A Multi-Task Learning Method for Relative Geologic Time, Horizons, and Faults With Prior Information and Transformer

Jiarun Yang, Xinming Wu, *Associate Member, IEEE*, Zhengfa Bi, and Zhicheng Geng

***Abstract*— Horizon extraction and fault detection are essential in seismic interpretation and are closely related to each other. Most existing methods tend to deal with these two tasks independently, and may not work well in interpreting seismic images with complex geologic structures. We propose a multi-task learning (MTL) network with two branches to extract all horizons and detect faults simultaneously by estimating a relative geologic time (RGT) map as well as computing a fault map. These two branches share training datasets, feature maps, and network parameters during the training. The RGT estimation branch, constructed with a transformer architecture, is more lightweight compared with previous convolutional neural network (CNN) methods but provides a larger and structure-oriented receptive field to adaptively capture global structural information for estimating a globally optimal RGT map. The fault detection branch is a simple CNN, which merges feature maps shared by the transformer and the derivatives of the estimated RGT to compute a fault map. The fault detection branch provides boundary control for the RGT estimation branch, while the latter provides global constraints for the former to improve its robustness to noise. Note that our RGT estimation by globally fitting all structures in a seismic image is a volumetric horizon interpretation method with which we are able to obtain a whole volume of horizons, all at once, by simply extracting contours of the RGT map. In our method, we further enable convenient human interactions by integrating manually interpreted horizons (or horizon segments) into the network, which imposes expert knowledge on the network to estimate reasonable RGT results from seismic images with complex fault systems, unconformities, and poor data quality. Moreover, when using 3-D horizons as constraints, we are able to decompose the computational expensive 3-D RGT estimation from a seismic volume into independently parallel 2-D estimations slice by slice and combine them to obtain a laterally consistent 3-D result.**

***Index Terms*— Fault, horizon, multi-task learning (MTL), relative geologic time (RGT), transformer.**

I. Introduction

EXTRACTING horizons and detecting faults from a seismic image are two important tasks of seismic interpretation. Horizons can be used for ancient sedimentary environment analysis [1], structural modeling [2], and stratigraphic interpretation [3]. Faults can be used for structural modeling [4], well location screening [5], and tectonic analysis [6].

The discontinuities of reflections are typical features of faults in a seismic image. Therefore, different methods are proposed for fault detection to highlight the reflection discontinuities in a seismic image by computing additional attributes, such as semblance [7], coherency [8], [9], [10], variance [11], curvature [12], and fault likelihood [13], [14]. However, these methods based on reflection discontinuities are sensitive to noise and other structures similar to faults in a seismic image. Therefore, some methods are proposed to further enhance fault-related features and suppress the noise in the attribute images, such as ant tracking [15] and optimal surface voting [16]. Some other authors [17] proposed to improve fault detection by using prior horizon information. Fault detection is also important for horizon interpretation. When we interpret the horizons on both sides of faults, the fault detection maps can provide boundary constraints.

Interpreters interpret horizons by tracking reflections in a seismic image. Because manual interpretation is time consuming, numerous automatic and semiautomatic methods are proposed for improving the interpretation efficiency. These horizon extraction methods are mostly based on seismic instantaneous phase unwrapping [18], [19], seismic waveform classification [20], [21], [22], and seismic local slopes [23], [24], [25], [26], [27], [28], [29], [30]. These methods can extract the horizon correctly along continuous reflections in the seismic image, but may not work well in tracking the reflections across faults. To solve this problem, some methods [31], [32] remove faults from a seismic image by computing fault locations and fault displacement vectors. Some authors [33] use manually picked points on both sides of faults to constrain the process of horizon extraction.

Estimating a relative geologic time (RGT) volume or map from a seismic image provides a volumetric method of extracting all horizons in the seismic image [18], [19], [34], [35]. In the RGT volume, every point is assigned an RGT value,

Manuscript received 16 December 2022; revised 6 March 2023; accepted 23 March 2023. Date of publication 11 April 2023; date of current version 26 April 2023. This work was supported by the National Science Foundation of China under Grant 41974121. *(Corresponding author: Xinming Wu.)*

Jiarun Yang, Xinming Wu, and Zhengfa Bi are with the School of Earth and Space Sciences, University of Science and Technology of China, Hefei 230026, China (e-mail: jryang@mail.ustc.edu.cn; xinmwu@ustc.edu.cn; zfbi@mail.ustc.edu.cn).

Zhicheng Geng is with the Machine Learning Solutions Lab, Amazon, Austin, TX USA (e-mail: zhichenggeng@utexas.edu).

Digital Object Identifier 10.1109/TGRS.2023.3264593

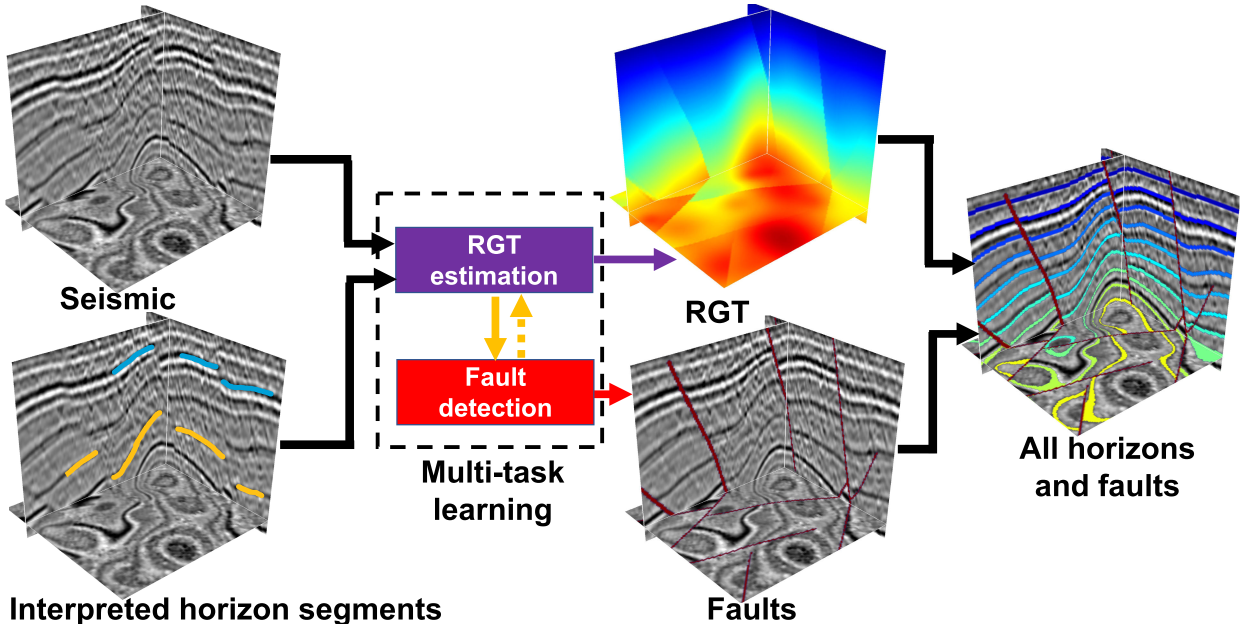

Fig. 1. We propose an MTL network to simultaneously extract all horizons and faults from a seismic image and interpreted horizon segments. The network includes RGT estimation and fault detection branches, which are related to each other.

and RGT values will increase with depth in general. The contours extracted from an RGT volume represent seismic horizons, and faults and unconformities are highlighted by discontinuities in the RGT volume. An RGT map can be also applied in sedimentologic interpretation [36], geologic body detection [36], structural implicit modeling [37], and missing well-log data prediction [38].

For RGT estimation, a straightforward but laboring method is to track or pick horizons and interpolate these horizon surfaces to obtain an RGT volume [39], [40]. Although the estimated RGTs can strictly fit the interpreted horizon surfaces, the RGT surfaces interpolated among these horizons are hardly estimated well. To solve this problem, the methods based on phase unwrapping [18], [19] are proposed to estimate RGT by unwrapping the seismic instantaneous phase. Fomel [41] uses a slope-based method to pick horizons and estimate RGT maps by using plane-wave destruction and setting seed points to further follow the local slopes.

In recent years, convolutional neural network (CNN)-based methods have successfully applied in seismic data processing [42], [43], seismic interpretation [44], [45], [46], and seismic inversion [47], [48]. Some methods based on CNN are proposed to estimate RGT maps from seismic images and obtain the structural information (e.g., faults and horizons) from the predicted RGT images [49], [50]. Geng et al. [49] use a 2-D end-to-end CNN for RGT estimation from seismic images. From the predicted RGT map, all horizons in the seismic image can be extracted efficiently by extracting RGT contours. Bi et al. [50] consider RGT estimation and fault detection to be related to each other, but deal with these two tasks separately. They use one 3-D end-to-end CNN to compute an RGT volume from a 3-D seismic volume and use another 3-D end-to-end CNN to detect faults from the predicted RGT volume. Although numerous methods have been proposed for both tasks of the seismic horizon and fault interpretation, most of the existing methods deal with them as two independent tasks. In addition, most of the existing methods may not be able to robustly estimate reasonable results from seismic images with complex fault systems, unconformities, and poor data quality. Therefore, it is still necessary to incorporate prior structural information into the horizon and fault interpretation methods, especially the deep-learning-based ones, to improve their robustness. Vision transformer (ViT) [51] is the first work to use self-attention in computer vision, and it achieved state-of-the-art performance in image classification. Unlike the traditional CNN architecture, the transformer architecture excels in capturing global features. Now, transformer architecture is obtaining greater attention as a viable block used in geophysics, such as earthquake detection [52], [53], deblending [54], and reservoir parameter prediction [55].

