Original Paper

Seismic impedance inversion based on cycle-consistent generative adversarial network

Yu-Qing Wang [a, b, c, d], Qi Wang [a, b, c, d], Wen-Kai Lu [a, b, c, d, *], Qiang Ge [e], Xin-Fei Yan [e]

[a] The Institute for Artificial Intelligence, Tsinghua University (THUAI), Beijing, 100084, China
[b] State Key Laboratory of Intelligent Technology and Systems, Tsinghua University, Beijing, 100084, China
[c] Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing, 100084, China
[d] The Department of Automation, Tsinghua University, Beijing, 100084, China
[e] The Research Institute of Petroleum Exploration and Development, China National Petroleum Corporation (CNPC), Beijing, 100083, China

ARTICLE INFO

ABSTRACT

Deep learning has achieved great success in a variety of research fields and industrial applications. However, when applied to seismic inversion, the shortage of labeled data severely influences the performance of deep learning-based methods. In order to tackle this problem, we propose a novel seismic impedance inversion method based on a cycle-consistent generative adversarial network (Cycle-GAN). The proposed Cycle-GAN model includes two generative subnets and two discriminative subnets. Three kinds of loss, including cycle-consistent loss, adversarial loss, and estimation loss, are adopted to guide the training process. Benefit from the proposed structure, the information contained in unlabeled data can be extracted, and adversarial learning further guarantees that the prediction results share similar distributions with the real data. Moreover, a neural network visualization method is adopted to show that the proposed CNN model can learn more distinguishable features than the conventional CNN model. The robustness experiments on synthetic data sets show that the proposed method can achieve better performances than other methods in most cases. And the blind-well experiments on real seismic profiles show that the predicted impedance curve of the proposed method maintains a better correlation with the true impedance curve.

1. Introduction

Seismic impedance inversion is an effective method used for stratigraphic interpretation and reservoir prediction, which can help engineers analyze the spatial structure and physical properties of underground strata. Over the last 50 years, a variety of researches have been conducted to solve the seismic inversion problem. We can divide them into three kinds of methods: model-based methods, data-based methods, and learning-based methods. Model-based methods assume that the relationship between seismic data and inversion targets obeys some deterministic model such as the Robinson seismic convolution model (RSCM). It is the earliest way to solve the inversion problem and is now widely adopted in the seismic exploration industry. Recursive inversion (Lindseth, 1979) and trace integration (Ferguson and Margrave, 1996) compute impedance directly from the predicted reflection coefficients. Generalized linear inversion (GLI) (Cooke and Schneider, 1983) and constrained sparse spike inversion (CSSI) (Sacchi, 1997) iteratively improve the inversion model according to the reconstruction error between the calculated seismic data and the recorded seismic data. However, the recorded seismic data have limited frequency bandwidth and the inversion target has broad bandwidth, leading to the inversion problem underdetermined. To avoid the influence of inaccurate initial models, some data-based methods are proposed, such as well-logging constrained methods (Carron, 1989), sparsity constrained method (Yuan et al., 2015), and geostatistical inversion methods (Gouveia and Scales, 1998). Another famous data-based model is the artificial neural network (Lu et al., 1996) based method. By supposing that there exists a nonlinear mapping between seismic data and the inversion target, ANN is proposed to model the inversion process without defining a

* Corresponding author. The Institute for Artificial Intelligence (THUAI), Beijing, 100084, China.
E-mail address: lwkmf@mail.tsinghua.edu.cn (W.-K. Lu).

Abbreviation		N_{batch}	Batch size
		$N_{\text{all}}^{\text{iter}}$	The number of training iterations
D	The data set of the observed near-well seismic data (labeled data set)	$N_{\text{g}}^{\text{iter}}$	The iteration number of generators
D^*	The data set of the observed seismic data (unlabeled data set)	$N_{\text{d}}^{\text{iter}}$	The iteration number of discriminators
		λ_1	The balance parameter for cycle-consistency loss
d	A sample from data set D	λ_2	The balance parameter for estimation loss
d^*	A sample from data set D^*	c	The number of feature maps
Z	The data set of acoustic impedance	n	The input and output size of CNN model
z	A sample from data set Z	L	The number of encoding layers
Z_H	The data set of the high frequency part of Z	fc	The fully connected layer
z_H	A sample from data set Z_H	e_i	The ith encoding layer
Z_L	The data set of the low frequency part of Z	d_i	The ith decoding layer
z_L	A sample from data set Z_L	$k_j^{e_i}$	The convolution kernel of the jth feature map at layer e_i
Z_L^*	The data set of the estimated low frequency part of impedance for unlabeled seismic data D^*	$k_j^{d_i}$	The convolution kernel of the jth feature map at layer d_i
z_L^*	A sample from data set Z_L^*	W_i^{fc}	The weights of the ith fully connected layer
W_B	The weights of backward-CNN model	b_i^{fc}	The bias of the ith fully connected layer
W_F	The weights of forward-CNN model	r_i	The receptive field size of layer e_i
W_{D1}	The weights of discriminator D1	R_i	The receptive field of layer e_i
W_{D2}	The weights of discriminator D2		
α	Initial learning rate		

specific model which may introduce modeling errors. Recently, Deep learning (LeCun et al., 2015) has led to groundbreaking advancements in many fields such as image classification, image segmentation, and machine translation. Compared to other methods, the convolutional neural network (CNN) contains deeper layers and more sophisticated structures, which leads to stronger abilities of feature extraction and structure representation. The development of deep learning algorithms helps to improve the performance of data-driven seismic inversion methods by adopting convolutional neural networks. Das et al. (2018, 2019) propose a 1D CNN to predict high-resolution impedance data from seismic data and conduct robustness experiments on synthetic data sets. Wu et al. (2018) and Huang et al. (2018) propose to improve FWI based on CNN. Some more complex network structures such as Generative adversarial network (GAN) (Mosser et al., 2018; Li and Luo, 2019), recurrent neural network (RNN) (Richardson, 2018), Unet (a special fully convolutional neural network) (Xu et al., 2019; Wang et al., 2020) and long short-term memory (LSTM) (Guo et al., 2019; Sun et al., 2019; Alfarraj et al., 2019) are also adopted for seismic inversion. Besides, due to the sophisticated structure of CNN, the trained CNN models are always hard to interpret and it causes difficulty in measuring the performance of CNN on new data sets. To address this problem, Bayesian deep learning-based inversion methods (Choi et al., 2020; Luo et al., 2020) are proposed to estimate the uncertainty of inversion results and evaluate the effectiveness of CNN models on new data sets.

As a data-driven algorithm, the practical application of CNN is limited by the number of labeled samples. In order to enhance the performance of CNN when the number of tag datasets is limited, data augmentation (Ding et al., 2017; Jia et al., 2017) and semi-supervised learning (Chen et al., 2018; Chen et al., 2019; Xu et al., 2019) have also been proposed. In the actual seismic exploration process, the stratigraphic physical parameters are usually obtained by drilling. Due to the high cost of drilling and hard acquisition of well-logging parameters, the number of labeled samples that can be used for inversion is small. The application of CNN in seismic inversion is limited by the insufficient number of well-logging samples. Therefore, it is necessary to improve the generalization ability of CNN when only a few numbers of labeled training data are available.

This paper is an extension and improvement of Wang et al. (2019b). In this paper, we address the problem of seismic impedance inversion when the number of labeled samples is small and propose a novel inversion method based on a 1D cycle-consistent generative adversarial network (Cycle-GAN). The proposed 1D Cycle-GAN contains two 1D generative subnets and two 1D discriminative subnets. The generative subnets are used to model the seismic forward and inversion process and they are reversed for each other. The discriminative subnets can extract the distribution of the true data sets and constrain that the distributions of the generated data are identical to the distributions of the true data. And the experimental results on synthetic data show that the Cycle-GAN based inversion algorithm obtains better performance than other CNN based algorithms. The experiments on real seismic data also obtain promising inversion results. Besides, we adopt a neural network visualization method (Erhan et al., 2009; Mordvintsev et al., 2015; Wang et al., 2019a) to interpret the learned features in the trained CNN model and illustrate that the proposed Cycle-GAN model can learn more distinguishable features than conventional CNN model. The main contribution of this paper can be summarized as follows:

(1) We propose a novel seismic impedance inversion method based on a 1D cycle-consistent generative adversarial network.
(2) The robustness of the proposed method under different Signal-to-Noise Ratios (SNRs), numbers of labeled data, and cut-off frequencies of the estimated low-frequency impedances are studied. And different types of inversion methods are compared.
(3) We adopt a neural network visualization method to visualize the learned features in the trained CNN model and compare the visualized features of Cycle-GAN with the conventional open-loop CNN model.

The rest of the paper is organized as follows: First, we review the recent research works related to our paper. Second, we describe the proposed Cycle-GAN-based inversion method, the training procedure, and the adopted neural network visualization method in detail. Third, the experimental setups and results on both synthetic

and real data sets are demonstrated and other deep learning-based inversion methods are compared. Last, we draw conclusions based on the conducted experiments.

2. Related works

2.1. Convolutional neural network

Due to the powerful feature extraction capability of Convolutional Neural Networks (CNNs), it has been widely applied in many research fields and achieved superior performances (LeCun et al., 2015). CNN is usually composed of a feature extraction stage and a task-specific stage. The feature extraction stage is designed to extract features from the input data, and it usually contains stacked convolutional layers. The convolutional layers convolve the local regions of the input images with filter kernels and then are followed by non-linear activation layers to generate the output features. The adopted activation functions are essential for acquiring nonlinear expressions of the input signals and enhancing the representation ability. The task-specific stage differs with different types of tasks. For the classification tasks, the task-specific stage is usually composed of several fully-connected layers with Sigmoid or Softmax functions. And for the regression tasks, it usually contains convolutional layers with Tanh activation functions. The parameters of CNNs are optimized based on the defined loss functions and the backpropagation algorithms. More details of CNNs can be found in LeCun's work (LeCun et al., 2015).

2.2. CNN-based seismic inversion

CNNs have been widely adopted in seismic inversion. Das et al. (2018, 2019) propose an effective synthetic data generating method based on geostatistical simulation and train a simple two-layer CNN model on the generated data sets which share statistically similar facies proportions, rock-physics relations, and source wavelet properties with the true well-logging data. Benefit from the geostatistical simulation method, enough labeled data can be generated to train the CNN model and it partially resolves the small sample size problem. Alfarraj et al. (2019) and Wang et al. (2020) both adopt a semi-supervised learning framework for seismic impedance inversion. They propose to simultaneously model the seismic forward process and the inversion process based on CNN, thus the forward CNN model can serve as a geophysical constrain to the inversion CNN model and the solution space can be reduced. Similarly, a recent work by Sun et al. (2020) trains a hybrid network, which contains a theory-guided wave propagation network to regularize the inversion CNN model, for pre-stack seismic inversion and can effectively reduce the degree of freedom within the neural network. In this paper, we propose a Cycle-GAN-based inversion method. Compared to the previous works (Alfarraj et al., 2019; Wang et al., 2020), we further adopt two discriminators to constrain the distribution consistency between the generated data and the true data.

