MDA GAN: Adversarial-Learning-Based 3-D Seismic Data Interpolation and Reconstruction for Complex Missing

Yimin Dou, Student Member, IEEE, Kewen Li, Hongjie Duan, Timing Li, Lin Dong, Member, IEEE, and Zongchao Huang

Abstract—The interpolation and reconstruction of missing traces are crucial steps in seismic data processing; moreover, it is also a highly ill-posed problem, especially for complex cases such as high-ratio random discrete missing, continuous missing, and missing in fault-rich or salt body surveys. These complex cases are rarely mentioned in current works. To cope with complex missing cases, we propose multidimensional adversarial generative adversarial network (MDA GAN), a novel 3-D GAN framework. It keeps the anisotropy and spatial continuity of the data after 3-D complex missing reconstruction using three discriminators. The feature splicing module is designed and embedded in the generator to retain more information of the input data. The tanh cross entropy (TCE) loss is derived, which provides the generator with the optimal reconstruction gradient to make the generated data smoother and continuous. We experimentally verified the effectiveness of the individual components of the study and then tested the method on multiple publicly available data. The method achieves reasonable reconstructions for up to 95% of random discrete missing and 100 traces of continuous missing. In fault and salt body enriched surveys, MDA GAN still yields promising results for complex cases. Experimentally, it has been demonstrated that our method achieves better performance than other methods in both simple and complex cases. Moreover, our network does not require training weights for each survey, the same weights it uses are applied to multiple surveys, significantly reducing time and computational costs, and we make the model publicly available on https://github.com/douyimin/MDA_GAN.

Index Terms—Adversarial learning, generative adversarial network (GAN), seismic complex missing, seismic data interpolation, seismic data reconstruction.

I. INTRODUCTION

COMPLETE seismic data are often difficult to acquire due to various constraints such as economic, physical, and other factors. Reconstructing missing seismic data is a critical and challenging task. There are currently two major types of methods for interpolation and reconstruction of seismic data: theory-driven and data-driven.

There are four theory-driven methods. The first is prediction filter-based methods [1], [2], [3], [4] that extend interpolation to the frequency–space (f–x) domain, which exploit the predictability of linear events in the (f–x) domain. Such methods must use a local windowing strategy when handling nonlinear events. The choice of window affects the performance of interpolation and reconstruction, and the choice of the optimal window is currently unrealistic [5]. The second is the wave equation-based approach [6], [7], which requires subsurface velocity as a priori; however, it is difficult to obtain accurate velocity models in practical engineering, which limits its extension. The third is the methods based on sparse constraints [8], [9], [10], [11], which assume that the seismic data are linear or quasi-linear, and even though it can handle nonlinear data through parameter tuning, it is impractical to obtain optimal parameters, which affects its performance [12]. Finally, there are low-order constraint methods [13], [14], [15], [16], which are based on compression sensing; however, most of them are only applicable to the missing case of random discrete.

Data-driven employing machine learning or deep learning, there are generally two types of methods, autoencoder (AE) and generative adversarial networks (GANs). AE-based methods include using AE [5], CAE [17], UNet [18], ResNet [19], and so on. AE methods use encoders to extract hidden variables from missing data, and the complete data are used to supervise the seismic data generated by the decoder, which leads to data being learned point-to-point and also results in less comprehensive global information acquisition, with weak global information and strong local information. Due to the lack of global information, the performance is weak for large-scale missing, so most such methods are suitable for handling random discrete missing [20], [21], [22], [23], [24], [25]. Yu and Wu [20] and Li et al. [21] introduced the attention mechanism to solve the weak global information problem of the AE method and achieved promising results in the case of continuous missing. He et al. [22] used a multistage UNet to step through the interpolation of consecutive missing.

Additional to this are GAN-based methods, but they are less studied. GAN adds a discriminator to AE (generator), which
introduces regional or global information, so it should have better performance, especially for continuous and large-scale deficiencies [26]. Oliveira et al. [27] achieved interpolation and reconstruction of Netherlands Offshore F3 seismic data via cGANs. These methods all employ $L_1$ as the reconstruction loss of the generator, but GAN generally uses tanh as the activation function of the output layer to ensure that the value domain of the generator is at $[-1, 1]$, and the gradient of $L_1$ under this activation function is nonpositively correlated with the loss [see (8)–(10)], which may lead to point-to-point learning that cannot be optimized. Since natural images are bounded by the three RGB channels to each other and allow for diversity inpainting, $L_1$ is possible as the reconstruction loss of natural image inpainting [28], [29], [30], [31]. However, seismic data only have single channel, so these GAN-based interpolation results for seismic data have some obvious splicing traces; the interpolation and reconstruction of seismic data require restoring the original seismic data as much as possible, and $L_1$ causes the results to deviate from the optimal solution. Wei et al. [32] changed the adversarial loss to Wasserstein loss based on cGAN and achieved the 2-D interpolation of up to 35 consecutive seismic traces missing. Kaur et al. [33], [34] used GAN and CycleGAN to interpolate 2-D synthetic seismic data. The existing methods are limited to the application of GAN in the field of seismic interpolation, so there is no significant difference in the performance of the current GAN-based and the AE-based methods. Moreover, most of the existing AE and GAN-based methods focus only on repairing 2-D slices, but reconstruction of 2-D slices one by one leads to discontinuities in brutal stacked [12] section when there are complex cases such as large scale or continuous missing in shot gathers. Therefore, it is crucial to research 3-D interpolation and reconstruction approaches.

