

Two-dimensional Seismic Velocity Inversion via Enhanced Multi-view Convolutional Neural Networks for Regression

Chuang Pan

School of Artificial Intelligence

School of Computer and Electronic Information

Nanjing Normal University

Nanjing, China

202243027@stu.njnu.edu.cn

Yiran Huang⁺

Department of Computer Science

School of Arts and Sciences

Boston University

Boston, USA

yrhuang@bu.edu

Qingzhen Wang

National Engineering Research Center

of Offshore Oil and Gas Exploration

CNOOC Research Institute Ltd

Beijing, China

wangqzh2@cnooc.com.cn

Jun Li

School of Artificial Intelligence

School of Computer and Electronic Information

Nanjing Normal University

Nanjing, China

lijuncst@njnu.edu.cn

Jianhua Xu[✉]

School of Artificial Intelligence

Nanjing Normal University

Nanjing, China

xujianhua@njnu.edu.cn

xujianhua1999@tsinghua.org.cn

Abstract—Seismic exploration is a mainstream technique to find subsurface raw oil and gas in energy industry, which stimulates seismic wave on the Earth’s surface, receives seismic waves from the subsurface layers, processes these seismic data and lastly infers oil and gas existence. All these steps are associated with a physical parameter: velocity. Nowadays, this parameter is estimated via a so-called seismic waveform inversion method, which is to derive a subsurface velocity model of the Earth by inverting seismic data observed at the surface. Besides this traditional physics-guided full waveform inversion, recently, more attention has paid on data-driven inversion techniques, mainly deep learning based pre-stack inversion, which builds a nonlinear mapping from a multi-shot record to a velocity profile for two-dimensional situation. Usually, a multi-shot record is widely simulated as a multiple channel color image. However, the multiple shots are located at the distinct surface positions, which is different from the fixed camera position of images, and whose shot gathers include less complementary and much redundant information. In order to characterize this situation, we regard each shot gather as a view in machine learning and propose an enhanced multi-view convolutional neural network for regression (MVCNNR) to velocity inversion, in this paper. Our MVCNNR generalizes multi-view convolutional neural network for 3D shape recognition into a pixel-level regression network that further is added both addition connections in residual network and skipping concatenation connections in U-net, to improve its performance. Experiments on four types of velocity datasets (Layered, Faulted, SaltBody, and SaltDome) show that our MVCNNR is superior to two representative data-driven techniques (FCNVMB and VelocityGAN), via quantitative evaluation and visualization analysis.

Index Terms—velocity inversion, multi-view convolutional neural network, concatenation connection, U-net, addition connec-

tion, residual network

I. INTRODUCTION

Seismic exploration [1], [2] is a mainstream technique for searching for raw oil and gas in energy industry, which consists of three stages: seismic data acquisition, processing and interpretation, as shown in Fig. 1 for two-dimensional land exploration:

In the acquisition stage, the vibration source is moved to stimulate reflected seismic waves that propagate in the subsurface layers and then are received by geo-phones or receivers uniformly along the surface. And each source location corresponds to a shot gather and many gathers construct a shot record, as shown in Fig 1(a).

In the data processing procedure, these collected records are processed via various methods and software, for example, de-noising, stacking, de-convolution, and migration to create many high-solution images to approximate the subsurface geological structure and geophysical parameters, as shown in Fig. 1(b).

Lastly, geologists and geophysicists interpret processed seismic data to achieve more accurate subsurface situation manually or automatically and then predict where and whether there exists an industrial oil or gas reservoir, which will be validated by drilling a borehole, as shown in Fig. 1(c) and (d).

It is noted that the difference between land exploration and marine one is the acquisition and validation, as shown in Fig. 2.

The entire seismic exploration procedure is greatly associated with a physical parameter: velocity, which measures

⁺Now Yiran Huang is a PhD candidate at Department of Data Science, New Jersey Institute of Technology, email: yh87@njit.edu

[✉]Corresponding author

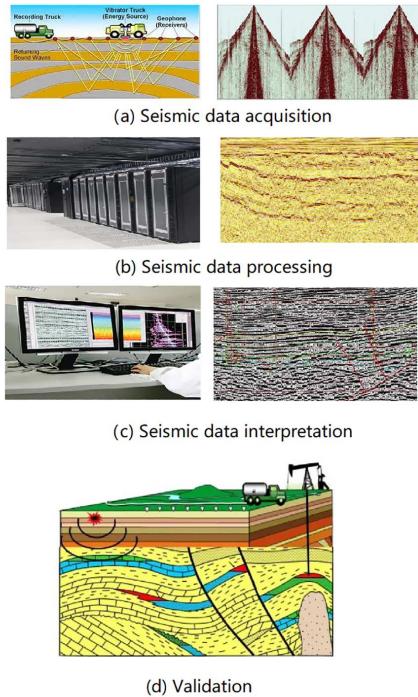


Fig. 1. Basic stages of seismic exploration for oil and gas on land.



Fig. 2. Acquisition and validation on marine exploration.

how fast the seismic waves propagate in the subsurface layers. Overall, it goes high as the depth from the Earth surface increases. More detailedly, this parameter not only depends on different kinds of rocks (for example, sand stone, mud stone, shale and so on), but also is associated with the oil and gas existence. Therefore, building accurate velocity models is of great significance to seismic exploration. Nowadays, so-called physics-guided full-waveform inversion (FWI) [3], [4] is a representative technique, which finds out a velocity model that minimizes a misfit function between synthetic and real observed seismic data. Although this kind of FWI methods has shown great success in many applications, it may be limited in some cases due to its solution non-uniqueness, high computational costs, and low robustness. Therefore, many researchers aim to search for an alternative strategy [5]–[7]. In recent five years, many novel velocity inversion methods have been developed in the framework of deep learning (DL), since these deep networks have a powerful non-linear mapping ability from seismic records to velocity models, which induces a novel kind of data-driven FWI for velocity inversion [5]–[7]

under the supervised learning paradigm [8].