2.3. Generative adversarial network

Generative adversarial network (GAN) (Goodfellow et al., 2014) includes a generative model and a discriminative model. By imposing an adversarial loss function, the generative model can learn to generate fake data as similar as the real data while the discriminative model tries to distinguish between the fake data and the real data. The loss function of GAN adopts a two-player minimax game strategy. It has been proven that the minimax game reaches its optimal when the discriminator is unable to distinguish between fake data and real data, and the generator can generate

fake data sharing the same distribution as the real data (Goodfellow et al., 2014). Since the proposal of GAN, a series of variants has been proposed to adapt different applications, such as cGAN (Mirza et al., 2014), DCGAN (Radford et al., 2015), InfoGAN (Chen et al., 2016), WGAN (Arjovsky et al., 2017) and Cycle-GAN (Zhu et al., 2017; Yi et al., 2017; Luo et al., 2017). Among them, Cycle-GAN has done a great job in unpaired image translation tasks. It contains two generative neural networks and two discriminative networks. The former is used to translate images between domain X to domain Y and the latter is used to constrain distribution consistency in the two domains. Compared with GAN, Cycle-GAN includes a new kind of loss function named cycle-consistency loss to ensure that the input image can be reconstructed after processed by the two generative networks. For the seismic inversion problem, the well-logging data can be obtained as labeled data. Therefore, we modify the loss function of Cycle-GAN to make it suitable for semi-supervised learning.

3. Methodology

In this section, we first introduce the proposed Cycle-GAN structure for seismic impedance inversion, the adopted generative CNN models, and the discriminative CNN models. Then the designed loss functions are described in detail. Later, we summarize the training procedure. And finally, the adopted neural network visualization method is introduced.

3.1. Network structures

The adopted Cycle-GAN structure for seismic impedance inversion is depicted in Fig. 1. Similar to the cycle-GAN structure in image translation (Zhu et al., 2017; Yi et al., 2017; Luo et al., 2017), the adopted Cycle-GAN model contains two signal domains: seismic domain and impedance domain. The seismic domain includes the seismic data and the impedance domain includes the acoustic impedance data. The main target of Cycle-GAN is to implement the translation between the seismic domain and the impedance domain. Two generative subnets and two discriminative subnets are used in the Cycle-GAN model. As shown in Fig. 1, the generative subnets are denoted as forward-CNN and backward-CNN, the discriminative subnets are denoted as D1 and D2. From the perspective of geophysics, the forward-CNN models the seismic forward process, which computes seismic data from impedance data: $z{\rightarrow}d$, and the backward-CNN models the seismic inversion process, which maps seismic data to the high-frequency part of impedance data: $d{\rightarrow}z_H$. Notice that in order to ease the inversion task and better control the inversion results, the low-frequency part of impedance data z_L is provided by the traditional interpolation method (Wang et al., 2020). Specifically, the low-frequency part of near-well traces are computed from well-logging data, and the low-frequency data between near-well traces are interpolated along seismic events. The forward process and inversion process are reversed for each other, thus during the training, the forward-CNN serves as a good regularization term for backward-CNN. According to the theory of GAN (Goodfellow et al., 2014), the discriminative network D1 is adopted to constrain the distribution consistency between the predicted impedance data and the true impedance data, the discriminator D2 is used to constrain the consistency between the predicted seismic data and the true seismic data.

The adopted neural network structures of generative subnets and discriminative subnets are illustrated in Fig. 2 and Fig. 3. As it shows in Fig. 2, a 1D U-net structure (Ronneberger et al., 2015; Wang et al., 2020) is adopted for both the forward-CNN model and
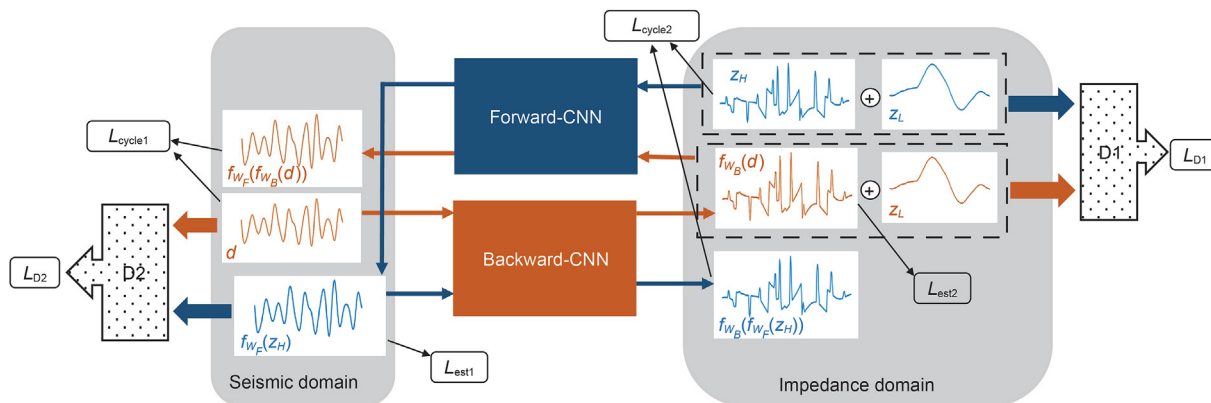
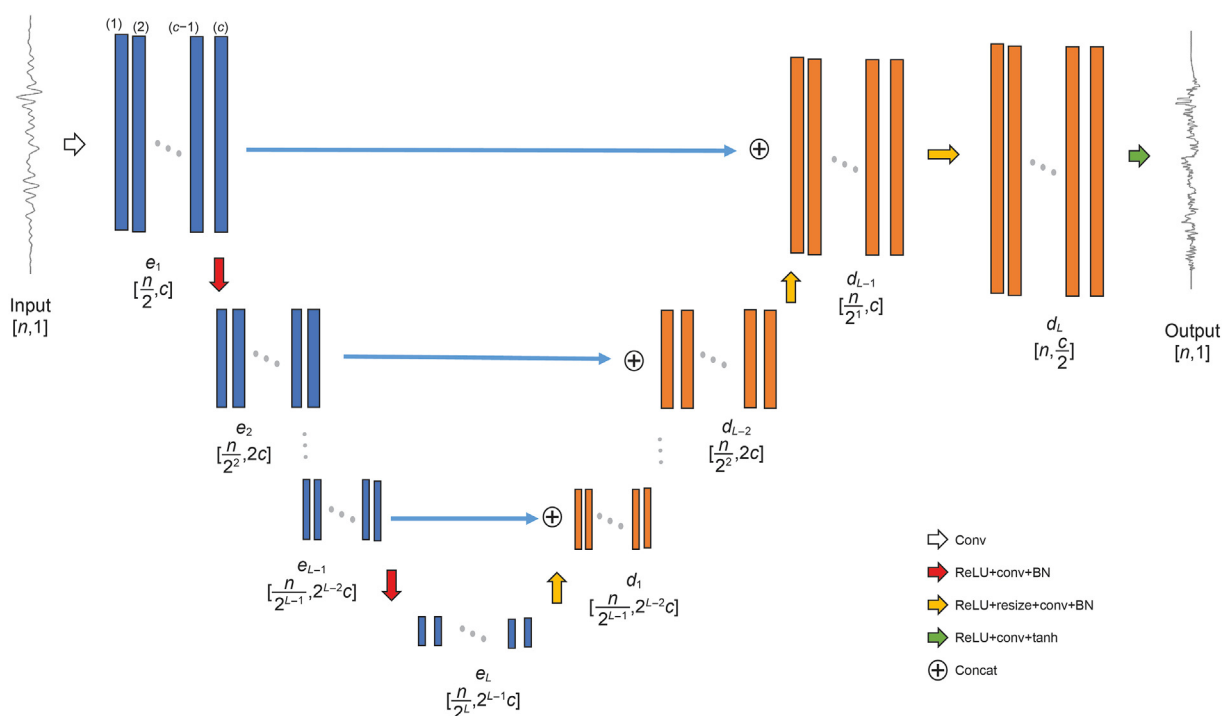**Fig. 1.** The architecture of Cycle-GAN.

Fig. 2. The neural network structure of generative subnets.
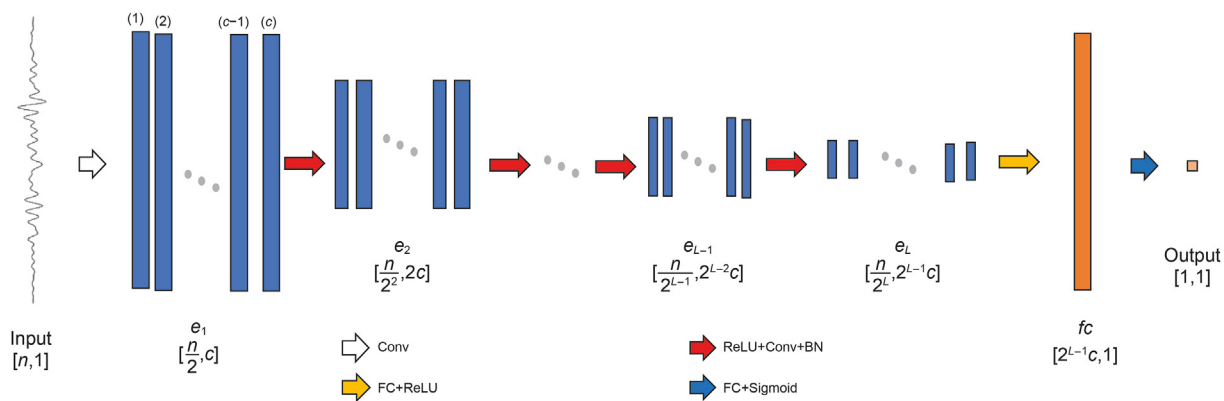

Fig. 3. The neural network structure of discriminative subnets.

backward-CNN model. The input size and output size are both set to be $n \times 1$. The 1D U-net structure includes an encoding process and a decoding process. The encoding layers are denoted as e_i, $i = 1, 2, \cdots, L$, where L represents the number of encoding layers. The white arrow in Fig. 2 represents the convolution operation:

$$e_1 = k_j^{e_1} * \text{Input}, \quad j = 1, 2, \cdots, c \tag{1}$$

where $k_j^{e_1}$ denotes the convolution kernel of the jth feature map at layer e_1, c denotes the number of feature maps. The size of the convolution kernel is set to 3 and the stride of convolution is set to 2. Therefore, the output size of layer e_1 is $\frac{n}{2} \times c$. The red arrow in Fig. 2 represents the "ReLU+conv+BN" operation:

$$e_i = \text{BN}\left(k_j^{e_i} * \text{ReLU}(e_{i-1})\right), \quad j = 1, 2, \cdots, 2^{i-1}c, \quad i = 2, \cdots, L \tag{2}$$

where $k_j^{e_i}$ represents the convolution kernel of the jth feature map at layer e_i, the number of feature maps at layer e_i is set to $2^{i-1}c$, $\text{BN}(\cdot)$ represents the batch normalization operation. Correspondingly, the decoding layers are denoted as d_i, $i = 1, 2, \cdots, L$. The yellow arrow in Fig. 2 represents the "ReLU+resize+conv+BN" operation:

$$d_i = \text{BN}\left(k_j^{d_i} * \text{resize}(\text{ReLU}(t).)\right), j = 1, 2, \cdots, 2^{L-i-1}c, i = 1, \cdots, L,$$
$$t = \begin{cases} e_L, & \text{if } i = 1 \\ \text{concat}(d_{i-1}, e_{L+1-i}), & \text{if } i > 1 \end{cases} \tag{3}$$

where $k_j^{d_i}$ denotes the convolution kernel of the jth feature map at layer d_i, the number of feature maps at layer d_i is set to $2^{L-i-1}c$, $\text{resize}(\cdot)$ is the up-sampling operation and $\text{concat}(\cdot)$ implements the skip connections to merge multi-scale information. The output of the generative subnets is obtained as follows:

$$\text{Output} = \tanh\left(k^o * \text{ReLU}(d_L)\right) \tag{4}$$

where k^o denotes the convolution kernel of the output layer. For the forward-CNN model, the input is acoustic impedance z and the output target is seismic data d. For the backward-CNN model, the input is seismic data d and the output is the high-frequency part of impedance z_H. By adding the predicted z_H with z_L, the absolute impedance can be obtained.