In this article, we propose a multi-task learning (MTL) network (Fig. 1) with the constraints of prior structural information to simultaneously compute an RGT map and a fault image from a seismic image as well as interpreted horizons. The MTL network consists of two coupled branches of RGT estimation and fault detection. We construct the first branch of RGT estimation with an encoder–decoder architecture. The encoder uses a transformer backbone to capture global structural information. The decoder uses a CNN backbone to extract local structural features and employs selective feature fusion (SFF) modules to merge both global (concatenated from the encoder) and local information for the RGT estimation. Compared with the previous CNN method [49], our RGT estimation branch is more lightweight but provides an adaptively anisotropic receptive field to help our network build RGT maps based on the structural information in seismic images. In this branch, the inputs include a seismic image and a

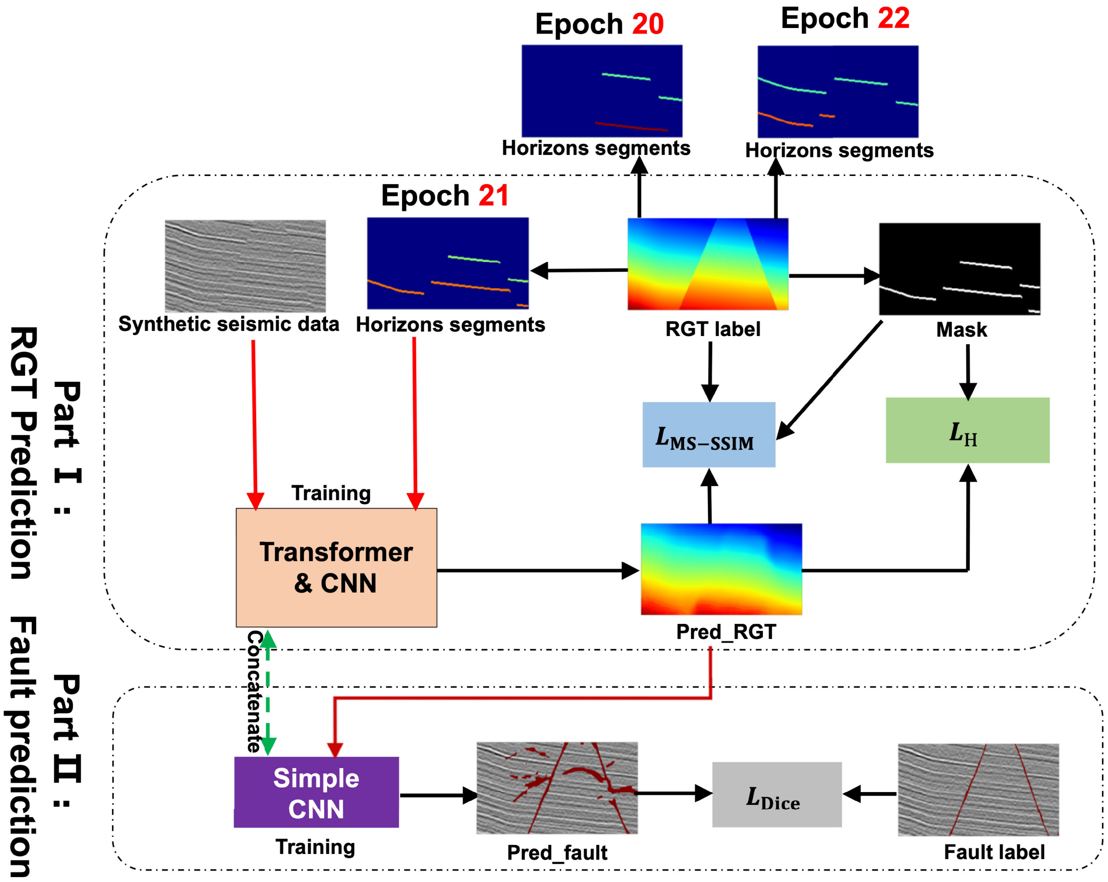

Fig. 2. Workflow of training our MTL model. In every epoch, the horizon segments and the corresponding mask are randomly selected from the RGT label, which can improve the diversity of training data. The RGT estimation network includes a transformer backbone and CNN, and the fault detection network is a simple CNN.

map containing interpreted horizon segments with preassigned RGT values, and the output is an estimated RGT map. The interpreted horizon segments with RGT values are used as one of the inputs and as a constraint in loss functions, which helps guide our network to estimate reasonable RGT results, especially in complicated cases. Our method allows inputting the prior constraints of incomplete horizons or even sparse horizon points that may be only interpreted near the complex structure areas (e.g., both sides of faults, unconformities, and geobody boundaries), without the need of using complete horizons as constraints. In this way, we can build a convenient interaction between interpreters and our network, where interpreters can modify the input horizon segments in real time to update RGT and fault estimations until reasonable results are obtained. Moreover, when using 3-D horizons as constraints, we are able to estimate 2-D RGT maps slice by slice in a 3-D seismic volume and further merge them to obtain a full 3-D RGT result with low computational time and memory. The contours extracted from an RGT volume represent seismic horizons, and we can obtain a whole volume of horizons from the result of the RGT estimation branch.

The second branch of fault detection is a simple end-to-end CNN architecture whose inputs consist of the derivatives of the RGT volume and feature maps shared from the encoder of the RGT estimation branch. With shared feature maps and network parameters, these two branches can provide controls for each other. In the process of forward propagation during the training, the first branch can transfer feature maps with sufficient seismic structural information and obvious fault structures to the second branch. These feature maps, shared by the first branch of RGT estimation, help improve the robustness of the fault detection branch to obtain more clear and more continuous fault maps. In the process of backward propagation during the training, the loss in the fault detection branch can provide boundary constraints for the first branch of RGT estimation to obtain sharper results near faults.

In summary, our contributions to this article are as follows.

1) We propose an MTL learning method for RGT estimation, horizon extraction, and fault detection, in which we use a lightweight transformer architecture to capture the global structure patterns in a seismic image.
2) In addition to seismic data, we propose to add prior horizon constraints to guide the network to obtain reasonable RGT and fault images in complicated cases.
3) In order to implement the 3-D RGT estimation task accurately and efficiently, we decompose the 3-D problem into independently parallel 2-D estimations slice by slice where we use 3-D horizons as constraints to

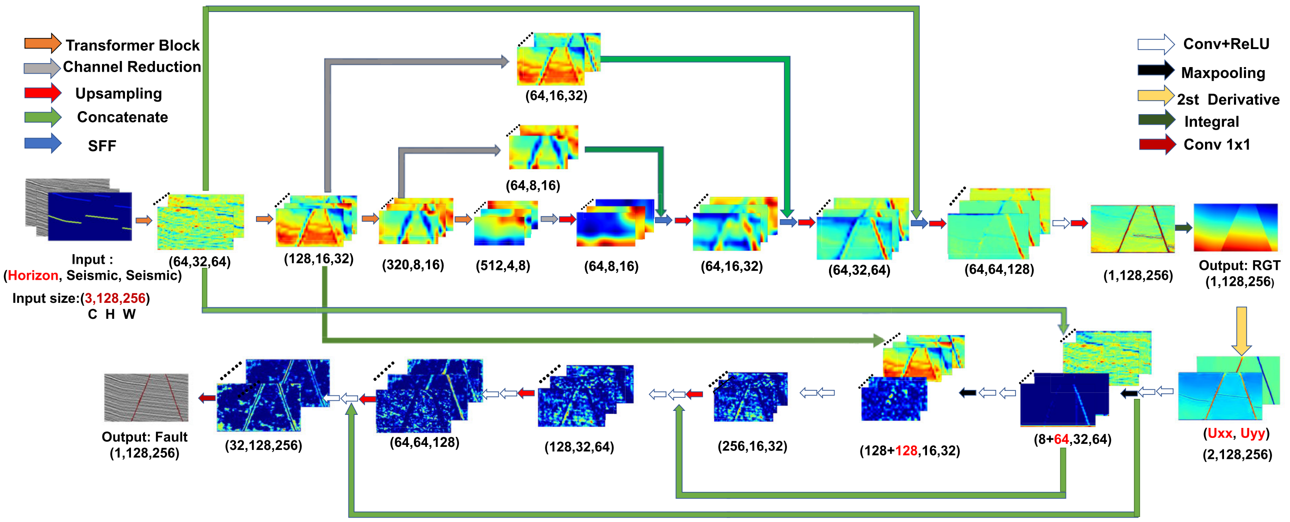

Fig. 3. Proposed MTL network architecture consists of two branches of RGT estimation and fault detection. The RGT estimation branch contains a transformer encoder to capture the global structural relationships from the input (horizon, seismic, and seismic) and a CNN decoder to compute the local information and fuse both global and local information by the module named SFF. The fault estimation branch is a simple end-to-end CNN architecture, and the inputs of this branch are the second derivatives of the RGT prediction and the feature maps from the transformer encoder of the first branch, which can provide clear fault features and more detailed seismic structural features, respectively.

maintain the lateral consistency of the independent 2-D estimations.

II. Framework of Multi-Task Learning

We use an MTL network (Fig. 2) with a transformer backbone and a simple end-to-end CNN to compute RGT and fault maps from seismic images and horizon segments. To integrate expert knowledge into our network, we use horizon segments with preassigned RGT values as the input of our network and the constraints of loss functions in the RGT estimation branch. We use the Dice loss to solve the highly imbalanced distribution of samples for fault detection. Finally, we quantitatively evaluate the estimated results by comparing them with manual interpretation.

A. RGT Estimation

Fig. 3 shows more detail of our MTL network architecture. The branch of RGT estimation is modified from a network with transformer blocks [56] named global–local path networks (GLP). The inputs of this branch are a single seismic image and horizon segments, and the output is an RGT map. The inputs are embedded as patches that are fed into the transformer block. The transformer block includes multiple sets of an efficient self-attention module, the feed-forward network (FFN), and a patch merging layer. The self-attention module can capture vertical and horizontal long-range structural information, and we consider this character beneficial for RGT estimation that requires globally fitting structural features in a seismic image. The vertical direction features in a seismic image represent the relative relations of layers, and the horizontal direction features in a seismic image represent the continuity of each layer. The FFN module provides positional information for the transformer block. Because of the patch merging layer, the encoder part can generate multi-scale features, which are also used in the decoder. Through four transformer blocks, we can obtain a multi-scale feature map with a size of $(H)/(2^{i+1}) \times (W)/(2^{i+1}) \times C_i$ ($i \in \{1, 2, 3, 4\}$).

In the lightweight decoder, we aim to obtain meaningful local information and integrate it with global information from the encoder with skip connections. The channels of input features from the encoder are reduced into 64 with a 1×1 convolution filter to decrease the computational complexity. Next, we use an upsampling layer to enlarge the size of the feature map from $(H)/(32) \times (W)/(32) \times 64$ to $(H)/(16) \times (W)/(16) \times 64$. Then, we use an SFF with an attention mechanism to select and fuse global and local features.