We suggest that introducing GAN into seismic interpolation requires addressing two major challenges. The first is how to design a reasonable GAN benchmark framework for 3-D seismic data interpolation and reconstruction to prevent mode collapse during training while ensuring that the anisotropy and spatial continuity of 3-D seismic data can be preserved even in the case of complex missingness. Second, it can ensure that the generator performs accurate point-to-point learning without distortion of each pixel even when the discriminator introduces global information and uses the tanh activation function.

There is no GAN-based method for 3-D seismic interpolation [25], [27], [32], [33], [34]. The interpolation of 3-D seismic data needs to ensure the spatial continuity and reliability of the seismic data in three directions (timeline, inline, and crossline) while maintaining the anisotropy of the seismic data, especially when we need to handle complex problems such as large-scale discrete missing or continuous missing. Moreover, compared with the reconstruction of 2-D data, the computational resources and parameters required for 3-D data have increased significantly [35], [36], especially the alternate training mode of generator and discriminator, which puts harsh requirements on the hardware conditions. The increase of parameters is more possible to cause the mode collapse of GAN [37], [38], so it is challenging to design a GAN-based interpolation and reconstruction framework for 3-D seismic data.

We propose a multidimensional adversarial GAN (MDA GAN) that uses three discriminators to extract regional or global features for 3-D volume, timeline, inline, and crossline of seismic data (inline and crossline share one discriminator and apply to both prestack and poststack data) to maintain anisotropy and spatial continuity of the reconstructed seismic data.

In response to the demanding hardware requirements imposed by 3-D data and the mode collapse problems that may be caused, a 3-D data generator for GAN is designed, which propagates the features using high and low resolution in parallel to prevent information loss, so that excessive width is not required to retain image information, and low (large scale) and high resolution interact during propagation, increasing the receptive field and fully extracting global information, so the network does not need to be deeper [39]. Meanwhile, we propose the feature splicing module (FSM) module and embed this generator to ensure the integrity of the unmissed parts during the model inference. The generator reduces the parameters and the load on the hardware while maintaining a high-quality reconstruction performance.

To enable accurate point-to-point learning of seismic data in the framework of GAN, we analyzed the gradients of reconstruction loss that are commonly used nowadays and thus derived the tanh cross entropy (TCE) loss that can perform accurate pixel-level learning under the tanh activation function for seismic data reconstruction by MDA GAN.

In general, our contributions include the following three points. First, a novel GAN framework, MDA GAN, is proposed to cope with more complex 3-D seismic missing cases to ensure the anisotropy as well as spatial continuity of the reconstructed 3-D data. Second, to cope with the burden on the hardware caused by the parameter proliferation brought by 3-D GAN, we elaborated the generator and proposed and embedded the FSM to accomplish end-to-end data generation and splicing. Finally, a TCE reconstruction loss is proposed so that the training process can provide smoother gradients, enable accurate point-to-point learning, and ensure that the reconstructed data are distortion-free.

II. APPROACH

In this section, we first introduce the baseline framework of MDA GAN, then illustrate the structure of FSM and how it is embedded and functions in the generator, and finally discuss the seismic voxel (pixel)-level reconstruction loss, TCE loss, and analyze its training gradient to illustrate the advancement of the method.

A. Baseline Framework

Our approach is based on GAN, and in the Appendix, we add a basic explanation for GAN and the current underlying framework for GAN-based image inpainting in computer vision (CV).

The major disadvantage of current 2-D seismic interpolation is that reconstruction of 2-D slices one by one leads to
discontinuities in brutal stacked section [12] when there are complex cases such as large scale or continuous missing in shot gathers. To solve this problem, we extend the seismic interpolation and reconstruction to three dimensions.

The generator is a 3-D CNN structure, and the input and output are 3-D data of the same size. This article describes the structure of the generator and its components detailed in II-B.

The 3-D seismic data have obvious anisotropy, where the seismic reflection axis is approximately perpendicular to the inline and crossline slices, and a clear seismic layered texture can be observed in these two directions, while the timeline direction shows irregular texture. Therefore, we use two 2-D discriminators $D_{\theta_{I}}$ and $D_{\theta_{m}}$, inline and crossline share discriminator $D_{\theta_{I}}$, and timeline direction uses the $D_{\theta_{m}}$ discriminator. The 2-D discriminators are employed for each directional slice to maintain the anisotropy of the restored seismic data, and we also use a 3-D discriminator $D_{\theta_{G}}$ to ensure the continuity of the results in 3-D space.

The framework consists of a 3-D generator, two 2-D discriminators, and a 3-D discriminator. Accordingly, we use three adversarial losses $L_{\theta_{G}}$, $L_{\theta_{I}}$, and $L_{\theta_{m}}$. In combination with the reconstruction (IV-B), the loss of the supervised generator can be expressed as follows:

$$L_{\theta_{G}} = L_{\theta_{G}} + L_{\theta_{I}} + L_{\theta_{m}} + \lambda L_{\text{rec}}$$  

(1)

where $L_{\text{rec}}$ is the reconstruction loss, which we will describe in detail in II-D, and $\lambda$ is a scaling factor to adjust the reconstruction loss to the same order of magnitude as the adversarial loss. Each adversarial loss is expressed as follows:

$$L_{\text{adv}} = \log \left(1 - D_{\theta_{G}} \left( G_{\theta_{G}} \left( I_{m} \right) \right) \right)$$  

(2)

where $I_{m}$ is the missing input data. This loss tries to make the generator fool the discriminator so that the generated hidden variables are approximated to the real data.

Each discriminator determines the authenticity of the generated tensor or matrix, using the cross-entropy loss for supervision [see (3)]

$$L_{D} = \frac{1}{2} \log \left(1 - D_{\theta_{D}} \left( I_{g} \right) \right) + \frac{1}{2} \log \left(- D_{\theta_{D}} \left( G_{\theta_{G}} \left( I_{m} \right) \right) \right).$$  

(3)

For reconstructing high-quality seismic data, our approach employs adversarial learning in multiple dimensions, hence the term MDA GAN. The overall framework of MDA GAN is shown in Fig. 1(a).