The structure of discriminative subnets is shown in Fig. 3. We adopt a 1D neural network structure similar to AlexNet (Krizhevsky et al., 2012). It contains an encoding process similar to the generative subnets, whose encoding process is implemented by equation (1) and (2), and followed by fully connected layers. The computation of the two fully connected layers can be written as follows:

$$fc = \text{ReLU}\left(W_1^{fc} \times \text{Flatten}(e_L) + b_1^{fc}\right) \tag{5}$$

$$\text{Output} = \text{Sigmoid}\left(W_2^{fc} \times fc + b_2^{fc}\right) \tag{6}$$

where W_i^{fc} denotes the weights of fully connected layer i, b_i^{fc} denotes the bias, $\text{Flatten}(\cdot)$ is the function that reshapes the input data into a vector, $\text{Sigmoid}(\cdot)$ is the non-linear activation function. The output size is set to 1. For discriminator D1, the input is the predicted impedance or the real impedance. For discriminator D2,

the input is the predicted seismic data or the real seismic data. The two generative subnets and two discriminative subnets are constructed according to the structure in Fig. 1.

3.2. Loss functions

The originally proposed Cycle-GAN (Zhu et al., 2017; Yi et al., 2017; Luo et al., 2017) structure is used to tackle unpaired image translation problems and it mainly includes two kinds of losses: cycle-consistent loss and adversarial loss. For the seismic inversion problem, the well-logging data can provide labeled data for training. Therefore, the proposed Cycle-GAN-based impedance inversion method is actually under the semi-supervised learning framework. A new kind of loss, named estimation loss, is adopted to constrain the prediction consistency on the near-well traces.

In this paper, we denote the labeled seismic data set as D and its corresponding target impedance as Z, whose number is small. The unlabeled seismic data set is denoted as D^*. As displayed in Fig. 1, the cycle-consistency loss is used to constrain that the forward-CNN and backward-CNN are reversed for each other. It can be calculated both on labeled data and unlabeled data:

$$L_{\text{cycle1}}(W_B, W_F) = \left\|d^* - f_{W_F}\left(f_{W_F}(d^*) + Z_L^*\right)\right\|_2^2 + \left\|d - f_{W_F}\left(f_{W_F}(d) + Z_L\right)\right\|_2^2 \tag{7}$$

$$L_{\text{cycle2}}(W_B, W_F) = \left\|z_H - f_{W_B}\left(f_{W_F}(z_H + z_L)\right)\right\|_2^2 \tag{8}$$

where $f_{W_B}(\cdot)$ denotes the computation of backward-CNN with weights W_B, $f_{W_F}(\cdot)$ denotes the computation of forward-CNN with weights W_F, $d \in D$ represents the labeled seismic data, $d^* \in D^*$ represents the unlabeled seismic data, $z_H \in Z_H$ represents the high-frequency part of the impedance data, $z_L \in Z_L$ represents the low-frequency part of the impedance data, $z_L^* \in Z_L^*$ represents the estimated low-frequency part for the unlabeled seismic data. From the above equation, we can see that the information contained in unlabeled data can be extracted during the training process.

The estimation loss is calculated based on the predicted results and the labeled targets and it can be written as:

$$L_{\text{est1}}(W_F) = \left\|d - f_{W_F}(z_H + z_L)\right\|_2^2 \tag{9}$$

$$L_{\text{est2}}(W_B) = \left\|z_H - f_{W_B}(d)\right\|_2^2 \tag{10}$$

The estimation loss is used to constrain the convergence on labeled data sets.

The adversarial loss encourages the generative subnets to generate fake data similar to real data and encourages the discriminative subnets to distinguish between the generated fake data and real data. The adversarial game obtains its optimum solution when the distribution of the generated data is identical to the distribution of the true data and the discriminators cannot distinguish between the generated data and the real data (Goodfellow et al., 2014). The adversarial loss can be written as:

$$\min_{W_B} \max_{W_{D1}} L_{D1} = \log f_{W_{D1}}(z_H) + \log\left(1 - f_{W_{D1}}\left(f_{W_B}(d)\right)\right) + \log\left(1 - f_{W_{D1}}\left(f_{W_B}(d^*)\right)\right) \tag{11}$$

Table 1
Cycle-GAN training procedure.

Require: seismic data sets D, D^*, impedance data sets Z_H, Z_L, Z_L^*, forward-CNN model with parameters W_F, backward-CNN with parameters W_B, discriminator D1 with parameters W_{D1}, discriminator D2 with parameters W_{D2}, batch size N_{batch}, the number of training iterations N_{all}^{iter}, the iteration number of generators N_g^{iter}, the iteration number of discriminators N_d^{iter}, balance parameters λ_1, λ_2, initial learning rate α.
1: Randomly initialize parameters W_F, W_B, W_{D1}, W_{D2}
2: for N_{all}^{iter} steps do
3:      repeat
4:          for N_g^{iter} steps do
5:              Randomly sample data $\left\{d^{(i)}\right\}_{i=1}^{N_{batch}/2} \subseteq D$, $\left\{z_H^{(i)}\right\}_{i=1}^{N_{batch}/2} \subseteq Z_H$, $\left\{z_L^{(i)}\right\}_{i=1}^{N_{batch}/2} \subseteq Z_L$, $\{d^{*(i)}\}_{i=1}^{N_{batch}/2} \subseteq D^*$, $\{z_L^{*(i)}\}_{i=1}^{N_{batch}/2} \subseteq Z_L^*$
6:              Update W_F and W_B according to (13) (14)
7:          end for
8:          for N_d^{iter} steps do
9:              Randomly sample data $\left\{d^{(i)}\right\}_{i=1}^{N_{batch}/2} \subseteq D$, $\left\{z_H^{(i)}\right\}_{i=1}^{N_{batch}/2} \subseteq Z_H$, $\left\{z_L^{(i)}\right\}_{i=1}^{N_{batch}/2} \subseteq Z_L$, $\left\{d^{*(i)}\right\}_{i=1}^{N_{batch}/2} \subseteq D^*$, $\left\{z_L^{*(i)}\right\}_{i=1}^{N_{batch}/2} \subseteq Z_L^*$
10:            Update W_{D1} and W_{D2} according to (11)(12)
11:        end for
12:    until all of the labeled data are sampled
13: end for
14: Output: Optimized parameters W_F^*, W_B^*, W_{D1}^*, W_{D2}^*

$$\min_{W_F}\max_{W_{D2}} L_{D2} = \log f_{W_{D2}}(d) + \log f_{W_{D2}}(d^*) \\ + \log\left(1 - f_{W_{D2}}\left(f_{W_F}(z_H + z_L)\right)\right) \tag{12}$$

where $f_{W_{D1}}(\cdot)$ denotes the computation of discriminative subnet D1 with weights W_{D1}, $f_{W_{D2}}(\cdot)$ denotes the computation of discriminative subnet D2 with weights W_{D2}. The adversarial loss L_{D1} can guarantee that the inversion results of the unlabeled seismic data share similar distributions with the true impedance data.

In summary, the overall loss functions for the two generative subnets can be written as:

$$L(W_F) = \lambda_1\left(L_{cycle1} + L_{cycle2}\right) + \lambda_2 L_{est1} \\ - \log\left(f_{W_{D2}}\left(f_{W_F}(z_H + z_L)\right)\right) \tag{13}$$

$$L(W_B) = \lambda_1\left(L_{cycle1} + L_{cycle2}\right) + \lambda_2 L_{est2} \\ - \log\left(f_{W_{D1}}\left(f_{W_B}(d)\right)\right) - \log\left(f_{W_{D1}}\left(f_{W_B}(d^*)\right)\right) \tag{14}$$

where $\lambda_1, \lambda_2 > 0$ are constant parameters to balance different loss terms. And the overall loss functions for discriminators are shown in equation (11) and (12).

3.3. Training procedure

The training procedure for Cycle-GAN is illustrated in Table 1. Similar to the training process of GAN (Goodfellow et al., 2014), we iteratively update the generative subnets and discriminative subnets. The iteration number of generative subnets and

discriminative subnets are set to N_g^{iter}, N_d^{iter}, respectively. The overall number of training iterations is N_{all}^{iter}. Adam (Kingma and Ba, 2014) with initial learning rate α is adopted to update the weights of subnets. After optimization, the trained Cycle-GAN is used to obtain the inversion results.

3.4. Neural network visualization

In order to understand what kind of features the neural networks have learned by stacked convolutional layers, we adopt a neural network visualization method to visualize the trained inversion model (Erhan et al., 2009; Mordvintsev et al., 2015; Wang et al., 2019a). One of the important features in CNNs is that a unit in the feature maps is only related to a region of the input, and this region is named as the "receptive field" of the unit (Luo et al., 2016). Take the CNN model in Fig. 2 for example, when the size of the convolution kernel is set to 3 and the stride is set to 2, the receptive field size of each layer can be computed as follows:

$$r_i = 3r_{i-1} - 2r_{i-2}, \ r_0 = 1, \ r_{-1} = 0, \ i = 1, 2, \cdots, L \tag{15}$$

where r_i represents the receptive field size of layer e_i. It can be seen that as the layer goes deeper, the receptive field size grows. The goal of neural network visualization is to find the special patterns in the receptive field that trigger the unit on the feature maps of each layer. Therefore, the target function of neural network visualization can be designed as follows:

$$R_i^* = \arg\max_{R_i} h_{W_B^*}(R_i), \ i = 1, 2, \cdots, L \tag{16}$$

where W_B^* denotes the optimized parameters of backward-CNN, R_i is defined as the receptive field of a specific unit on layer e_i and $h_{W_B^*}(R_i)$ computes the activation value of the unit. We adopt a gradient ascent algorithm to solve the above maximum optimization problem:

$$R_i^{t+1} \leftarrow R_i^t + \eta \frac{\partial h_{W_B^*}(R_i^t)}{\partial R_i^t} \tag{17}$$

where η represents the learning rate. However, the visualization results of deeper layers are severely affected by checkerboard artifacts (Odena et al., 2016). In order to mitigate these artifacts, we initialize the gradient ascent method by selecting samples from the training data sets D and D^*:

$$R_i^0 = \arg\max_{R_i} h_{W_B^*}(R_i)$$

$$s.t. \ R_i \subseteq \{\varphi(D), \varphi(D^*)\} \tag{18}$$

where $\varphi(\cdot)$ represents the cropping operation. To deal with the above problem, we repeatedly select samples from data sets and choose the kind of patterns that maximize the target function. After the optimization of (18) and (16), the obtained receptive field pattern R_i^* can be used to interpret the unit on the feature map of layer e_i.