Usually, there are two main stages in the supervised learning setting, i.e., preparing a batch of training instances and training a proper learner. In the former stage, we traditionally collect many instances that then are annotated manually or automatically. However, this principle does not hold in seismic inversion, since geologists and geophysicist can not directly annotate those collected seismic records accurately. To this difficulty, a trade-off strategy is to build velocity profiles as labels at first and then to solve acoustic or elastic wave equation to create their corresponding multi-shot records as instances. This implies that existing data-driven inversion is based on synthetic data rather than real-world data, which becomes a special point in data-driven inversion methods. So far, to best of our knowledge, there are four types of synthetic datasets: Layered [9], Faulted [10], SaltBody [11] and SaltDome [12], all of which will be used in our experiments.

On the other hand, many researchers aim to find some proper deep learning architectures to fulfil seismic inversion task. In [13], the first data-driven velocity inversion method was proposed, which is based on traditional convolutional neural networks (CNN) [14]–[16], that is, several convolutional layers cascading two full-connected layers. Specially, each original multi-shot record is converted into a velocity semblance as its input, and each two-dimensional velocity profile is reshaped into a vector as its output. More efficiently, zero-offset (single channel) gathers are feeded to CNN in [17].

Subsequently, many researchers switch to full convolutional networks (FCN), which consist of convolutional layers only and are originally applied in image segmentation [18]. In [19] the FCN is directly used as an inversion network. A representative FCN is U-net [20], [21], which has an encoder-decoder architecture and skipping concatenation connections from lower level feature maps of encoder to high level maps of decoder, to improve network performance. FCNVMB [11] is a typical U-net based inversion technique. InversionNet [22] is a special U-net, its encoder and decoder is linked via a fully connected layer. This network is generalized to a multi-scale inversion case including a low-resolution InversionNet and a high-resolution one [23], and an InversionNet3D for 3D velocity situation [24].

Generative adversarial network (GAN) was proposed in 2014 [25], [26], which consists of two sub-networks: a generator and a discriminator, and then is trained via minmax game principle. The trained generator could act as an image generator. In VelocityGAN [9], its generator has a encoder-decoder architecture, which is used a velocity inversion network.

To the best of our knowledge, all aforementioned data-driven velocity inversion methods regard a multi-shot seismic record as a multi-channel image, as shown in Fig. 3. As we known, a color image is described using three complementary components (i.e., RGB), and the camera position is fixed. However, different shot gathers are acquired at distinct source positions, as shown in Fig. 3, among which there is less complementary and more redundant information. This situation is much close to multi-views in multi-view learning [27], [28].

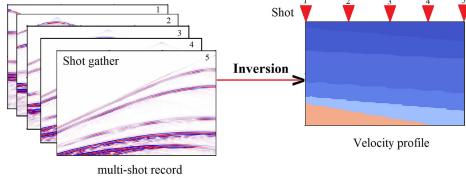


Fig. 3. Existing data-driven inversion framework, regarding a multi-shot record as a multi-channel color image.

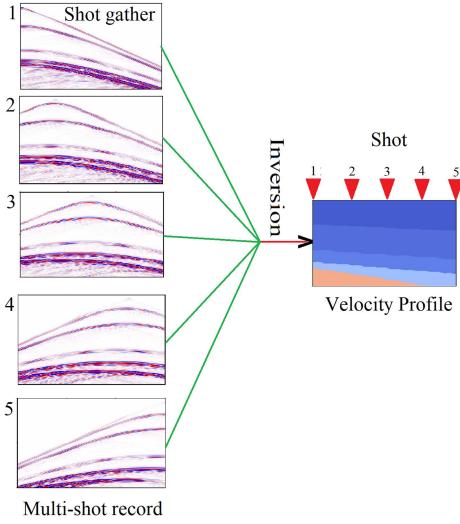


Fig. 4. Our data-driven inversion framework, where a multi-shot record is regarded as multi-view images.

In this study, we apply multi-view learning paradigm to seismic velocity inversion, as shown in Fig. 4. The first multi-view convolutional neural networks were proposed in [29] for 3D shape recognition. Each view passes through a convolutional neural network (CNN), multiple view feature maps are fused via view pooling, and finally the fused feature maps are dealt with using a CNN and a fully connected layer. At first, in this paper, we extend this classification network into a multi-output regression one for velocity inversion. Further, we add two connections: addition and concatenation from residual network and U-Net, respectively. The former is to divide CNNs into several convolutional blocks and add shortcut addition connections used in residual networks [30], to avoid possible gradient vanishing. The latter is to embed skipping concatenation connections in U-Net from each view to fused CNN, to enhance network performance. This enhanced multi-view convolutional neural network for velocity inversion is referred to as MVCNNR, simply. The detailed experiments on four synthetic datasets (Layered, Faulted, SaltBody and SaltDome) demonstrate that our MVCNNR performs better than two typical inversion networks (i.e., FCNVMB [11] and VelocityGAN [9]). Further, on SaltBody dataset, we execute an ablation experiment to show how and whether two addition and concatenation operations work.

