of the Laplace-domain inversion depends on the offset and the choice of Laplace damping constants. Hence, numerous studies aimed to adjust the norm of the misfit function to minimize the occurrence of local minima. Ma and Hale [16] utilized the dynamic warping algorithm, mitigating FWI's reliance on an accurate initial model by estimating the travel time shift between observed and synthetic data. Pointed out by Wu et al. [17] and Luo and Wu [18], the seismic data's envelope potentially holds ultra low-frequency signals, offering an avenue to estimate long-wavelength velocity structures even in the absence of low-frequency information. Furthermore, Warner and Guasch [19] introduced the adaptive waveform inversion (AWI) to counter the cycle-skipping problem. This method employs a convolutional filter, transforming predicted data into observed data. Yang and Engquist [20] and Yang et al. [21] explored the optimal transport approach, measuring amplitude differences and global phase shifts, offering a solution to circumvent cycle-skipping issues. Sun and Alkhalifah [22] showed that AWI is part of a more general framework based on matching filters and optimal transport. Leveraging well logging data, Li et al. [23] devised a structure-guided interpolation algorithm to construct the background velocity model for FWI. Yao and Wang [24] crafted a full-waveform inversion starting model from wells, employing dynamic time warping and convolutional neural networks. Another prominent technique for velocity model construction is the reflection waveform inversion (RWI) [25], requiring migration and demigration to update the wavepath. RWI typically incorporates a correlation-type objective function to address amplitude challenges encountered without true amplitude [26, 27, 28, 6, 29, 30, 31]. However, RWI demands significantly higher computational resources compared to conventional FWI methods.

Recently, machine learning has been demonstrated as significant potential in the field of seismic processing due to its ability to approximate any nonlinear function [32, 33, 34, 35, 36, 37], which also showed a potential in velocity model building. As a result, some researchers have proposed replacing traditional physics-based FWI with convolutional neural networks (CNNs) to directly learn a mapping from seismic data to velocity models in a data-driven manner [38, 39, 40]. These trained CNN models can then be applied directly to unseen test data to predict velocity models. Typically, the training of such CNNs requires a large dataset and is conducted in a supervised learning (SL) manner. For example, Yang and Ma [41] employed the classic U-Net network architecture as a baseline, trained it on a self-constructed dataset, and then tested it on 2D simulated models and the SEG salt model, demonstrating its potential in constructing velocity models. Wu and Lin [42] integrated a CNN with a conditional random field to form a hybrid network model, aiming to enhance the network's ability to refine the velocity fields near faults and boundaries. Zhang and Gao [43] suggested the use of images in the common-shot domain, replacing shot gathers, and then established an iterative deep CNN to produce high-resolution velocity models. Yang et al. [44] also utilized low-resolution velocity models, images, and well velocity constraints to reconstruct high-resolution velocity models. While these data-driven approaches offer a new direction for seismic inversion, they often encounter challenges in generalization due to the complexity of real subsurface structures and property distributions. Moreover, the constructed training datasets often fail to cover real underground features. To address the lack of physical interpretability of purely data-driven approaches, some researchers have proposed integrating a forward simulator based on the wave equation into the neural network (NN)-based seismic inversion [45, 46, 47]. This simulator is employed to forward model the subsurface properties predicted by the NN. The discrepancy between the resulting seismic records and the original observations is then used as a misfit to update the network. This approach not only eliminates the need for labeled data but also enhances generalization to some extent. However, these methods have still not achieved satisfactory results on field data and often face challenges, such as unclear inversion boundaries and overly blurred background velocities in situations when low-frequencies are absent [41].

Instead of using NNs to approximate an inversion operator, can we leverage their low-frequency extrapolation capabilities, and thus, extend the low-frequency information of observed data. Compared to the daunting task of directly approximating the inversion operator with NNs, using them to learn low-frequency extension might be more realistic. Some researchers have initiated preliminary explorations and applied this concept within FWI. Ovcharenko et al. [48] were among the first to suggest using a feed-forward artificial neural network (ANN) to approximate the nonlinear

relationship between frequency-wavenumber spectra with limited bandwidth and the low-frequency data and, thus, recovering low-frequency information. To tackle the inherent challenge of ANNs, wherein the number of trainable parameters surges with increasing input data, Ovcharenko et al. [49] further transitioned from ANNs to CNNs and applied it for low-frequency extrapolation in multi-offset seismic data. Instead of working in the frequency-wavenumber domain, Sun and Demanet [50, 51] proposed using a CNN to directly reconstruct low-frequency components trace-by-trace from band-limited data in the time domain. Furthermore, Sun and Demanet [52] extended their proposed framework to the low-frequency extrapolation of multicomponent elastic wave data. Hu et al. [53] and Jin et al. [54] developed progressive transfer learning algorithms, which embed a physics-based FWI module to iteratively update the training dataset, thereby reducing the feature discrepancy between it and the test dataset. These studies have provided validation of the feasibility of using NNs for seismic data’s low-frequency extrapolation. However, they only tested on synthetic data and rarely showcased applications on field data. Thus, the performance on field data is our ultimate goal.

Achieving good low-frequency extension results on field data using data-driven methods remains a challenge. Consequently, relatively little attention has been devoted to enhancing the performance of NNs in low-frequency extrapolation on field data. Fabien-Ouellet [55] utilized recurrent CNN to simultaneously achieve seismic denoising and low-frequency generation, validating the method’s performance on both synthetic and field data. Additionally, Nakayama and Blacquiere [56] developed a multi-task seismic processing framework based on NNs, capable of concurrently suppressing the blending noise, interpolating missing traces, and recovering low-frequency information. Fang et al. [57] employed a CNN architecture based on convolutional autoencoders to learn the relationship between low-frequency and high-frequency data, thereby predicting low-frequency components from high-frequency data. They showcased the effectiveness of low-frequency recovery in land field data and conducted a comparative analysis between FWI and RTM products using both the original dataset and the dataset enhanced with NN-predicted low-frequency components. In a similar vein, Wang et al. [58] applied a dense CNN methodology to broaden the low-frequency spectrum of prestack viscoacoustic seismic data, yielding promising outcomes when tested on marine field data. Ovcharenko et al. [59] developed a multi-task processing framework, which not only recovers low-frequency information but also provides a smooth subsurface background model. This was applied to marine data involving the elastic assumption. Although these methods have demonstrated a certain ability in recovering low frequencies on field data, their performance significantly lags behind their results on synthetic data. This discrepancy is attributed to their reliance on an SL approach where training occurs solely on synthetic data before being directly applied to field data. It is evident that field and synthetic data have considerable feature differences due to physical approximations made during synthetic data generation. Consequently, the features captured by the NN on synthetic data do not generalize well when applied to field data. To bridge the feature gap between the synthetic and field data, Alkhalifah et al. [60] introduced a domain-adaptive method, namely MLReal, which embeds real features of field data into synthetic data to enhance low-frequency extrapolation performance on field data. However, this preprocessing might eliminate some characteristics of seismic data during the conversion, such as phase. Therefore, a more viable alternative is to train directly on field data. It allows the network to extract frequency features from field data more directly, and thus, contributes to low-frequency tasks on field data.

Indeed, predicting low frequency data using unsupervised learning poses significant challenges. To our knowledge, we have only come across two studies addressing this issue. One study by Sun et al. [61] developed a semi-supervised learning approach by incorporating a domain adaptation approach. They trained a generative model, CycleGAN, using synthetic low-frequency shot gathers and paired field band-limited shot gathers. The trained CycleGAN demonstrated its capability to extrapolate low-frequency components during testing on field data. However, during its training phase, CycleGAN still captures features inherent to the synthetic data, which might pose generalization issues when predicting field data. For instance, field data may exhibit anisotropic effects, but synthetic data simulated under the isotropic assumption may not accurately represent the field data. Meanwhile, training CycleGANs is not trivial. Since it requires simultaneous training of both discriminator and generator models, it relies on expert tuning and hyperparameter selection. Another work was proposed by Wang et al. [62], employing a self-supervised learning (SSL) method for low-frequency extension in seismic data. They first trained an NN using paired seismic data sets missing the low-frequency components. These paired data sets share the same high-frequency cutoff but differ in their low-frequency components. The dataset with the relative low-frequency component, sourced from original data, serves as labels. In contrast, the data devoid of the relative low-frequency components, achieved by filtering out the low-frequency part from the original data, acts as the NN’s input. Subsequently, they employed the trained network to generate low-frequency components. However, since the training data for this method only extracts labels from the original data missing low frequencies, its performance in low-frequency reconstruction relies on the low-frequency range of the original seismic data. Moreover, downsampling significantly reduces the time sampling length of the seismic data, leading to a loss of high-frequency information and a consequent decline in time resolution.

In line with the work by Wang et al. [62], our study introduces a Lesslow2Low (L2L) framework. This approach involves applying a high-pass filter to the initial observed seismic data, which already lacks low-frequency components, thereby creating input data with even fewer low-frequency contents. The original observed data serve as pseudo-labels.

In fact, the initial inspiration for L2L came from the classic denoising algorithm Noisier2Noise [63], which adds noise to the original noisy data to create data with stronger noise. This noisier data is considered as input, while the original noisy data acts as pseudo-labels, allowing the network to be trained in an SSL fashion. The subtle distinction between the L2L framework and the method of [62] lies in our ability to apply high-pass filters with varying cutoff frequencies during the epoch progression, as opposed to a fixed approach. However, the L2L framework also faces similar problems: the network’s low-frequency extrapolation performance is limited by the low-frequency range of the original observed data. We present an example that substantiates this point and leads to two important findings: 1. The reason for the L2L framework’s limited performance, compared to the SL method, is the frequency information bias between the training set employed in L2L and that used in the SL method. 2. The NN, which is trained using the L2L framework, demonstrates a certain degree of low-frequency extension capability with the original observed data. Although this capability is limited, if we can iteratively refine the frequency information bias between the original observed data and the ideal ground truth, we can gradually approach the low-frequency extrapolation performance of the SL method.

Motivated by these findings, we develop an effective SSL low-frequency extrapolation algorithm and apply it to FWI. In our approach, the NN training is divided into two stages: warm-up and iterative data refinement (IDR). The difference between these stages lies in the training sets used. During the warm-up stage, the original observed data acts as the pseudo-labels, and the input data is generated by applying a high-pass filter to the observed data. In the IDR stage, the pseudo-labels for the current epoch are derived from the predictions made by the network trained in the previous epoch on the original observed data, and the input data are obtained by high-pass filtering these predicted pseudo-labels. The warm-up stage, which precedes the IDR stage, aids in stabilizing the network and initially capturing data characteristics, thus providing some degree of low-frequency extrapolation capability. The IDR stage gradually narrows the gap between the predicted pseudo-label and the ideal ground truth, thereby enhancing the network’s low-frequency extrapolation performance. We first conduct tests on synthetic data to validate the effectiveness of our method and the ability to avoid cycle-skipping local minima. Subsequently, we explore the application potential of our method in exploration field data to improve the accuracy of FWI. Lastly, we implement our algorithm on extensive earthquake seismogram data to conduct low-frequency extrapolation testing, assessing its capability to extend the spectrum to extremely low-frequency components.

2 Method

2.1 FWI

Using the whole information of seismic data, FWI can construct a high-resolution subsurface model by iteratively minimizing the misfit between observed and simulated data. The conventional misfit objective function is given by the l_2 norm as follows:

$$\chi(\mathbf{m}) = \frac{1}{2} \|d^{obs}(\mathbf{x}_r, \mathbf{x}_s) - d^{syn}(\mathbf{x}_r, \mathbf{x}_s)\|_2^2, \quad (1)$$

where the superscripts *obs* and *syn* denote the observed and simulated seismic data, respectively, \mathbf{x}_r and \mathbf{x}_s represent the coordinates of the receivers and sources, respectively. FWI attempts to find a medium capable of generating simulated data that closely align with the observed ones. Since FWI is a waveform matching process, there would be mismatching problems between the simulated and observed seismograms when the initial model is poor, leading to a time delay greater than $T/2$ (where T is the period of a wavelength), that is, the so-called cycle-skipping problem. The poor initial model often leads FWI to local minima.

In order to overcome this problem, researchers have proposed many of the aforementioned algorithms, mostly by modifying the misfit function norm [16, 17, 19, 20, 64]. Actually, the root cause of the cycle-skipping problem is the velocity model lacking the long-wavelength components, which is often constructed with the low-frequency data information [65]. Unfortunately, in real seismic exploration, data with frequencies below 5 Hz are usually contaminated by noise, thus rendering unusable low-frequency data. Unlike the aforementioned methods, we try to predict the low-frequency data information for the observed data using an innovative machine learning algorithm.

To enhance our focus on the phase information and to avoid the amplitude challenge, we adopt a convolution-based misfit function [66], which has a similar performance to the correlative norm in RWI.

$$\chi(\mathbf{m}) = \frac{1}{2} \|d^{obs}(\mathbf{x}_r, \mathbf{x}_s) * d^{syn}(x_{ref}, \mathbf{x}_s) - d^{syn}(\mathbf{x}_r, \mathbf{x}_s) * d^{obs}(x_{ref}, \mathbf{x}_s)\|_2^2, \quad (2)$$

where the subscript *ref* denotes a reference trace, which is selected from the near-offset traces with a high signal-to-noise ratio. Moreover, to improve the robustness against background noise, we further introduce a hybrid-norm

objective function into equation (2).

$$\chi_{hybrid}(\mathbf{m}) = \sqrt{1 + \frac{\|\Delta\hat{d}(\mathbf{x}_r, \mathbf{x}_s)\|^2}{\epsilon^2}} - 1, \quad (3)$$

where $\Delta\hat{d}(\mathbf{x}_r, \mathbf{x}_s) = \hat{d}^{obs}(\mathbf{x}_r, \mathbf{x}_s) - \hat{d}^{syn}(\mathbf{x}_r, \mathbf{x}_s)$, \hat{d}^{obs} and \hat{d}^{syn} represent the first and second convolution terms in equation (2). As for the damping coefficient, we follow the criterion as

$$\epsilon = c \cdot \text{mean}(|\hat{d}^{obs}|), \quad (4)$$

where c is a constant ranging between 0.1 \sim 100.0. The smaller c is the greater the added gain to the weak signal, which helps enhance the robustness of FWI to noise.

2.2 Supervised low-frequency extrapolation

When seismic waves travel through the Earth's interior, the stratigraphic filtering effect restricts the frequency content of the observed data. This restriction is especially obvious at lower frequencies, resulting in a conspicuous absence of low-frequency components in the observed data. Therefore, the recorded seismic data can be expressed as

$$d^{obs} = H[d^{real}], \quad (5)$$

where d^{obs} represents the observed data with an absence of low-frequency components, $H[\cdot]$ symbolizes the high-pass filter effect of the Earth's strata on the seismic waves, and d^{real} denotes the broad band ground-truth data.

The feasibility of an NN-based data-driven approach for low-frequency extrapolation has been validated [49, 51]. Generally, we can employ the SL technique to train an NN that provides a non-linear mapping. This mapping translates data restricted in frequency bandwidth, particularly lacking in low-frequency components, to a dataset that encompasses the low-frequency components. We can represent this operation as follows:

$$d^{real} = \text{NN}(d^{obs}, \theta), \quad (6)$$

where θ is the learnable parameters of network NN. Due to the unavailability of labels in field data, a synthetic dataset is typically constructed to facilitate the training of the NN.

2.3 LessLow2Low

Compared to training on synthetic data, direct training on real data can significantly enhance the generalizability of NNs, thereby yielding superior low-frequency extrapolation on field data. However, real data typically lacks low-frequency information, which raises the question: how can we conduct effective training under these constraints? Inspired by a denoising method from the machine learning community, known as Noisier2Noise [63], we can devise a strategy, called LessLow2Low (L2L), that facilitates training directly on real data. Within the Noisier2Noise framework, we are equipped solely with the original noisy data, which serves as a pseudo-label. The input data is generated by introducing additional noise to the already noisy original data. By constructing a noisier-noisy dataset in this manner, we can provide a substrate for training the NN. Analogously, we, in our L2L framework, assume we only have a waveform dataset that lacks low-frequency components, in which such data can be treated as pseudo-label data. Subsequently, a high-pass filter is applied to this pseudo-label data to obtain the input for the NN. In this context, the high-pass filtered data, relative to the original waveform data, possess less low-frequency components, thereby enabling the establishment of an SSL regime on real data. Nevertheless, considering the higher nonlinearity in the relation between low and high frequency data, this paradigm often provides limited capabilities for low-frequency extrapolation. In the following, we will present an example to illustrate this point.