4. Results

In this section, we describe the parameter settings of our method, demonstrate the inversion results, and compare them with other methods on both synthetic data sets and real data sets. Moreover, we conduct robustness experiments on the influences of
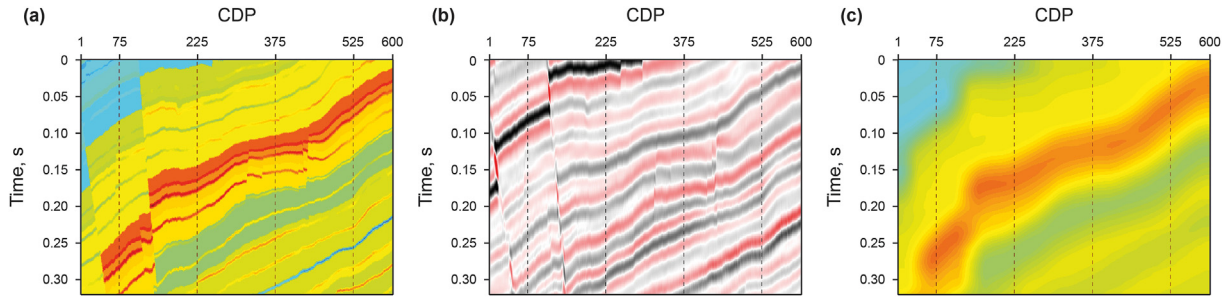
Fig. 4. The synthetic profile. (a) The synthetic impedance profile. (b) The synthetic seismic profile. (c) The low frequency impedance profile.

Table 2
Training parameter settings.

	$N_{\text{all}}^{\text{iter}}$	N_{batch}	$N_{\text{g}}^{\text{iter}}$	$N_{\text{d}}^{\text{iter}}$	α	λ_1	λ_2
Cycle-GAN-0	3000	4	6	4	1×10^{-4}	200	0
Cycle-GAN						200	200

Table 3
Parameter settings of generative subnets and discriminative subnets.

	L	n	c
backward-CNN	5	256	128
forward-CNN	3	256	128
D1 and D2	5	256	64

noise, the numbers of labeled samples, and the cut-off frequencies of z_L. Also, the neural network visualization results are displayed and compared.

4.1. Synthetic data examples

A 2D synthetic profile, shown in Fig. 4, is adopted to examine our method. Fig. 4a displays the synthetic impedance profile, the traces located at CDP 75, 225, 375, 525 are regarded as labeled traces Z and the rest traces are unlabeled traces. Fig. 4b displays the synthetic seismic profile, which is generated from Fig. 4a by convolving with a 20 Hz Ricker wavelet, the 4 traces at CDP 75, 225, 375, 525 are labeled traces D and the rest traces are unlabeled traces D^*. Fig. 4c displays the low-frequency impedance profile whose cut-off

frequency is 5.3 Hz, the 4 traces at CDP 75, 225, 375, 525 are labeled traces Z_L and the rest traces are unlabeled traces Z_L^*.

Table 2 illustrates the training parameter settings, including the setting of total training epochs $N_{\text{all}}^{\text{iter}}$, the training batch size N_{batch}, the iteration number of generators $N_{\text{g}}^{\text{iter}}$, the iteration number of discriminators $N_{\text{d}}^{\text{iter}}$, the initial learning rate α and the weighting parameters λ_1, λ_2. We give two parameter setting strategies in Table 2, where "Cycle-GAN-0" represents the parameter settings of the originally proposed Cycle-GAN (Zhu et al., 2017; Yi et al., 2017; Luo et al., 2017) and "Cycle-GAN" represents the parameter settings of our proposed Cycle-GAN. Furthermore, we refer to "Cycle-GAN-Refine" as firstly adopting "Cycle-GAN-0" to obtain an initial inversion model and then fine-tuning it with "Cycle-GAN". In the experiments, we find that Cycle-GAN-Refine tends to get better inversion results than other methods.

The parameter settings of generative subnets and discriminative subnets are shown in Table 3, where L denotes the number of encoding layers, n denotes the input and output size, c denotes the number of feature maps at layer e_1. Notice that parameter L of backward-CNN is set to 5 while L of forward-CNN is set to 3, it is because that the seismic forward process is based on a simple convolution model and is much easier to model than the seismic inversion process. The loss curves of Cycle-GAN during training are displayed in Fig. 5a, from which we can see that the loss values decrease as the epoch increases, and after 500 epochs, the decrease of loss values slows down.

We compare our method with model-based inversion (Russell et al., 1991) and other deep learning-based inversion methods,
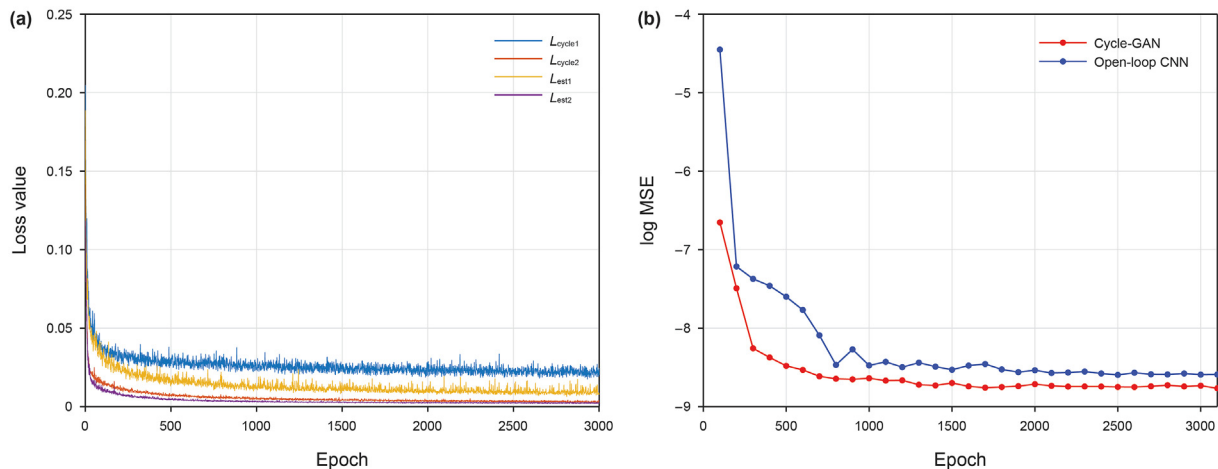

Fig. 5. The loss and log MSE curve during the training. (a) The loss curve of Cycle-GAN. (b) The log MSE curves of the predicted impedance on unlabeled data sets during training (Red curve: Cycle-GAN, blue curve: Open-loop CNN). It can be seen that the testing error of Cycle-GAN decreases faster than those of Open-loop CNN.

Table 4
Comparison of different methods based on the average MSE of 5 repeated experiments. The optimal result is highlighted with **bold red font** and the suboptimal result is highlighted with **bold font**.

Method	Average MSE ($\times 10^{-4}$) of the predicted impedance data	Average MSE ($\times 10^{-4}$) of the reconstructed seismic data
Model-based	5.5819	Not applied
Semi-CRNN	2.8114	6.9437
LSTM	2.6272	Not applied
Open-loop CNN	1.8829	Not applied
Closed-loop CNN	1.5842	9.5252
Cycle-GAN-0	4.8991	19.6074
Cycle-GAN	**1.5680**	**5.7962**
Cycle-GAN-Refine	**1.4905**	**4.0804**
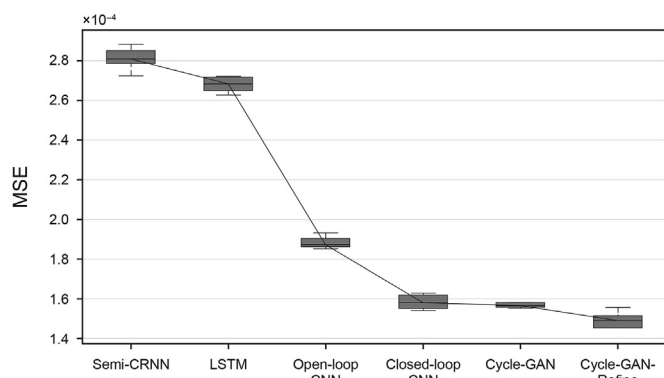
Fig. 6. Box plot of the MSE value in the 5 repeated experiments shown in Table 4.

including Semi-CRNN (Alfarraj et al., 2019), LSTM, Open-loop CNN, and Closed-loop CNN (Wang et al., 2020), where LSTM and Open-loop CNN adopts the conv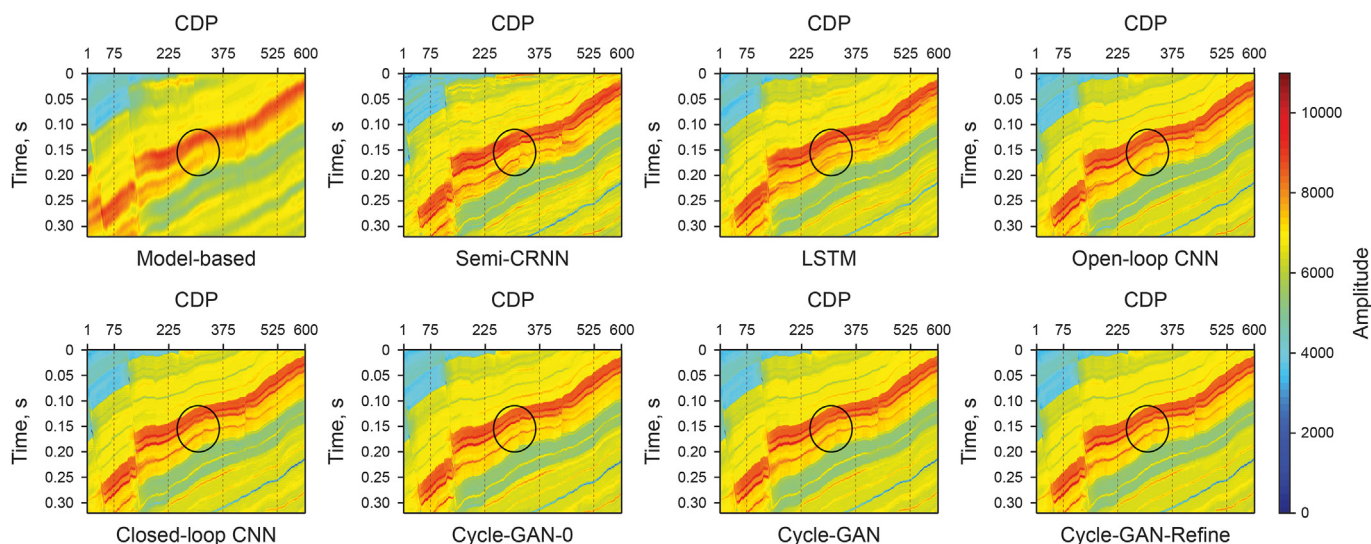entional CNN training strategy that only use one single deep learning model to learn the inversion process based on the loss function in (10), the inversion CNN model of Open-loop CNN is the same as the one used in our Cycle-GAN model. We repeat the experiments 5 times under different initialization weight parameters. The average Mean square error (MSE) of the inversion results are illustrated in Table 4 and the box plots of MSE in the 5 repeated experiments are displayed in Fig. 6. It can be seen that Cycle-GAN-Refine obtains the lowest average MSE and Cycle-GAN obtains the sub-lowest average MSE. The deep learning-based inversion methods achieve better inversion results than the traditional model-based inversion method. Cycle-GAN-0 is actually trained on unpaired data sets, thus the inversion error of Cycle-GAN-0 is higher than other methods. From Fig. 6, we can see that the inversion errors of Closed-loop CNN and Cycle-GAN are quite close, but Cycle-GAN-Refine always tends to get better inversion results than other methods. Furthermore, we compare the testing errors of Open-loop CNN and Cycle-GAN during the training process. As it is shown in Fig. 5b, the testing error of Cycle-GAN decreases faster than Open-loop CNN, it can be inferred that benefit from the semi-supervised learning scheme and the constraints of