Summarily, there are two main contributions in this study: (a) to construct an enhanced multi-view convolutional neural network for velocity inversion, (b) to validate our proposed inversion technique, via comparing it with two existing methods on four datasets.

The remaining parts of this paper are organized as follows. The details of MVCNNR are described in Section II. Section III shows our experimental results and comparison with two existing methods, as well as an ablation experiment. Finally, our conclusions are outlined in Section IV.

II. METHODOLOGY

In this section, we regard a multi-shot seismic record as multiple views in multi-view learning [27], and then propose an enhanced multi-view convolutional neural network for velocity inversion (MVCNNR) after introducing three basic networks: multi-view CNN for classification, residual network for classification and U-net for image segmentation.

A. Three Related Deep Neural Networks

In this subsection, we introduce three deep neural networks: multi-view convolutional neural network, U-net and residual network, which is related to our MVCNNR.

1) *Multi-view convolutional neural network*: Convolutional neural network (CNN) is one of widely-mentioned deep neural networks [15], [16] for classification, which usually consists of several convolutional layers and a few fully connected layers. Multi-view convolutional neural networks (MVCNN) [29] is a variant of traditional CNN, which splits the beginning of CNN into several sub-CNNs in parallel. Each image in the multi-view data passes through its sub-CNN to extract its individual feature representation, independently. These view specific features are integrated via a view pooling operation to construct a multiple channel feature map, which is subsequently dealt with via the second sub-CCN. The final softmax layer executes a classification task. This architecture could make full use of complementary information hidden in multi-view data, and remove much redundant information among different view images. Such an MVCNN has become a landmark CNN architecture in 3D shape recognition direction [28].

2) *U-Net*: The U-Net [20] is a typical FCN for image segmentation [31], as a pixel-level binary classification problem, which has two attractive characteristics: encoder-decoder architecture and skipping concatenation connection.

The entire architecture consists of an encoder, a bridging block and a decoder. Its encoder decreases the height and width of feature maps and increases the number of channels, step by step. For its decoder, the reverse operations are applied.

The most specific point in U-net is its skipping concatenation connections, which concatenates the output of encoders with that of decoders, symmetrically. This strategy could copy the low-level information to the high-level ones, and thus could build an effective information path from low-levels to high levels, which facilitates gradient backward propagation for training procedure and compensates the high-level information via low-level detailed information.

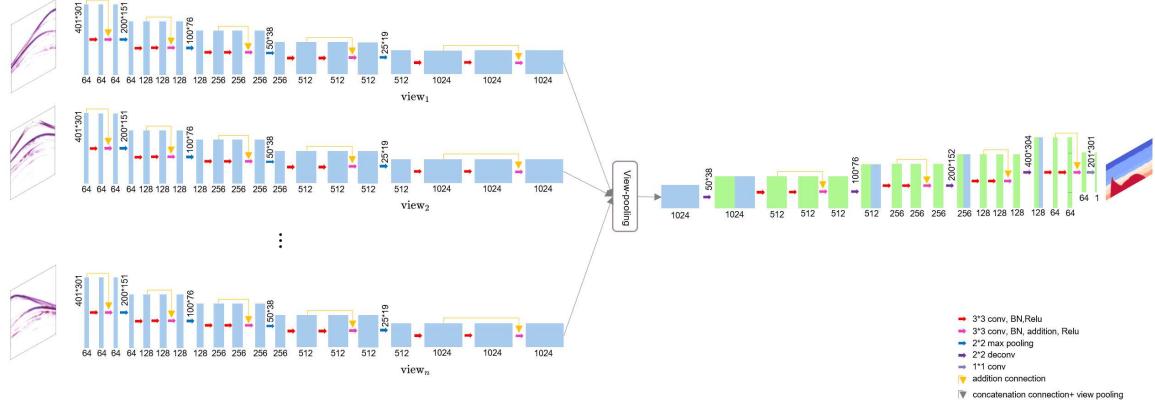


Fig. 5. Main architecture of our MVCNNR, including multi-view architecture and addition connection.

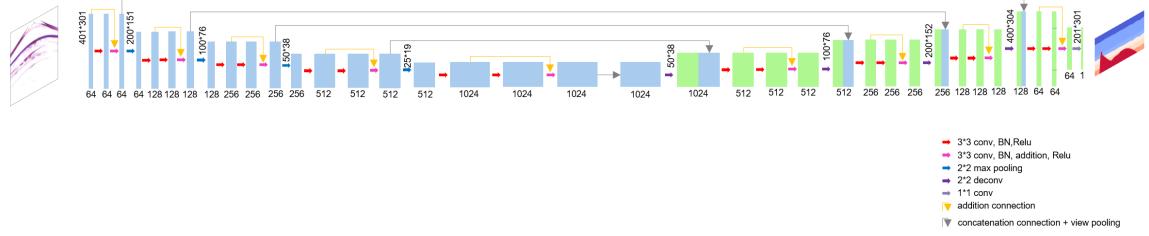


Fig. 6. Concatenation connection from one view to decoder in our MVCNNR.

3) Residual Network: In deep learning, as the number of layers increases continuously, there exists a possible phenomenon: gradient vanishing, which implies that the gradients tend to zeros and thus the weights could not be updated correspondingly. This problem makes training procedure extremely difficult. To deal with this limitation, in residual network [30], many convolutional layers are divided into several convolutional blocks, which include plain and residual convolutional blocks. In each residual block, its input or output of first convolutional layer is connected to the output of the current block via shortcut addition operation, which also builds a information pathway from low-levels to relatively high-levels, similarly in U-Net. Such a trick speeds up training procedure efficiently and improves the performance for classification and regression further.