The structure of SFF is shown in Fig. 4, where first use 1×1 convolution filter to reduce the dimensions of global features to 64, which is equal to the dimensions of local features. Then, these two-part features are concatenated with the size of $(H)/(16) \times (W)/(16) \times 128$. Next, two convolution modules are used to integrate information from these features. After the information integration, we obtain two attention maps from a convolution layer and a sigmoid layer. The two maps represent what the local features and global features are focusing on. In the attention map of global features [Fig. 4(b)], the areas of horizons and faults appear as larger values in the attention maps of global and local features, respectively. This is consistent with our expectation that horizons represent global relationships and faults represent detailed information. Finally, the two maps are multiplied with original global and local features, and are further added together to obtain a hybrid feature.

After three SFF modules and four upsampling layers, we obtain an $H \times W \times 1$ feature map which estimates the vertical derivative of an RGT map. We further integrate

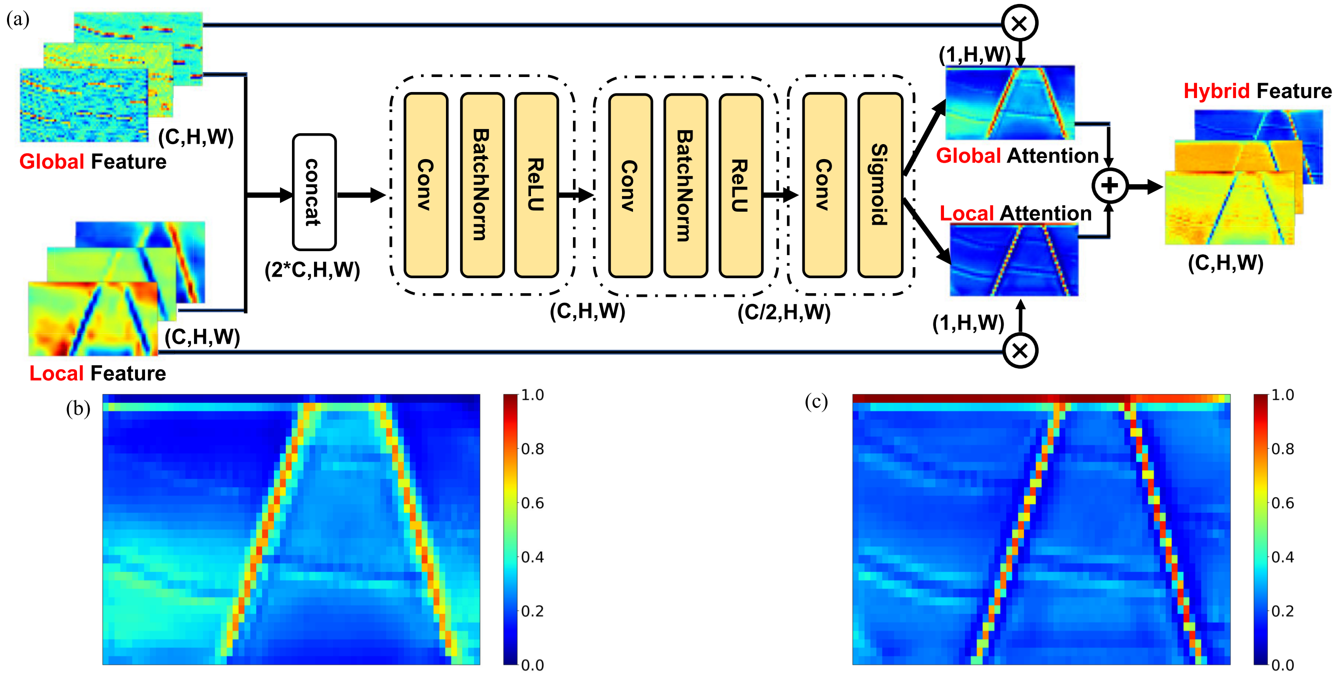

Fig. 4. (a) Structure of the SFF module. (b) Attention map of global features. (c) Attention map of local features.

this feature map in the vertical direction to obtain a final RGT map. The reason why we design such an intermediate layer of integration instead of directly outputting an RGT map in our network is that the integration will accumulate potential prediction errors and amplify the loss misfit during the training, and it can help our network learn RGT features better. In Section IV, we use examples to demonstrate the advantage of the integral process.

B. Fault Detection

We use a simple CNN (button part in Fig. 3) to detect faults by the shared features from the RGT estimation branch. First, we observe that fault features are implicitly indicated as discontinuities in the estimated RGT map. Therefore, we calculate the second derivatives (U_{xx} and U_{yy}) of the predicted RGT map to obtain features for highlighting faults. In the fault detection network, the two derivative features are used as inputs followed by one convolution block and a max pooling layer with stride 4. The convolution block includes two sets of a 3×3 convolution layer and an ReLU activation. As shown in Fig. 3, the outputs of the first convolution and max pooling layers are eight feature maps containing obvious fault structures. These feature maps computed only from the RGT map show clear fault structures in global scales but may miss detailed features of small-scale faults in the original seismic image. Because the transformer encoder of the RGT estimation branch can extract meaningful features from the seismic image, we merge the output feature maps of the first and second transformer blocks into the encoder of the fault detection branch. Then, we use two convolution blocks and one max pooling layer with stride 2 to obtain the other multiscale feature maps in the encoder of the fault detection. In the decoder part of the fault detection, we use three convolution blocks, three 2×2 upsampling layers, and two concatenations with feature maps from the encoder to restore the original resolution as well as fuse fault features between the encoder and decoder. A fault probability map is finally computed, through a 1×1 convolution layer followed by a sigmoid layer.

C. Loss Function

1) RGT Loss: During the training process, the RGT estimation is supervised by two loss functions based on the known horizon segments and RGT labels, respectively. We expect that the estimated RGT values at each known horizon segment should be exactly the same. We impose a such constraint on the RGT prediction during the training by defining the following loss $\mathcal{L}_{\mathrm{H}}$:

$$\mathcal{L}_{\mathrm{H}} = \sum_{i=0}^{N} \sum_{j=0}^{M_i} (y_{i,j} - \bar{y}_i)^2 \tag{1}$$

where N is the number of horizon segments, and M is the number of points at each horizon. $y_{i,j}$ represents an RGT value estimated at the jth point of the ith horizon, and $\bar{y}_i$ is the average of the RGT values estimated at the ith horizon. In this loss measurement, we do not care about specific RGT values estimated at the known horizon segments, and we only make sure the RGT values estimated at each horizon are identical. With the known positions of the horizon segments, this loss provides a self-supervision during the training.

Another loss function supervising the RGT estimation is based on RGT labels and we define it with the structural similarity (SSIM) metric [57]. SSIM has been widely used in the regression problem, such as super-resolution [58], depth estimation [59], and image restoration [60]. Compared with pixel-wise measurements, such as MSE or MAE, SSIM focuses more on the structural information. Our RGT map is a combination of geological structures, such as horizons and faults. Therefore, we use the criterion based on multi-scale SSIM (MS-SSIM) as the second loss function in the RGT estimation branch, and MS-SSIM can be written as follows:

$$\text{MS-SSIM}(x, y) = [l_M(x, y)]^{\alpha M} \cdot \prod_{j=1}^{M} [c_j(x, y)]^{\beta_j} [s_j(x, y)]^{\gamma_j} \tag{2}$$

x and y are two images that we want to evaluate. With M scales of one image, this method can measure image features at different resolutions. This measurement mimics the process of seismic structural interpretation by geophysicists or geologists who typically zoomed-in view and zoomed-out view the seismic image to observe the structures in different scales. In the RGT estimation problem, l, c, and s represent the measurements of RGT values, a sharp change in an RGT map, and relative relationships among the contours (corresponding to horizons in a seismic image) in an RGT map, respectively. We calculate the SSIM in one part of an image, and we slide an 11×11 Gaussian window pixel by pixel to compute the mean SSIM of all local images. In addition, we replace the predicted RGT values with ground truth values at the horizons segments before calculating this MS-SSIM loss. These replaced values, if consistent with the predicted ones, will affect the surrounding pixels by applying the Gaussian window which makes the network more sensitive to the input horizon segments. The range of SSIM is from 0 to 1, and the loss function of MS-SSIM can be written as follows:

$$\mathcal{L}_{\text{MS-SSIM}} = 1 - \text{MS-SSIM}(x, y). \tag{3}$$

The total loss function for RGT estimation is a hybrid one defined as follows:

$$\mathcal{L}_{\text{RGT}} = (\mathcal{L}_{\text{MS-SSIM}} + \mathcal{L}_{\text{H}})/2. \tag{4}$$

2) Fault Loss: In the fault detection branch, we consider it to be a binary segmentation problem. Binary cross-entropy (BCE) is a common loss function in binary segmentation problems, which can be written as follows:

$$\mathcal{L}_{\text{BCE}} = -\sum_{i=1}^{i=N} y_i \log p_i - \sum_{i=1}^{i=N} (1 - y_i) \log (1 - p_i) \tag{5}$$

where N represents the number of points in the fault image. y_i represents the positive label and p_i represents the prediction possibility, which is calculated by the sigmoid layer in the network.

BCE cannot work well in the dataset with an imbalanced distribution of positive and negative samples. The positive (fault) samples are much less the negative (non-fault) samples in a fault image [44]. When training with the common BCE loss, the network can easily make only negative predictions everywhere to minimize the training objective. Several loss functions have been proposed in recent years to solve this problem. The dice coefficient score (DCS) [61] is a measurement used to evaluate the similarity between prediction and label and is widely used in the task with an imbalanced training dataset. DCE can be written as follows:

$$\text{DCS} = 2 \times \frac{| P \cap G |}{| P | + | G |} \tag{6}$$

where P and G denote the prediction and ground truth, respectively. $| P \cap G |$ represents the intersection of P and G. $| P |$ and $| G |$ represent the pixel number of P and G. The range of DCS is from 0 to 1, and the larger the value is, the more similar prediction and label are. We define the loss function for fault detection based on the DCS as follows:

$$\mathcal{L}_{\text{Fault}} = 1 - \text{DCS}. \tag{7}$$

Finally, the total loss function of training the MTL network can be written as follows:

$$\mathcal{L}_{\text{Total}} = (\mathcal{L}_{\text{RGT}} + \mathcal{L}_{\text{Fault}})/2. \tag{8}$$

D. Evaluation Metric

We propose a metric to measure the RGT estimation, which is similar to the one discussed in [50], and call this metric horizon extraction error by RGT values (HEERV). In calculating the metric, we first uniformly select several RGT values R_i and extract the corresponding contours from the RGT label. Then, we compute the average values $\tilde{R}_i$ of the RGT values estimated at the extracted contours and further extract the corresponding contours of $\bar{R}_i$ from the estimated RGT map. Finally, we compute the vertical distances between the contours extracted from the RGT label and the predicted RGT map. The distances from our HEERV metric to evaluate the accuracy of the RGT estimation or horizon extraction.