We simulate synthetic data for the Marmousi2 model with a dominant frequency of 15 Hz, followed by the preparation of three distinct training datasets. These datasets share the same input data, derived by subjecting the simulated synthetic data to a high-pass filter with a cutoff of 10 Hz. However, the label data for each set are unique. For the first dataset, the label data come from the original synthetic dataset, designated as low-full. The second dataset's label data are obtained by applying a high-pass filter with a cutoff frequency at 5 Hz to the original synthetic data, identified as lesslow-low 1. The third dataset's label data are filtered to remove frequencies below 7 Hz from the original synthetic data, termed lesslow-low 2. The first dataset serves as the foundation for the SL of the NN. The rationale behind constructing the latter two datasets is to consider the reality that real data often lacking low-frequency components. We train NNs using these three datasets, all under the same training configuration as elucidated in subsequent sections.

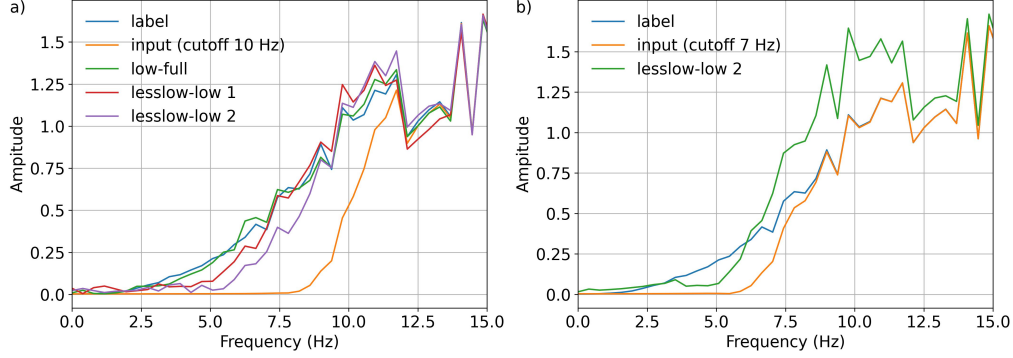


Figure 1: (a) Comparison of amplitude spectrum curves from the prediction results of networks trained on different datasets. (b) The prediction results of the network trained on the lesslow-low 2 dataset for the original input data, which lack frequency components below 7 Hz, are compared with the amplitude spectrum curves of the input data and their corresponding labels.

We employ a test dataset, from which frequencies below 10 Hz have been filtered out, to assess the low-frequency extrapolation capabilities of networks trained on three distinct training sets. We extract a single trace from their prediction results for the test data and plot the corresponding amplitude spectra (see Figure 1a), comparing them with the test data and their associated labels. We observe that the network trained on the low-full dataset achieves superior low-frequency extrapolation, which is definitively attributed to the network being trained in an SL fashion. In contrast, the NNs trained on the two lesslow-low datasets exhibit somewhat diminished performance. This substantiates the preceding contention that a training paradigm which uses data lacking low-frequency components as labels can only furnish a limited capacity for low-frequency extrapolation. For example, the label data in the lesslow-low 1 dataset are devoid of frequency components below 5 Hz. Therefore, they do not include information in their prediction distribution for frequencies below 5 Hz contained in the test data. An additional finding is that the network, trained on lesslow-low 1, exhibits better low-frequency extrapolation performance than the network trained on lesslow-low 2. This can be attributed to the fact that the lesslow-low 1 dataset has a lower frequency information bias relative to the lesslow-low 2 dataset when compared with the low-full dataset.

An additional test involving using the network trained on the lesslow-low 2 dataset to predict the corresponding pseudo-label data, which lack below 7 Hz frequencies. The single-trace amplitude spectrum from the prediction is compared with the pseudo-label data, as well as with the original synthetic data, which is depicted in Figure 1b. We can see that the network trained on lesslow-low data is capable of extending the frequency to a certain extent from the original pseudo-label data.

The two illustrative examples impart two key insights: First, if we can mitigate the frequency information bias between the lesslow-low and low-full datasets, we can incrementally approach the frequency extrapolation performance of a network trained on the low-full dataset. Second, the networks trained on lesslow-low datasets exhibit a certain capacity to extrapolate lower frequencies in the pseudo-label data involved in their training. From these insights, we can infer that if we iteratively refine the low-frequency component information of the pseudo-label data using the network trained on the lesslow-low data, we could ultimately approximate the frequency extrapolation performance of an SSL framework on real data. Motivated by this inference, in the following section, we will present our SSL framework for low-frequency extrapolation.

2.4 Self-supervised low-frequency extrapolation

Drawing on the key findings from the previous section, we develop an SSL framework for low-frequency extrapolation. Our framework principally consists of two components: a warm-up and an iterative data refinement (IDR) phases. The detailed algorithmic procedure is delineated in Algorithm 1. In the following, we will elucidate the key components therein.

Firstly, the NN undergoes a warm-up phase. We begin by simply employing an L2L procedure to apply a high-pass filter to the original seismic data $\{x_i\}_{i=1}^N$, which lacks low-frequency components, thereby creating a lesslow-low dataset $\{H[x_i], x_i\}_{i=1}^N$. Subsequently, the NN is subjected to multiple epochs of optimization training on this dataset. The objective of this process is to enable rapid stabilization of the neural network, allowing it to preliminarily capture the characteristics of seismic data. Furthermore, the second finding from the previous section indicates that this pre-trained

network, denoted as NN_w , exhibits a certain degree of low-frequency extrapolation ability with respect to the original label data. This capability forms the groundwork for the iterative refinement of the training set in subsequent stages.

Subsequently, the NN enters the IDR phase. In this stage, we first leverage the pre-trained network NN_w to predict the original seismic data. The predictions serve as the initial pseudo-labels for the IDR phase, with corresponding inputs derived from applying a high-pass filter to these predictions. This procedure facilitates the creation of a new lesslow-low dataset, employed for the first epoch of training during the IDR stage:

$$NN_0 \leftarrow \{(H[NN_w(x_i)], NN_w(x_i))\}_{i=1}^N, \quad (7)$$

where the network NN_0 is directly initialized from the pre-trained network NN_w .

Informed by the first insight from the previous section, we anticipate that after training for one epoch on the new training set, the low-frequency extrapolation capability of NN_0 will slightly surpass that of NN_w . This improvement is attributed to the fact that, compared to the lesslow-low dataset used during the warm-up phase, the new lesslow-low dataset exhibits lower frequency representation in the labels compared to the low-full dataset. During the subsequent training, we iteratively perform this process to progressively diminish the frequency information bias between the lesslow-low and low-full datasets. Specifically, for each training epoch, we first employ the network trained in the previous epoch (e.g., NN_{j-1}) to predict the original data $\{x_i\}_{i=1}^N$, and thus, obtain the frequency-extended pseudo-labels $\{NN_{j-1}(x_i)\}_{i=1}^N$. Then, we apply a high-pass filter to these pseudo-labels to generate corresponding input data $\{H[NN_{j-1}(x_i)]\}_{i=1}^N$. The resulting lesslow-low training set $\{(H[NN_{j-1}(x_i)], NN_{j-1}(x_i))\}_{i=1}^N$ will optimize the network NN_j for one epoch, for example,

$$NN_j \leftarrow \{(H[NN_{j-1}(x_i)], NN_{j-1}(x_i))\}_{i=1}^N. \quad (8)$$

After conducting multiple epochs of training in the IDR phase, our network incrementally aligns with the low-frequency extrapolation capabilities of a model trained on the low-full dataset.

Algorithm 1 Self-supervised low-frequency extrapolation

Input: Raw band-limited seismic data $\{x_i\}_{i=1}^N$.

Input: Neural network model NN.

Input: High-pass filter $H[\cdot]$.

Input: E_{warmup} : The number of epochs during the warm-up phase.

Input: E_{idr} : The number of epochs during the iterative data refinement phase.

Warm-up phase

Output: Pre-trained model NN_w

- 1: Randomly initialize network parameters θ
- 2: **for** j **in** E_{warmup} **do**
- 3: Perform the high-pass filter on raw seismic data $H[x_i]$
- 4: Construct the lesslow-low dataset $\{H[x_i], x_i\}_{i=1}^N$
- 5: Optimize the network NN_w on the lesslow-low dataset
- 6: $NN_w \leftarrow \{H[x_i], x_i\}_{i=1}^N$
- 6: **end for**

Iterative data refinement phase

Output: Final low-frequency extrapolation model $NN_{E_{idr}}$

- 7: Predict the raw band-limited seismic data $\{NN_w(x_i)\}_{i=1}^N$
 - 8: Construct the new refined lesslow-low dataset
 - 9: $\{H[NN_w(x_i)], NN_w(x_i)\}_{i=1}^N$
 - 9: Initialize the network $NN_0 = NN_w$
 - 10: Optimize the network NN_0 on the new lesslow-low dataset for one epoch
 - 10: $NN_0 \leftarrow \{H[NN_w(x_i)], NN_w(x_i)\}_{i=1}^N$
 - 11: **for** $j \rightarrow 1$ **in** E_{idr} **do**
 - 12: Generate the low-frequency extended data $\{NN_{j-1}(x_i)\}_{i=1}^N$
 - 13: Construct the new refined lesslow-low dataset
 - 13: $\{H[NN_{j-1}(x_i)], NN_{j-1}(x_i)\}_{i=1}^N$
 - 14: Initialize the network $NN_j = NN_{j-1}$
 - 15: Optimize the network NN_j on the new lesslow-low dataset for one epoch
 - 15: $NN_j \leftarrow \{H[NN_{j-1}(x_i)], NN_{j-1}(x_i)\}_{i=1}^N$
 - 16: **end for**
-

2.5 Network architecture and training procedure

Our SSL framework employs a classic network architecture, namely U-Net [67], which has been widely utilized in the NN-based seismic processing workflows [32, 41, 37]. Figure 2 comprehensively details the network architecture utilized in this study. It's noteworthy that we did not strictly follow the classic U-Net structure, but instead make several modifications to suit our specific requirements. The first significant change is in scaling: rather than adopting the four scales typical of the traditional U-Net, we incorporate five scales, involving five 2x2 downsampling and 2x2 upsampling operations. The rationale behind this modification is that different scales often represent different frequency component information. By extracting features across multiple scales, we aim to train an NN that is more attuned to various frequency components, which in turn is expected to enhance the network's capability in low-frequency extrapolation. The second modification involves the skip connections at the network's maximum scale: they directly receive the input data, rather than data that have undergone transformations through an input layer. This strategy is adopted to circumvent the loss of original signal information that can be caused by the network's depth. Since CNNs could potentially create a smoothing effect, which might entail the loss of high-frequency signals, we deliberately avoid impairing these high-frequency signals during the process of low-frequency extrapolation. The third modification exists in the convolution layers: whereas the classic U-Net baseline employs a 3x3 convolution layer followed by batch normalization (BN) and a Leaky Rectified Linear Unit (LeakyReLU) activation function, our experiments revealed that incorporating BN would produce unstable low-frequency signals during the IDR phase. Therefore, we omit BN to ensure that the network provides a reliable solution.

During the training process, we present a hybrid loss function to co-optimize the network. This hybrid loss function consists of a data loss \mathcal{L}_d and an amplitude spectrum loss \mathcal{L}_a . The data loss measures the difference between the NN's outputs O_i , $i = 1, \dots, N$ and the corresponding pseudo-labels L_i , $i = 1, \dots, N$, using the mean absolute error (MAE) as follows:

$$\mathcal{L}_d(L, O) = \frac{1}{N} \sum_{i=1}^N |L_i - O_i|. \quad (9)$$

The amplitude spectrum loss \mathcal{L}_a is obtained by computing the misfit in amplitude spectra between the network's output and the pseudo-labels, also employing the MAE metric as follows:

$$\mathcal{L}_a(L, O) = \frac{1}{N} \sum_{i=1}^N |\text{AS}(L_i) - \text{AS}(O_i)|, \quad (10)$$

where the symbol $\text{AS}(\cdot)$ represents the operation for obtaining the amplitude spectrum.

The total loss function is defined as

$$\mathcal{L}(L, O) = \mathcal{L}_d(L, O) + \epsilon \cdot \mathcal{L}_a(L, O), \quad (11)$$

where ϵ is a hyperparameter, which is used to regulate the proportion of the amplitude spectrum loss within the total loss. In our numerical examples, we set it to a constant value of 0.01. The role of loss \mathcal{L}_a and the setting of the hyperparameter ϵ will be thoroughly analyzed in the discussion section. We utilize the AdamW optimization algorithm [68] in our training stage. The implementation employs a GeForce RTX 8000 graphics processing unit and leverages the PyTorch framework.

3 Numerical examples

3.1 Synthetic Data

In order to verify the reasonability and accuracy of the proposed algorithm, we first use the simulated data generated with Marmousi model to perform the following tests. The modified model size is $2.20\text{km} \times 6.11\text{km}$ with a spatial interval of 10.0m along both horizontal and vertical directions. Figure 3 displays the true Marmousi velocity model with the modified dimensions. A Ricker wavelet with dominant frequency of 15 Hz is used as the source signal to generate the mimic observed seismic data. We generate a total of 122 shot gathers, each recording 2401 time steps with a time interval of 0.002s. The distance between shots is 50m, with the initial shot located at the model's far left at zero km. From these 122 shot gathers, we extract a total of 8200 data patches, each sized 128x128. For each patch we randomly choose the cutoff low-frequencies between 5 ~ 15 Hz, forming our original observed dataset. During the warm-up phase, we randomly filter out frequency components below 6 ~ 30 Hz from the original observed data to generate the input data. As previously described, the pseudo-label data at this time is the original observed data. In the IDR phase, the high-pass filter cutoff frequency range for the initial 50 epochs is set between 5 ~ 7 Hz. Then, every 50 epochs, we increase the maximum cutoff frequency by 2 Hz, while the minimum cutoff frequency remains at 5 Hz.

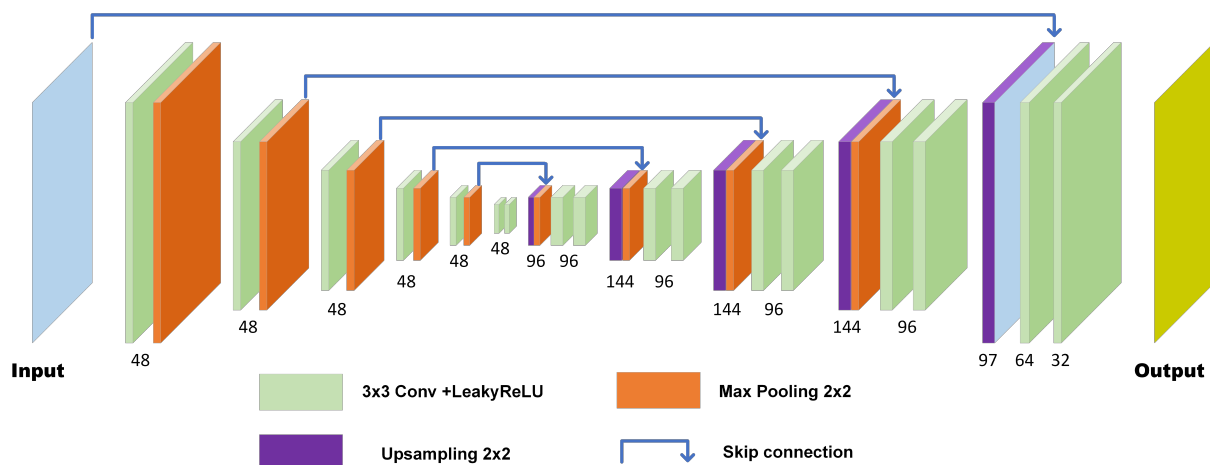


Figure 2: The neural network architecture used in our study.

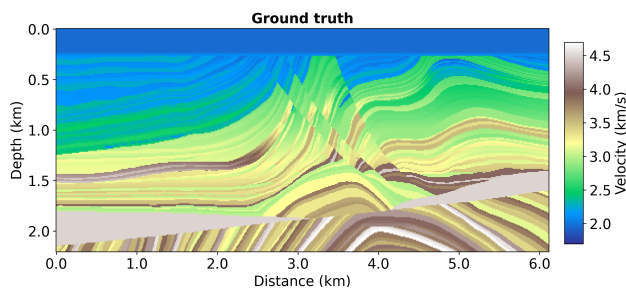


Figure 3: True Marmousi velocity model.