B. Enhanced Multi-view Convolutional Neural Network for Regression

In this sub-section, we integrate and extend the aforementioned three networks to construct an enhanced multi-view convolutional neural network for regression to deal with our seismic velocity inversion task, as shown in Figs. 5 and 6.

At first, as shown in the beginning of Fig. 5, each shot gather in a multi-shot record is simulated as a view in MVCNN to characterize the possible complementary and redundant information. Further, our velocity inversion is regarded as pixel level regression, where a 2D convolutional layer is applied to

replace the softmax operation, to fit 2D velocity profiles, as shown in the end of Fig. 5.

Before the view pooling of our MVCNNR, each sub-CNN is designed as an encoder independently. This encoder consists of five residual blocks whose number of channels is doubled gradually from 64 to 1024. Each residual block includes three convolutional layers and a maximal pooling layer, where the first three layers do not change the size of feature maps (including height, width and channels) and the last pooling operation indeed reduces the height and width of feature maps by half. For each residual block, its shortcut addition connection links the outputs of the first and second layers. In Fig. 5, we show the height and width of feature maps and the number of channels, and convolutional layer and its auxiliary operations in the lower right corner.

The view pooling layer fuses the feature information from all view feature maps. After that, there is a decoder cascading four residual blocks, as shown in Fig. 5. Each block is comprised of a de-convolutional layer and three convolutional layers. Here the de-convolutional layer enlarges the height and width doubly and reduces the number of channels by half. Three subsequent convolutional layers remain the size of feature maps.

In Fig. 6, we show the skipping concatenation connection from one view of encoder to decoder symmetrically. Essentially, all views have this connection. It is noted that this connection needs the equal height and width only. Finally, we refer to this extended architecture as an enhanced multi-view

convolutional neural network for regression or MVCNNR simply.

C. Training Strategy

Let our training set of size N be $\{(\mathcal{X}_i, \mathbf{V}_i), i = 1, \dots, N\}$, where \mathcal{X}_i is a third-order tensor to describe the i -th multi-shot record, and \mathbf{V}_i is a matrix to indicate the i -th velocity profile. The seismic velocity inversion can be formulated as

$$\mathbf{V} = f^{-1}(\mathcal{X}) \quad (1)$$

where f^{-1} denotes our inversion operator from seismic multi-shot record to 2D velocity profile. In this case, the loss function is defined as a regularized mean square error, i.e.,

$$J(\mathbf{w}) = \frac{1}{N} \sum_{i=1}^N \|\hat{\mathbf{V}}_i - f^{-1}(\mathcal{X}_i)\|_F^2 + \frac{\lambda}{2N} \|\mathbf{w}\|_2^2 \quad (2)$$

where \mathbf{w} indicates all weights and bias, $\lambda > 0$ is the regularization constant $\|\cdot\|_F$ represents the Frobenius norm of matrix, and $\|\cdot\|_2$ is the 2-norm of vector. In our experiment, this loss function is minimized by ADAM algorithm [32].

III. EXPERIMENTS

In order to verify the effectiveness of our proposed inversion method, we will conduct a comprehensive experimental comparison with two existing techniques (FCNVMB [11] and VelocityGAN [9]) on four datasets, in this section.

A. Four Datasets and Two Data-driven Inversion Methods

We collected four datasets: Layered, Faulted, SaltBody and SaltDome, which are originally designed and tested respectively in [9]–[12], and whose velocity models are available via drawing emails to their authors.

We reshape these velocity models into 3000m in width and 2000m in depth. Each velocity profile has 5-12 layers, whose interfaces are slightly curved. The velocity values vary from 2000 to 4000m/s and increase in depth. This is the first type of Layered models. The other velocity models are all based on this Layered model. When a few layers are broken and then slipped, which can induces normal or reverse fault, and thus is referred to as Faulted model. A salt body of any shape is embedded at any position with a velocity of 4500m/s, which is the third type of SaltBody models. In the forth type of SaltDome models, a salt dome with a velocity of 4500m/s like a mushroom is involved, which simulates a magmatic exhalation result. Via 10m interval, each model is grided in a matrix of 201×301 .

Subsequently, we solve acoustic wave equation to synthesize their five-shot seismic records using finite difference method, where 25 Hz Ricker wavelet is applied and 1ms time sampling interval is used. Five shots are located at 0, 750, 1500, 2250 and 3000m, respectively. The 301 receivers are uniformly inserted at 301 positions from 0 to 3000m with 10m offset. In this case, each 5-shot record is formulated as third tensor of $2001 \times 301 \times 5$, as shown in Table I. In Figs. 7 to 10, we illustrate a velocity model from each dataset and its five-shot seismic record. In addition, to compare all three inversion

TABLE I
TWO DATASETS USED IN OUR EXPERIMENTS.

Dataset	#Train	#Test	Seismic record	Velocity profile
Layered	1800	200	$2001 \times 301 \times 5$	201×301
Faulted	1800	200	$2001 \times 301 \times 5$	201×301
SaltBody	1600	100	$2001 \times 301 \times 5$	201×301
SaltDome	1900	100	$2001 \times 301 \times 5$	201×301

methods more fairly, we down-sample the time sampling points from 2001 to 401, just as in FCNVMB [11]. Moreover, we separate these datasets into training and testing subsets, as shown in Table I.