III. Training and Validation

In this section, we first illustrate the workflow of the training data generation. Then, we introduce the detail of training and validation. Finally, we show the performance of our model on synthetic validation data.

A. Training Data

As a supervised learning method, our proposed MTL network needs a large number of samples (seismic images and horizon segments) and labels (RGT and fault images) to train a model with stability and robustness. The fault labels obtained from field seismic images by manually picking are time consuming and highly subjective and estimating RGT maps accurately from field seismic images are almost impossible.

Therefore, we use a workflow to automatically build a synthetic training dataset [62]. First, we build an initial flat reflectivity model and add some folding and faulting controlled by some parameters. Next, we convolve the model with a Ricker wavelet to obtain a clean synthetic seismic image and add some noise extracted from field data with a random SNR to make the synthetic seismic image [Fig. 5(a)] more similar to field data. The reflectivity, structure parameters, and the peak frequency of wavelet and SNR are all randomly chosen in a reasonable range. To obtain the corresponding fault labeling

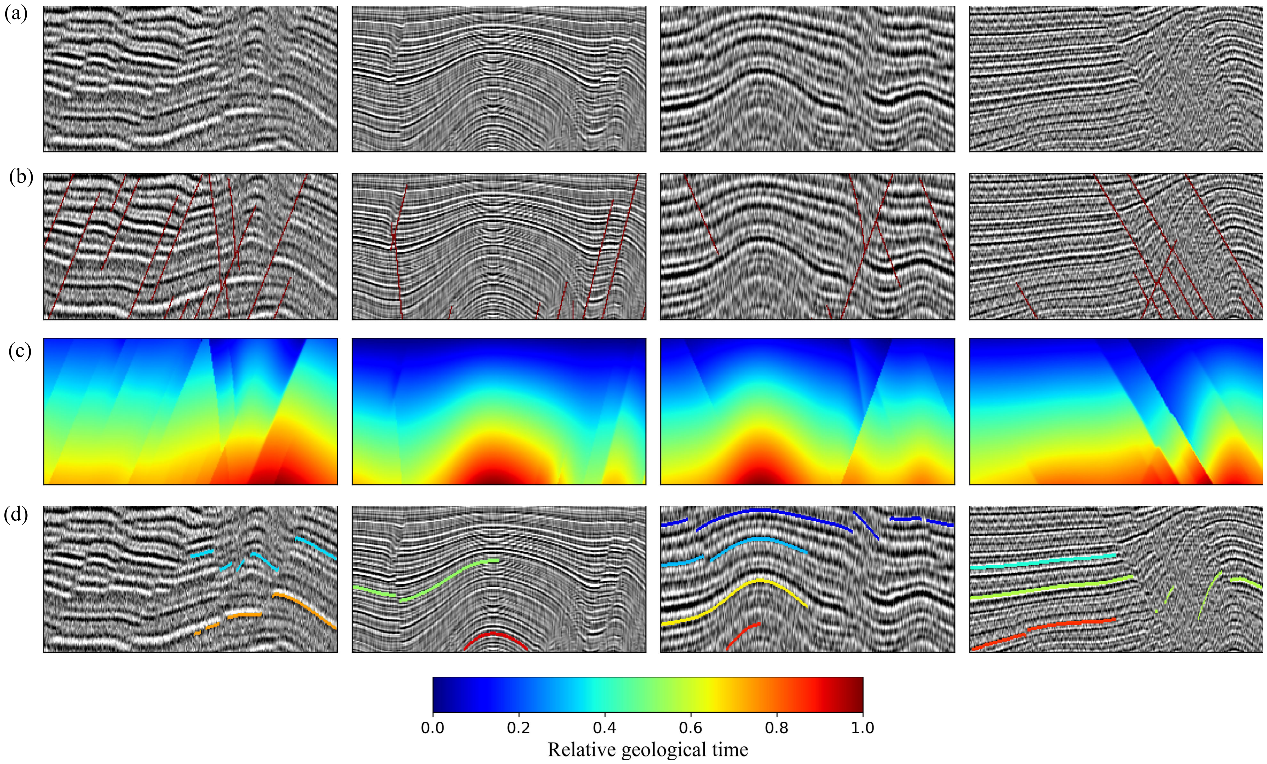

Fig. 5. Four of 10 000 synthetic training samples. The training dataset includes (a) seismic images, (b) corresponding faults, and (c) RGT labels. (d) Horizons segments are randomly extracted from the RGT labels and are input together with seismic images into our neural network to provide constraints.

image, we select the points between the hanging wall or footwall sides as fault labels. The fault label points and other points, which are, respectively, assigned with ones and zeros, form the binary fault label image [Fig. 5(b)] corresponding to the seismic image.

To obtain the corresponding RGT label, an initial flat RGT model is built with vertically monotonic increasing values. Then, we use the same structure parameters, which are chosen for the reflectivity model, to simulate the same folding and faulting structures in this RGT model [Fig. 5(c)]. By using this workflow, we randomly choose the parameters in the above-mentioned description and generate 10 000 pairs of 2-D seismic images and corresponding fault and RGT labeling maps with the size of 128 × 256. Finally, we randomly extract several horizons (RGT contours) [Fig. 5(d)] from each RGT map in every training epoch and use them as part of the network inputs to provide prior constraints of known horizons. In defining a known horizon, we choose a partial or complete contour randomly, because we may not be able to obtain a whole horizon of a field seismic image [63]. In this way, the model is trained to be able to deal with situations of various horizon segments as input.

B. Training Details

We train our MTL network by using 9600 pairs of synthetic seismic images and corresponding RGT and fault images, and another 400 pairs are used for validation. To relieve the effects of the seismic amplitude variations between field and synthetic seismic images, we use the normalization that seismic images are subtracted by their means and divided by their standard deviations. To define the values of horizon segments, every RGT map is normalized by its minimum and maximum, and the range of RGT values is from 0 to 1. We use the pretrained model, which was trained by common RGB images with three channels. To be consistent, we define our inputs with three channels (horizon segments, seismic, and seismic). We set the training epoch to 800 and the batch size to 80. From our experience, a transformer network needs a large amount of training data and a relatively small learning rate (LR) compared with a CNN model to obtain an excellent result, and we use a StepLR with the initial value of 0.001 and decay the LR by 0.5 after every 100 epochs [Fig. 6(c)]. Our proposed network is optimized with the Adam optimizer whose weight decay is set to 0.0001.

The RGT loss curves [Fig. 6(a)] and the fault loss curves [Fig. 6(b)] converge to less than 0.004 and 0.3 after 800 epochs, respectively. As shown in Fig. 7, the reflection features are highly contaminated by noise, and the fault structures are complex [Fig. 7(a)]. However, the predicted RGT [Fig. 7(d)] and fault images [Fig. 7(h)] are almost the same as the labels [Fig. 7(c) and 7(g)]. The contours [Fig. 7(e)] extracted from predicted RGT maps [Fig. 7(d)] represent horizons, which accurately follow seismic reflections, even across the faults [Fig. 7(d)]. As shown in Fig. 7(f), the extracted horizons can also fit the known horizon segments well. This indicates that our network predicts RGT maps confirming the known horizon segments, which enables human interactions in complex field examples to compute reasonable RGT results.

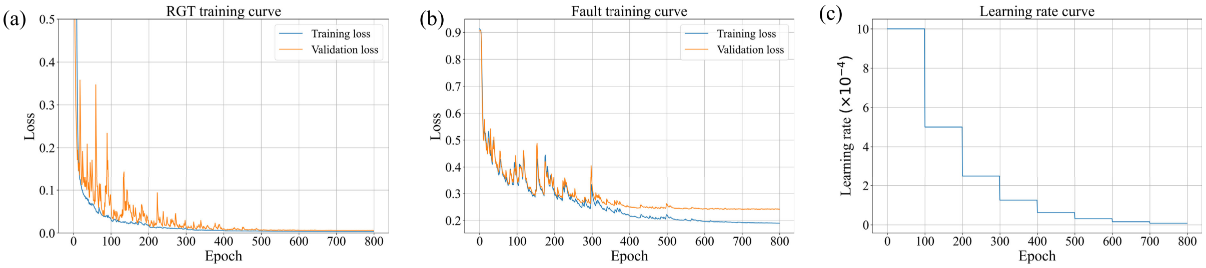

Fig. 6. (a) Training (blue) and validation (orange) loss curves for RGT estimation. (b) Training (blue) and validation (orange) loss curves for fault detection. These two training processes are simultaneously performed. (c) LR curve is updated at every 100 epochs.

TABLE I
COMPARISON OF DIFFERENT NETWORK IN EFFICIENCY

Network	FLOPs ↓ (GB)	Trainable parameters ↓ (MB)
Geng *et al.* [64]	129.739	214.780
Our RGT estimation branch	**7.148**	**61.152**

IV. COMPARISON EXPERIMENTS

In this part, we first perform several field data tests to verify the superiority of our RGT estimation with transformer over the previous CNN-based method [64]. We show that the transformer architecture and horizon constraints significantly improve the RGT estimation, and we also show the improvement by adding an integral layer before outputting the RGT map. Furthermore, with 3-D horizon constraints, we can decompose the 3-D RGT estimation to 2-D predictions slice by slice. Finally, we show the results containing predicted RGT and fault images from the whole MTL network.

A. CNN Versus Transformer

The previous CNN method [64] for 2-D RGT estimation obtains an excellent performance, and we can use it to extract horizons easily from a seismic image. But this method also has some limitations: 1) the network in this method has high costs in computation. After four residual blocks in the encoder part, the number of feature maps steeply rises to 2048. 2) The network in this method has a relatively small receptive field. Maintaining a large receptive field is important for the RGT estimation problem to capture global structural information.

The hierarchical and efficient encoder in our RGT estimation branch helps solve these problems. Floating point operations (FLOPs) and trainable parameters indicate the efficiency and complexity of a network, respectively. As shown in Table I, our RGT estimation branch is much more efficient and lightweight than the one proposed in [64]. Next, we further compare the receptive field and RGT estimation accuracy of our method with the CNN-based method.