Once the maximum cutoff frequency reaches 15 Hz, it is kept fixed. The network is trained for a total of 550 epochs, with the warm-up phase accounting for 50 epochs. The learning rate start with $3e-4$, then decreased by a factor of 0.8 at the 65, 130, 195, 260, and 325 epochs.

Figure 4a shows the seismogram of a simulated shot gather. We use a high-pass filter to cut off different frequency components, as shown in Figures 4b, 4d and 4f filtered with 5 Hz, 10 Hz, 15 Hz high-pass filters, respectively. After processing with our trained network, we can recover the missing components and the corresponding recovered data are shown in Figures 4c, 4e and 4g.

To clearly verify the performance of the proposed algorithm in reconstructing the low-frequency components, we perform a spectral analysis of the traces at different locations between the original, filtered and predicted data, as displayed in Figures 5-7. The predicted amplitude spectrum of data with cutoff frequency of 15 Hz (Figures 5) and 10 Hz (Figure 6) exhibit a good recovery of the filtered out components after applying the proposed algorithm. The reconstructed low-frequency components (green line) commendably match the original true information (orange line) at the missing band range. As for the more practical case with cutoff frequency of 5 Hz (Figure 7), the proposed extrapolation algorithm can still recover the missing low-frequency information even to 1.5 Hz. Therefore, Figures 5-7 prove the accuracy, stability and reasonability of the proposed SSL low-frequency extrapolation algorithm.

With the predicted data after low-frequency extrapolation, we perform the following FWI tests. Figure 8 shows the inversion results with cutoff frequency of 10 Hz. Figure 8a displays the smoothed initial velocity model, which is far away from the true model in Figure 3. We choose 5 frequency bands with dominant frequencies of 3 Hz, 4 Hz, 5 Hz, 8 Hz and 15 Hz to perform the multi-scale inversion, Figure 8b is the corresponding result of the original full-band data, matching the true model perfectly. If we directly use the data with cutoff frequency of 10.0 Hz to conduct the test, the inversion result is contaminated with artifacts because of the missing low-frequency information (see Figure 8c). In comparison, we can reconstruct the subsurface velocity model well with the predicted seismic data (see Figure 8d). To precisely show the inversion results, we choose the profiles at $X=2.5$ km, 3.5 km and 4.5 km and exhibit them in Figure 9. We can see that the direct inversion of filtered data (red line) is far away from the true model (blue line).

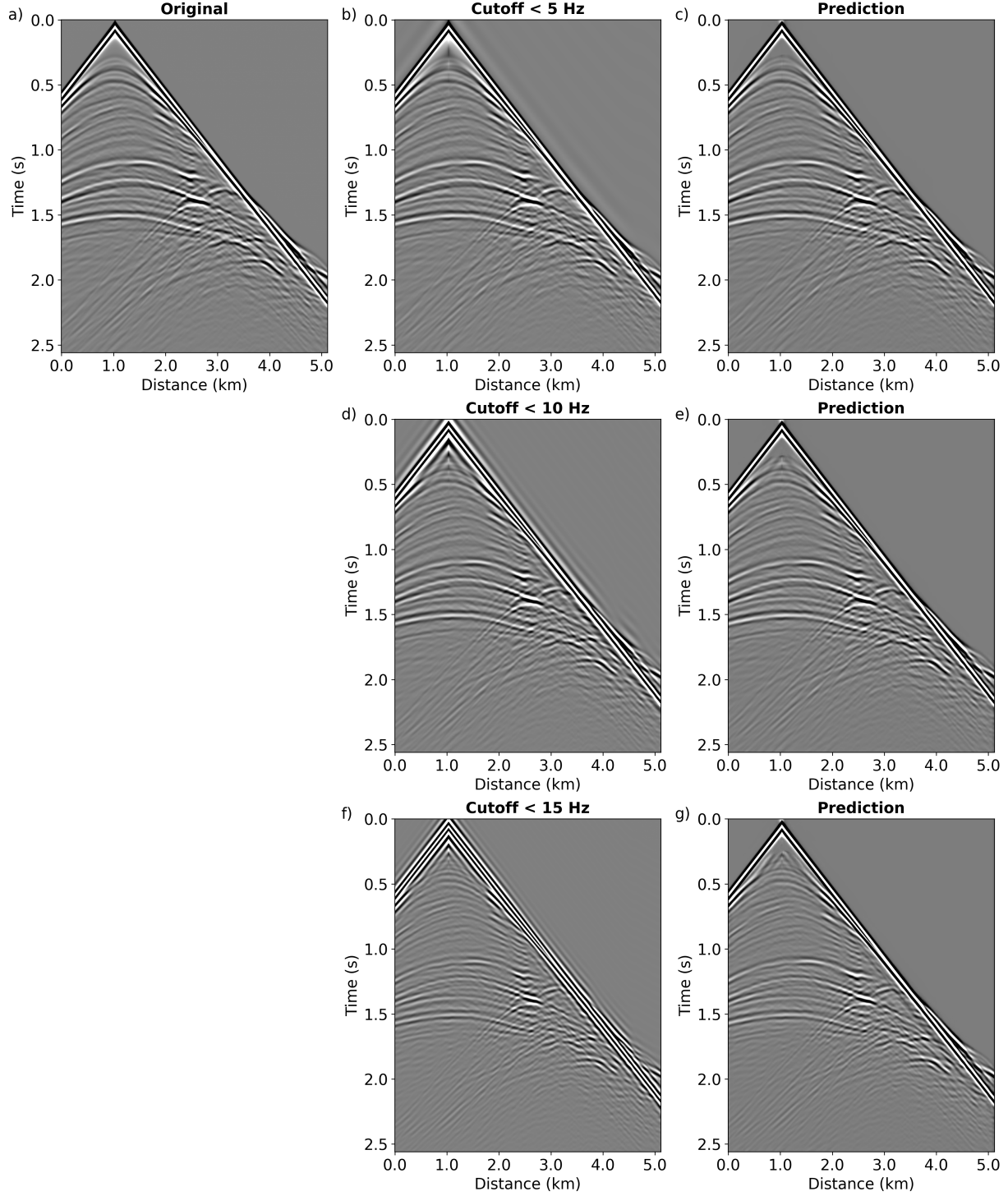


Figure 4: The extrapolation results for the test shot gathers missing different low-frequency components. (a) the original full band data; data filtered by different high-pass filters: (b) 5 Hz, (d) 10 Hz and (f) 15 Hz; the corresponding data recovered with the proposed SSL algorithm: (c) 5 Hz, (e) 10 Hz and (g) 15 Hz.

Encouragingly, the inversion result of the predicted data (purple line) is similar to that of the full-band original data (green line), both of which match the true model perfectly.

Self-supervised low-frequency extrapolation

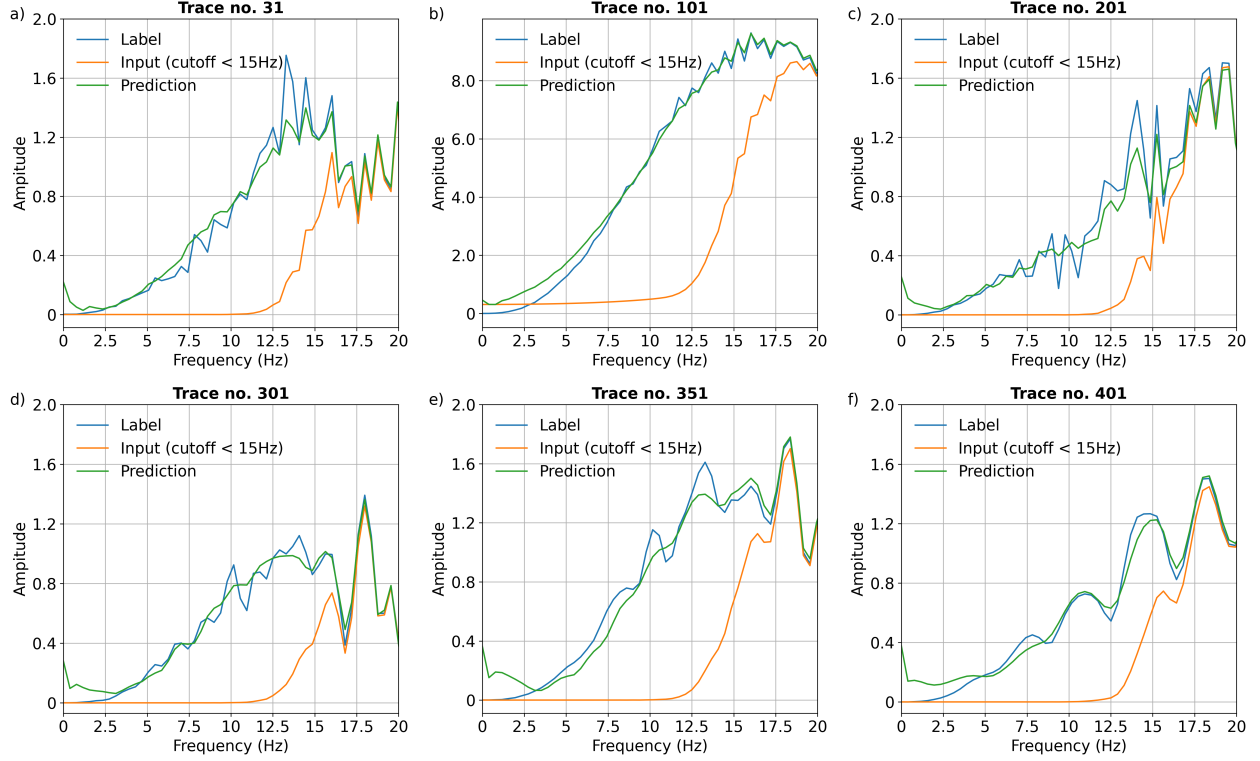


Figure 5: The amplitude spectrum comparison at different locations. The 10 Hz high-pass filter is used.

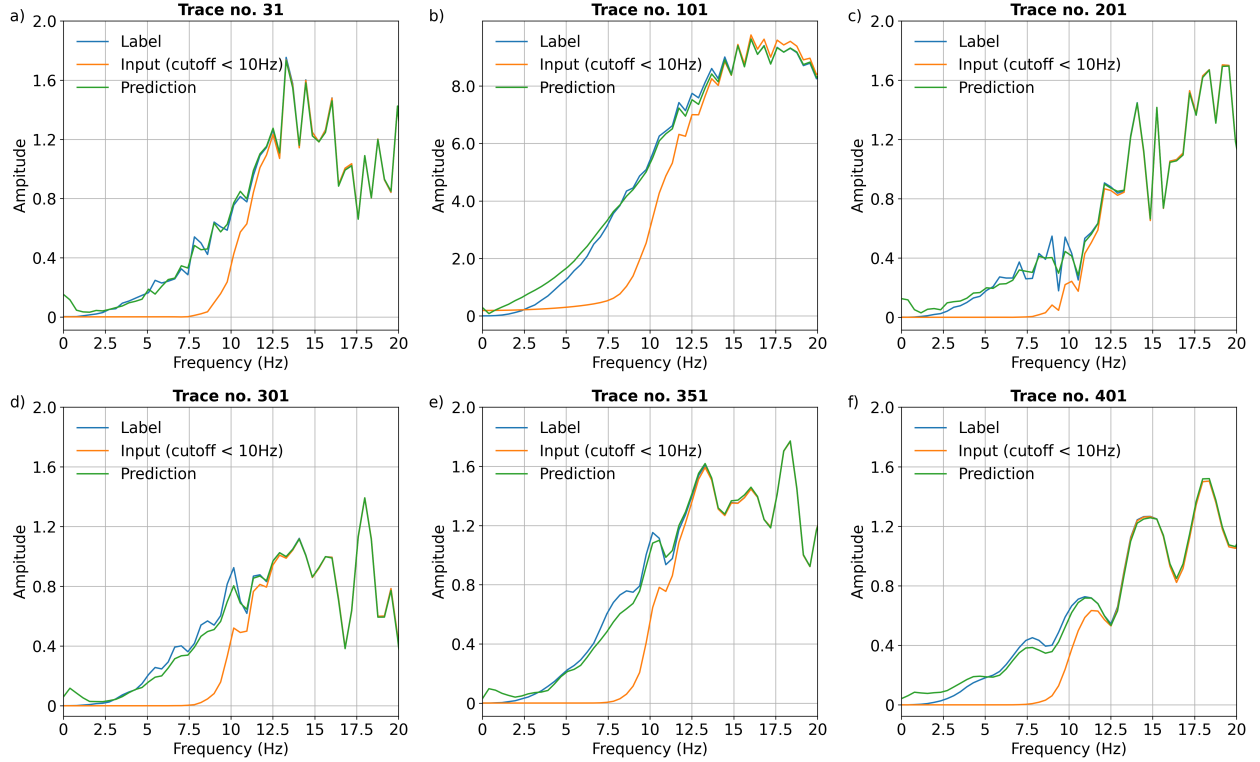


Figure 6: The amplitude spectrum comparison at different locations. The 15 Hz high-pass filter is used.

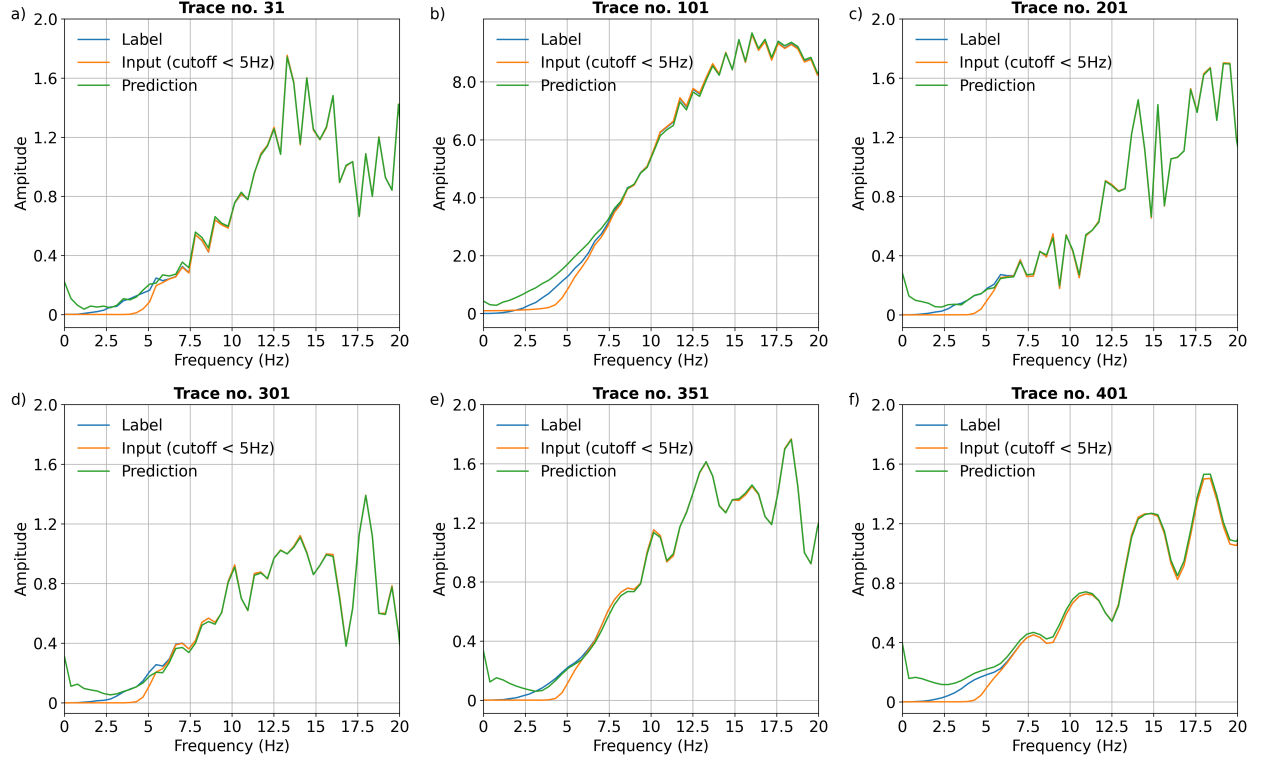


Figure 7: The amplitude spectrum comparison at different locations. The 5 Hz high-pass filter is used.