B. Generator

1) Backbone Network: Traditional generators normally take the form of concatenating high-resolution features with low-resolution features, i.e., encoder–decoder structures, and related models are available for various vision tasks [28], [29], [30], [31], [41]. However, such methods downsample the features several times and return the original resolution via the decoder, which causes the loss of information, while the process of restoring the resolution requires plenty of parameters and eats up many video memory resources. The new model design idea of parallel propagation of high resolution with low resolution was proposed by Sun et al. [42] and Wang et al. [43], and our previous work demonstrated that this structured network can achieve higher performance with few hardware resources [39]. We designed the generator based on this structure, using fewer computational resources to ensure the generation of high-quality data.

Resblock [40] is selected as the base block of the generator. The network is downsampled only twice, and then, a low-resolution branch is added for parallel propagation, which is structured, as shown in Fig. 1(b). Because the network keeps the high-resolution features, we do not need an excessive width and depth to ensure that the network recovers from the low resolution, so we use fewer parameters and save much bandwidth.

2) Feature Splicing Module: To adequately preserve the unmissing information of the input data $I_{m}$, many works perform splicing of $I_{m}$ with $I_{g}$ (output) via mask, and the
splicing process is expressed as follows:

$$I_g \approx I'_g = (1 - M) \times I_g + M \times I_m$$  \hspace{1cm} (4)$$

where $I_g$ is the ground truth and $I'_g$ is the spliced data; $M$ is mask, which is a matrix of the same size as the inputs, with 0 indicating missing data and the opposite with 1. However, for seismic data, it is difficult to obtain masks by a single step (threshold method) and some manual intervention is needed to splice or introduce other algorithms. Not splicing will cause the resulting $(1 - M) \times I_g$ to differ from the original data $(1 - M) \times I_g$, as shown in Fig. 9(a-5)–(c-5). We want to find a solution to replace the role of $M$ and to form end-to-end training and inference.

For this purpose, we presented FSM. Our approach is to splice the low-level features of the network (representing the original nonmissing part) with the high-level features (representing the generated part) so that the network automatically completes this process and then reconstructs the spliced features to output the result. The existing work such as UNet [30], which concatenates the high- and bottom-level features and then fuses them using convolution, does not express the splicing process explicitly, and in the fusion process, the missing and unmissing of the bottom-level features have the same weight; this causes the generated unmissing part $(1 - M) \times I_g$ to be blurred and differ from the original data, making it necessary to manually splice the generated with the original data.

The module can be represented by the following equations:

$$F^{\text{branch}} = F^\text{cat} (F^l, F^h)$$

$$\mathcal{W}_m = \sigma(F^\text{conv} (F^{\text{branch}}, 3, 1, 2C_1, C_1), 1, 1, C_1, 2C_1))$$  \hspace{1cm} (5)$$

$$F^{\text{expl}} = F^\text{conv} (\mathcal{W}_m \odot F^{\text{branch}}, 1, 1, 2C_1, C_1).$$  \hspace{1cm} (6)$$

$$\mathcal{W}_m \approx \mathcal{M} \left(\frac{\text{conv}^{l} - x_l}{\text{cout}}\right),$$  \hspace{1cm} (7)$$

In (5), the features are concatenated with the high-level feature $F^h$ and the low-level feature $F^l$ to obtain $F^{\text{branch}}$, and the channel after concatenation is changed from $C_1$ to $2 \times C_1$. $F^l$ uses the features after the first convolution and $F^h$ uses the features before the last residual block, so they both keep the original feature resolution. $F^l$ has not been downsampled and convolved multiple times, so it retains all the information of the original data, and $F^h$ has been filled with the missing parts after the generator network.

We want to express the splicing process of the features explicitly so that the network automatically expresses (4), and we compress and weight $F^{\text{branch}}$ by a convolution, then recover it to the original channel, and restrict its value domain to $[0, 1]$ by the sigmoid ($\sigma(x_i) = 1/(1 + \exp(-x_i))$) activation function to obtain $\mathcal{W}_m$. The above process is represented by (6).

After (7), the FSM can be updated via standard backpropagation, and our experiments show that the module is able to converge $\mathcal{W}_m$ so that it generates a feature mask that resembles $M$ and applies it to the splicing of features. The entire process can be represented in Fig. 2.

We visualized $\mathcal{W}_m$, which has channel $2C_1$ and has two components, $\mathcal{W}_l$ ($C_1$ channels) and $\mathcal{W}_h$ ($C_1$ channels), with $\mathcal{W}_l$ corresponding to the $F_l$ part of the $F^{\text{branch}}$ and $\mathcal{W}_h$ corresponding to the $F_h$ part of the $F^{\text{branch}}$, as shown in Fig. 9(a-5)–(c-5). We want to find $\mathcal{W}_m$ indicating missing data and the opposite with 1. However, for seismic data, it is difficult to obtain masks by a single step (threshold method) and some manual intervention is needed to splice or introduce other algorithms. Not splicing will cause the resulting $(1 - M) \times I_g$ to differ from the original data $(1 - M) \times I_g$, as shown in Fig. 9(a-5)–(c-5). We want to find $\mathcal{W}_m$ by (7). After (7), the FSM can be updated via standard backpropagation, and our experiments show that the module is able to converge $\mathcal{W}_m$ so that it generates a feature mask that resembles $M$ and applies it to the splicing of features. The entire process can be represented in Fig. 2.