A unit in a feature map of a neural network depends on a region of the input image, and the region can be defined as the receptive field for this unit. With a bigger size of the receptive field, a network is able to better extract high-level features and fuse global information more effectively. A large receptive field is important for the task of RGT estimation that requires globally fitting structures in an input seismic image. The receptive field is theoretically dependent on the network architecture (e.g., convolutional kernel size, number of layers, and downsampling) but the effective receptive field (ERF) can also be affected by the training process and the network inputs [65]. Compared with the theoretical receptive field (TRF), ERF is related to the features of the input image, and it can more directly show the ability of a trained network in extracting and fusing global information in practice. Fig. 8(a) is a seismic image fed into CNN [64] and our network with a transformer backbone, and Fig. 8(b) and (c) is the ERFs of the center unit [red points in Fig. 8(b) and (c)] in these two networks. We can observe that the RGT estimation branch of our network shows a larger ERF than CNN. Moreover, the CNN receptive field [Fig. 8(b)] is isotropic in space, while the one of our network [Fig. 8(c)] appears an anisotropic shape reasonably follow the layered structure [denoted by white curves in Fig. 8(c)] and even the dislocation across the fault [denoted by the red line in Fig. 8(c)] in the input seismic image. We think the anisotropic receptive field, with a long-distance extension following seismic structures, would be essential for estimating an accurate RGT result from a seismic image.

We use a field example (128 [vertical] × 256 [inline] samples) in Fig. 9 to demonstrate the advantage of the RGT estimation branch of our network. The reflections in this field seismic image [Fig. 9(a) and (b)] appear significant lateral variations and are dislocated by several faults, which poses challenges to the RGT estimation from the seismic image. In this example, we manually interpret two horizons [solid curves in Fig. 9(a) and (b)] to validate the predictions by networks. The second row of Fig. 9 shows the RGT results predicted by the previous CNN [Fig. 9(c)] and our RGT estimation branch [Fig. 9(d)]. They look quite similar but our result appears sharper discontinuities across the faults denoted by black arrows [Fig. 9(d)]. To better evaluate the CNN-based and our RGT results, from which we extract two horizons (corresponding to the manual ones) and display them with dotted curves in Fig. 9(a) and (b), respectively. The horizons extracted from our RGT results visually better fit the

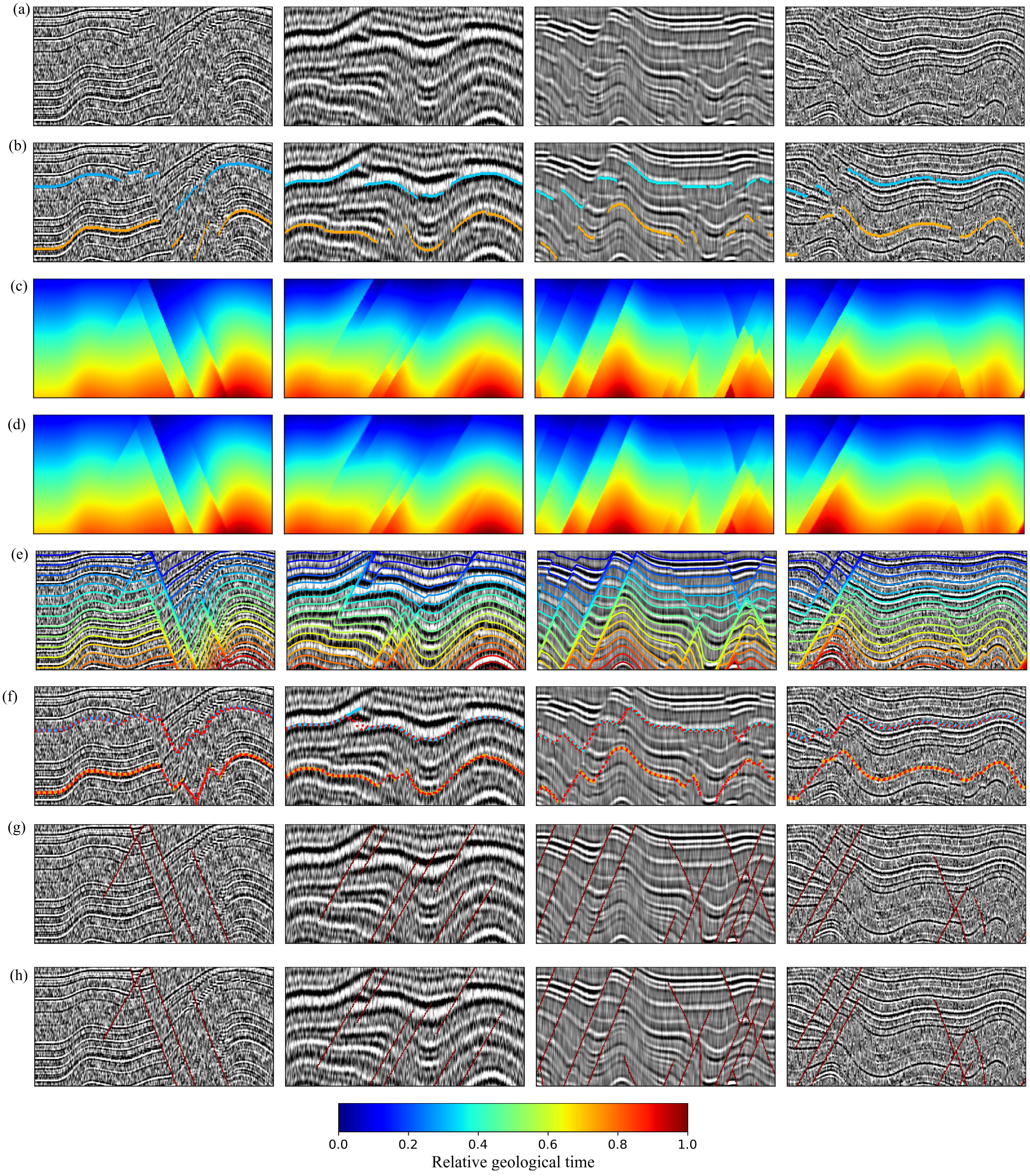

Fig. 7. We show several inputs from the validation dataset and the corresponding results, including (a) seismic images, (b) horizon segments, (c) RGT labels, (d) predicted RGT maps, and (e) contours extracted from (d). (f) Contours (purple dotted curves) extracted from (d) perfectly fit the input horizon segments (b). (h) Fault images, predicted at the same time as the RGT estimation and also fit (g) fault labels well.

manual horizons than the ones extracted from the CNN-based RGT. The quantitative metric HEERV of our result (5.957) is also lower than the CNN result (7.712). The HEERV values indicate that our network is able to obtain a more reasonable horizon and fault interpretation result from a seismic image compared with the CNN-based method. Finally, we can easily

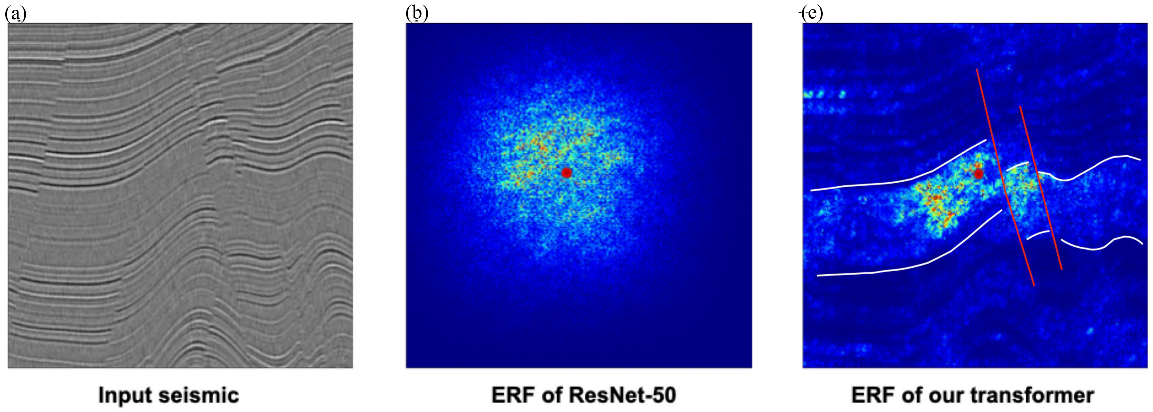

Fig. 8. With the same input seismic image (a), the previous CNN-based method yields an isotropic ERF centered at the target point (denoted by the red dot), while our network generated a geologically more reasonable ERF which is spatially anisotropic and follows seismic reflection and fault structures (write curves and red lines).