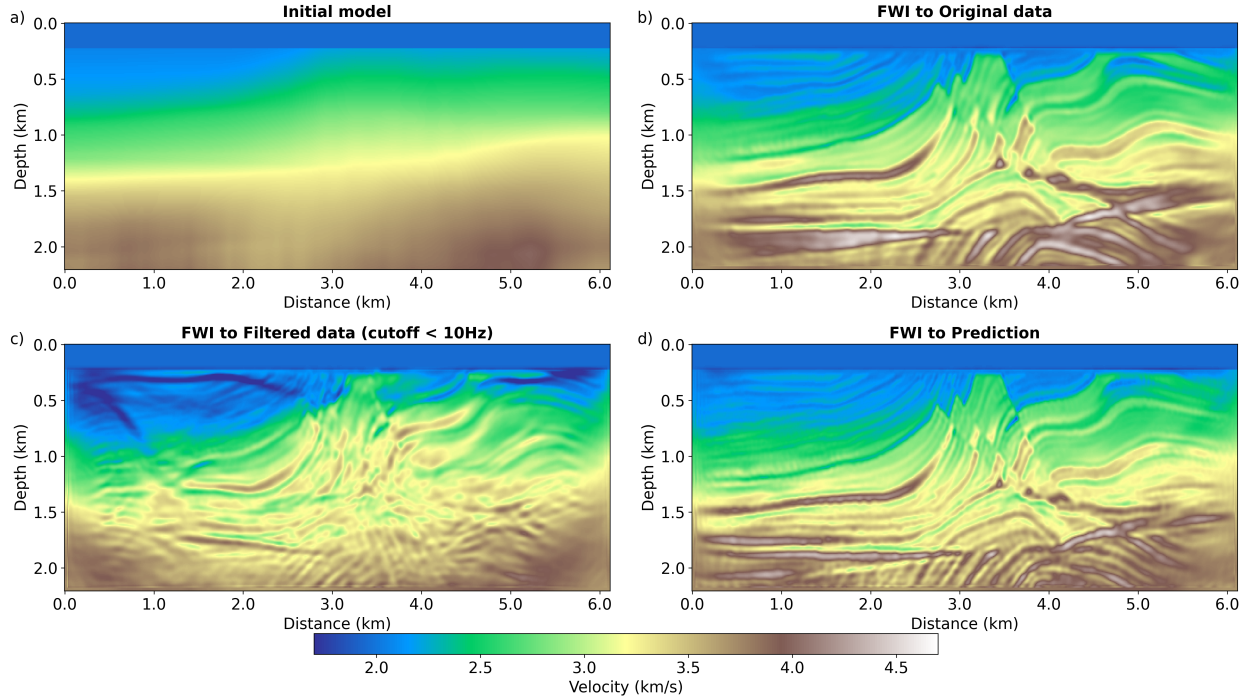


Figure 8: (a) The initial velocity model and inversion results generated with different data sets: (b) original full band data, (c) filtered data with a cutoff frequency of 10 Hz and (d) the predicted data with the proposed algorithm.

In line with real seismic exploration scenarios, we also perform the tests on the data missing frequency information below 5 Hz. In order to highlight the importance of the low frequency components, we use a smoother velocity as the

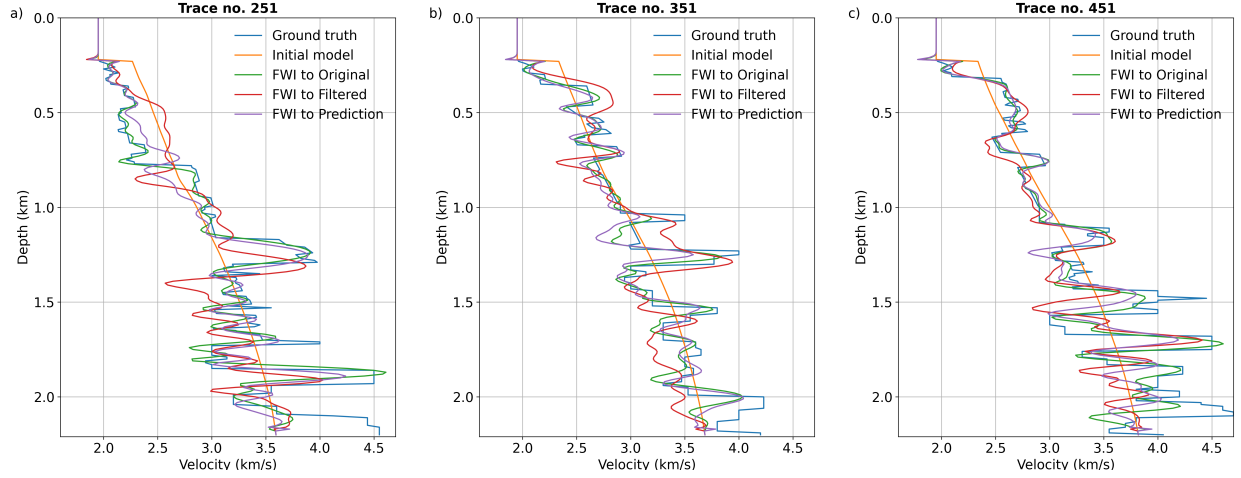


Figure 9: Profiles at different locations: (a) $X=2.5$ km, (b) $X=3.5$ km and (c) $X=4.5$ km. The cutoff frequency is 10 Hz.

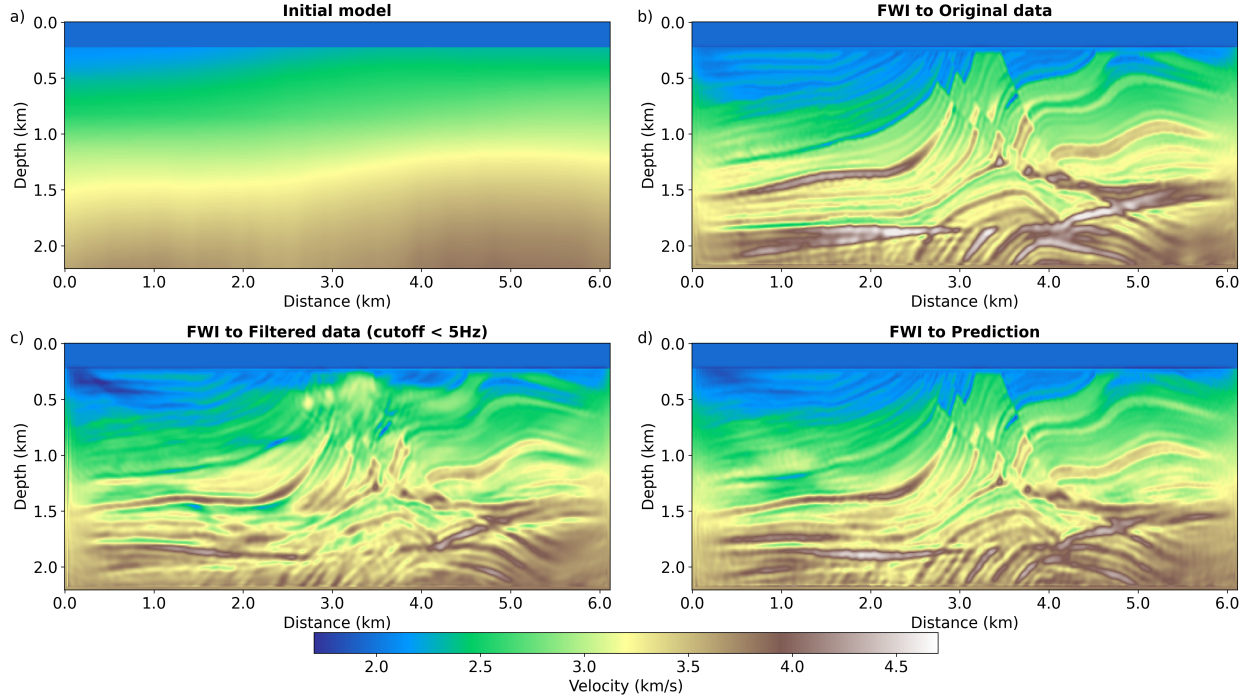


Figure 10: (a) The initial velocity model and inversion results generated with different data sets: (b) original full band data, (c) filtered data with a cutoff frequency of 5 Hz and (d) the predicted data with the proposed algorithm.

initial model (Figure 10a). Five frequency bands with dominant frequencies of 2 Hz, 3 Hz, 5 Hz, 8 Hz and 15 Hz are selected to conduct the multi-scale inversion. Figure 10b displays the result of the inversion applied to the original full-band data, which also nicely matches the true model. When using the filtered data missing frequency information below 5 Hz to conduct the test, the inversion result is poor due to the cycle-skipping issue. With the proposed algorithm to extrapolate the filtered data, the inversion result can recover the velocity well. Figure 11 displays the profiles at $X=2.5$ km, 3.5 km and 4.5 km, from which we can see that the result of the filtered data has both numerical and depth errors compared to the true model. In contrast, the result of using the predicted data fits the true model very well similar to the inversion applied to the original data.

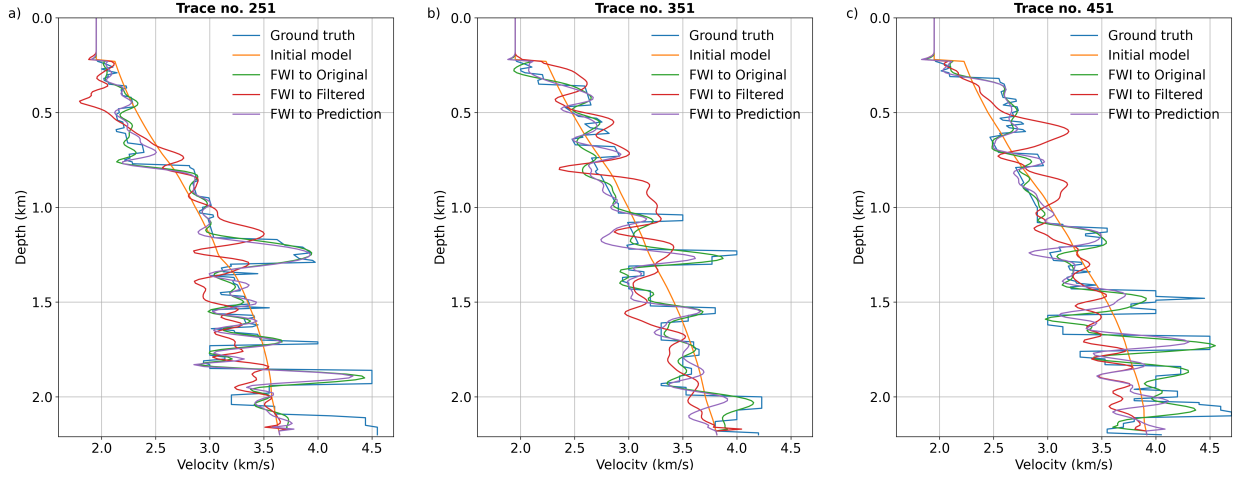


Figure 11: Profiles at different locations: (a) $X=2.5$ km, (b) $X=3.5$ km and (c) $X=4.5$ km. The cutoff frequency is 5 Hz.

3.2 Field Data

After verifying the accuracy and reasonability of the proposed algorithm, we attempt to apply it to real marine seismic data. The data were acquired in North West Australia using a variable depth streamer. The original dataset comprises 1824 shots spaced approximately 18.75 m apart laterally. For our test, we select 201 shots, with 56m lateral spacing, to lower the computational burden. Each shot gather is acquired through steamer cables with a maximum offset of 8.0 km and a 12.5 m recording spacing.

The original acquired data included frequency components as low as 2.5 Hz. In our experiment, to consider a more general scenario where typically collected data lack frequencies below 5 Hz, we filtered out frequencies below 5 Hz from the original acquired data, treating it as our available observed data. We extract 18500 data patches from the filtered shot gathers, each sized 128×128 . During the warm-up phase, the network undergoes pre-training for 50 epochs, where the input data is prepared by performing a high-pass filter with a cutoff frequency of 10 Hz on the observed data.. In the IDR phase, the initial range for the high-pass filter cutoff frequency is set between $1 \sim 2$ Hz, and then increases by 0.5 Hz for both the upper and lower limits every 20 epochs. Eventually, the filter cutoff frequency is fixed between $4 \sim 5$ Hz. Instead of starting from scratch, here the network is initialized from the model trained on synthetic data. The network is trained with a total of 200 epochs. The initial learning rate is $2e-4$, which is reduced by a factor of 0.8 at the 25, 50, 75, 100, 125, and 150 epochs.

Figure 12a denotes the original seismogram of the first shot. Figure 12b shows the corresponding filtered shot gather, where we cut off the information below 5 Hz. Figure 12c is the result after low-frequency extrapolation with the proposed algorithm. Figure 12d-12f displays the energy below 5 Hz of the original, filtered and predicted data, from which we find that the original signal below 5 Hz has weaker energy than the predicted one. This is because the prediction does not have limitations in the low frequency limit, whereas the data contain limited energy below 2.5 Hz. As shown in Figures 12g-12i, we manage to predicted low frequency information below 2.5 Hz. Figure 12j shows that there still exists some signal of reflected waves, which means that the proposed algorithm could extrapolate to frequencies as low as 1.75 Hz.

Figure 13a displays the initial velocity model, Figures 13b and 13d are the inversion results of the original and predicted data with starting dominant frequency of 2.5 Hz. The corresponding inversion results are similar to a large degree, indirectly proving that the frequency components within $2.5 \sim 5.0$ Hz predicted by our algorithm are consistent with the original data. However, the result of predicted data shows some additional details, which correspond to the additional components less than 2.5 Hz after the frequency extrapolation. If we directly use the filtered data to perform the inversion, the corresponding inversion result is shown in Figure 13c. We can see that two big false faults appear at $X=3$ km and 8 km caused by the cycle-skipping problem. To emphasize the positive performance of our proposed low-frequency extrapolation algorithm, we conduct a comparative inversion test with an initial dominant frequency of 1.75 Hz. The corresponding outcome is presented in Figure 13e. Contrasting the outcomes displayed in Figures 13b and 13d, the result depicted in Figure 13e reveals a more comprehensive representation of the background information.

To demonstrate the accuracy of the different velocity models shown in Figure 13, we perform a reverse time migration. Figure 14 shows the corresponding migration images. Compared to the image from the initial model (Figure 14a),

those of the FWI results (Figure 14b-14d), to some degree, show the stratigraphic relief. To evaluate the accuracy of the images, we choose three common image gathers at $X=2.15$ km, 5.5 km and 6.9 km to make a detailed analysis, as shown in Figure 15. Compared to the other images, that of the predicted data with starting dominant frequency of 1.75 Hz resulted in less noise (depicted by the dashed square area) and more flat gathers (indicated by the red arrows), proving the accuracy of the proposed algorithm.

3.3 Application on earthquake seismogram data

We finally select an earthquake seismogram data to demonstrate the generalizability and feasibility of our method for extending the frequency band to ultra-low frequency components for large-scale collected data. This earthquake seismogram data were collected in western Sichuan, China. The distribution of dense seismic stations and the structure of the collection area are shown in Figure 16. The waveforms used were recorded from September 2006 to July 2009.

In our experiment of low-frequency extrapolation of earthquake seismogram, we implement several modifications compared to processing in exploration data: 1. Instead of extracting two-dimensional data patches to construct datasets, we extract one-dimensional data, each comprising 128 time steps, directly from the acquired data; 2. Given that our training data are one-dimensional, we adapt our network architecture to include one-dimensional convolutional baselines; 3. In the initial training epochs, due to the network's instability, the predicted low-frequency components include some ultra-low-frequency numerical artifacts. Consequently, in IDR phase, frequencies below 0.1 Hz are filtered out from the network's predicted pseudo-labels to ensure the stability of the network training process. We select two teleseismic events, specifically events 51 and 53, from the collected earthquake seismogram. The original collected data contain frequencies below 0.25 Hz. To consider the scenario of missing low-frequencies, we applied a high-pass filter with a cutoff frequency of 0.25 Hz to the original data. The filtered data, assumed as known, are used to construct our training dataset. Notably, the original collected data are not included in our training set, as our approach operates on an SSL basis. During the training's warm-up and IDR phases, we employ a similar strategy, progressively increasing the high-pass filter cutoff frequency range, ultimately fixing it at 0.25 Hz. The network undergoes training for a total of 300 epochs. The initial learning rate $1e-4$, and a scheduler reduces it by a factor of 0.8 at the 30, 60, 90, 120, 150, and 180 epochs.