We compare our MVCNNR with FCNVMB [11] and VelocityGAN [9], where the former is downloaded at ¹, while the latter is recoded by us. For three methods, the batch size is set to 10, the number of epochs 100, and the learning rate 0.001. For our MVCNN, the regularization constant is equal to 0.001. Further, we use three performance evaluation metrics: root mean squared error (RMSE), peak signal noise ratio (PSNR) and structural similarity (SSIM), where the first metric is mainly used for multi-output regression [33], and the last two for image equality assessment [34], [35].

B. Experimental Results

After training three inversion techniques via training instances, we use testing instances to calculate three metrics, where the mean and standard derivation are reported in Table II. From Table II, we observe that: (a) on three metrics, our MVCNNR performs the best among three compared inversion methods; (b) such three techniques are sorted as MVCNNR, FCNVMB and VelocityGAN in the descending order.

In Figs. 11 to 14, we illustrate eight predicted velocity profiles for each type of velocity models. Visually, we find out that our MVCNNR (the last column) achieve the most clear velocity profiles, which are very close to the ground true velocity models (the first column). Additionally, FCNVMB (the second column) is also better than VelocityGAN (the third column).

C. Ablation Experiment

Our MVCNNR is extended from MVCNN for classification at first, and then is embedded in concatenation connection in U-net and addition connection in residual network. It is argued whether the last two connections work. Therefore we conduct an ablation experiment on SaltBody dataset here.

Our baseline architecture is a MVCNN for regression to deal with two-dimensional velocity profiles. The basic performance is listed in Table III, denoted by Baseline.

When the concatenation (Cat) and addition (Add) connections are added, respectively and independently, the performance has been improved greatly, as shown in Table III. It is attractive that the addition operation could obtain more improvement than the concatenation one.

¹<https://github.com/YangFangShu/FCNVMB>

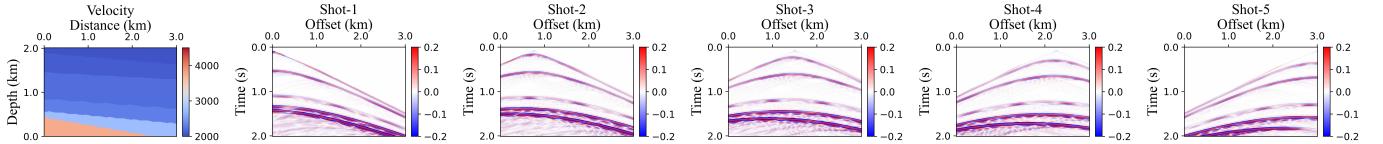


Fig. 7. Layered velocity model and its 5-shot seismic record.

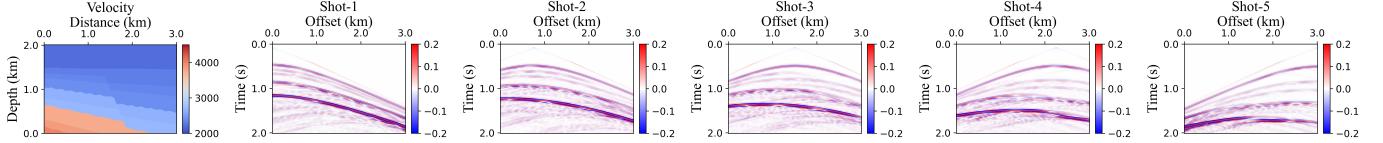


Fig. 8. Faulted velocity model and its 5-shot seismic record.

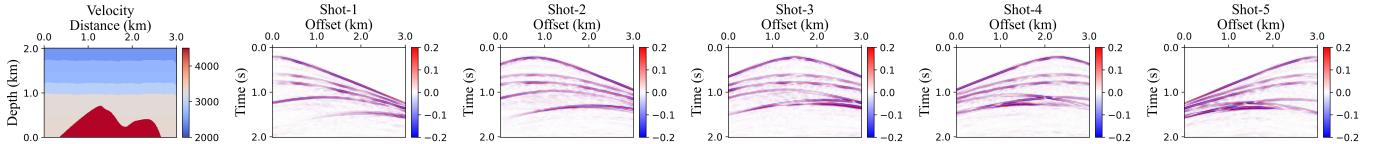


Fig. 9. SaltBody velocity model and its 5-shot seismic record.

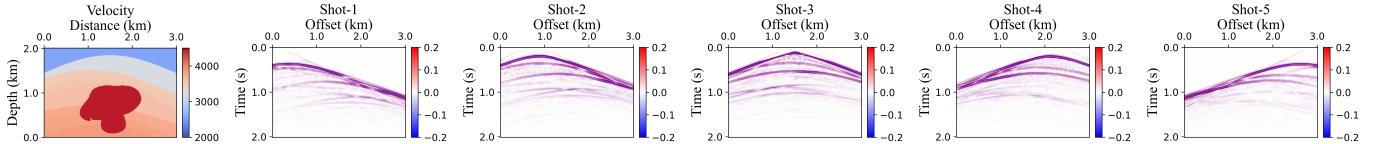


Fig. 10. SaltDome velocity model and its 5-shot seismic record.

TABLE II
THE QUANTITATIVE RESULTS OF THREE DIFFERENT NETWORKS.