C. Discriminator

The discriminator network uses the structure of a standard encoder. The discriminator consists of five convolutional layers, and each layer employs the LeakyReLU [37] activation function and also incorporates the spectral normalization [44] to stabilize the training of the GAN.

D. TCE Loss

Most of the current reconstruction losses are based on $L_1$ and $L_2$ [45], [46], [47], [48], where $L_1$ provides stable gradients and $L_2$ has the more stable solution, but both loss functions have the common disadvantage of unsmoothed gradients in GANs [see (10) and (13)]. Generator networks generally use tanh as the final activation layer to keep the network’s value domain at $[-1, 1]$ while ensuring training stability [37]. Next, we derive the gradient expressions for $L_1$ and $L_2$ under the tanh activation function.
First, we derive the gradient of $L_1$ loss backpropagation to backbone

$$L_{L_1} = \frac{1}{N_w \times N_h \times N_d} \sum_{i} \sum_{j} \sum_{k} |y_{i,j,k} - \hat{y}_{i,j,k}|$$

$$\hat{y}_t = \text{Tanh}(\hat{x}) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}, \quad \frac{d\hat{y}_t}{d\hat{x}} = 1 - \hat{y}^2$$

$$\frac{\partial L_{L_1}(y, \hat{y})}{\partial \hat{x}_t} = \frac{\partial L_{L_1}(y, \hat{y})}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial \hat{x}_t} = \begin{cases} 1 - \hat{y}_t^2, & \text{if } y_t - \hat{y}_t \geq 0 \\ \hat{y}_t^2 - 1, & \text{if } y_t - \hat{y}_t < 0. \end{cases}$$

The most reasonable relationship between $\partial L_{L_1}/\partial \hat{x}_t$ and $\hat{y}_t$ should be the one that triggers the minimum gradient (minimum absolute value) when $\hat{y}_t = y_t$. As $\hat{y}_t$ and $y_t$ become farther apart, the greater the gradient they cause should be, and this increasing relationship should be linear, i.e., smooth, its slope should not change with $\hat{y}_t$. We define this optimal assumption as follows:

$$\frac{\partial L(y, \hat{y})}{\partial \hat{x}_t} = y_t - \hat{y}_t = \varepsilon.$$  \hspace{1cm} (11)

Fig. 4 shows a sharp fluctuation in the neighborhood of $L_1$ loss when $\hat{y}_t = y_t$. Although it provides the correct gradient direction, which leads to it being trainable, but its values are coarse and the incremental relationship is nonlinear, so it obviously does not satisfy the above optimality assumption.

Equations (12) and (13) illustrate the gradient form of $L_2$ backpropagation to backbone

$$L_{L_2} = \frac{1}{N_w \times N_h \times N_d} \sum_{i} \sum_{j} \sum_{k} \|y_{i,j,k} - \hat{y}_{i,j,k}\|^2$$

$$\frac{\partial L_{L_2}(y, \hat{y})}{\partial \hat{x}_t} = \frac{\partial L_{L_2}(y, \hat{y})}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial \hat{x}_t} = 2(y_t - \hat{y}_t)(1 - \hat{y}_t^2).$$

In Fig. 5, the three curves before reaching the extreme value point execute a relatively reasonable trend; although it is not smooth, the magnitude of their absolute values is proportional to the distance from the point $\hat{y}_t = y_t$. However, when they are in the position after the extreme point, i.e., the area marked by the red line segment, they show an unreasonable trend and the absolute values that should be large are becoming reduced. The $L_2$ loss forms a parabola between $\hat{y}_t = y_t$ and $\hat{y}_t = 1$. Although this can still be trained and does not affect the propagation of the gradient, we still hope to find a more reasonable gradient induced by the loss.

Both the $L_1$ and $L_2$ losses are deficient in the form of the gradient passed to the backbone. We give the optimal gradient form assumption in (11). We want $\varepsilon$ to be directly used as the gradient form of the loss to backbone, and the resulting loss will satisfy the optimal gradient assumption, so we integrate it. Then, the loss function we expect can be expressed by

$$L_{TCE}(y, \hat{y}_t) = \int \varepsilon \cdot \frac{\partial \hat{x}_t}{\partial \hat{y}_t} d\hat{x}_t = \int y_t - \hat{y}_t d\hat{x}_t$$

$$= -y_t\left(\log|\hat{y}_t| + 1 - \log|\hat{y}_t - 1|\right)$$

$$- \frac{1}{2} \log|1 - \hat{y}_t^2| + \log 2 + C'$$

$$= -\left[(\frac{1 + y_t}{2})\log\left(\frac{1 + \hat{y}_t}{2}\right) + (\frac{1 - y_t}{2})\log\left(\frac{1 - \hat{y}_t}{2}\right)\right].$$

The loss of all voxels should be averaged during training so that the loss expression is

$$L_{TCE}(y, \hat{y}) = \frac{1}{N_w \times N_h \times N_d} \sum_{i} \sum_{j} \sum_{k} \left[(\frac{1 + y_{i,j,k}}{2})\log\left(\frac{1 + \hat{y}_{i,j,k}}{2}\right) + (\frac{1 - y_{i,j,k}}{2})\log\left(\frac{1 - \hat{y}_{i,j,k}}{2}\right)\right].$$

Its form is similar to cross entropy, so we named it TCE. This loss causes the gradient at training as in the following equation:

$$\frac{\partial L_{TCE}(y, \hat{y})}{\partial \hat{x}_t} = y_t - \hat{y}_t = \varepsilon.$$  \hspace{1cm} (16)

In Fig. 6, TCE shows the smoothed gradient of its feedback to backbone. When $\hat{y}_t = y_t$, the gradient is 0 and the distance between $\hat{y}_t$ and $y_t$ is linear with the gradient size, which is the optimal loss we want to obtain.