Fig. 7. The 2D inversion results of different methods. The top row displays the inversion results of model-based inversion, Semi-CRNN, LSTM, and Open-loop CNN. The bottom row displays the inversion results of Closed-loop CNN, Cycle-GAN-0, Cycle-GAN and Cycle-GAN-Refine.
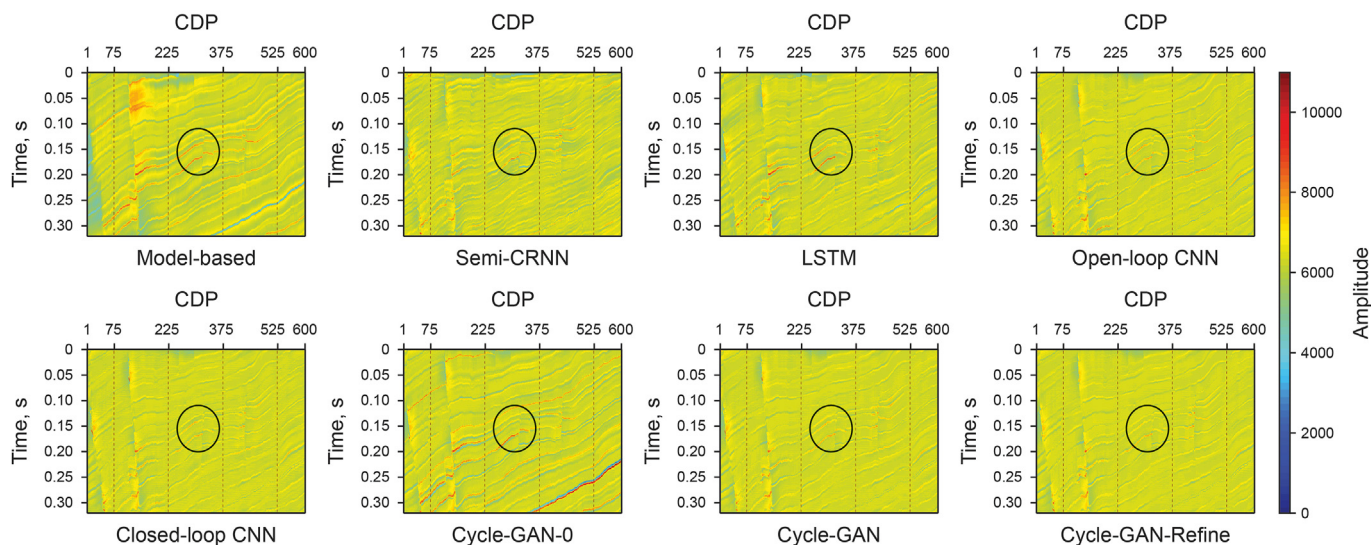
Fig. 8. The inversion error maps of different methods. The top row displays the error maps of model-based inversion, Semi-CRNN, LSTM, and Open-loop CNN. The bottom row displays the error maps of Closed-loop CNN, Cycle-GAN-0, Cycle-GAN and Cycle-GAN-Refine.
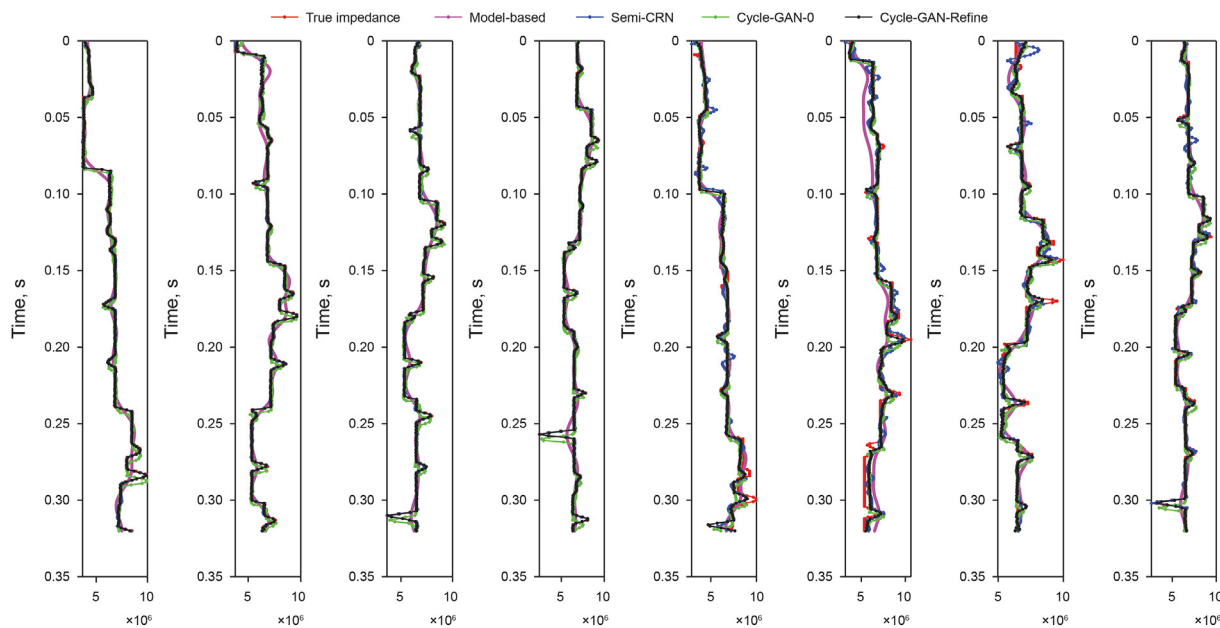

Fig. 9. The inversion results of different methods at CDP 75, 225, 375, 525, 50, 150, 300, and 400. The red lines represent the true impedance, the magenta, blue, green, and black lines represent the inversion results of model-based inversion, Semi-CRNN, Cycle-GAN-0 and Cycle-GAN-Refine.

discriminators, Cycle-GAN can learn reasonable inversion models faster than Open-loop CNN.

Figs. 7–9 display inversion results of the above methods. The top rows in Figs. 7 and 8 show the inversion results of model-based inversion, Semi-CRNN, LSTM, and Open-loop CNN. And the bottom rows in Figs. 7 and 8 show the inversion results of Closed-loop CNN, Cycle-GAN-0, Cycle-GAN, and Cycle-GAN-Refine. As marked by the black circles in Fig. 7, there are some discontinuities along seismic events in the inversion results of Semi-CRNN, LSTM, and Open-loop CNN, and there are some blurs in the inversion results of model-based inversion, Closed-loop CNN, and Cycle-GAN. The inversion profile of Cycle-GAN-Refine is clearer and more continuous along seismic events. Besides, it can be seen from the error maps in Fig. 8 that Cycle-GAN-Refine obtains lower inversion errors than other

methods. Fig. 9 shows the predicted impedance sequences of different methods at 4 labeled traces (CDP 75, 225, 375, 525) and 4 unlabeled traces (CDP 50, 150, 300, 400). For labeled traces, the inversion results of Semi-CRNN and Cycle-GAN-Refine correlate well with the true impedance, while the predicted impedance sequences of model-based inversion have lower resolution, and those of Cycle-GAN-0 show some misfits with the true impedance sequence. For unlabeled traces, the misfitting between the inversion results and the true impedance increases. However, the predicted sequences of Cycle-GAN-Refine correlate better with the true impedance than other methods.

Moreover, we further reconstruct the seismic profile based on the predicted impedance profile. If the inverted impedance profile is correct and the forward model is reliable, the reconstructed
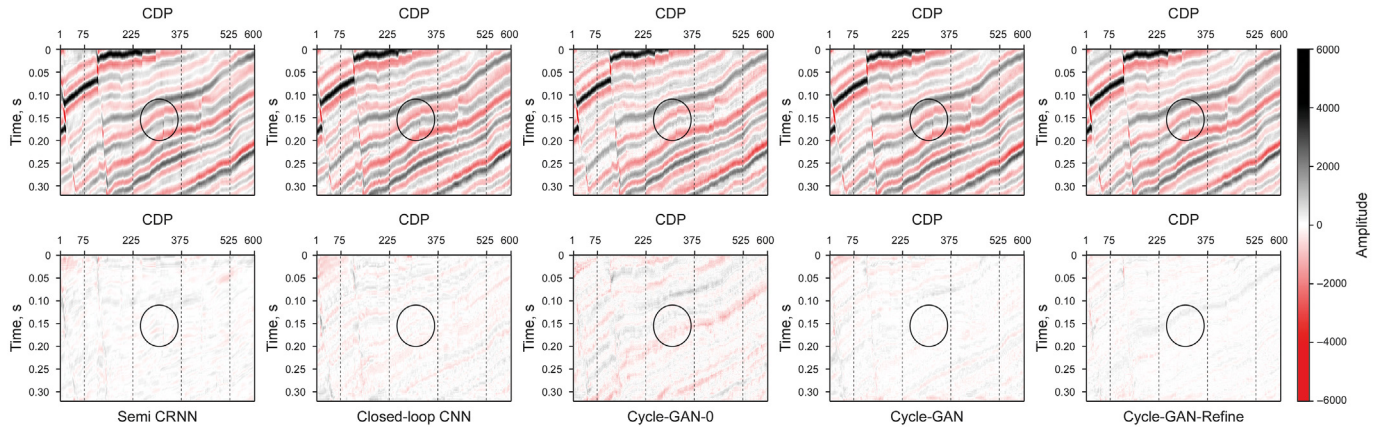
Fig. 10. The reconstructed seismic data and corresponding error maps of different methods. The top row displays the reconstructed seismic data of Semi-CRNN, Closed-loop CNN, Cycle-GAN-0, Cycle-GAN and Cycle-GAN-Refine. The bottom row displays the corresponding error maps.