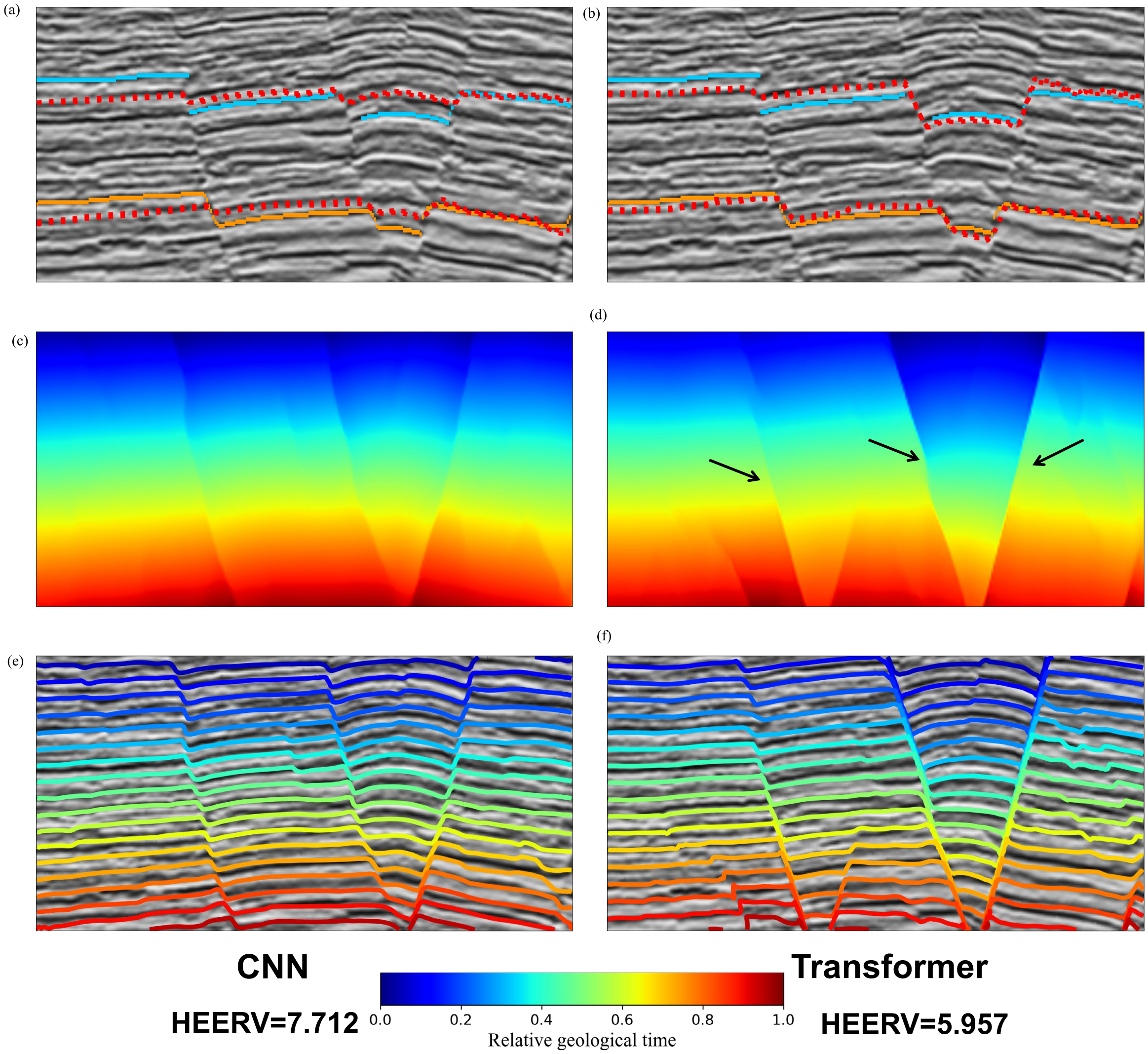

Fig. 9. Previously proposed CNN-based [49] (left column) and our transformer-based (right column) RGT estimation methods are applied to a field example with complex faults. Compared with the results of the CNN-based method, our RGT result appears sharper discontinuities across the faults as denoted by black arrows. In addition, the horizons (dashed curves) extracted from our RGT map better fit the manually interpreted horizons (solid curves), as shown in the first row.

extract any number of contours from the RGT results to obtain a set of dense horizons that follow reflections and dislocate across faults, as shown in Fig. 9(c) and (d).

B. Horizon Constraints

It is a common challenge to compute geologically reasonable horizons or faults from a seismic image with complex

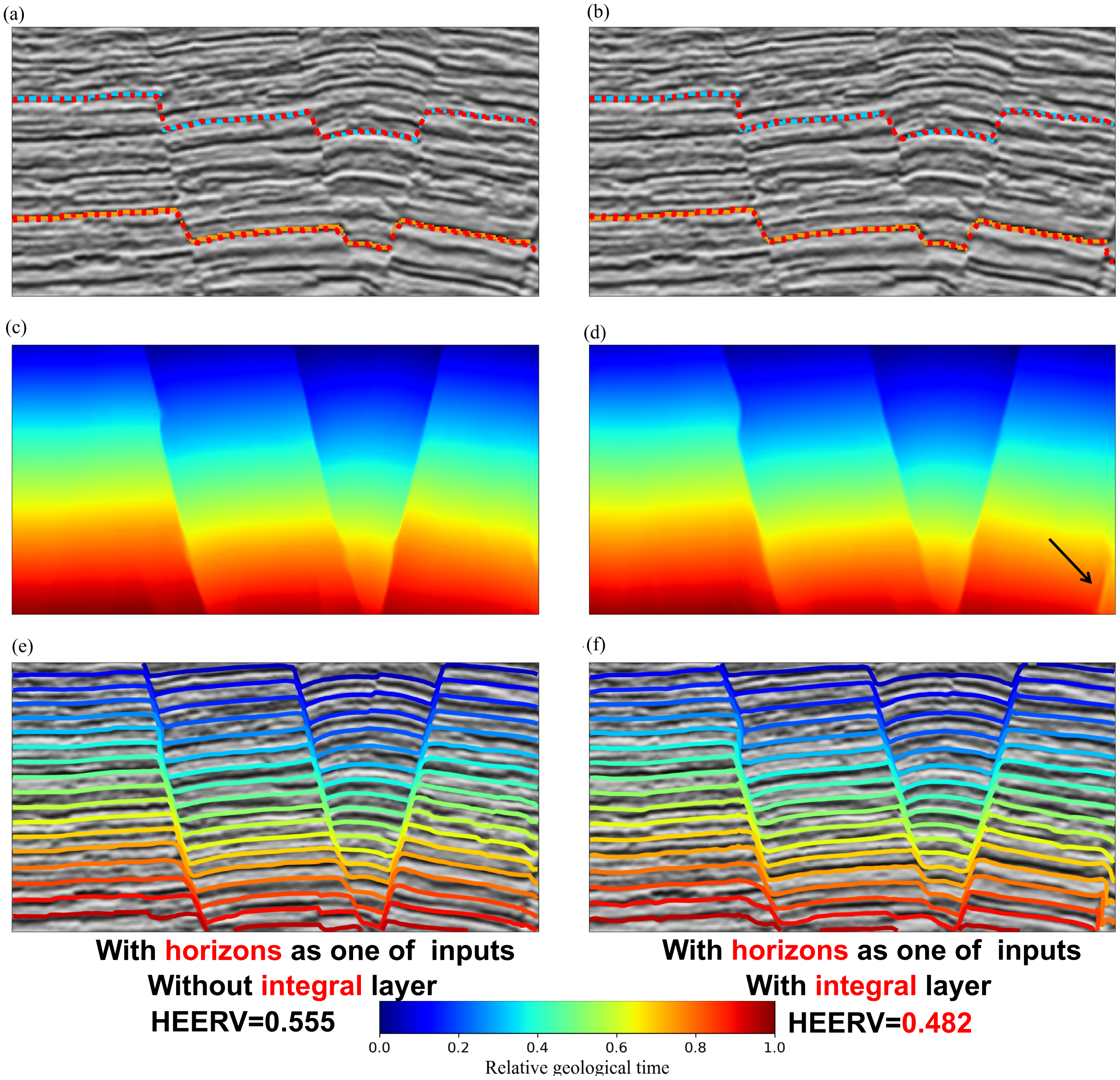

Fig. 10. For the same example in Fig. 9, we use the interpreted horizons (solid curves in the first row) as constraints to improve the RGT estimation (c), from which the extracted horizons or contours more accurately follow seismic reflections (e) and the manual interpretation (a) than those in Fig. 9. We add an integral layer, as discussed in [66], at the end of the RGT estimation network to further improve the RGT estimation and (the black arrow in d) obtain a lower HEERV (0.482).

structures, artifacts, and low SNR for existing automethods [19], [20], [24], [50], [64]. In these cases, experts typically need to interpret with their prior knowledge of geological background instead of simply tracking seismic geometric features which may be unclear or even geologically untrue due to limited resolution and artifacts of seismic imaging. Similarly, it is necessary to incorporate prior expert geological knowledge into automatic seismic structural interpretation methods, including our MTL method, to deal with complicated cases. Inputting manually interpreted horizons into our network is an effective way to incorporate prior constraints to help estimate reasonable RGT maps from seismic images with complex fault systems, poor data quality, and unconformities. Moreover, with the constraints of 3-D horizons, we are able to decompose a 3-D RGT estimation task into computationally cheap 2-D predictions slice by slice in a 3-D seismic volume but still maintain lateral consistency among the 2-D RGT results.

1) Complex Fault Systems: For the same example in Fig. 9, we input the two manually interpreted horizons together with the 2-D seismic image into our network and obtain an RGT result and its associated horizons (contours) in the left column of Fig. 10. As shown in Fig. 10(a), the horizons (dotted curves) extracted from the estimated RGT map fit well with the manual ones used as constraints, which indicates that the network performs RGT estimation honoring the input horizon constraints as what we expect. As shown in Fig. 10(e), a set of dense horizons (contours) extracted from the RGT map follows seismic reflections, which indicates the network also tends to fit seismic structures in estimating the RGT map. Moreover, the input horizon constraints not only ensure accurate RGT estimation near the horizon positions but also improve and stabilize the RGT estimation in other areas, which we will illustrate in more examples in the following sections.

As discussed in [66], we can further improve the performance of our network by adding a layer of vertical integration operation before the output layer as denoted by the dark green arrow in the RGT estimation branch in Fig. 3. With this layer, we assume the network learns to estimate RGT derivatives from the inputs and the integration converts the derivatives to an RGT map. Such integration will accumulate prediction errors, if exist, and therefore, emphasize the misfit between the predictions and labels during the training process, which would be helpful to better train the network. In addition, the integration is a global operator that is helpful for the network to fuse global features for the RGT estimation. Although the RGT results and their associated contours (horizons) of the network without (left column of Fig. 10) and with (right column) the integration layer are visually similar, the HEERV metric of the later is relatively smaller (0.482) than the former (0.555). In addition, the latter [Fig. 10(f)] can accurately produce a discontinuity across the fault [denoted by the black arrow in Fig. 10(f)], while the former missed that.

2) Poor Data Quality: In addition to complex faults, the horizon can also guide the network to track reflections in a seismic image with low SNR and artifacts. The first row of Fig. 11 shows a low SNR dataset (128 [vertical] × 256 [inline] samples) from the Soda Lake geothermal field, where some of the reflections are almost invisible and, therefore, are hard to track for an automatic horizon interpretation method. As shown in Fig. 11(a), if we use only the seismic image as the input, the horizons computed by our network are quite messy and do not fit the manual ones [solid curves in Fig. 11(a)], which indicates that our network cannot compute a reasonable RGT map from a seismic image with unclear reflections. Therefore, we gradually add more and more horizon segments as constraints to illustrate how the horizon constraints affect the RGT estimation from a seismic with low data quality. As denoted by the solid curves of the second column of Fig. 11(a)–(f), we track eight horizons by manually picking. By inputting one horizon [solid curve in the first image of Fig. 11(b)] as constraints into the network, we are able to compute much more stable horizons [curves in the third image of Fig. 11(b)], compared with those in Fig. 11(a). However, the computed horizons still do not fit well with the manual ones except near the constraint horizon, as shown in the second image of Fig. 11(b). As shown in Fig. 11(c), we further add two more horizons as constraints and obtain more reasonable horizons that fit well with most of the interpreted horizons, even in areas far away from the constraint horizons. To further improve the result, we add one more horizon in the region with blurry reflections [the green one in the first image of Fig. 11(d)], which leads to geologically reasonable horizons everywhere in the seismic image.