We present the original seismic waveforms of different teleseismic events (Figure 17 for event 20 and Figure 18 for event 73), along with the waveforms filtered to exclude frequency components below 0.25 Hz, and compared these with the predicted waveforms under various low-pass filters. Due to the complexity of the coda wave following the direct P phase, we establish a variable time window to better demonstrate the restoration of low-frequency components in the predicted waveforms relative to the original recordings. This time window is indicated by the light green shaded areas in Figures 17 and 18 (panel (a)). Within this window, we extract the band-passed original recordings (black), filtered waveforms (blue), and predicted waveforms (red). We then compute their normalized cross-correlation coefficients (NCC) across different filtering bands (0.01-0.15 Hz, 0.01-0.2 Hz, 0.01-0.25 Hz, 0.01-0.3 Hz). The NCC values thus reflect the degree of fit from the onset of the P-wave to the later coda waves. As observed in panels (b) of Figures 17 and 18, for different low-pass filters, the predicted waveforms exhibit a better restoration of the original low-frequency components (the red NCC curves are consistently above the blue ones). These results demonstrate the effectiveness of our method in the restoration of ultra-low-frequency content in earthquake waveform data.

4 Discussion

In this paper, we have concurrently validated the efficacy of our developed low-frequency extrapolation method on both exploration seismic and earthquake seismogram data. Owing to our method's adoption of a self-supervised learning (SSL) paradigm, it effectively obviates the need for labels, thereby enhancing its practical application potential. Subsequently, we will discuss the robustness of our method against noisy data, the role of the amplitude spectrum loss, and the strategies for the effective setting of filtering frequencies.

4.1 Robustness to noise

It is widely recognized that observed seismic data are invariably contaminated by noise. Therefore, a critical examination is the robustness of our method against noise in the data. Recently, an SSL paradigm for noise reduction have proven its efficacy in attenuating various types of seismic noise [69]. By integrating a noise reduction functionality into our SSL low-frequency extrapolation approach, we can present a unified SSL framework. This framework is capable of performing both denoising and low-frequency extrapolation processing on seismic data simultaneously. In the following, we will meticulously explore how to make fine-tuned adjustments to the developed SSL algorithms (Algorithm 1) to fulfill this purpose.

Initially, during the warm-up phase, we need to alter the generation mode of input data in the lesslow-low dataset. This entails adding noise to the original noisy data after it has been high-pass filtered, for example,

$$I_i = H[x_i] + n_i, \quad i = 1, \dots, N, \quad (12)$$

where the I_i represents the input data, and n_i denotes the added noise. The pseudo labels, however, remain the original noisy data x_i . It is important to note that the original noisy data also lack low frequency content. In other words, the training of our coupled framework for low-frequency extrapolation and denoising is conducted under an SSL paradigm.

Similarly, during the IDR stage, we create a new lesslow-low dataset's input by adding noise to the network's predicted results after high-pass filtering, such as

$$I_i = H[NN(x_i)] + n_i, \quad i = 1, \dots, N. \quad (13)$$

The pseudo labels are still the network's predictions of the original noisy data $NN(x_i)$. With these modifications, we can see that the input data, compared to the original noisy data, not only lack more low-frequency components but also contain a stronger noise. Consequently, the network is trained to optimize two objectives simultaneously: denoising and low-frequency extrapolation.

We share an example to validate the performance of the coupled SSL approach. We continue to employ the synthetic data generated in Section 3.1. The original noisy data exhibit a deficiency in the low-frequency range of (5, 10)Hz, and they include random noise with the levels ranging from 5 to 30, generated by the subsequent equation:

$$n_i = 0.01 \cdot \xi \cdot std(L_i) \cdot rand(0, 1), \quad i = 1, \dots, N, \quad (14)$$

where the ξ is the noise level, $std(L_i)$ represents the standard deviation of the pseudo label L_i , and $rand(0, 1)$ is the standard normal distribution.

In the warm-up phase, the network is pre-trained for 50 epochs. When we create the lesslow-low dataset, the original noisy data are filtered within the range of 6 ~ 25Hz. The noise level added to the post-filtered results, ranging from 5 to 30, is used to construct the input data. In the IDR phase, the initial 50 epochs involve a random high-pass cutoff frequency ranging from 5 to 6 Hz. Subsequently, every 50 epochs, the upper limit of the high-pass cutoff frequency is increased by 1 Hz. This cutoff frequency range is fixed when it reaches the range of 5 ~ 10Hz. Throughout this entire phase, the level of noise added also varies randomly within the range of 5 to 30. The network in total undergoes training across 290 epochs. The initial learning rate is 2e-4, and a scheduler reduces it by a factor of 0.8 for every 65 epochs.

Figure 19 presents the test results for a noisy shot gather. Panel (a) displays the original simulated data. Panel (b) shows the data from panel (a) with frequencies below 10Hz removed and random noise of level 15 added. The corresponding prediction results are shown in panel (c), with the residuals between the prediction and the original simulated data displayed in panel (d). Compared to panel (b), panel (e) contains a shot gather with stronger noise, the corresponding prediction results for which are shown in panel (f), and the discrepancy between the prediction and the original simulated data is displayed in panel (g). It is evident that our framework is effective in removing noise. Undoubtedly, our method also attenuates some of the signal energy, particularly the deep signals in scenarios with strong noise. This occurs because we lack frequency components below 10Hz. In such instances, the added noise significantly overwhelms the weaker deep reflection signals, leading to the attenuation of these signals' energy while removing noise. To further validate the frequency extension performance, we plot the amplitude spectrum curve of the prediction (panel (f)), which is shown in Figure 20, comparing it with the original simulated data and the test data from panel (e). From the figures, we can see that in the frequency range below 10 Hz, the input test data exhibits random fluctuations in the amplitude spectrum due to noise contamination. However, our framework successfully removes these noise-induced artifacts and effectively extends the low-frequency components. We can clearly see that frequencies below 10 Hz have been restored. Furthermore, we make comparisons to the FWI results with different data sets, as shown in Figure 21. Compared to the inversion result (Figure 21b) of the noisy data missing the low-frequency information, the result (Figure 21c) of the predicted data after low-frequency extrapolation can recover the subsurface velocity favorably, especially in the middle area with adequate acquisition coverage. Figure 22 displays the corresponding profiles at X=2.5 km, 3.5 km and 4.5 km. Obviously, the predicted inversion results (red line) match the true velocity (blue line) very well, while the noisy inversion results have a large error.

4.2 The role of the amplitude spectrum loss

In the process of optimizing network training, we introduce an amplitude spectrum loss to work alongside the data loss in measuring the misfit between the network output and the pseudo labels. We do this to ensure the network focuses not only on data reconstruction but also on maintaining consistency in the frequency domain, as our goal is to effectively recover low-frequency components. Integrating an amplitude spectrum loss compels the network to more accurately capture frequency characteristics during training.

Table 1: The comparison of low-frequency extrapolation performance using different hyperparameter ϵ , where the MAE metric has been used to measure the misfit between the prediction and the original simulated data.

Hyperparameter setting	Cutoff 5 Hz	Cutoff 10 Hz	Cutoff 15 Hz
$\epsilon = 0$	0.000556	0.000357	0.000251
$\epsilon = 0.001$	0.000541	0.000325	0.000207
$\epsilon = 0.01$	0.000486	0.000286	0.000198
$\epsilon = 0.1$	0.000951	0.000818	0.000788
$\epsilon = 1$	0.000619	0.000473	0.000217

Consequently, a pertinent question arises: how do we adjust the weights of the data and frequency losses? To address this issue, we conduct a test to compare how different hyperparameter ξ settings influence the low-frequency extrapolation performance of our method. We employ the same training configuration as in Section 3.1, with the sole difference being the ϵ is set to 0, 0.001, 0.01, 0.1, and 1, where 0 corresponds to using only the data loss. To quantitatively assess the low-frequency extrapolation performance, we use the MAE metric to measure the discrepancy between the prediction and original simulated data. The MAE metrics for networks trained with different ϵ settings are displayed in Table 1. It demonstrates that incorporating the amplitude spectrum loss contributes to the enhanced performance in low-frequency extrapolation. For example, the network trained with setting $\epsilon = 0.01$ achieves better performance in recovering low-frequency components than that using only data loss, as evidenced by a lower MAE metric.

However, it’s crucial to recognize that careful setting of the hyperparameter ϵ is necessary, as excessively high values can lead to a decline in network performance. For example, compared to using only data loss, the hyperparameter settings of $\epsilon = 0.1$ and $\epsilon = 1$ result in poorer low-frequency recovery. Based on our experience, we find that a hyperparameter setting of $\epsilon = 0.01$ demonstrates the robustness across various datasets, consistently enhancing low-frequency extrapolation performance compared to using data loss alone. Therefore, we recommend setting this parameter to 0.01, as adopted in our numerical examples.

4.3 High-pass filter setting

Our method draws inspiration from the Noisier2Noise concept, where we create input data that lack even more low-frequency components by applying a high-pass filter to the pseudo-label data. In the process of applying this high-pass filter, we need to provide a range for high-pass cutoff frequencies. It is important to emphasize that the cutoff frequencies range is not arbitrarily set; it involves a certain trick. We will share insights on setting the cutoff frequency range, which can aid in enhancing the network’s low-frequency extrapolation performance.

First, we need to analyze the frequency distribution range of the seismic data we have, as this will provide the prior knowledge required for setting the high-pass filter. Taking the synthetic data test in Section 3.1 as an example, we can determine the range of low frequencies missing in the original data to be 5 ~ 15 Hz by plotting the amplitude spectrum curve. Subsequently, during the warm-up phase, our experience in setting the cutoff frequency range is guided by the principle that the maximum cutoff frequency should not exceed twice the highest missing frequency, and the minimum cutoff frequency should be slightly above the lowest missing frequency. Therefore, as observed, in Section 3.1, we set the filtering cutoff frequency range for the pseudo-labels to 6 ~ 30 Hz during the warm-up phase.

In the IDR phase, the optimal approach is to gradually increase the upper limit of the filter cutoff frequency while keeping the lower limit consistent with the lowest frequency missing in the original seismic data. For example, in Section 3.1, the range of the filter cutoff frequency for the initial 50 epochs varied randomly between 5 to 7 Hz. Then, every 50 epochs, we increased the maximum filter cutoff frequency by 2 Hz. Once the maximum filter cutoff frequency aligned with the highest frequency missing in the original seismic data, we maintained this range for training until the predefined maximum training epochs. The rationale behind this is our intention to progressively enhance the network’s low-frequency extrapolation performance. For example, extending the low-frequency range of data missing frequencies below 10 Hz is simpler compared to data missing frequencies below 5 Hz. Hence, we aim for the network to commence learning from these simpler tasks and then progressively increase the difficulty during training by incorporating data missing more low frequencies. This approach to learning is more stable and significantly improves the network’s low-frequency extrapolation performance.

We, here, present an example to validate the effectiveness of this training strategy. Compared to Section 3.1, we only modify the settings of the filter cutoff frequency range. Specifically, in the IDR phase, we abandoned the gradually

increasing cutoff frequency range setting strategy used in Section 3.1 (referred to as Strategy 1), and instead, directly set the cutoff frequency range to 5 ~ 15 Hz (referred to as Strategy 2). We compare the predictions of the network models at different training epochs on three test data, each missing frequencies below 5 Hz, 10 Hz, and 15 Hz, against the network trained in Section 3.1. Figure 23 displays the MAE metrics of the networks using two different cutoff frequency setting strategies on three test data. The panels a, b, and c correspond to the seismic data missing frequencies below 5 Hz, 10 Hz, and 15 Hz, respectively. It is evident that the network employing Strategy 1 shows gradually improving performance as the training epochs increase. In contrast, the network using Strategy 2 seems to exhibit a declining performance trend on all three test data, indicating non-convergent characteristics. These results highlight the effectiveness of our proposed training strategy, proving its significant enhancement in training stability and low-frequency extrapolation capabilities of the network.

5 Conclusion

We developed a novel neural network (NN)-based seismic low-frequency extrapolation method in a self-supervised learning (SSL) fashion. Under our framework, the NN sequentially undergoes two stages: warm-up and iterative data refinement (IDR). In the warm-up stage, we construct a lesslow-low dataset, using the original observed data, which lack low-frequency components, as pseudo-labels. The input data are obtained by applying a high-pass filter to these pseudo-labels, resulting in a further loss of low-frequency content. The NN rapidly warms up on this constructed dataset, initially extracting the original data's frequency characteristics and providing a degree of low-frequency extension capability. During the IDR stage, we update the lesslow-low dataset in each training epoch, where the pseudo-labels are derived from the network's predictions of the original observed data from the previous epoch, and the input data are created by applying a high-pass filter to these predicted pseudo-labels. Continually updating the training dataset allows us to gradually reduce the frequency information bias between the network's predictions and the ideal ground truth, thereby steadily enhancing the network's low-frequency extrapolation performance. We validated our method's effectiveness on both synthetic and field data in exploration circumstances. The results demonstrated that our method effectively extrapolates low-frequency components, enabling us to address the main challenge of full waveform inversion, specifically cycle-skipping. Further testing on earthquake seismogram demonstrated our method's applicability to extending ultra-low-frequency content in large-scale collected data.

Acknowledgments

This publication is based on work supported by the King Abdullah University of Science and Technology (KAUST). The authors thank the DeepWave sponsors for supporting this research. They also thank CGG for sharing the field seismic data. This work utilized the resources of the Supercomputing Laboratory at King Abdullah University of Science and Technology (KAUST) in Thuwal, Saudi Arabia.

References

- [1] Hamed Ben-Hadj-Ali, Stéphane Operto, and Jean Virieux. An efficient frequency-domain full waveform inversion method using simultaneous encoded sources. *Geophysics*, 76(4):R109–R124, 2011.
- [2] Yunseok Choi and Tariq Alkhalifah. Application of multi-source waveform inversion to marine streamer data using the global correlation norm. *Geophysical Prospecting*, 60(4):748–758, 2012.
- [3] John E Anderson, Lijian Tan, and Don Wang. Time-reversal checkpointing methods for rtm and fwi. *Geophysics*, 77:S93–S103, 2012.
- [4] Qingchen Zhang, Hui Zhou, Anxin Zuo, Muming Xia, and Jie Wang. Efficient boundary storage strategy for 3d elastic fwi in time domain. In *SEG Technical Program Expanded Abstracts 2014*, pages 1142–1146. Society of Exploration Geophysicists, 2014.
- [5] Qingchen Zhang, Hui Zhou, Qingqing Li, Hanming Chen, and Jie Wang. Robust source-independent elastic full-waveform inversion in the time domain. *Geophysics*, 81(2):R29–R44, 2016.
- [6] Qiang Guo and Tariq Alkhalifah. Elastic reflection-based waveform inversion with a nonlinear approach. *Geophysics*, 82(6):R309–R321, 2017.
- [7] Qingchen Zhang, Weijian Mao, Hui Zhou, Hongjing Zhang, and Yangkang Chen. Hybrid-domain simultaneous-source full waveform inversion without crosstalk noise. *Geophysical Journal International*, 215(3):1659–1681, 2018.