Table 5
Comparison under different numbers of labeled traces. The optimal results are highlighted with **bold red font** and the suboptimal results are highlighted with **bold font**.

Method	Number of labeled traces (MSE: $\times 10^{-4}$)					
	1	2	4	10	20	50
Open-loop CNN	4.6108	4.3264	1.8314	1.9314	1.3913	0.9077
Closed-loop CNN	**3.7618**	**2.6007**	1.5562	1.2391	**0.8673**	0.8640
Cycle-GAN-0	7.6978	5.6374	5.7582	6.1340	7.7942	7.4307
Cycle-GAN	5.8231	2.9948	**1.5391**	**1.2373**	0.9219	**0.7783**
Cycle-GAN-Refine	**3.5121**	**2.5124**	**1.4517**	**1.1801**	**0.8593**	**0.8256**

seismic profile should consist well with the synthetic seismic profile. We compare the reconstruction results of Semi-CRNN, Closed-loop CNN, Cycle-GAN-0, Cycle-GAN, and Cycle-GAN-Refine, the comparison results are illustrated in Table 4 and Fig. 10. From Table 4, we can see that Cycle-GAN-Refine achieves the lowest reconstruction errors. And similar conclusions can be drawn from Fig. 10. The lower reconstruction errors of our methods indicate that the predicted impedance profile of our method is more reliable than other methods.

4.2. Robustness experiments

In order to examine the robustness of our method, we conduct robustness experiments on synthetic data sets and compare our method with Open-loop CNN, Closed-loop CNN, and Cycle-GAN-0. Table 5 and Fig. 11a illustrate the comparison results under different numbers of labeled traces. The optimal results are highlighted with the bold red font in Table 5 and the suboptimal results are highlighted with bold font. As the number of labeled traces increases,
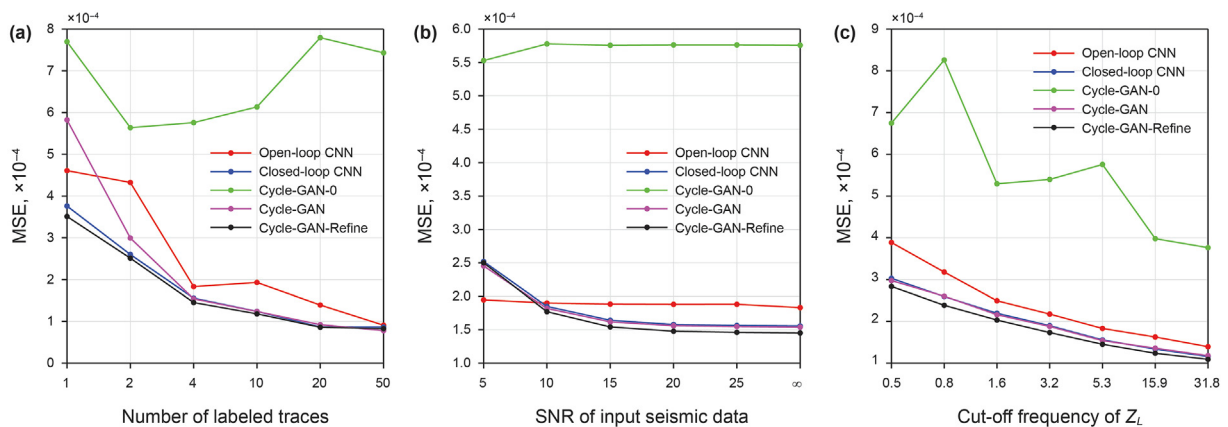

Fig. 11. Comparisons of MSE under different settings. (a) The MSE curves of different methods under different numbers of labeled traces. (b) The MSE curves under different SNR values of the input seismic data. (c) The MSE curve under different cut-off frequencies of Z_L.

Table 6
Comparison under different SNR values of the input seismic data. The optimal results are highlighted with bold red font and the suboptimal results are highlighted with **bold font**.

Method	SNR of input seismic data (MSE: $\times 10^{-4}$)					
	5	10	15	20	25	∞
Open-loop CNN	1.9465	1.9001	1.8838	1.8804	1.8810	1.8314
Closed-loop CNN	2.5204	1.8488	1.6419	1.5780	1.5637	1.5562
Cycle-GAN-0	5.5278	5.7786	5.7575	5.7631	5.7631	5.7582
Cycle-GAN	**2.4546**	**1.8189**	**1.6177**	**1.5601**	**1.5476**	**1.5391**
Cycle-GAN-Refine	2.5010	1.7698	1.5420	1.4780	1.4609	1.4517

Table 7
Comparison under different cut-off frequencies of Z_L. The optimal results are highlighted with bold red font and the suboptimal results are highlighted with **bold font**.

Method	Cut-off frequency of Z_L (Hz) (MSE: $\times 10^{-4}$)						
	0.5	0.8	1.6	3.2	5.3	15.9	31.8
Open-loop CNN	3.8880	3.1806	2.4947	2.1749	1.8580	1.6262	1.3938
Closed-loop CNN	3.0288	**2.5948**	2.1946	1.8980	1.5562	**1.3368**	**1.1603**
Cycle-GAN-0	6.7472	8.2586	5.2957	5.3992	5.7582	3.9806	3.7648
Cycle-GAN	**2.9760**	2.6030	**2.1595**	**1.8787**	**1.5391**	1.3577	1.1795
Cycle-GAN-Refine	2.8358	2.3854	2.0323	1.7325	1.4517	1.2368	1.0914

the inversion errors of Open-loop CNN, Closed-loop CNN, Cycle-GAN, and Cycle-GAN-Refine tend to decrease. Because that Cycle-GAN-0 is trained under the unsupervised scheme, the number of labeled traces has no obvious correlations with the inversion performance of Cycle-GAN-0. It can be seen from Table 5 and Fig. 11a that Cycle-GAN-Refine obtains better inversion results than other methods except that when the number of labeled traces equals 50, the inversion error of Cycle-GAN is lower than Cycle-GAN-Refine. And when the number of labeled traces is larger than 2, the performances of Closed-loop CNN and Cycle-GAN are similar. However, the performance of Cycle-GAN gets worse when the number of labeled traces becomes smaller. We can conclude that Cycle-GAN is not robust enough in the case of small numbers of labeled traces and Cycle-GAN-Refine shows better robustness under different numbers of labeled traces.

Table 6 and Fig. 11b show the comparison results under different Signal-to-Noise Ratios (SNRs) of the input seismic data. Random Gaussian noise with different variances is added to the input seismic data to obtain different levels of SNRs. From Table 6 and Fig. 11b, we can see that when the SNR value is higher than 10 dB, the inversion accuracy of Cycle-GAN-Refine and Cycle-GAN is higher than other methods, which means the trained Cycle-GAN model is robust when the input seismic data contains little noise. But when the SNR value is low, Open-loop CNN obtains better inversion results than other methods, and the performance of Cycle-GAN and Cycle-GAN-Refine drops significantly. Therefore, if Cycle-GAN and Cycle-GAN-Refine are applied to noisy seismic data, noise attenuation methods are first required to enhance the quality of the seismic data.

Table 7 and Fig. 11c display the comparison results under different cut-off frequencies of Z_L. With the increasing of cut-off frequency, the inversion errors of different methods decrease. The
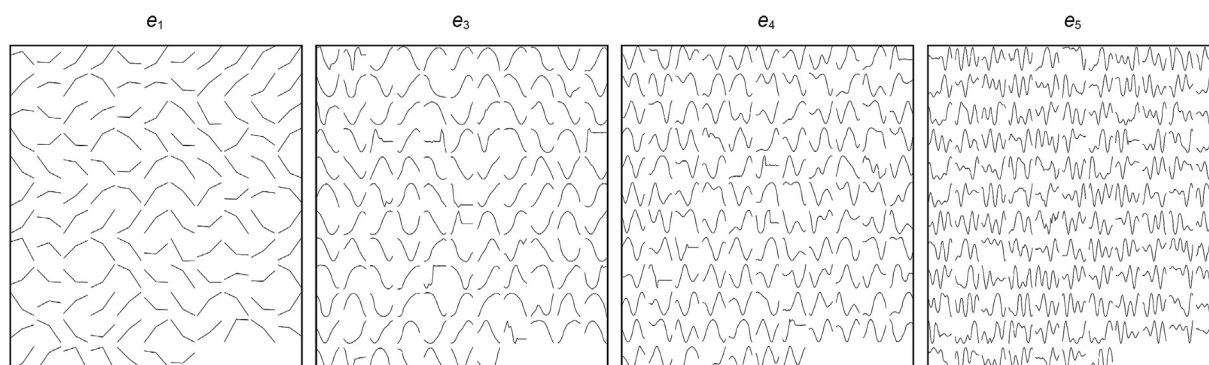

Fig. 12. The visualization features of different encoding layers. From left to right are the visualization results of the first 128 features in the layer e_1, e_3, e_4 and e_5, respectively.
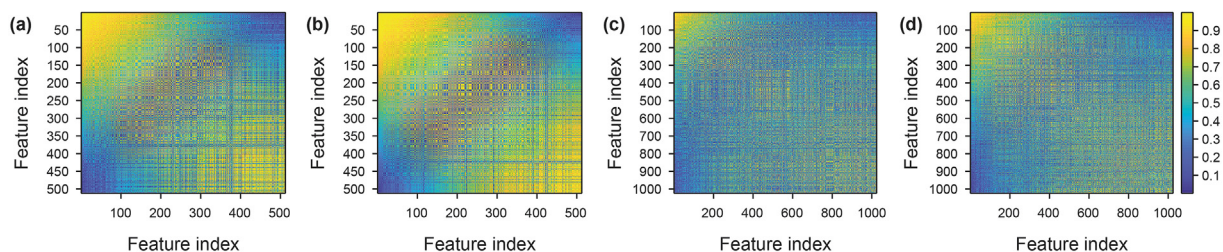
Fig. 13. The correlation matrices of visualized features. (a) and (b) are the correlation matrices of Cycle-GAN and Open-loop CNN at encoding layer e_3. (c) and (d) are the correlation matrices of Cycle-GAN and Open-loop CNN at encoding layer e_4.
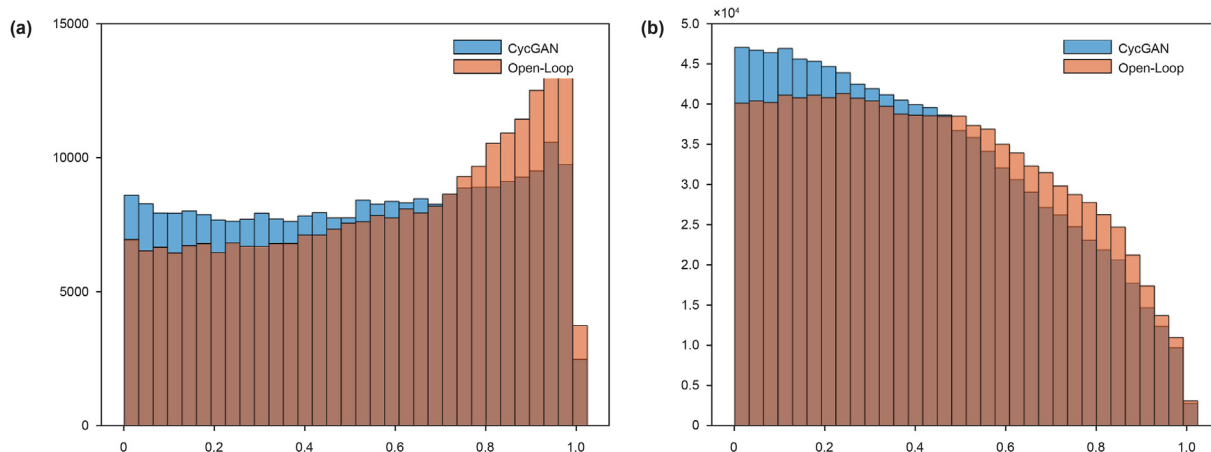

Fig. 14. The histograms of correlation matrices. (a) The histogram of the encoding layer e_3. (b) The histogram of the encoding layer e_4. The blue bars display the histogram of Cycle-GAN and the red bars display the histogram of Open-Loop CNN.