Method	RMSE(\downarrow)	PSNR(\uparrow)	SSIM(\uparrow)
Layered			
FCNVMB	110.69 \pm 43.46	32.72 \pm 2.93	0.9625 \pm 0.0118
VelocityGAN	170.84 \pm 45.75	28.69 \pm 2.15	0.9605 \pm 0.0122
MVCNNR	102.81 \pm 28.94	33.13 \pm 2.28	0.9687 \pm 0.0104
Faulted			
FCNVMB	123.12 \pm 42.49	31.66 \pm 2.54	0.9541 \pm 0.0123
VelocityGAN	232.95 \pm 56.14	25.95 \pm 1.99	0.9119 \pm 0.0137
MVCNNR	107.51 \pm 28.12	32.72 \pm 2.19	0.9639 \pm 0.0101
SaltBody			
FCNVMB	214.95 \pm 64.88	26.74 \pm 2.27	0.9153 \pm 0.0156
VelocityGAN	312.90 \pm 78.19	23.41 \pm 2.09	0.9144 \pm 0.0137
MVCNNR	167.48 \pm 43.45	28.86 \pm 2.15	0.9330 \pm 0.0137
Saltdome			
FCNVMB	181.95 \pm 52.09	28.13 \pm 2.02	0.9324 \pm 0.0132
VelocityGAN	260.99 \pm 94.05	25.13 \pm 2.48	0.9187 \pm 0.0143
MVCNNR	164.71 \pm 32.16	28.87 \pm 1.50	0.9382 \pm 0.0114

At the last row of Table III, we add both concatenation and addition operations simultaneously, the performance has been enhanced further. This ablation experiment demonstrates that two embedded connections could improve the performance for seismic velocity inversion effectively.

TABLE III
ABALATION EXPERIMENTS FOR TWO OPERATIONS ON SALTBODY.

Cat	Add	RMSE	PSNR	SSIM
Baseline		325.97 \pm 88.20	23.14 \pm 2.50	0.8900 \pm 0.0282
\checkmark		246.54 \pm 75.95	25.63 \pm 2.66	0.9050 \pm 0.0257
	\checkmark	203.25 \pm 57.33	27.24 \pm 2.41	0.9182 \pm 0.0211
\checkmark	\checkmark	167.48 \pm 43.45	28.86 \pm 2.15	0.9330 \pm 0.0137

IV. CONCLUSIONS

In this paper, we propose an enhanced multi-view convolutional neural network for seismic velocity inversion, which integrating and generalizing multi-view convolutional neural network for classification, U-Net for image segmentation and residual network for classification. This method characterizes complementary information and reduces redundant information among different shot gathers of a seismic record. On four benchmark datasets (Layered, Faulted, SaltBody and Salt-Dome), we illustrate the effectiveness of our proposed technique, via comparing it with two existing methods (FCNVMB and VelocityGAN), according three evaluation metrics (i.e., root mean squared error, peak signal noise ratio and structural similarity). The ablation experiment shows how concatenation and addition connections work in our MVCNNR further. In short, our work provides a novel effective deep learning

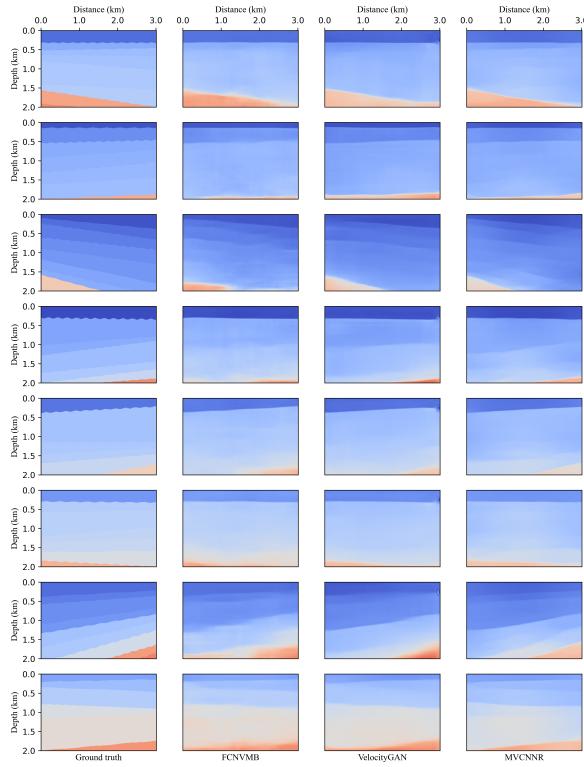


Fig. 11. Eight predicted velocity models from three methods on Layered.

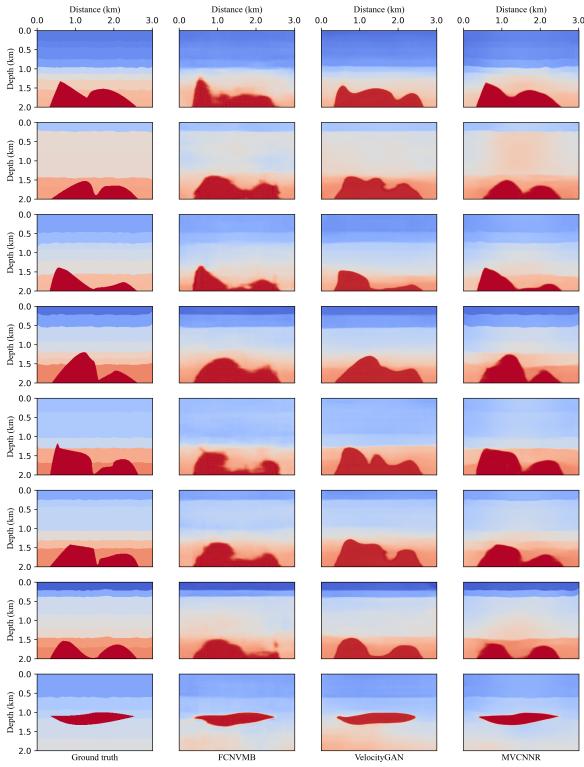


Fig. 13. Eight predicted velocity models from three methods on SaltBody.