Although $L_1$ and $L_2$ can give correct feedback to the generator in the gradient direction, their gradient values are coarse and they do not reasonably reflect the distance between GT and the predicted value, while the gradient passed from TCE to the generator backbone is the residual of GT and the predicted value, which perfectly captures the change of the distance between the output and the target.
The overarching training framework follows Fig. 1. The optimizer uses Adam [49]. In general, the training of the discriminator should be ahead of the generator, which is more conducive to producing high-quality data. We use the TTUR strategy for training [50], setting the learning rate of discriminators $\eta_d = 0.0004$ and the generator $\eta_g = 0.0001$, and finally alternating the training of the generator and discriminators.

### III. EXPERIMENTS

#### A. Experimental Grouping

We designed three control groups and one experimental group using the current most common UNet as the baseline model (Group 1). UNet has an excellent performance in interpolating seismic data, which has been demonstrated in several works [see [18], [20], [22], and so on]. Group 2 is the replacement of the backbone of Group 1 with the generator we proposed in Fig. 1. Group 2 does not apply adversarial learning and therefore does not require the tanh activation function and TCE loss, and this group can verify the validity of the generator and the FSM compared to Group 1. Group 3 incorporates multidimensional adversarial learning, which differs from the experimental group in that it uses $L_1$ as the reconstruction loss. This group can be compared with Group 2 to verify the effectiveness of multidimensional adversarial and with the experimental group to verify the effectiveness of TCE loss. Group 4 is the experimental group, which uses exactly the method proposed in this article. The settings of the four groups of experiments are shown in Table I.

| Method           | Explain                                      |
|------------------|----------------------------------------------|
| Group 1          | UNet                                         |
| Group 2          | MDA Generator, Regression by generators only |
| Group 3          | MDA GAN ($L_1$)                              |
| Group 4          | MDA GAN (TCE)                                |

#### B. Dataset

The current experimental approach of most data-driven seismic interpolation studies is to train a new set of weights for each survey [5], [17], [18], [19], [20], [21], [22], [25], [27], [33], [34], but retraining the neural network is extremely time-consuming and resource-wasting, resulting in a much higher time and resource cost than the theory-driven approach. Furthermore, the traces of the gathers are often highly similar to each other, so the neural network weights obtained from training for a specific survey may be overfitted, and this overfitting tends to show extremely high performance for the test set divided by the same survey, so this way of dividing the dataset may not objectively reflect the performance of the model.

In this study, we want to train a general model that can be applied to the reconstruction of most seismic data without having to retrain new weights for different surveys, which is the current trend in artificial intelligence (AI) development, CV and natural language processing (NLP) scientists are currently working on building general models for various tasks [51], [52]. We have used multiple surveys (both prestack and poststack) to jointly train one model in the expectation that it will generalize to more data, and we have made the final model publicly available on Github.\(^1\)

The usage of the data is shown in Table II.

| Surveys           | Purpose         |
|-------------------|-----------------|
| SEG C3            | 3/5 train, 1/5 val, 1/5 test |
| Mobil Avo Viking Graben Line 12 | 3/5 train, 1/5 val, 1/5 test |
| F3 Netherlands    | 3/5 train, 1/5 val, 1/5 test |
| New Zealand Kerry | 3/5 train, 1/5 val, 1/5 test |
| New Zealand Parhaka | 3/5 train, 1/5 val, 1/5 test |
| New Zealand Opunake | All train      |
| Sinoopec surveys  | All train       |

\(^1\)Available: https://github.com/douyimin/MDA_GAN
C. Evaluation Metrics

All experiments in this article were evaluated quantitatively using the peak signal-to-noise ratio (PSNR) and SSIM metrics [53], and all data were normalized to [0, 1] prior to evaluation. To facilitate comparison and reproduction, the quantitative metrics are calculated directly using the public library.
Fig. 9. (a) Case of 40 consecutive missing traces. (a-1)–(a-4) Interpolation results of UNet, MDA generator, MDA GAN ($L_1$), and MDA GAN methods for this case, respectively. (a-5)–(a-8) are the SSIM map corresponding to each method. (b) Case of 70 consecutive missing traces. (b-1)–(b-4) Interpolation results of UNet, MDA generator, MDA GAN ($L_1$), and MDA GAN methods for this case, respectively. (b-5)–(b-8) SSIM map corresponding to each method. (c) Case of 100 consecutive missing traces. (c-1)–(c-4) Interpolation results of UNet, MDA generator, MDA GAN ($L_1$), and MDA GAN methods for this case. (c-5)–(c-8) SSIM map corresponding to each method.

scikit-image’s implementation\(^2\) of PSNR and SSIM, and all hyperparameters in SSIM are default.

1) Peak Signal-to-Noise Ratio : It is one of the most common methods for image quality assessment and is expressed as follows:

\[
F_{\text{PNSR}}(I_g', I_g) = 10 \cdot \log_{10} \left( \frac{F_{\text{max}}(I_g')^2}{F_{\text{mse}}(I_g', I_g)} \right) \tag{17}
\]

where $F_{\text{max}}(\cdot)$ is a function of the maximum value from the data and $F_{\text{mse}}(\cdot)$ is the mean square error (mse) of the two images.