Table 8
The average correlation coefficients of the visualized features on different encoding layers.

Layer	Open-loop CNN	Cycle-GAN
e_1	0.6744	0.6360
e_2	0.6337	0.6139
e_3	0.5671	0.5187
e_4	0.4337	0.4059
e_5	0.3313	0.3135

inversion accuracy of Cycle-GAN and Closed-loop CNN is quite close under different cut-off frequencies, and the performance of Cycle-GAN-Refine is better than other methods. According to the slop of the MSE curves in Fig. 11c, it can be concluded that Cycle-GAN-Refine shows higher robustness than other methods under different cut-off frequencies of Z_L.

4.3. Neural network visualization results

Based on the neural network visualization methods introduced in 3.4, we demonstrate the visualization results of the backward-CNN in Cycle-GAN and compare it with Open-loop CNN in this section. According to equation (17), the receptive field sizes of encoding layers e_1, e_2, e_3, e_4, e_5 can be computed as 3, 7, 15, 31, 62. And according to the parameter setting in Table 3, the feature map numbers of e_1, e_2, e_3, e_4, e_5 are 128, 256, 512, 1024, 2048, respectively. Fig. 12 displays the first 128 visualization features of layer e_1, e_3, e_4 and e_5, from which we can see that as the layer goes deeper, the learned features become more complicated and structured.

Moreover, we compare the visualization results of Cycle-GAN and Open-loop CNN in Fig. 13, Fig. 14, and Table 8. The correlation coefficients of the visualized features at each layer are computed. A larger correlation coefficient indicates the higher similarity between the two visualized features. Fig. 13a and b shows the correlation matrices of the visualized features at layer e_3. Comparing Fig. 13a and b, it can be seen that Open-loop CNN has more large correlation coefficients than Cycle-GAN, and a similar situation can be found in Fig. 13c and d. We further draw the histograms of the correlation matrices in Fig. 14, from the distribution of correlation coefficients we can see that there are more coefficients close to 1 in the Open-loop histograms than in the Cycle-GAN histograms. Table 8 illustrates the average correlation coefficient values of each encoding layer. The average coefficient values of Cycle-GAN is lower than Open-loop CNN. From the above visualization results, it can be concluded that the learned features of Cycle-GAN are more abundant and diverse than the features of Open-loop CNN. The employment of unlabeled data during the training and the use of discriminators can enrich the features learned by CNN.

4.4. Real data examples

In this section, we adopt a 2D real seismic profile to testify our method. The real seismic profile is shown in Fig. 15a. The traces located at CDP 38, 314, 489, and 603 are near-well traces whose target impedance can be computed from well-logging data. And the rest traces are regarded as unlabeled traces. In order to evaluate our inversion results quantitatively, we further consider the trace located at CDP 489 as blind well. The model parameter settings and the training parameter settings are the same as synthetic data experiments shown in Tables 3 and 4.
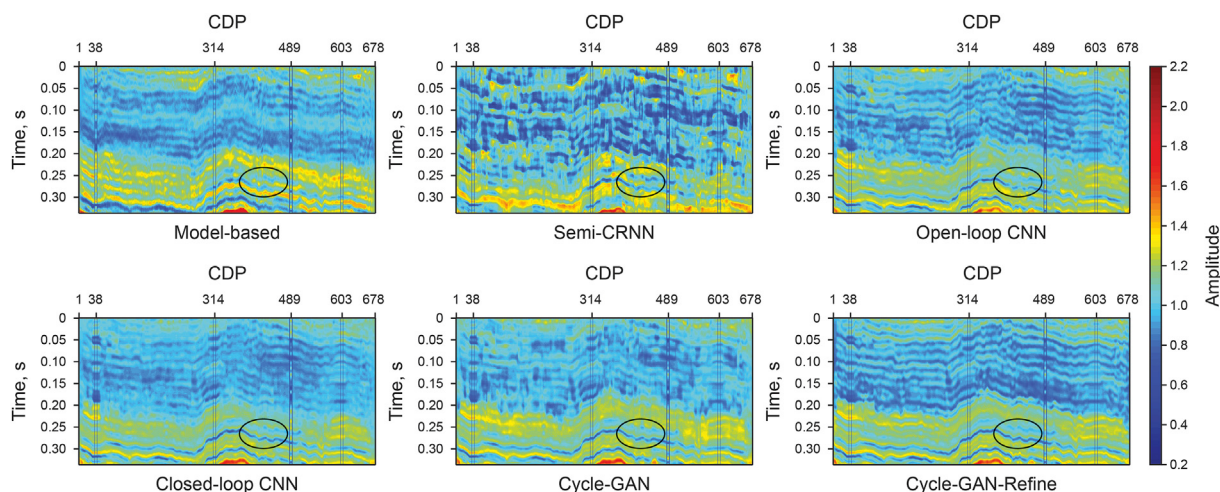
Fig. 15. The inversion results of the real 2D seismic profile. Top row displays the predicted impedance of model-based inversion, Semi-CRNN and Open-loop CNN. Bottom row displays the predicted impedance of Closed-loop CNN, Cycle-GAN and Cycle-GAN-Refine.
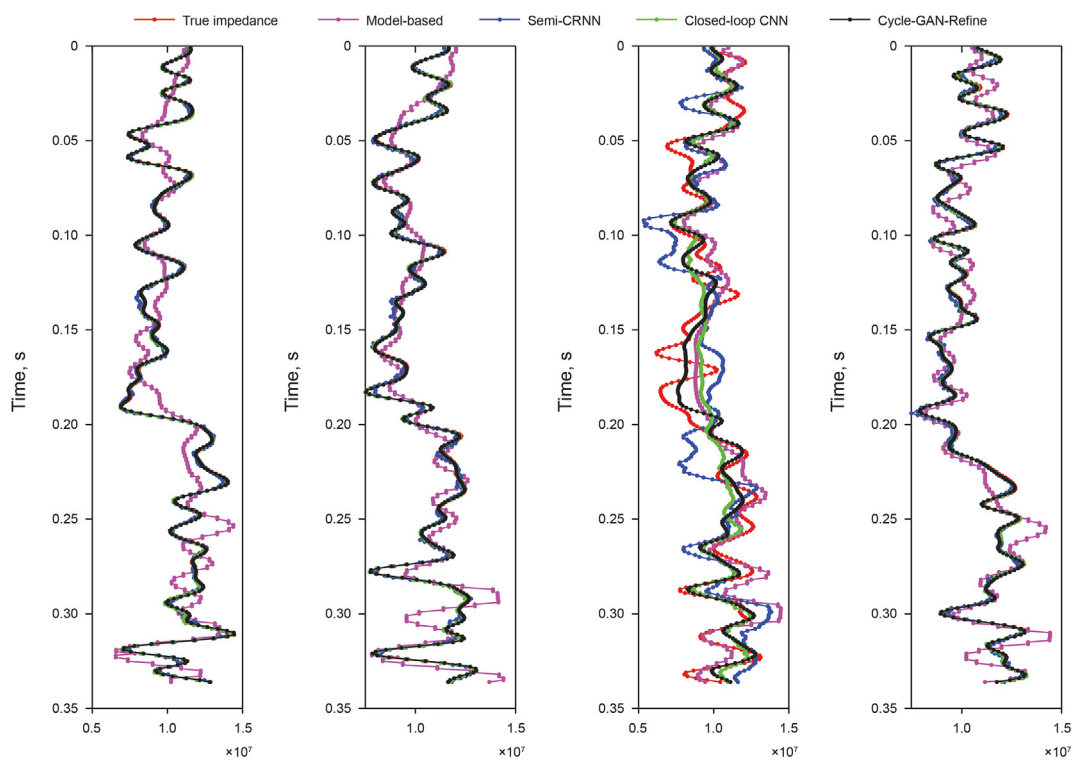

Fig. 16. The predicted impedance of different methods at CDP 38, 314, 489, 603. The red lines represent the true impedance, the magenta, blue, green, and black lines represent the inversion results of model-based inversion, Semi-CRNN, Closed-loop CNN and Cycle-GAN-Refine.

We compare our method with model-based inversion (Russell et al., 1991), Semi-CRNN (Alfarraj et al., 2019), Open-loop CNN, and Closed-loop CNN (Wang et al., 2020). The predicted impedance profiles are displayed in Fig. 15. Fig. 16 compares the predicted impedance of different methods on labeled near-well traces (CDP 38, 314, 603) and unlabeled blind-well trace (CDP 489). As marked by the black circles in Fig. 15, compared with other methods, the inversion results of Cycle-GAN-Refine are more continuous and clearer. We can further see from Fig. 16 that, on labeled near-well traces, the inversion results of Semi-CRNN, Closed-loop CNN and Cycle-GAN-Refine are much more accurate than the results of

model-based inversion. And on blind well, the predicted impedance curve of Cycle-GAN-Refine correlates better with the true impedance curve, especially between 0.15s and 0.2s. In addition, Table 9 illustrates the comparison of correlation coefficients and MSE on blind well, from which we can see that the predicted impedance of Cycle-GAN-Refine obtains the optimal correlation with the true impedance and Cycle-GAN achieves the suboptimal inversion results. Furthermore, in Fig. 17, we investigate the reconstructed seismic profile calculated from the predicted impedance of Cycle-GAN-Refine. It can be seen that the reconstructed seismic profile is well consistent with the real seismic

Table 9
The prediction correlation coefficients and MSE of different methods on blind well. The optimal result is highlighted with **bold red font** and the suboptimal result is highlighted with **bold font**.