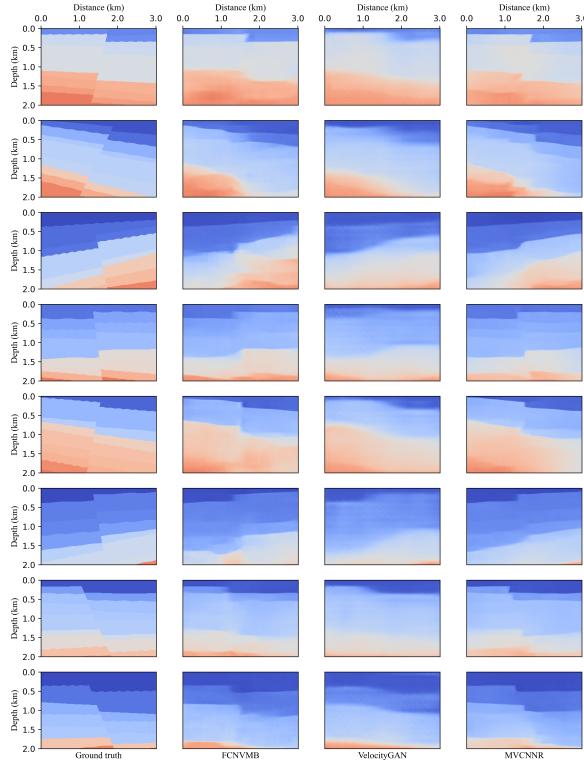


Fig. 12. Eight predicted velocity models from three methods on Faulted.

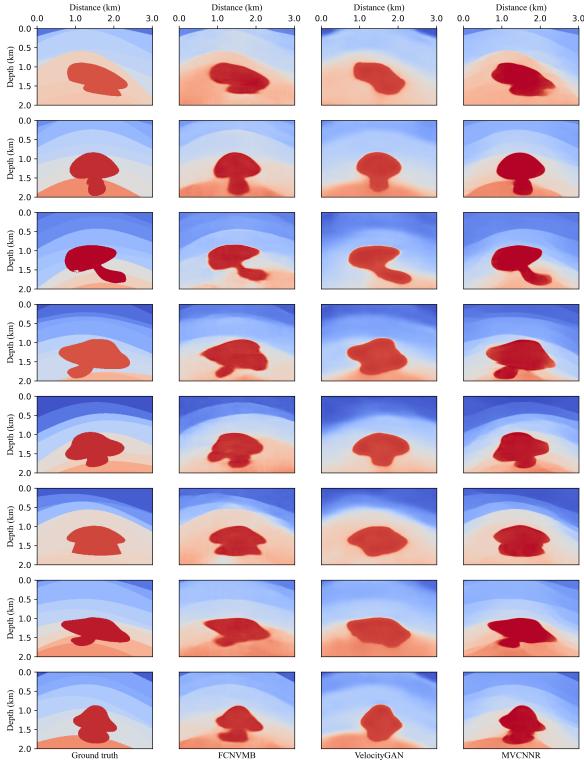


Fig. 14. Eight predicted velocity models from three methods on SaltDome.

architecture for seismic inversion. In the future work, we will validate our proposed technique with more synthetic data and apply our method to real-world data via transfer learning and domain adaptation technique [36]–[38].

ACKNOWLEDGMENT

This work is supported by the National Science Foundation of China (NSFC) under Grant 62076134 and 62173186, and the Open Fund of National Engineering Research Center of Offshore Oil and Gas Exploration. We thank two reviewers for their constructive comments to help us for polishing our paper further.