- [8] Albert Tarantola. Inversion of seismic reflection data in the acoustic approximation. *Geophysics*, 49(8):1259–1266, 1984.
- [9] Albert Tarantola. A strategy for nonlinear elastic inversion of seismic reflection data. *Geophysics*, 51(10):1893–1903, 1986.
- [10] Carey Bunks, Fatimetou M Saleck, S Zaleski, and G Chavent. Multiscale seismic waveform inversion. *Geophysics*, 60(5):1457–1473, 1995.
- [11] Chaiwoot Boonyasiriwat, Paul Valasek, Partha Routh, Weiping Cao, Gerard T Schuster, and Brian Macy. An efficient multiscale method for time-domain waveform tomography. *Geophysics*, 74(6):WCC59–WCC68, 2009.
- [12] Z. Gao, Z. Pan, J. Gao, and R. Wu. Frequency controllable envelope operator and its application in multiscale full-waveform inversion. *IEEE Transactions on Geoscience and Remote Sensing*, 57(2):683–699, Feb 2019.
- [13] Zhiming Ren and Yang Liu. Elastic full-waveform inversion using the second-generation wavelet and an adaptive-operator-length scheme. *Geophysics*, 80(4):R155–R173, 2015.
- [14] Wansoo Ha and Changsoo Shin. Laplace-domain full-waveform inversion of seismic data lacking low-frequency information. *Geophysics*, 77(5):R199–R206, 2012.
- [15] Youngseo Kim, Changsoo Shin, Henri Calandra, and Dong-Joo Min. An algorithm for 3d acoustic time-laplace-fourier-domain hybrid full waveform inversion. *Geophysics*, 78(4):R151–R166, 2013.
- [16] Yong Ma and Dave Hale. Wave-equation reflection traveltime inversion with dynamic warping and full-waveform inversion. *Geophysics*, 78(6):R223–R233, 2013.
- [17] Ru-Shan Wu, Jingrui Luo, and Bangyu Wu. Seismic envelope inversion and modulation signal model. *Geophysics*, 79(3):WA13–WA24, 2014.
- [18] Jingrui Luo and Ru-Shan Wu. Seismic envelope inversion: reduction of local minima and noise resistance. *Geophysical Prospecting*, 63(3):597–614, 2015.
- [19] Michael Warner and Lluís Guasch. Adaptive waveform inversion: Theory. *Geophysics*, 81(6):R429–R445, 2016.
- [20] Yunan Yang and Björn Engquist. Analysis of optimal transport and related misfit functions in full-waveform inversion. *Geophysics*, 83(1):A7–A12, 2018.
- [21] Yunan Yang, Björn Engquist, Junzhe Sun, and Brittany F Hamfeldt. Application of optimal transport and the quadratic wasserstein metric to full-waveform inversion. *Geophysics*, 83(1):R43–R62, 2018.
- [22] Bingbing Sun and Tariq Alkhalifah. The application of an optimal transport to a preconditioned data matching function for robust waveform inversion. *Geophysics*, 84(6):R923–R945, 2019.
- [23] Qingqing Li, Qingchen Zhang, Qizhen Du, and Shijun Cheng. Well-guided multisource elastic full-waveform inversion. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2022.
- [24] Jiashun Yao and Yanghua Wang. Building a full-waveform inversion starting model from wells with dynamic time warping and convolutional neural networks. *Geophysics*, 87(2):R223–R230, 2022.
- [25] Sheng Xu, D Wang, F Chen, Yu Zhang, and G Lambare. Full waveform inversion for reflected seismic data. In *74th EAGE Conference and Exhibition incorporating EUROPEC 2012*, pages cp–293. European Association of Geoscientists & Engineers, 2012.
- [26] Benxin Chi, Liangguo Dong, and Yuzhu Liu. Correlation-based reflection full-waveform inversion. *Geophysics*, 80(4):R189–R202, 2015.
- [27] Tariq Alkhalifah and Zedong Wu. Multiscattering inversion for low-model wavenumbers. *Geophysics*, 81(6):R417–R428, 2016.
- [28] Zedong Wu and Tariq Alkhalifah. Efficient scattering-angle enrichment for a nonlinear inversion of the background and perturbations components of a velocity model. *Geophysical Journal International*, 210(3):1981–1992, 2017.
- [29] Guanchao Wang, Shangxu Wang, Jianyong Song, Chunhui Dong, and Mingqiang Zhang. Elastic reflection traveltime inversion with decoupled wave equation. *Geophysics*, 83(5):R463–R474, 2018.
- [30] Zhen-dong Zhang and Tariq Alkhalifah. Local-crosscorrelation elastic full-waveform inversion. *Geophysics*, 84(6):R897–R908, 2019.
- [31] Gang Yao, Di Wu, and Shang-Xu Wang. A review on reflection-waveform inversion. *Petroleum Science*, 17:334–351, 2020.
- [32] Xinming Wu, Luming Liang, Yunzhi Shi, and Sergey Fomel. Faultseg3d: Using synthetic data sets to train an end-to-end convolutional neural network for 3d seismic fault segmentation. *Geophysics*, 84(3):IM35–IM45, 2019.

- [33] Siwei Yu, Jianwei Ma, and Wenlong Wang. Deep learning for denoising. *Geophysics*, 84(6):V333–V350, 2019.
- [34] S Mostafa Mousavi and Gregory C Beroza. Deep-learning seismology. *Science*, 377(6607):eabm4470, 2022.
- [35] Randy Harsuko and Tariq A Alkhalifah. Storseismic: A new paradigm in deep learning for seismic processing. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2022.
- [36] Shijun Cheng, Xingchen Shi, Weijian Mao, Tariq Alkhalifah, Tao Yang, Yuzhu Liu, and Heping Sun. Elastic seismic imaging enhancement of sparse 4c ocean-bottom node data using deep learning. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.
- [37] Shijun Cheng, Randy Harsuko, and Tariq Alkhalifah. Meta-processing: A robust framework for multi-tasks seismic processing. *arXiv preprint arXiv:2307.14851*, 2023.
- [38] Mauricio Araya-Polo, Joseph Jennings, Amir Adler, and Taylor Dahlke. Deep-learning tomography. *The Leading Edge*, 37(1):58–66, 2018.
- [39] Shucai Li, Bin Liu, Yuxiao Ren, Yangkang Chen, Senlin Yang, Yunhai Wang, and Peng Jiang. Deep-learning inversion of seismic data. *IEEE Transactions on Geoscience and Remote Sensing*, 58(3):2135–2149, 2020.
- [40] Meng Du, Shijun Cheng, and Weijian Mao. Deep-learning-based seismic variable-size velocity model building. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2022.
- [41] Fangshu Yang and Jianwei Ma. Deep-learning inversion: A next-generation seismic velocity model building method. *Geophysics*, 84(4):R583–R599, 2019.
- [42] Yue Wu and Youzuo Lin. Inversionnet: An efficient and accurate data-driven full waveform inversion. *IEEE Transactions on Computational Imaging*, 6:419–433, 2019.
- [43] Wei Zhang and Jinghui Gao. Deep-learning full-waveform inversion using seismic migration images. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–18, 2021.
- [44] Senlin Yang, Tariq Alkhalifah, Yuxiao Ren, Bin Liu, Yuanyuan Li, and Peng Jiang. Well-log information-assisted high-resolution waveform inversion based on deep learning. *IEEE Geoscience and Remote Sensing Letters*, 20:1–5, 2023.
- [45] Peng Jin, Xitong Zhang, Yinpeng Chen, Sharon Xiaolei Huang, Zicheng Liu, and Youzuo Lin. Unsupervised learning of full-waveform inversion: Connecting cnn and partial differential equation in a loop. *arXiv preprint arXiv:2110.07584*, 2021.
- [46] Yuxiao Ren, Bin Liu, Senlin Yang, Duo Li, and Peng Jiang. Seismic data inversion with acquisition adaptive convolutional neural network for geologic forward prospecting in tunnels. *Geophysics*, 86(5):R659–R670, 2021.
- [47] Youzuo Lin, James Theiler, and Brendt Wohlberg. Physics-guided data-driven seismic inversion: Recent progress and future opportunities in full-waveform inversion. *IEEE Signal Processing Magazine*, 40(1):115–133, 2023.
- [48] Oleg Ovcharenko, Vladimir Kazei, Daniel Peter, and T Alkhalifah. Neural network based low-frequency data extrapolation. In *3rd SEG FWI workshop: What are we getting*, 2017.
- [49] Oleg Ovcharenko, Vladimir Kazei, Mahesh Kalita, Daniel Peter, and Tariq Alkhalifah. Deep learning for low-frequency extrapolation from multioffset seismic data. *Geophysics*, 84(6):R989–R1001, 2019.
- [50] Hongyu Sun and Laurent Demanet. Low frequency extrapolation with deep learning. In *SEG Technical Program Expanded Abstracts 2018*, pages 2011–2015. Society of Exploration Geophysicists, 2018.
- [51] Hongyu Sun and Laurent Demanet. Extrapolated full-waveform inversion with deep learning. *Geophysics*, 85(3):R275–R288, 2020.
- [52] Hongyu Sun and Laurent Demanet. Deep learning for low-frequency extrapolation of multicomponent data in elastic fwi. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2021.
- [53] Wenyi Hu, Yuchen Jin, Xuqing Wu, and Jiefu Chen. Progressive transfer learning for low-frequency data prediction in full-waveform inversion. *Geophysics*, 86(4):R369–R382, 2021.
- [54] Yuchen Jin, Wenyi Hu, Shirui Wang, Yuan Zi, Xuqing Wu, and Jiefu Chen. Efficient progressive transfer learning for full-waveform inversion with extrapolated low-frequency reflection seismic data. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–10, 2021.
- [55] Gabriel Fabien-Ouellet. Low-frequency generation and denoising with recursive convolutional neural networks. In *SEG Technical Program Expanded Abstracts 2020*, pages 870–874. Society of Exploration Geophysicists, 2020.
- [56] Shotaro Nakayama and Gerrit Blacquière. Machine-learning-based data recovery and its contribution to seismic acquisition: Simultaneous application of deblending, trace reconstruction, and low-frequency extrapolation. *Geophysics*, 86(2):P13–P24, 2021.

- [57] Jinwei Fang, Hui Zhou, Yunyue Elita Li, Qingchen Zhang, Lingqian Wang, Pengyuan Sun, and Jianlei Zhang. Data-driven low-frequency signal recovery using deep-learning predictions in full-waveform inversion. *Geophysics*, 85(6):A37–A43, 2020.
- [58] Zhiyong Wang, Guochang Liu, Jing Du, Chao Li, and Jiao Qi. Low-frequency extrapolation of prestack viscoacoustic seismic data based on dense convolutional network. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13, 2022.
- [59] Oleg Ovcharenko, Vladimir Kazei, Tariq A Alkhalifah, and Daniel B Peter. Multi-task learning for low-frequency extrapolation and elastic model building from seismic data. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–17, 2022.
- [60] Tariq Alkhalifah, Hanchen Wang, and Oleg Ovcharenko. Mlreal: Bridging the gap between training on synthetic data and real data applications in machine learning. *Artificial Intelligence in Geosciences*, 3:101–114, 2022.
- [61] Hongyu Sun, Yen Sun, Rami Nammour, Christian Rivera, Paul Williamson, and Laurent Demanet. Learning with real data without real labels: a strategy for extrapolated full-waveform inversion with field data. *Geophysical Journal International*, 235(2):1761–1777, 2023.
- [62] Meixia Wang, Sheng Xu, and Hongbo Zhou. Self-supervised learning for low frequency extension of seismic data. In *SEG Technical Program Expanded Abstracts 2020*, pages 1501–1505. Society of Exploration Geophysicists, 2020.
- [63] Nick Moran, Dan Schmidt, Yu Zhong, and Patrick Coady. Noisier2noise: Learning to denoise from unpaired noisy data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12064–12072, 2020.
- [64] Peng Yong, Romain Brossier, Ludovic Métivier, and Jean Virieux. Localized adaptive waveform inversion: theory and numerical verification. *Geophysical Journal International*, 233(2):1055–1080, 2023.
- [65] Jean Virieux and Stéphane Operto. An overview of full-waveform inversion in exploration geophysics. *Geophysics*, 74(6):WCC1–WCC26, 2009.
- [66] Yunseok Choi and Tariq Alkhalifah. Source-independent time-domain waveform inversion using convolved wavefields: Application to the encoded multisource waveform inversion. *Geophysics*, 76(5):R125–R134, 2011.
- [67] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.
- [68] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
- [69] Shijun Cheng, Zhiyao Cheng, Chao Jiang, Weijian Mao, and Qingchen Zhang. An effective self-supervised learning method for various seismic noise attenuation. *arXiv preprint arXiv:2311.02193*, 2023.

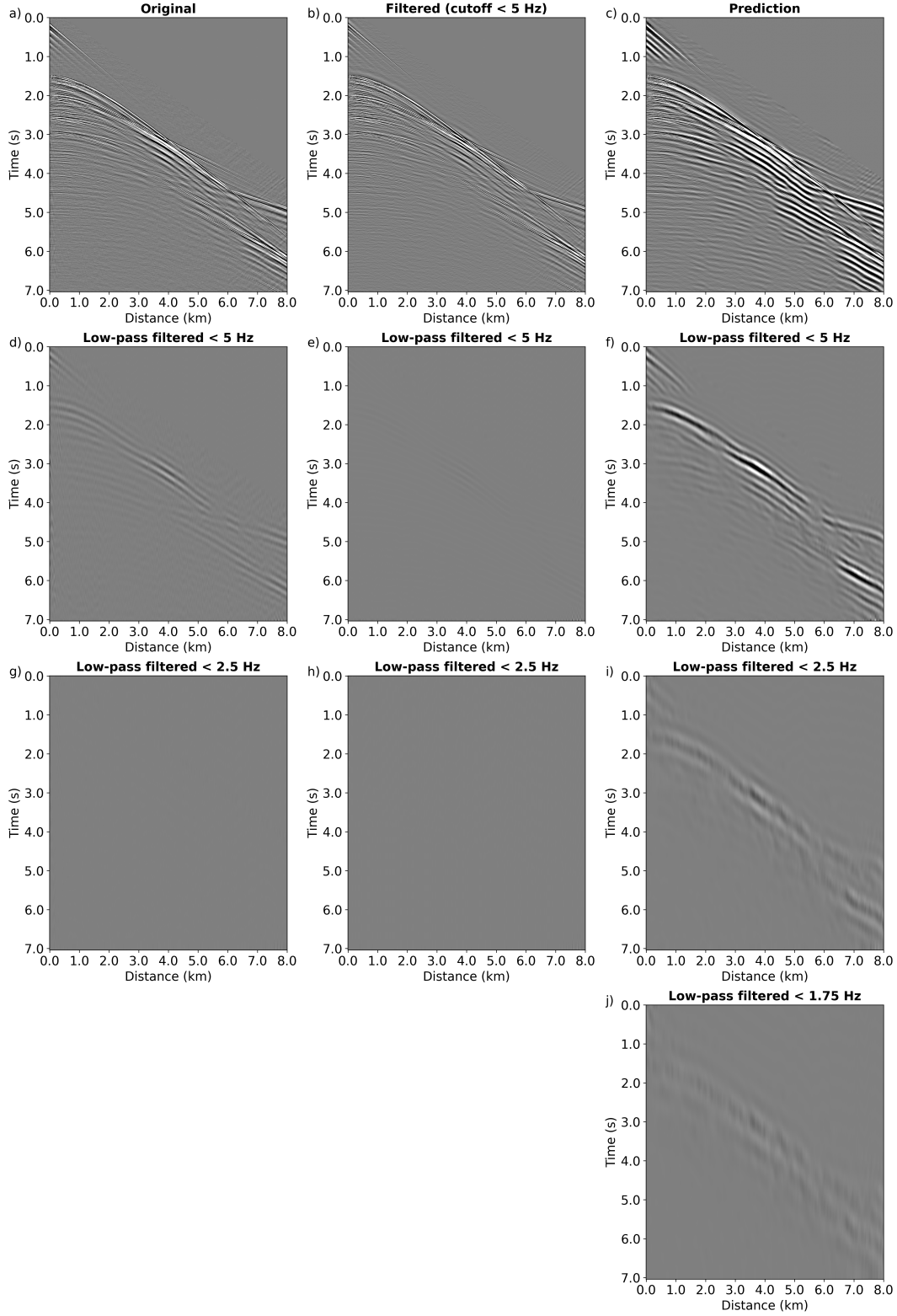


Figure 12: Real marine seismic data, which were acquired from North West Australia. (a) the original observed data, (b) the data filtered by 5 Hz high-pass filters, (c) the predicted data with the proposed algorithm. The frequency components less than 5 Hz: (d) original data, (e) filtered data and (f) predicted data. The frequency components less than 2.5 Hz: (g) original data, (h) filtered data and (i) predicted data. The frequency components less than 1.75 Hz: (j) predicted data.