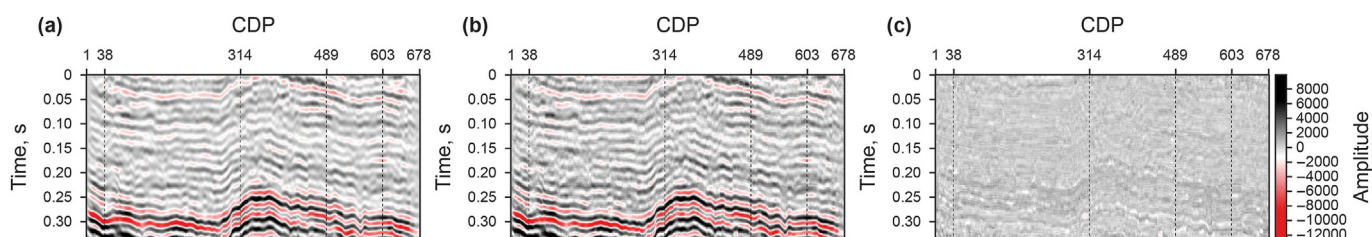| Method	Correlation Coefficient of Z_H^*	MSE
Model-based	0.6730	0.003947
Semi-CRNN	0.2452	0.008312
Open-loop CNN	0.6851	0.003627
Closed-loop CNN	0.7142	0.003584
Cycle-GAN	**0.7276**	**0.003174**
Cycle-GAN-Refine	**0.7828**	**0.002657**

Fig. 17. The reconstructed seismic data of Cycle-GAN-Refine. (a) The real seismic profile. (b) The reconstructed seismic profile of Cycle-GAN-Refine. (c) The corresponding error map of (b).

profile and the reconstructed error shown in Fig. 17c is quite low. Therefore, we can conclude that our method can be effectively applied to real seismic profiles.

5. Conclusion

In this article, we propose a seismic impedance inversion method based on a cycle-consistent generative adversarial network (Cycle-GAN). The proposed Cycle-GAN model adopts two generative subnets to model the seismic forward and inversion process and adopts two discriminative subnets to guarantee the distribution consistency between the prediction results and the target data sets. Unlabeled data is used during the training procedure to compensate for the shortage of labeled data. Robustness experiments are conducted on synthetic data sets, including the influence of different SNR values, the number of labeled traces, and the cut-off frequencies of low-frequency impedance. The experimental results show that the proposed method achieves better performance than other deep learning-based methods in most cases. And we adopt a neural network visualization method to interpret the learned features of Cycle-GAN and compare them with the features of Open-loop CNN, the visualization results illustrate that Cycle-GAN can learn more distinguishable and diverse features than Open-loop CNN. Finally, the blind-well experiment on real seismic profile shows that the inversion results obtained by the proposed method are better correlated with the well-logging impedance data than other methods.

References

Alfarraj, M., AlRegib, G., 2019. Semi-supervised learning for acoustic impedance inversion. In: SEG Technical Program Expanded Abstracts, vol. 2019. Society of Exploration Geophysicists, pp. 2298—2302. https://doi.org/10.1190/segam2019-3215902.1.

Arjovsky, M., Chintala, S., Bottou, L., 2017. Wasserstein generative adversarial networks. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70, vol. 2017, pp. 214—223 arXiv preprint arXiv:1701.07875.

Carron, D., June 1989. High Resolution Acoustic Impedance Cross-Sections from Wireline and Seismic Data. SPWLA 30th Annual Logging Symposium, Denver, Colorado.

Chen, X., Duan, Y., Houthooft, R., et al., 2016. Infogan: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. Advances in Neural Information Processing Systems, pp. 2172—2180 arXiv preprint arXiv:1606.03657.

Cooke DA and Schneider WA. Generalized linear inversion of reflection seismic data. Geophysics, 48(6), 665—676. https://doi.org/10.1190/1.1441497.

Choi, J., Kim, D., Byun, J., 2020. Uncertainty Estimation in Impedance Inversion Using Bayesian Deep Learning. SEG Technical Program Expanded Abstracts, vol. 2020. Society of Exploration Geophysicists, pp. 300—304. https://doi.org/10.1190/segam2020-3428098.1.

Das, V., Pollack, A., Wollner, U., Mukerji, T., 2018. Convolutional Neural Network for Seismic Impedance Inversion. SEG Technical Program Expanded Abstracts, vol. 2018. Society of Exploration Geophysicists, pp. 2071—2075. https://doi.org/10.1190/segam2018-2994378.1.

Das, V., Pollack, A., Wollner, U., et al., 2019. Convolutional neural network for seismic impedance inversion[J]. Geophysics 84 (6), R869–R880. https://doi.org/10.1190/geo2018-0838.1.

Erhan, D., Bengio, Y., Courville, A., et al., 2009. Visualizing Higher-Layer Features of a Deep Network, vol. 1341. University of Montreal, p. 1, 3.

Ferguson, R.J., Margrave, G.F., 1996. A simple algorithm for band-limited impedance inversion. CREWES Research Report 8 (21), 1–10.

Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu B, D., et al., 2014. Generative Adversarial Nets. Advances in Neural Information Processing Systems, 2672–2680. ACM, 26722680.

Gouveia, W.P., Scales, J.A., 1998. Bayesian seismic waveform inversion: parameter estimation and uncertainty analysis. J. Geophys. Res.: Solid Earth 103 (B2), 2759–2779. https://doi.org/10.1029/97JB02933.

Guo, R., Zhang, J., Liu, D., et al., 2019. Application of bi-directional long short-term memory recurrent neural network for seismic impedance inversion. In: 81st EAGE Conference and Exhibition 2019, vol. 2019. European Association of Geoscientists & Engineers, pp. 1–5. https://doi.org/10.3997/2214-4609.201901386, 1.

Huang L, Polanco M and Clee TE. Initial experiments on improving seismic data inversion with deep learning. IEEE Scientific Data Summit, 1–3. doi:10.1109/NYSDS.2018.8538956.

Kingma, D.P., Ba, J., 2014. Adam: a Method for Stochastic Optimization arXiv preprint arXiv:1412.6980.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. 25, 1097–1105.

LeCun Y, Bengio Y and Hinton G. Deep learning. Nature. 521. 436–444. doi:10.1038/nature14539.

Lindseth, R.O., 1979. Synthetic sonic logs–A process for stratigraphic interpretation. Geophysics 44, 3–26. https://doi.org/10.1190/1.1440922.

Li, Q., Luo, Y., 2019. Using GAN priors for ultrahigh resolution seismic inversion. SEG Technical Program Expanded Abstracts. Society of Exploration Geophysicists 2453–2457. https://doi.org/10.1190/segam2019-3215520.1.

Lu WK, Li YD and Mu YG. Seismic inversion using error-back-propagation neural network. Chin. J. Geophys., 39, 292– 301. (in Chinese).

Luo, Y., Chen, Y., Jin, R., et al., 2020. High-resolution Bayesian adaptive impedance inversion method and application. SEG Technical Program Expanded Abstracts 2020. Society of Exploration Geophysicists 290–294. https://doi.org/10.1190/segam2020-3415697.1.

Luo, P., Wang, G., Lin, L., Wang, X., 2017a. Deep dual learning for semantic image segmentation. In: Proceedings of the IEEE Conference on Computer Vision, vols. 21–26 arXiv:1701.04128.

Luo, W., Li, Y., Urtasun, R., Zemel, R., 2017b. Understanding the effective receptive field in deep convolutional neural networks. In: Conference on Neural Information Processing Systems. Barcelona, Spain. arXiv:1701.04128.

Mirza, M., Osindero, S., 2014. Conditional Generative Adversarial Nets, p. 2014 arXiv preprint arXiv:1411.1784.

Mordvintsev, A., Olah, C., Tyka, M., 2015. Inceptionism: Going Deeper into Neural Networks. https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html.

Mosser, L., Dubrule, O., Blunt, M., 2018. Stochastic Seismic Waveform Inversion Using Generative Adversarial Networks as a Geological Prior. First EAGE/PESGB Workshop Machine Learning, 03720 arXiv:1806.

Odena, A., Dumoulin, V., Olah, C., 2016. Deconvolution and checkerboard artifacts. Distill 1 (10), e3.

Radford, A., Metz, L., Chintala, S., 2015. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks arXiv preprint arXiv:1511.06434.

Richardson, A., 2018. Seismic Full-Waveform Inversion Using Deep Learning Tools and Techniques, 07232 arXiv:1801.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 234–241.

Russell, B., Hampson, D., 1991. Comparison of poststack seismic inversion methods. In: SEG Technical Program Expanded Abstracts. Society of Exploration Geophysicists, pp. 876–878. https://doi.org/10.1190/1.1888870.

Sun, J., Niu, Z., Innanen, K.A., et al., 2019. A theory-guided deep learning formulation of seismic waveform inversion. SEG Technical Program Expanded Abstracts. Society of Exploration Geophysicists 2343–2347. https://doi.org/10.1190/segam2019-3215762.1.

Sun, J., Innanen, K., Huang, C., 2020. Training a physics-guided convolutional neural network for seismic inversion. Earth ArXiv. https://doi.org/10.31223/osf.io/57n2w.

Sacchi, Mauricio D., 1997. Reweighting strategies in seismic deconvolution. Geophys. J. Int. 129 (3), 651–656. https://doi.org/10.1111/j.1365-246X.1997.tb04500.x.

Wang, Y., Lu, W., Liu, J., et al., 2019a. Random seismic noise attenuation based on data augmentation and CNN. Chinese J. Geophy. Chinese Edition 62 (1), 421–433. https://doi.org/10.6038/cjg2019M0385 (in Chinese).

Wang, Y., Ge, Q., Lu, W., et al., 2019b. Seismic impedance inversion based on cycle-consistent generative adversarial network. SEG Technical Program Expanded Abstracts 2019. Society of Exploration Geophysicists 2498–2502. https://doi.org/10.1190/segam2019-3203757.1.

Wang, Y., Ge, Q., Lu, W., Yan, X., Aug. 2020. Well-logging constrained seismic inversion based on closed-loop convolutional neural network. IEEE Trans. Geosci. Rem. Sens. 58 (8), 5564–5574. https://doi.org/10.1109/TGRS.2020.2967344.

Wu, Y., McMechan, G.A., 2018. Feature-capturing full waveform inversion using a convolutional neural network. 88th Annual International Meeting, SEG, Expanded Abstracts. https://doi.org/10.1190/segam2018-2963265.1, 2061–2065.

Xu, P., Lu, W., Tang, J., et al., 2019. High-resolution reservoir prediction using convolutional neural networks. 81st EAGE Conference and Exhibition 2019. European Assoc. Geosci. Eng. 2019 (1), 1–5. https://doi.org/10.3997/2214-4609.201901392.

Yi Z, Zhang H, Tan P and Gong M. Dualgan: UnsupeRvised dual learning for image-to-image translation. Proceedings of the IEEE International Conference on Computer Vision, 2849–2857. doi:10.1109/ICCV.2017.310.

Yuan, S., Wang, S., Luo, C., et al., 2015. Simultaneous multitrace impedance inversion with transform-domain sparsity promotion. Geophysics 80 (2), R71–R80. https://doi.org/10.1190/geo2014-0065.1.

Zhu, J.Y., Park, T., Isola, P., Efros, A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, vols. 2223–2232 arXiv:1703.10593.