REFERENCES

- [1] R. L. Sengbush, *Seismic Exploration Methods*. Dordrecht, Netherlands: Springer, 2012.
- [2] H. N. Alsadi, *Seismic Hydrocarbon Exploration: 2D and 3D Techniques*. Cham, Germany: Springer, 2017.
- [3] J. Virieux and S. Operto, “An overview of full-waveform inversion in exploration geophysics,” *Geophys.*, vol. 74, no. 6, pp. WCC127–152, Nov./Dec. 2009.
- [4] G. Yao, D. Wu, and S.-X. Wang, “A review on reflection-waveform inversion,” *Pet. Sci.*, vol. 17, no. 2, pp. 334–351, Mar. 2020.
- [5] A. Adler, M. Araya-Polo, and T. Poggio, “Deep learning for seismic inverse problems: Toward the acceleration of geophysical analysis workflows,” *IEEE Signal Process. Mag.*, vol. 38, no. 2, pp. 89–119, Mar. 2021.
- [6] Y. Lin, J. Theiler, and B. Wohlberg, “Physics-guided data-driven seismic inversion: Recent progress and future opportunities in full-waveform inversion,” *IEEE Signal Process. Mag.*, vol. 40, no. 1, pp. 115–133, Jan. 2023.
- [7] S. M. Mousavi, G. C. Beroza, T. Mukerji, and M. Rasht-Beheshti, “Applications of deep neural networks in exploration seismology: A technical survey,” *Geophys.*, vol. 89, no. 1, pp. WA95–115, Jan. 2024.
- [8] M. Sugiyama, *Introduction to Statistical Machine Learning*. Singapore: Elsevier Press, 2016.
- [9] Z. Zhang and Y. Lin, “Data-driven seismic waveform inversion: A study on the robustness and generalization,” *IEEE Trans. Geosc Remote Sens.*, vol. 58, no. 10, pp. 6900–6913, Mar. 2020.
- [10] S. Li, B. Liu, Y. Ren, Y. Chen, S. Yang, Y. Wang, and P. Jiang, “Deep learning inversion of seismic data,” *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 2135–2149, Mar. 2020.
- [11] F. Yang and J. Ma, “Deep-learning inversion: A next-generation seismic velocity model building method,” *Geophys.*, vol. 84, no. 4, pp. R583–R599, Jun. 2019.
- [12] G. Liu, L. Zhang, Q. Wang, and J. Xu, “Data-driven seismic prestack velocity inversion via combining residual network with convolutional autoencoder,” *J. Appl. Geophys.*, vol. 207, pp. Article–104 846, Dec. 2022.
- [13] M. Araya-Polo, J. Jennings, A. Adler, and T. Dahlke, “Deep-learning tomography,” *The Leading Edge*, vol. 37, no. 1, pp. 58–66, Jan. 2018.
- [14] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, “Backpropagation applied to handwritten zip code recognition,” *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989.
- [15] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge MA, USA: MIT Press, 2016.
- [16] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, “A survey of convolutional neural networks: Analysis, applications, and prospects,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 12, pp. 6999–7019, Dec. 2022.
- [17] B. Mao, L.-G. Han, Q. Feng, and Y.-C. Yin, “Subsurface velocity inversion from deep learning-based data assimilation,” *J. Appl. Geophys.*, vol. 167, pp. 172–179, Aug. 2019.
- [18] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, “Image segmentation using deep learning: A survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3523–3542, Jul. 2022.
- [19] W. Wang, F. Yang, and J. Ma, “Velocity model building with a modified fully convolutional network,” in *Proc. SEG Tech Program Extended Abstract*, 2018, pp. 2086–2090.
- [20] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in *Proc. MICCAI*, 2015, pp. 234–241.
- [21] N. Siddique, S. Paheding, C. P. Elkin, and V. Devabhaktuni, “U-Net and its variants for medical image segmentation: A review of theory and applications,” *IEEE Access*, vol. 9, pp. 82 031–82 057, Jun. 2021.
- [22] Y. Wu and Y. Lin, “Inversionnet: an efficienct and accurate data-driven full waveform inversion,” *IEEE Trans. Comput. Imaging*, vol. 6, pp. 419–433, Jun. 2020.
- [23] S. Feng, Y. Lin, and B. Wohlberg, “Multiscale data-driven seismic full waveform inversion with field data study,” *IEEE Trans. Geosci Remote Sens.*, vol. 60, pp. Article–4 506 114, 2022.
- [24] Q. Zeng, S. Feng, B. Wohlberg, and Y. Lin, “Inversionnet3D: efficient and scalable learning for 3D full waveform inversion,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. Article–4 506 816, 2022.
- [25] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Proc. NIPS*, 2014, pp. 2672 – 2680.
- [26] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, “A review on generative adversarial networks: Algorithms, theory, and applications,” *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 4, pp. 3313–3332, Apr. 2023.
- [27] X. Yan, S. Hu, Y. Mao, Y. Ye, and H. Yu, “Deep multi-view learning methods: A review,” *Neurocomputing*, vol. 448, pp. 106–129, Aug. 2021.
- [28] S. Qi, X. Ning, GuoWeiYang, L. Zhang, P. Long, W. Cai, and W. Li, “Review of multi-view 3D object recognition methods based on deep learning,” *Displays*, vol. 69, pp. Article–102 053, 2021.
- [29] H. Su, S. Maji, E. Kalogerakis, and E. G. Learned-Miller, “Multi-view convolutional neural networks for 3D shape recognition,” in *Proc. ICCV*, 2015, pp. 945–953.
- [30] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. CVPR*, 2016, pp. 770–778.
- [31] G. Du, X. Gao, J. Liang, X. Chen, and Y. Zhan, “Medical image segmentation based on U-net: A review,” *J. Imaging Sci. Technol.*, vol. 64, no. 2, pp. Article–020 508, Mar./Apr. 2020.
- [32] D. P. Kingma and J. L. Ba, “ADAM: A method for stochastic optimization,” in *Proc. ICLR*, 2015, pp. 1–15.
- [33] H. Borchani, G. Varando, C. Bielza, and P. Larrañaga, “A survey on multi-output regression,” *WIREs Data Min. Knowl. Discovery*, vol. 5, no. 5, pp. 216–233, Sep./Oct. 2015.
- [34] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [35] D. I. M. Setiadi, “PNSR vs SSIM: imperceptibility quality assessment for image steganography,” *Multimedia Tool Appl.*, vol. 80, no. 28–29, pp. 8423–8444, Nov. 2021.
- [36] S. Feng, H. Wang, C. Deng, Y. Feng, Y. Liu, M. Zhu, P. Jin, Y. Chen, and Y. Lin, “EFWI: Multiparameter benchmark datasets for elastic fullwaveform inversion of geophysical properties,” in *Proc. NeurIPS*, 2023, pp. 1–13.
- [37] W. M. Kouw and M. Loog, “A review of domain adaptation without target labels,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 3, pp. 766–785, Mar. 2021.
- [38] L. Zhang and X. Gao, “Transfer adaptation learning: A decade survey,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 35, no. 1, pp. 23–44, Jan. 2024.