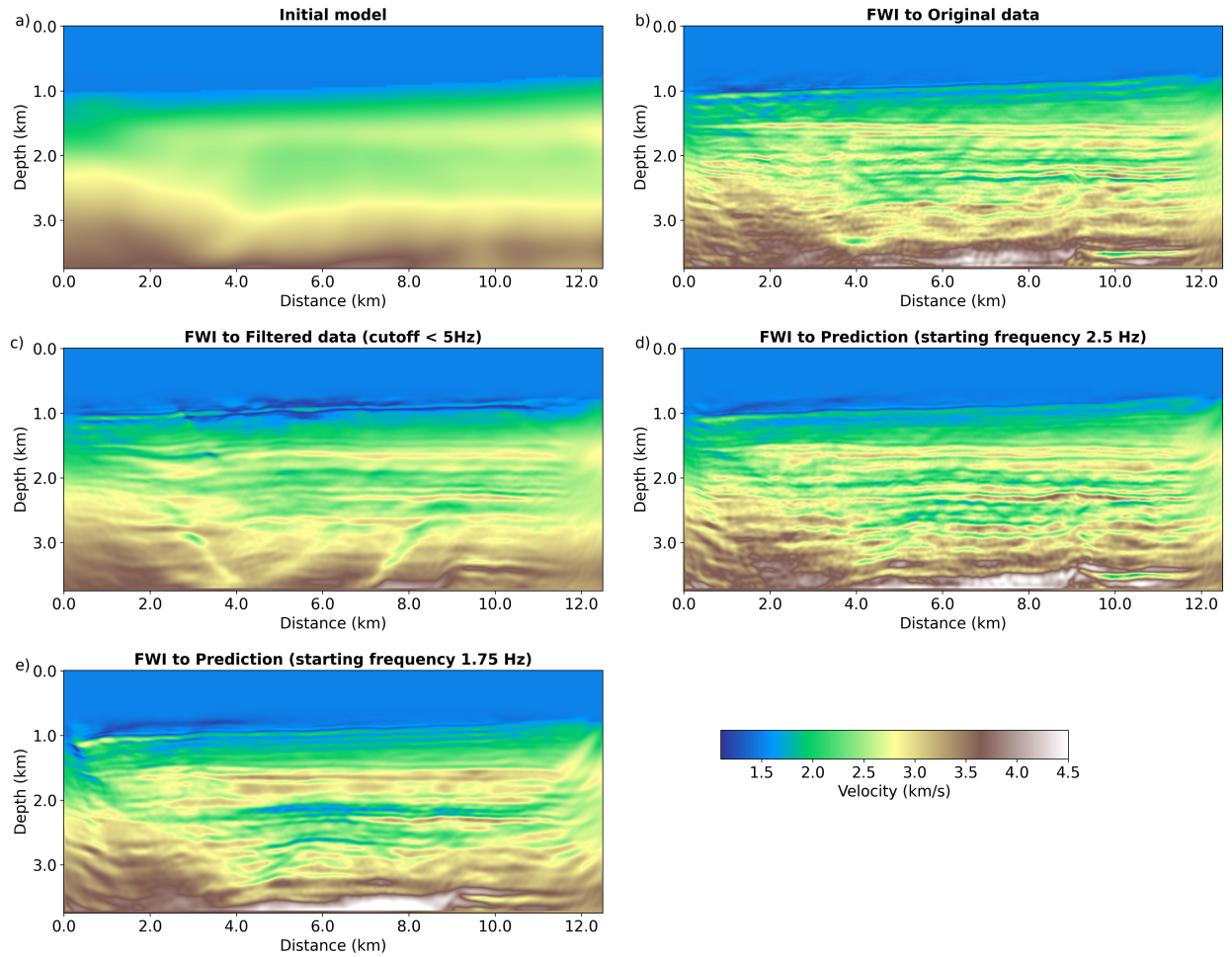


Figure 13: (a) The initial velocity model. Inversion results of different data sets: (b) the original observed data, (c) the data with cutoff frequency of 5 Hz, (d) the predicted data (with starting frequency of 2.5 Hz), (e) the predicted data (with starting frequency of 1.75 Hz).

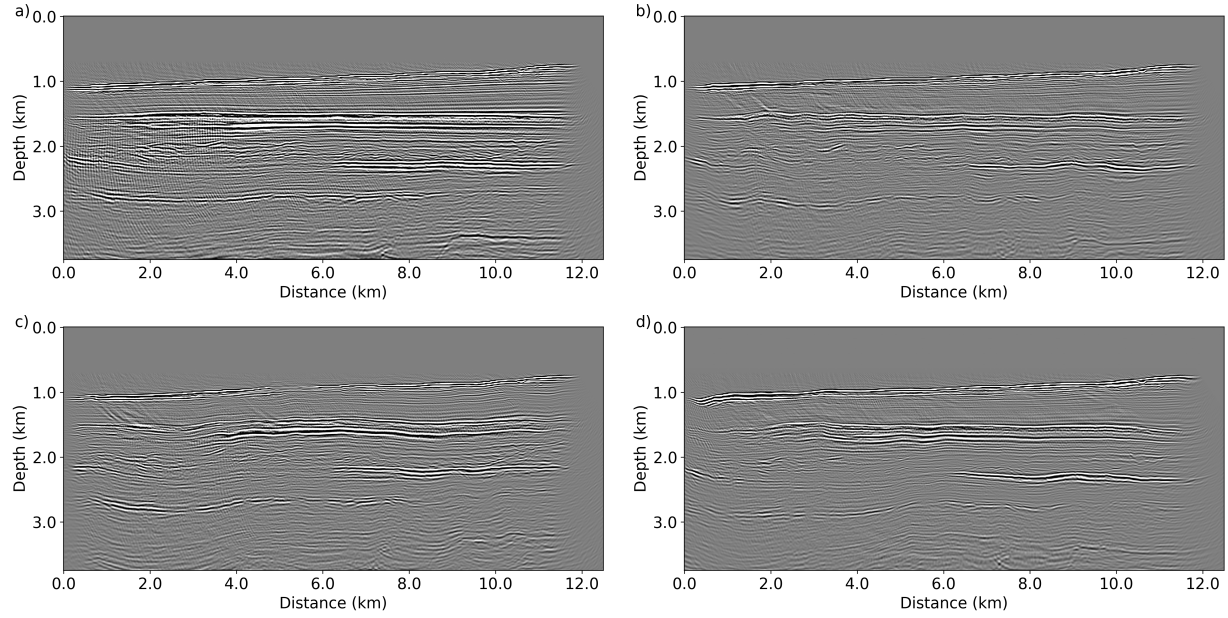


Figure 14: Reverse time migration images generated with different velocity models: (a) the initial velocity model, (b) the inversion result of original observed data, (c) the inversion result of data with cutoff frequency of 5 Hz, (d) the inversion result of the predicted data (with starting frequency of 1.75 Hz).

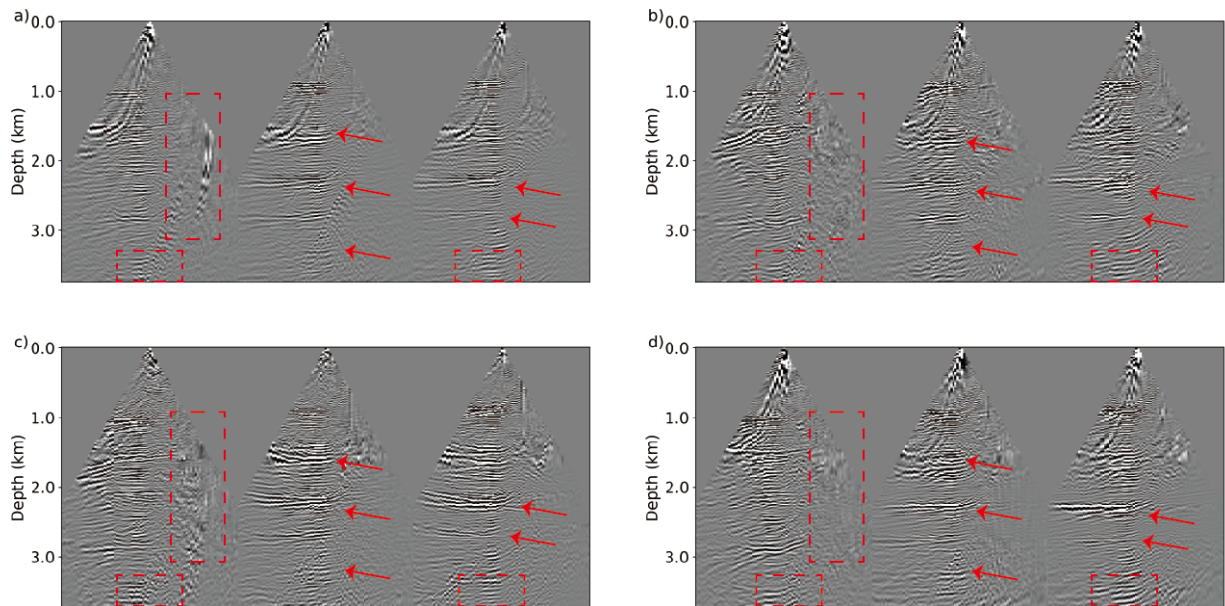


Figure 15: Common image gathers at X=2.15 km, 5.5 km and 6.9 km using: (a) the initial velocity model, (b) the inversion result of original observed data, (c) the inversion result of data with cutoff frequency of 5 Hz, (d) the inversion result of the predicted data (with starting frequency of 1.75 Hz).

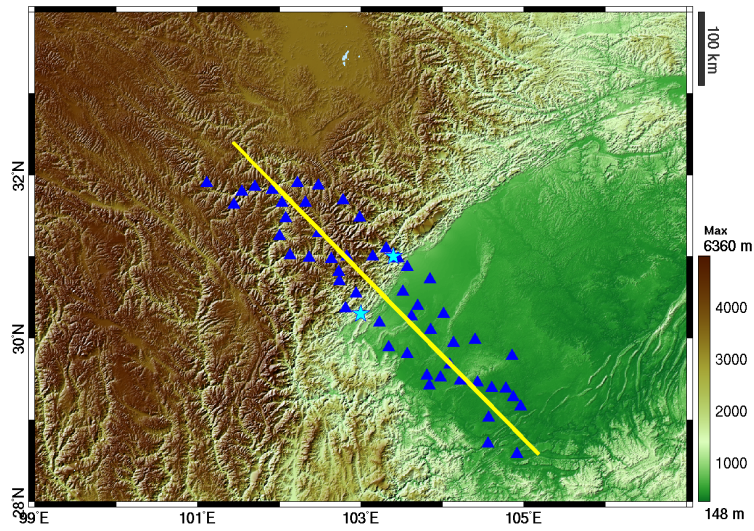


Figure 16: The local structure and stations distribution in the selected western Sichuan for the study. The blue triangles in the figure represent the two-dimensional seismic array used in this research. All stations are arranged along a reflection profile: the Aba-Zigong measurement line (indicated by the thick yellow line). The epicenters of the 2008 Wenchuan earthquake (top right) and the 2013 Lushan earthquake (bottom left) are marked with cyan pentagrams.

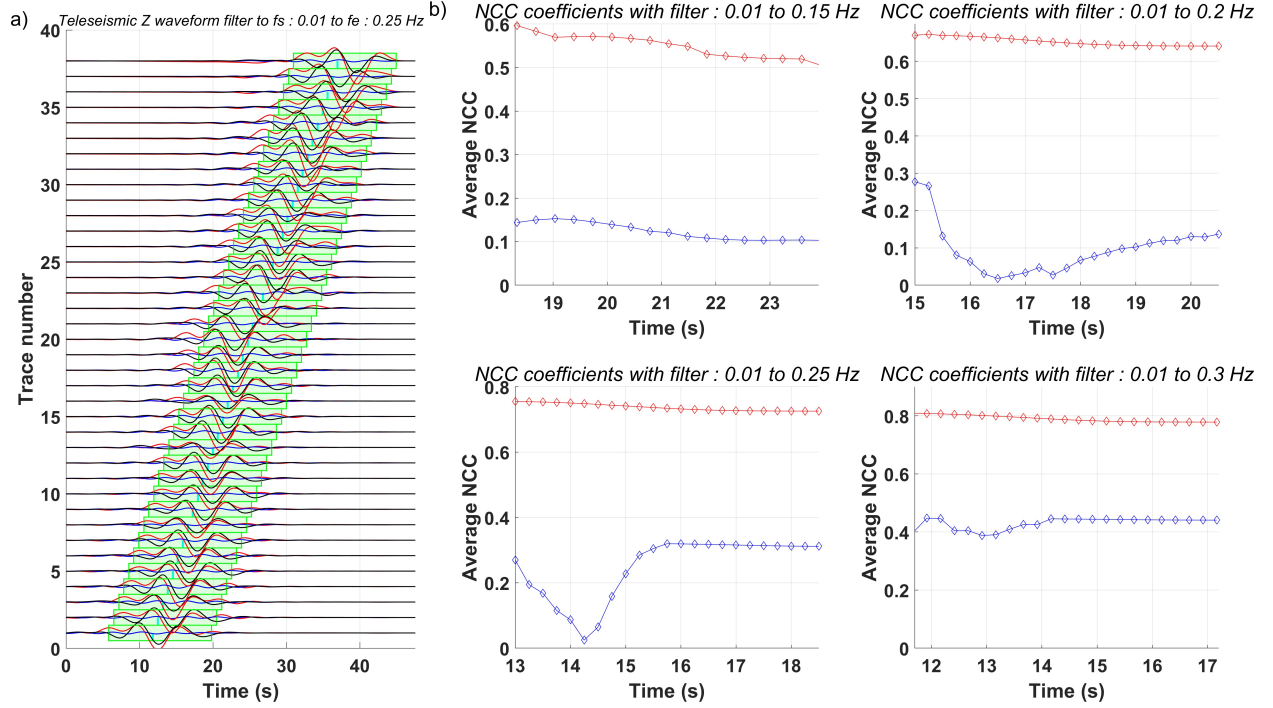


Figure 17: The original recordings of the Z component received from teleseismic event 20, shown in black lines in panels (a), following a band-pass filter (0.01 to 0.25 Hz). The waveforms after filtering out frequencies below 0.25 Hz and then applying the same band-pass filter (0.01 to 0.25 Hz), illustrated in blue lines in panels (a). The predicted results to the filtered waveforms without low-frequency components below 0.25 Hz, followed by the same band-pass filter (0.01 to 0.25 Hz), represented by red lines in panels (a). The panel (b) illustrates a comparison of the normalized cross-correlation coefficients (NCC) for waveforms with and without low-frequency components below 0.25 Hz and predicted waveforms under four different band-pass filtering operations (from top left to bottom right: 0.01-0.15 Hz, 0.01-0.2 Hz, 0.01-0.25 Hz, 0.01-0.3 Hz). This comparison is conducted for different time windows. These windows are defined from two minimum periods ($2/f_e$, where f_e is the highest cutoff frequency of the respective band-pass filter) before the maximum P-wave amplitude of the broadband waveform (indicated by the cyan dashed line) to varying lengths afterwards (ranging from 1s to a maximum of $2.25/f_e$ s). The NCC coefficients are calculated from the band-pass filtered results of the original waveforms and either the predicted waveforms or those without low-frequency components below 0.25 Hz (represented by red and blue curves, corresponding to the colors in panels (a)).

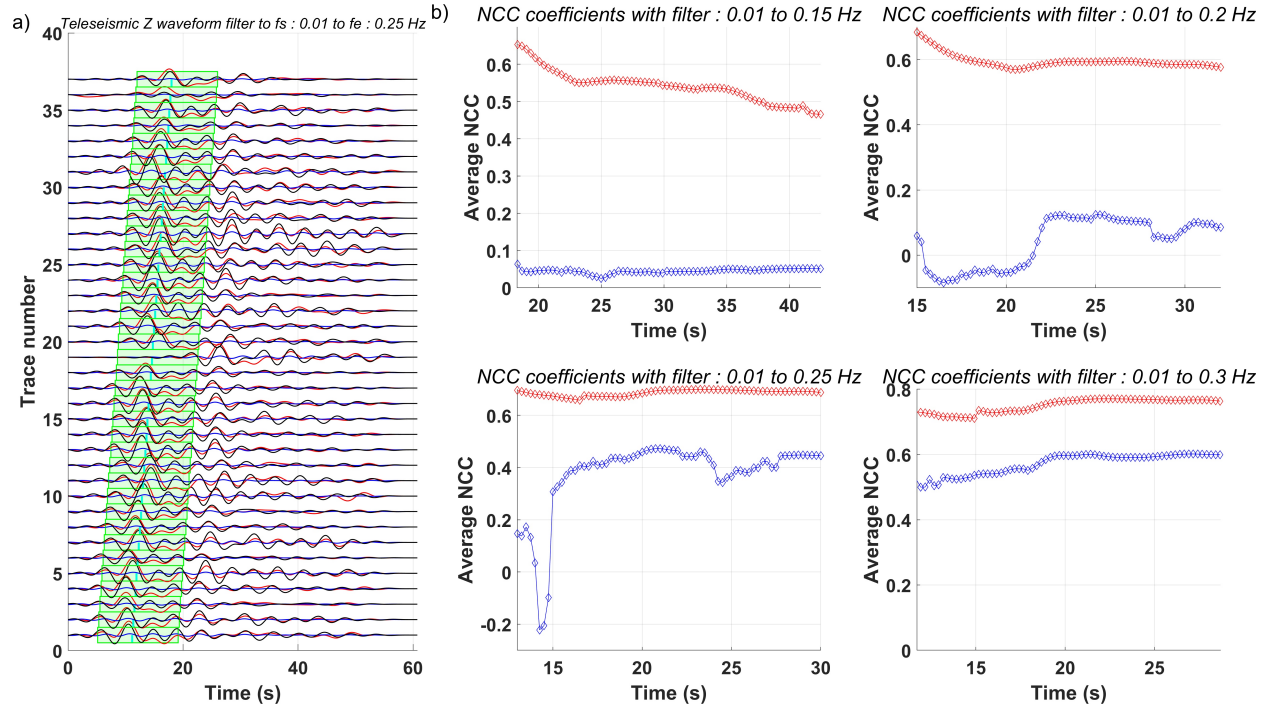


Figure 18: The original recordings of the Z component received from teleseismic event 73, shown in black lines in panel (a), following a band-pass filter (0.01 to 0.25 Hz). The waveforms after filtering out frequencies below 0.25 Hz and then applying the same band-pass filter (0.01 to 0.25 Hz), illustrated in blue lines in panel (a). The predicted results of the filtered waveforms without low-frequency components below 0.25 Hz, followed by the same band-pass filter (0.01 to 0.25 Hz), represented by red lines in panels (a). The other figures carry explanatory notes consistent with Figure 17.