2) Structural Similarity : Unlike PSNR’s estimation of pixel point-to-point, SSIM incorporates consideration of interpixel dependence and prefers to evaluate the spatial similarity of two images. The expression is given as follows:

\[
F_{\text{SSIM}}(I_g', I_g) = \frac{\left(2\mu_{I_g'}\mu_{I_g} + c_1\right)\left(2\sigma_{I_g'I_g} + c_2\right)}{\left(\mu_{I_g'}^2 + \mu_{I_g}^2 + c_1\right)\left(\sigma_{I_g'}^2 + \sigma_{I_g}^2 + c_2\right)} \tag{18}
\]

where $\mu_{I_g'}$ is the mean of $I_g'$, $\mu_{I_g}$ is the mean of $I_g$, $\mu_{I_g'}^2$ is the variance of $I_g'$, $\mu_{I_g}^2$ is the variance of $I_g$, $\sigma_{I_g'I_g}$ is the covariance of $I_g'$ and $I_g$, and $c_1$ and $c_2$ are two variables to stabilize the division with weak denominator, see literature [53] for details.

In skit-image, the equation is implemented by a sliding window of size $7 \times 7 \times 7$, i.e., the local SSIM is calculated.

\(^2\)Available: https://scikit-image.org
within each window to obtain the SSIM map. Averaging the SSIM map is the final SSIM value, while the visualization of the SSIM map can reflect the spatial distribution of the difference between the two data. In our experiments, we used SSIM to quantify the experimental results while visualizing the SSIM map to clearly reflect the spatial distribution of the reconstructed differences.

D. Test on SEG C3

In SEG C3, we stack gather into 3-D, as shown in Table II; 3/5 of the data were involved in training, 1/5 were used for validation, and we used the last 1/5 for testing. Certainly, there are other surveys involved in the training.

We present it using one of the 192 × 240 × 64 grid points (Fig. 7), which is obtained by interpolating (resize) the original data 168 × 205 × 31.

1) Discrete Missing: The upper limit of missing in most of the current studies is 50%. The present work hopes to address more complex missing cases, and we use 50% as the baseline and gradually expand the percentage of missing to 95% (50%, 75%, 90%, and 95%). Fig. 8 shows the qualitative results for each group of experiments in the missing randomized discrete case, and this figure also illustrates the visualization of quantitative results using SSIM metrics. Table III shows the SSIM and PSNR values on this test data.

In the case of 50% random missing, all four groups of experiments demonstrated promising results, and when the percentage of missing was expanded to 75%, UNet’s interpolation results started to show a discontinuity and its SSIM map showed a clear region of discrepancy, which was also shown at the same position in Group 2. When the percentage of missing continues to expand to 90%, the reliable interpolation is completed only for Groups 3 and 4 using multidimensional adversarial, and the regression-based approach is difficult to work with a high level of missingness.

With an extremely high percentage of missing 95%, regression methods, such as Groups 1 and 2, did not work at all, and interpolation was accomplished using multidimensional adversarial Groups 3 and 4, which differed by using different reconstruction losses. Qualitatively, the results are smoother using TCE. Quantitatively, the method using \( L_1 \) has higher SSIM values and TCE shows a higher PSNR. From Fig. 8(a-3)–(d-3), \( L_1 \) also shows some discontinuities and distortions in other missing ratios, the phenomenon that is also present in many GAN-based works [27], [34]. As we analyzed in Section II-D, the gradient provided by \( L_1 \) to the GAN is not optimal, which leads to interpolation results with some distortion. This is compensated well by TCE, which outperforms the currently most commonly used \( L_1 \) loss in most qualitative and quantitative experiments (except the SSIM metric under 95% missing).

2) Continuous Missing: Next, we compare the performance of the four methods in the continuous missing case. It should be noted that the best repair record for missing continuous seismic traces is currently by Yu and Wu [20]. They interpolated 30% of the missing continuous traces of the 128 × 128 2-D seismic image with an interpolated area of 128 × 38, i.e., 38 traces.

![Fig. 10. Mobil Avo Viking Graben Line 12 test data.](image)

**TABLE III**

| Discrete Missing Ratio | SSIM | PSNR |
|------------------------|------|------|
|                        | 50%  | 75%  | 90%  | 95%  |
| **SSIM**               |      |      |      |      |
| UNet                   | 0.9739 | 0.9356 | 0.8599 | 0.8189 |
| MDA Generator          | 0.9911 | 0.9490 | 0.8803 | 0.8377 |
| MDA GAN \( L_1 \)     | 0.9772 | 0.9612 | 0.9474 | 0.8901 |
| MDA GAN (TCE)          | 0.9847 | 0.9651 | 0.9465 | 0.8854 |
| **PSNR**               |      |      |      |      |
| UNet                   | 36.88 | 31.20 | 26.05 | 22.22 |
| MDA Generator          | 36.69 | 31.79 | 26.19 | 22.66 |
| MDA GAN \( L_1 \)     | 37.02 | 32.55 | 30.21 | 25.96 |
| MDA GAN (TCE)          | 37.14 | 32.81 | 31.00 | 26.29 |

Fig. 9 shows three consecutive missing, 40, 70, and 100 traces. In the case of consecutive missing cases, Groups 1 and 2 show unsatisfactory reconstructions with significant voids, and it can be observed that the regression method is only an extension of the data in the neighborhood of the missing site and is limited in scope. Table IV also shows that the quantitative metrics based on the MDA method are significantly higher than the other methods.

The MDA GAN method completes the reconstruction of the continuous missing. This experiment also shows the effectiveness of the TCE, where the interpolation result of \( L_1 \) has obvious splicing traces and discontinuity at the red arrow, while the method using TCE is smoother and continuous.

It is worth noting that the reconstruction results of UNet at the red box in Fig. 9 show significant differences compared to ground truth, while the model using FSM shows less or even no differences, precisely because the FSM in the generator comes into action, and as described in Section II-B and Fig. 3, FSM is able to retain more information about the unmissing part \((1 - \mathcal{M}) \times I_c\).