In addition, we find that the result constrained by four horizons [Fig. 11(d)] is almost identical to the result constrained by all eight horizon segments [Fig. 11(f)]. This indicates that the constraint of these four horizons is sufficient for the network to estimate an accurate RGT map from this seismic image with low data quality. In some situations, experts may not interpret a complete horizon, which means that only partially interpreted horizons or horizon segments are available to be used as constraints. We mimic this case in Fig. 11(e), where we use only the left part of the green horizon as constraints. In this case, we still obtain accurate horizons [Fig. 11(e)], which are almost identical to those computed with the constraint of the entire green horizon in Fig. 11(d). This test indicates that our method allows using incompletely interpreted horizons or horizon segments as constraints to improve the RGT estimation, which facilitates its applications in practice.

In our experiments, we use two methods to determine the RGT values of horizon patches. In most cases, we calculate the relative locations of horizon patches within the vertical space, using values ranging from 0 to 1 as the RGT values. If the layer in the area is inclined or folded significantly, the RGT values may be unreasonable. In this instance, we may need to interpret complete horizons through the entire survey to decide reasonable RGT assignments. To solve this problem, we need to first manually determine the geologic time order of the constraint horizons or horizon patches based on the geologic background of the survey or the stratigraphic sequence that appears in the seismic volume. We then assign some reasonable RGT values to the horizons in order. Our method is designed to automatically compute an RGT map with an adaptive range of values honoring the RGT assignments at the horizons, and we only need to make sure the RGT assignments are in a geologically reasonable sequence, that is the assigned RGT values of vertically shallower horizons should be relatively smaller than the deeper ones. Note that here we assume no overturning layers appear in the seismic images.

3) Unconformities: Unconformity [e.g., the region between dark blue and light green solid curves in Fig. 12(a)] can be regarded as a boundary where seismic reflectors terminate. It is indicated by vertical geologic time discontinuity in the corresponding RGT map. The reflector terminations or geologic time discontinuities near unconformities pose another type of challenge for the task of automatic horizon interpretation in a seismic image. It is also a challenge for our network trained by a dataset without any unconformity features. By inputting the seismic images (128 [vertical] × 256 [inline] samples) [Fig. 12(a)] into our network, we obtain the corresponding RGT maps in Fig. 12(b), where we do not observe any vertical discontinuities near the unconformities as expected. To be able to reasonably deal with the unconformities, one potential way is to retrain the network by seismic images with unconformities and the corresponding RGT labels with discontinuities at the unconformities. Here, however, we provide a more convenient way by incorporating some horizon constraints without the need of retraining the network.

We manually interpret four horizon segments [solid curves in Fig. 12(a)] and use them as constraints in our network for the RGT estimation. Note that the horizons used as constraints need to be geologically isochronous. Therefore, the light blue and cyan horizons are interpreted to be terminated at unconformities where the geologic time is varying. Fig. 12(c) shows the

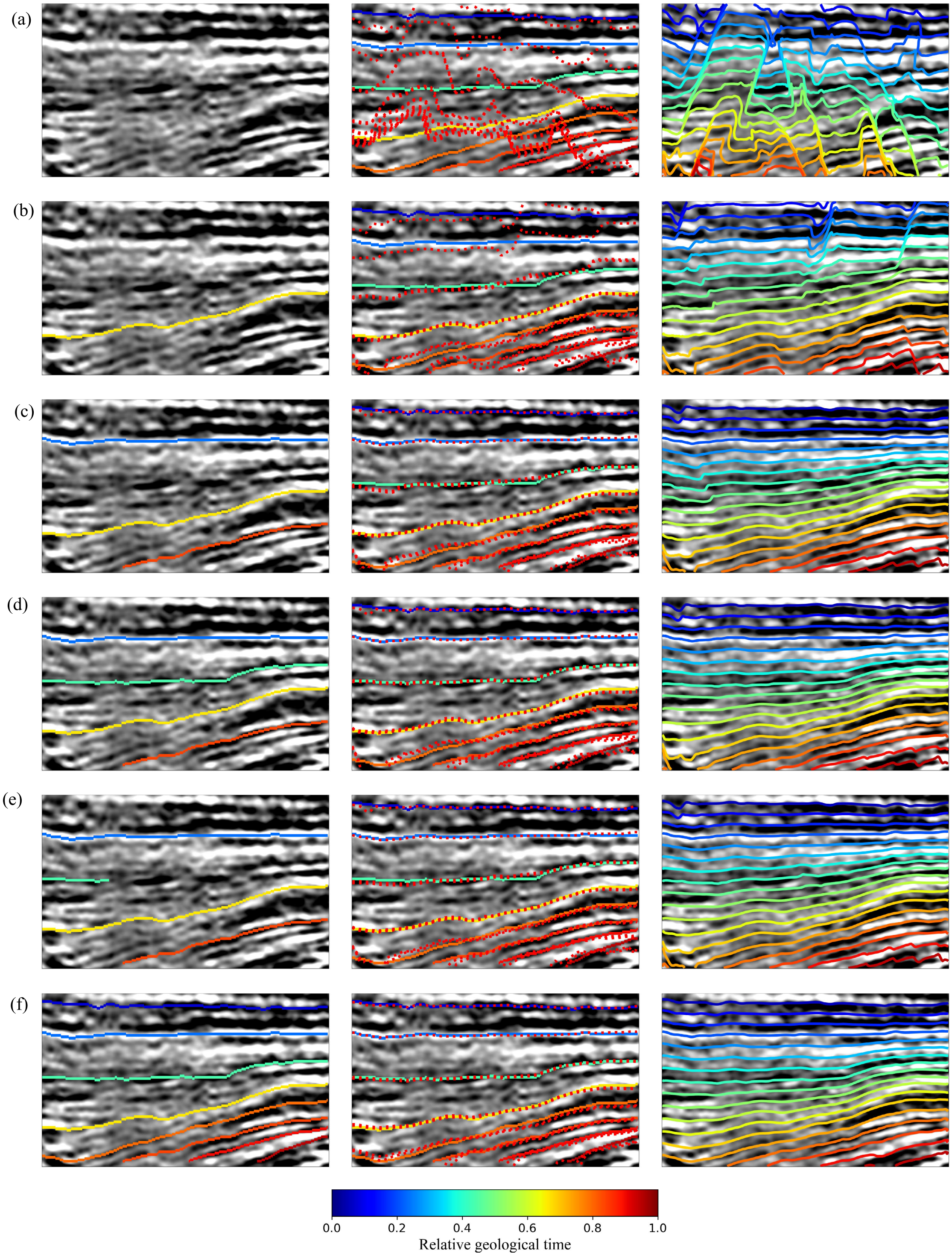

Fig. 11. Comparison study of using different numbers of horizons as constraints. In the first column, a seismic image with a different number of horizons is input to our network for RGT estimation. In (a)–(d), as we increase the number of constraint horizons from 0 to 4, the contours or horizons, extracted from the estimated RGT maps, become increasingly better fit to the interpreted horizons (solid curves in the second column) and seismic reflections (third column). The contours (d) from the RGT map constrained by four horizons are similar to those (f) constrained by eight horizons. Compared with using constraints from all four complete horizons (d), we can still obtain a quite similar RGT map by making one [the left part of the green horizon in (e)] of the constraint horizons incomplete.

RGT maps computed by using the horizon constraints, where we observe sharp vertical discontinuities near the unconformities, as denoted by the yellow and red arrows. Fig. 12(d) shows horizons (contours) extracted from the estimated RGT maps, from which we observe geologically reasonable horizon onlaps and downlaps at the unconformities denoted by the yellow and red arrows, respectively. This example shows that using horizon constraints, our network is able to reasonably estimate the geologic time discontinuities corresponding to the onlap and downlap of the unconformity, although there are

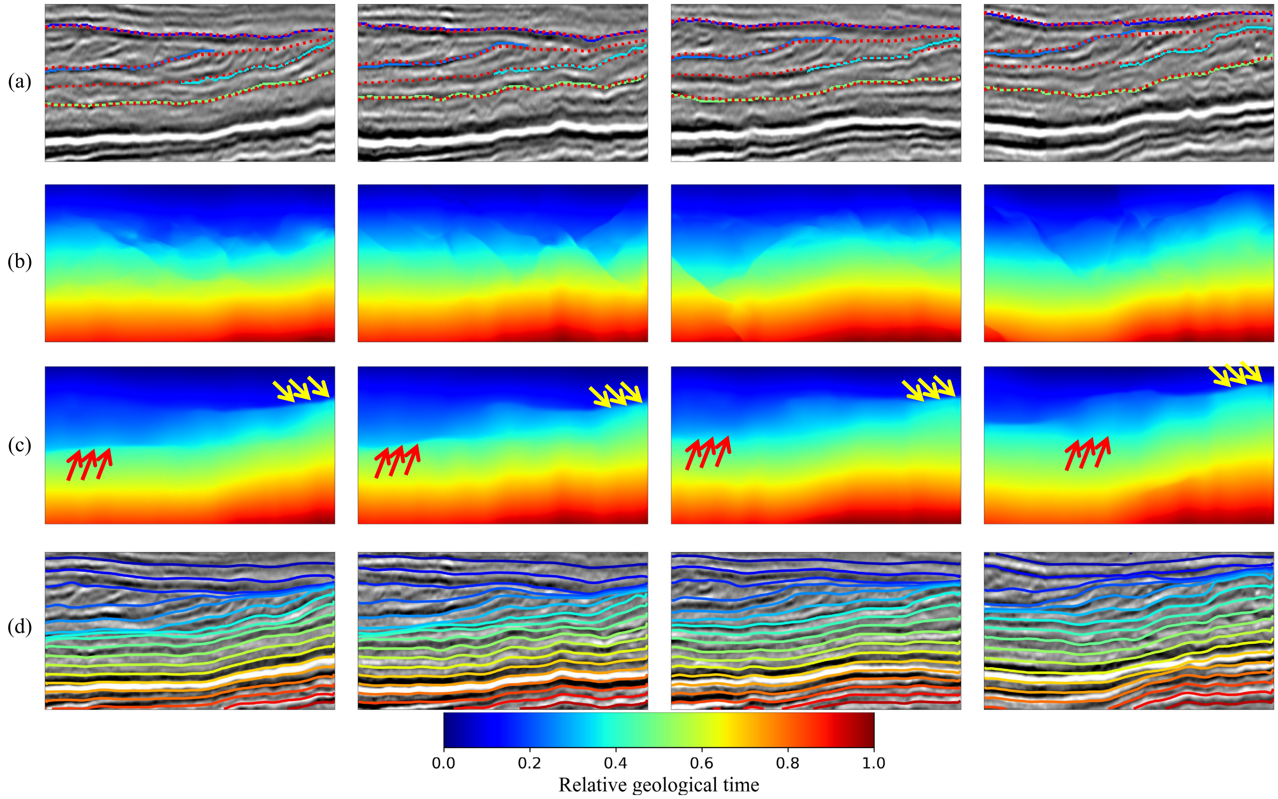

Fig. 12. RGT estimation in (a) seismic images with unconformities. The network does not work well to estimate (b) RGT maps from seismic images with unconformities because it is trained with synthetic datasets without unconformity features. However, we can input some horizons (solid curves in the first row) into the network as constraints to improve the (c) RGT estimation and (d) horizon extraction, where we observe reasonable discontinuities near the unconformities as denoted by yellow and red arrows.

no structures of unconformities in our training dataset. It is hard to prepare a truly diverse training dataset that contains all kinds of geologic structures or features, which typically limits the generalization of a trained network in various field datasets. However, the idea of using horizon constraints provides a convenient way to involve human interactions in the inference step of a trained network and effectively improve its generalization in various structures that may not be included in the training datasets.