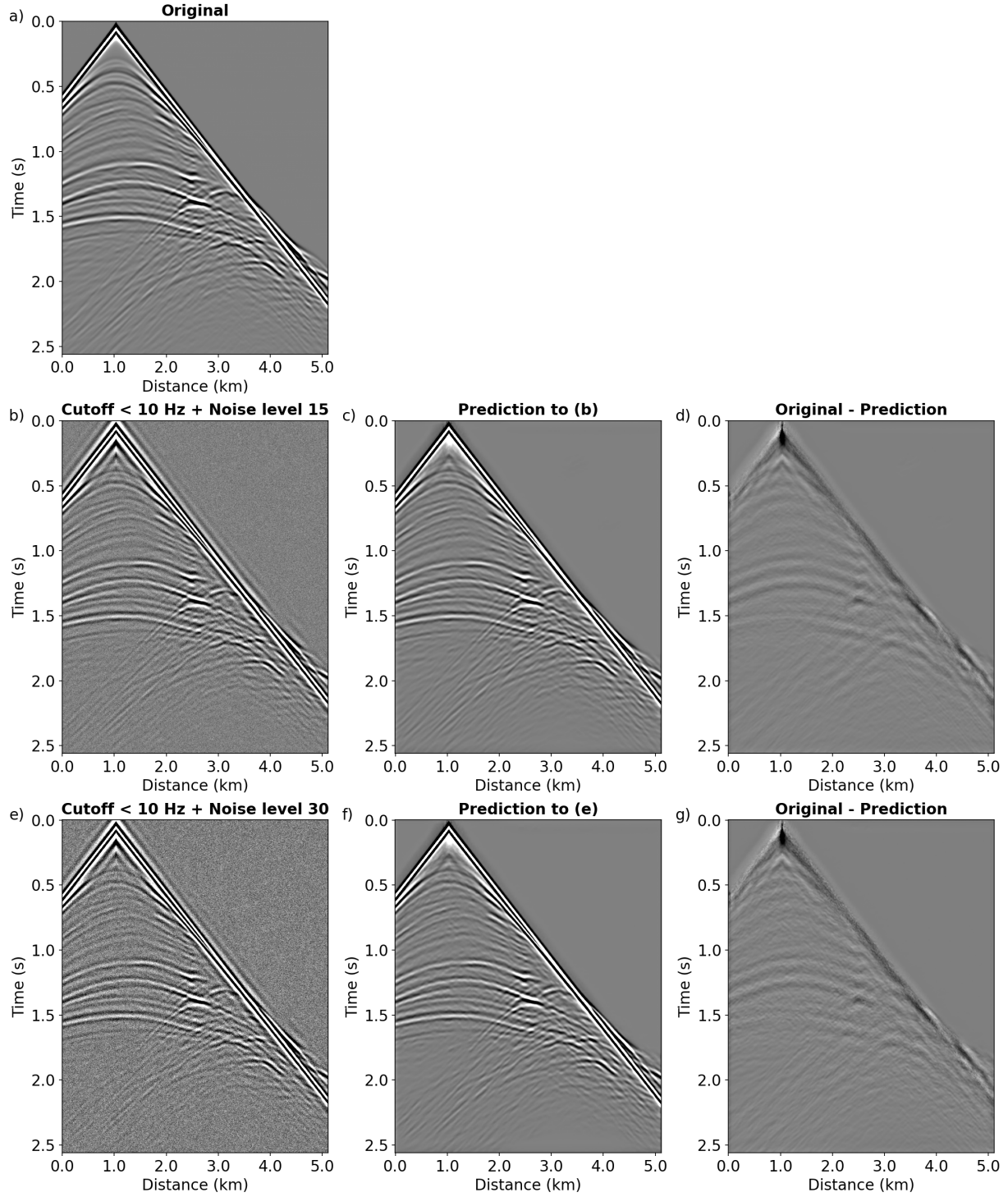


Figure 19: Testing the capability of our method to simultaneously restore low-frequency components and denoise the noisy data with missing low frequencies: (a) the mimic observed seismogram generated with the true velocity model, (b) and (e) are the test data missing low frequencies below 10 Hz but containing noise with a level of 15 and 30, respectively, (c) and (f) correspond the network's prediction results to (b) and (e), respectively, (d) and (g) are the residuals between their prediction results and the mimic observed seismogram (a), respectively.

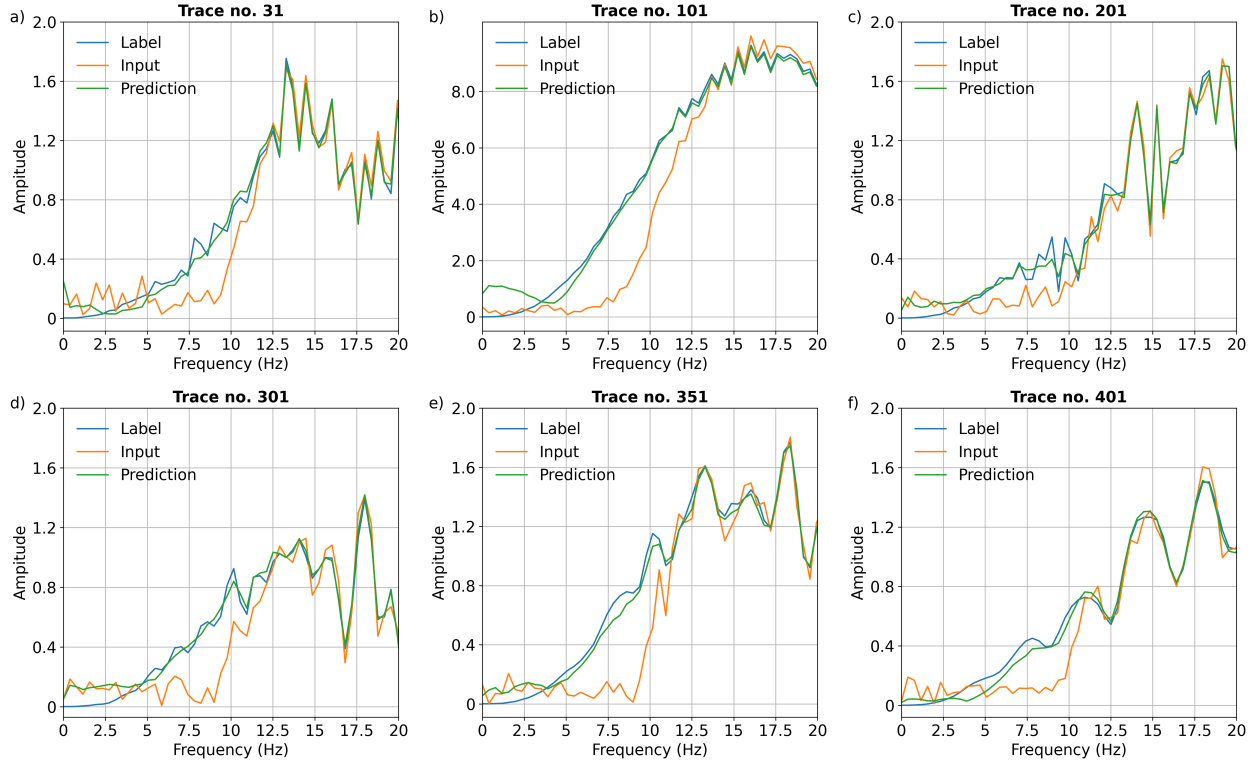


Figure 20: The amplitude spectrum comparison at different locations: (a) $X=0.3$ km, (b) $X=1$ km, (c) $X=2$ km. (d) $X=3$ km, (e) $X=3.5$ km, and (f) $X=4$ km. The test input data lack frequency components below 10 Hz and contain random noise with a noise level of 30.

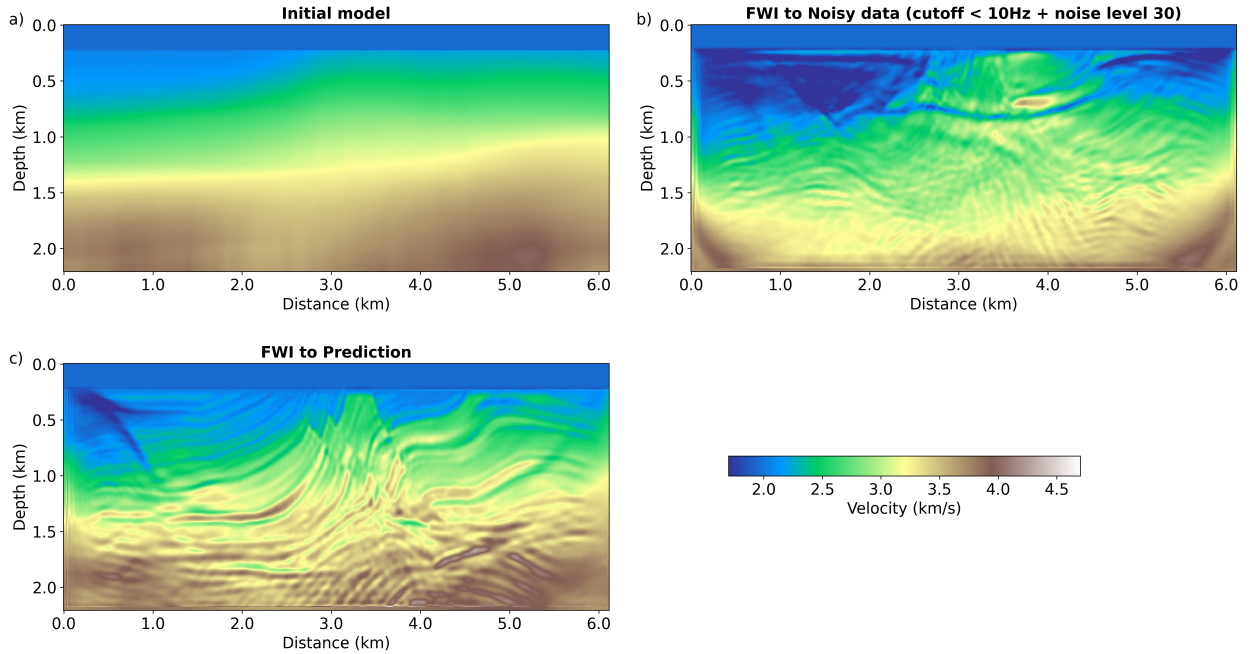


Figure 21: (a) Initial velocity model. Inversion results of (b) the data with a noise level of 30 and (c) the predicted data.

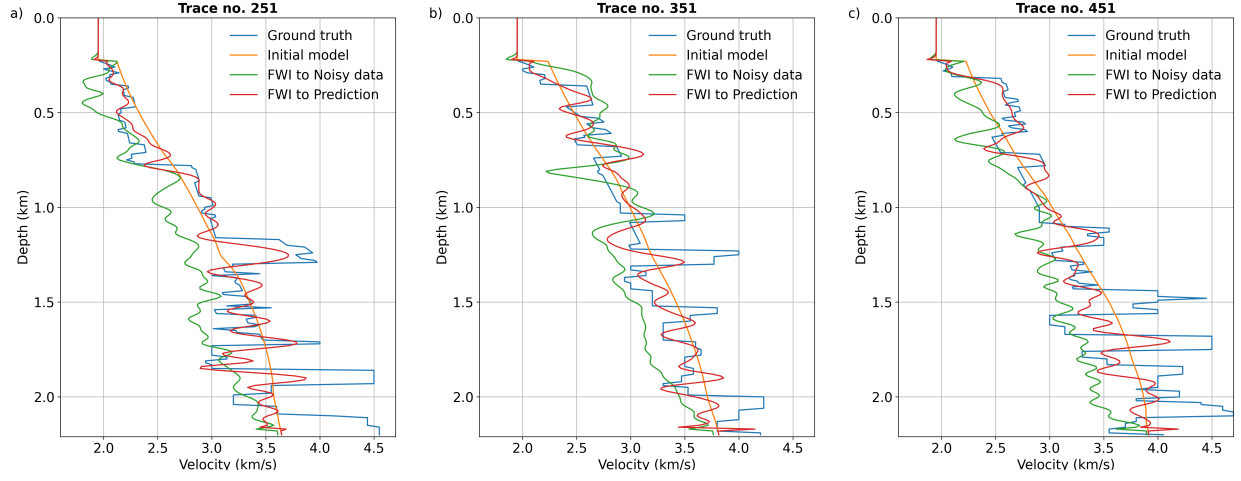


Figure 22: Profiles at different locations: (a) X=2.5 km, (b) X=3.5 km and (c) X=4.50 km.

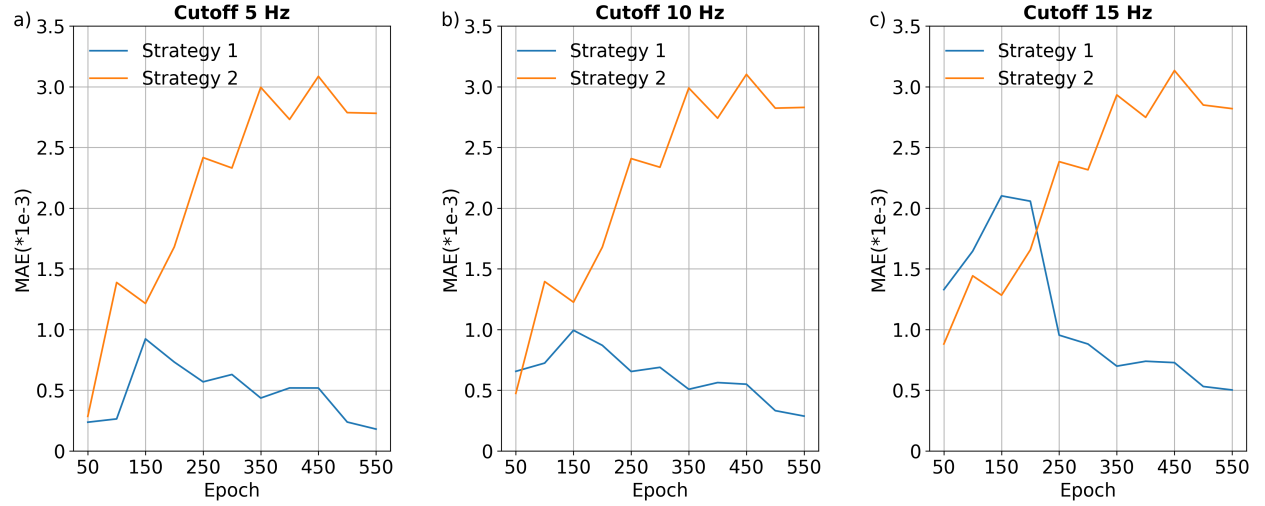


Figure 23: Comparison of the low-frequency extrapolation performance of networks trained with different high-pass filter cutoff frequency setting strategies. Strategy 1 denotes gradually increasing the upper limit of the cutoff frequency during the IDR stage, while Strategy 2 involves setting a fixed cutoff frequency range. The panels (a), (b), and (c) correspond to the MAE metrics of networks trained with these two strategies on three test data, each missing frequencies below 5 Hz, 10 Hz, and 15 Hz, respectively.