E. Test on Mobil Avo Viking Graben Line 12

We fold the prestack gather into 3-D and intercept 512 × 256 × 128 grid points in the test data as a sample for qualitative analysis, as shown in Fig. 10.

Fig. 11 shows the qualitative interpolation results of MDA GAN and UNet, and Table V shows the corresponding quantitative results.
At 50% discrete deficiency, both MDA GAN and UNet are able to perform the interpolation task well. When the ratio comes to 75%, UNet starts to be overwhelmed, while our method still shows high reliability. With more complex 90% and 95% missing, UNet invalidates, while our method still yields promising results. In the case of consecutive missing cases, all interpolation results of UNet are failures. Fig. 11 shows that the qualitative results of our method in complex cases are significantly higher than those of UNet, which is currently the most used, and the same conclusion can be obtained in the quantitative results in Table V.

**F. Test on New Zealand Kerry and F3 Netherlands**

This work focuses on the interpolation and reconstruction of complex cases. The above experiments validate the reconstruction performance of our method under complex
missing for the set of prestack gather. The purpose of this section is to verify the performance of our method with complex surveys rich in anomalies such as faults and salt bodies.

The missing in original engineering often occurs in shot gathers, and anomalous bodies also appear in brutal stack sections [12] formed by shot gathers, but the poststack data containing anomalous bodies are more abundant in the current public data, so we use the poststack as the test data in this section.

1) New Zealand Kerry (Fault Rich): Kerry contains abundant faults perpendicular to the timeline, which is a great challenge for the interpolation task. Fig. 12(a) shows the original data, which sampled at a total of $224 \times 512 \times 192$ grid points. Due to the limited data available for Kerry, part of the data used for the presentation was involved in the training [which has been marked in Fig. 12(a)].

In Fig. 12(b), we removed 80% of slices in both crossline and timeline directions, accumulating 95.62% of the missing voxels. Because of the large proportion of missing traces, some areas form continuous missing, and most of the faults are covered by missing traces.

UNet does not complete the reconstruction for most of the missing faults, and there are some voids at the yellow arrows in Fig. 12(c), while at the corresponding positions, MDA GAN fills these voids as faults. In the region marked by the red arrow in Fig. 12(d), although MDA GAN restores the amplitude, the fault information is lost because the continuous missing here removes the fault completely, while MDA GAN shows a very high interpolation performance for the missing where a portion of the fault information is retained.

2) F3 Netherlands (Salt Bodies): Salt bodies are common in marine seismic data, which have closed characteristics and distinct boundaries. In Fig. 13, we used the salt body part at the bottom of the F3 as qualitative data, which was all used as validation and test sets without involvement in training, which sampled at a total of $128 \times 384 \times 384$ grid points.

Fig. 12(a) shows the original data. In Fig. 12(b), we removed 80% of slices in both crossline and timeline directions, accumulating 95.76% of the missing voxels. In Fig. 12(c), the internal structure of the salt body is lost at (1) of the UNet reconstruction result, and the boundary information of the salt body is not reconstructed at (2), while the corresponding position in Fig. 12(d) shows a high-quality reconstruction result. MDA GAN demonstrates excellent reconstruction performance for anomalies such as faults and salt bodies.

### TABLE VI

| Discrete Missing Ratio | 50%  | 75%  | 90%  | 95%  |
|------------------------|------|------|------|------|
| **SSIM**               |      |      |      |      |
| UNet                   | 0.9588 | 0.9217 | 0.8801 | 0.8701 |
| MDA Generator          | 0.9620 | 0.9355 | 0.8980 | 0.8789 |
| MDA GAN ($L_2$)        | 0.9673 | 0.9501 | 0.9289 | 0.9179 |
| MDA GAN (TCE)          | 0.9700 | 0.9528 | 0.9277 | 0.9137 |
| **PSNR**               |      |      |      |      |
| UNet                   | 34.61 | 29.36 | 23.50 | 22.29 |
| MDA Generator          | 34.72 | 30.45 | 24.38 | 23.01 |
| MDA GAN ($L_2$)        | 34.88 | 31.93 | 26.11 | 25.84 |
| MDA GAN (TCE)          | 35.37 | 32.49 | 27.25 | 26.12 |

G. Quantitative Metrics for the Total Test Set

Finally, we calculate the SSIM and PSNR metrics jointly for all the test sets shown in Table II. Because the size of the test parts differs for each survey, we calculate only the proportional discrete missing cases, and the results are shown in Table VI. For a more visual analysis of the performance of each group of experiments, we organized Table VI into Fig. 14.

In Fig. 14(a), the MDA-based metrics are remarkably higher than the other two groups, indicating the effectiveness of multidimensional adversarial. While using $L_1$ and TCE losses do not differ significantly in terms of SSIM, with $L_1$ slightly...
higher than TCE, which is one of the reasons why L1 is currently widely used as a reconstruction loss, the experiments from the above show that the qualitative results of TCE losses are more advanced (Figs. 8 and 9), and it can provide smoother and continuous reconstruction results. In Fig. 14(b), the PSNR of MDA GAN (TCE) was higher than the other groups. Similar to Fig. 14(a), there was no significant difference between TCE and L1, and TCE was slightly higher than L1 in terms of PSNR metrics. This may be because PSNR is more sensitive to pixel differences, while SSIM prefers to show structural differences in regions.

IV. CONCLUSION

This work focuses on complex case interpolation of 3-D seismic data and proposes an MDA seismic reconstruction framework and TCE reconstruction loss. The MDA framework emphasizes maintaining the anisotropy and spatial continuity of the reconstructed seismic data in all three directions, and thus, promising results can be obtained in multiple complex cases. The FSM embedded in the generator enables the reconstructed data to retain more information about the unmissing parts. The TCE loss is mathematically derived to provide the generator with the optimal reconstruction gradient under the tanh activation function to ensure that the reconstructed pixels are free of distortion. Qualitative experiments have demonstrated the effectiveness of the MDA framework by showing that the MDA GAN is able to accomplish promising reconstructions under large proportions of discrete and continuous missing as well as fault and salt-enriched data missing. With SSIM visualization, it is demonstrated that the FSM module is indeed able to retain more of the original input information. Multiple qualitative experiments also demonstrate the ability to generate smoother and more continuous reconstruction results with TCE loss. Quantitative experiments demonstrated significantly better reconstruction performance of MDA GAN over multiple surveys than the baseline model and the ablation comparison experimental group. In the test set consisting of multiple surveys, MDA GAN outperformed the baseline model by 0.03, 0.05, and 0.04 for SSIM metrics and 3.13, 3.75, and 3.83 for PSNR under large proportional deletions of 75%, 90%, and 95%, respectively. In the 50% simpler missing case, our method still has the advantage.

APPENDIX

A. Generative Adversarial Network

GANs are proposed by Goodfellow et al. [54]. It consists of a generator network $G_{θ_{G}}$ and a discriminator network $D_{θ_{D}}$, where the task of the generator is to yield images $x \in \mathcal{R}^{N_c \times N_h}$ with a latent noise prior vector, and $z \in \mathcal{R}^{d}$ as input $z$ is sampled from a known distribution, i.e., latent vector $z$, $z \sim \mathcal{U}[-1, 1]^d$ [54]. The task of the discriminator is to distinguish real images from generated ones. The generator and the discriminator play a zero-sum game in which the two networks learn from each other and the data obtained by the generator become increasingly close to the real data so that the desired data can be generated. The objective function of GANs can be expressed as follows:

$$\max_{θ_{G}} \min_{θ_{D}} V(G_{θ_{G}}, D_{θ_{D}}) = \mathbb{E}_{x \sim p_{data}(x)} \left[ \log D_{θ_{D}}(x) \right] + \mathbb{E}_{z \sim p_{z}(z)} \left[ \log \left( 1 - D_{θ_{D}}(G_{θ_{G}}(z)) \right) \right]$$

(19)

where $p_{data}(x)$ represents the distribution of the training set, $p_{z}(z)$ represents the normal distribution of the noise, and $D_{θ_{D}}(G_{θ_{G}}(z))$ represents the generated data. $G_{θ_{G}}$ expects $D_{θ_{D}}(G_{θ_{G}}(z))$ to be as large as possible, and when $V(G_{θ_{G}}, D_{θ_{D}})$ becomes smaller, thus the equation is minimized for $G_{θ_{G}}$. $D_{θ_{D}}(x)$ is enhanced with the discriminator performance; when $D_{θ_{D}}(G_{θ_{G}}(z))$ becomes smaller and $V(G_{θ_{G}}, D_{θ_{D}})$ becomes larger, thus the equation is maximized for $D_{θ_{D}}$.

B. Image Inpainting via GAN

The key to applying GAN to image inpainting is to join discriminator and adversarial loss on the base of encoder regression [28], [29], [30], [31]. The existing image inpainting work is for 2-D images, its loss can be expressed as (20), and the basic framework is shown in Fig. 15

$$L_{G} = L_{adv} + \lambda L_{rec}$$

(20)

where $L_{adv}$ is the adversarial loss, commonly used CE-GAN [54], LS-GAN [55], W-GAN [56], and so on, $\lambda L_{rec}$ is the reconstruction loss (regression loss), and $L_{1}$ and $L_{2}$ are generally used [28], [29], [30], [31].

Most of the current GAN-based image inpainting follows this framework.

ACKNOWLEDGMENT

The authors are very indebted to the anonymous referees for their critical comments and suggestions for the improvement of this article.
[49] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.
[50] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “GANs trained by a two time-scale update rule converge to a local Nash equilibrium,” in Proc. Adv. Neural Inf. Process. Syst., vol. 30, 2017, pp. 1–13.
[51] A. Dosovitskiy et al., “An image is worth 16×16 words: Transformers for image recognition at scale,” 2020, arXiv:2010.11929.
[52] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” 2018, arXiv:1810.04805.
[53] 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.
[54] I. Goodfellow et al., “Generative adversarial nets,” in Proc. Adv. Neural Inf. Process. Syst., vol. 27, 2014, pp. 1–8.
[55] G.-J. Qi, “Loss-sensitive generative adversarial networks on Lipschitz densities,” Int. J. Comput. Vis., vol. 128, pp. 1118–1140, Nov. 2019.
[56] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein GAN,” 2017, arXiv:1701.07875.

Yimin Dou (Student Member, IEEE) received the B.S. degree in environmental engineering and the M.S. degree in computer technology from the University of Jinan, Jinan, China, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree with the China University of Petroleum (East China), Qingdao, China. His main research interests include interdisciplinary research in geosciences and machine learning.

Kewen Li received the Ph.D. degree in information management and information system from Tianjin University, Tianjin, China, in 2012. He is currently a Professor with the College of Computer and Communication Engineering, China University of Petroleum (East China), Qingdao, China. He is also a Visiting Scholar with Florida Atlantic University, Boca Raton, FL, USA. He has published more than 100 papers in reputable international conferences and journals. His research interests are in artificial intelligence, software engineering, data mining, and machine learning.

Hongjie Duan, photograph and biography not available at the time of publication.

Timing Li, photograph and biography not available at the time of publication.

Lin Dong (Member, IEEE), photograph and biography not available at the time of publication.

Zongchao Huang, photograph and biography not available at the time of publication.