4) Obtain a 3-D RGT Volume From 2-D Predictions: Field seismic data today are mostly in 3-D and it is necessary for an algorithm to be able to deal with 3-D volumes. However, training a 3-D deep neural network with 3-D convolutional kernels and feature maps is much more time consuming than training a 2-D network, even using relatively small volumes of training data. In addition, predicting an RGT volume from a large 3-D seismic volume is also expensive in computational time and memory for a trained network, which limits its applications in practice. Bi et al. [50] solve this problem by dividing a large 3-D volume into smaller subvolumes and making a prediction for each subvolume, which requires a complex postprocessing step to merge the predicted subvolumes of RGT to obtain a large RGT volume without merging artifacts.

We propose a more efficient and practical way to compute a 3-D RGT volume from a seismic volume by 2-D predictions slice-by-slice in the inline or crossline directions. The main challenge of this way is to maintain lateral consistency among these independent 2-D predictions. For example, Fig. 13(b) shows a 3-D RGT volume (128 [vertical] × 256 [inline] × 150 [crossline] samples) obtained by directly assembling 2-D predictions, where we observe obvious unreasonable lateral jumps in the predicting direction. We propose to maintain lateral consistency among the slice-by-slice 2-D predictions by introducing the lateral control of 3-D horizons. Specifically, for each 3-D horizon surface used for control, we assign the same RGT value to all points on the 3-D surface, as shown in Fig. 13(c). This ensures the horizon constraints input into the network for each 2-D prediction are consistent and, therefore, makes sure the RGT values predicated at the constraint horizons in all 2-D predictions are identical. In this way, we are able to compute laterally consistent 2-D RGT predictions that are directly assembled together in the predicting direction to form a reasonable 3-D RGT result, as shown in Fig. 13(d). In addition to the lateral consistency, the horizon constraints also improve the RGT estimation near the unconformities where we observe obvious vertical RGT discontinuities. As shown in Fig. 13(e) and (f), the 3-D horizons extracted from the 3-D RGT accurately follow seismic reflections and some of them reasonably terminate at the unconformity. This example demonstrates that we are able to decompose a 3-D RGT prediction task into much more efficient 2-D predictions but still maintain lateral consistency by introducing 3-D horizon constraints.

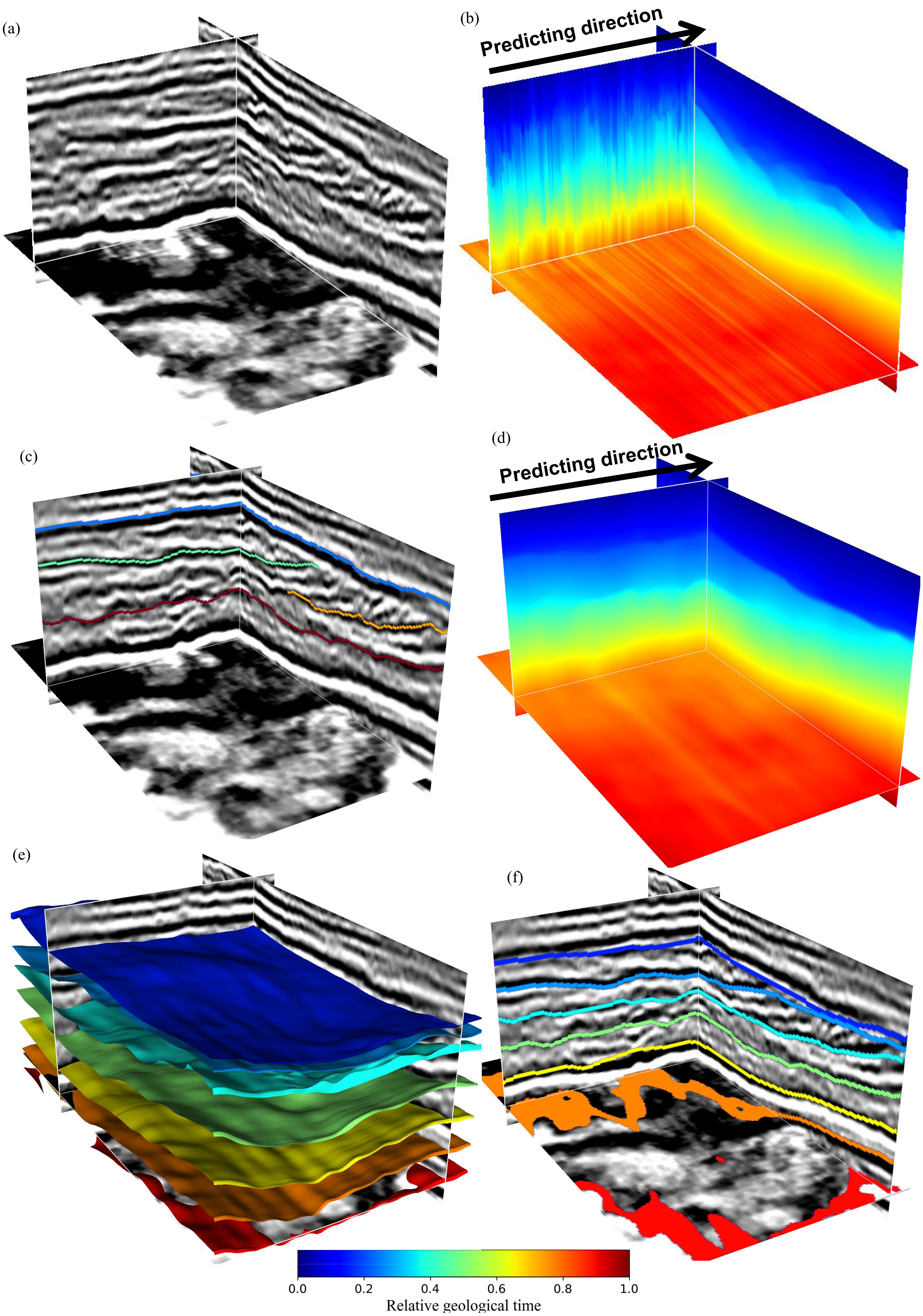

Fig. 13. Computing a 3-D RGT volume from a (a) 3-D seismic image by independent 2-D predictions slice-by-slice yields unreasonable lateral jumps in the (b) predicting direction. We use 3-D horizon surfaces [solid curves in (c)] as constraints to maintain lateral consistency of the 2-D predictions to obtain a (d) reasonable 3-D RGT volume. The 3-D horizons [(e) and (f)] extracted from the RGT volume fit seismic reflections well and reasonably terminate at the unconformity.

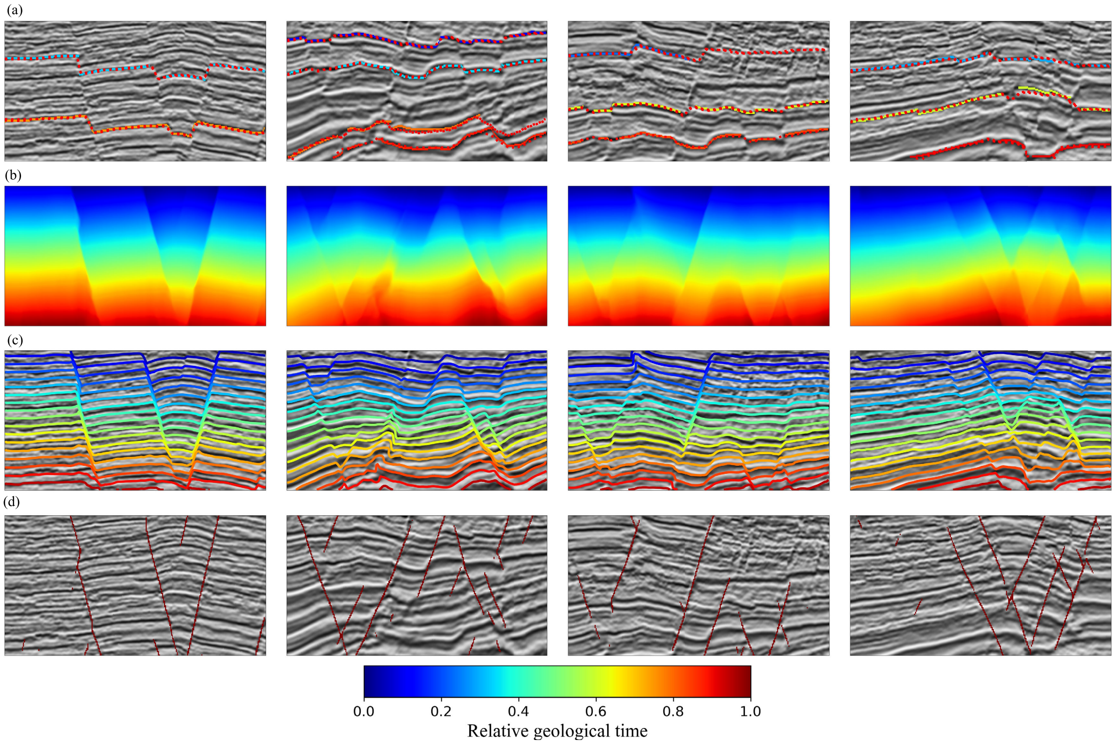

Fig. 14. There are obvious sharp edges in the (b) RGT maps, which correspond to the faults in (d) fault predictions, and the (c) contours extracted from (b) RGT maps fit seismic reflectors well.

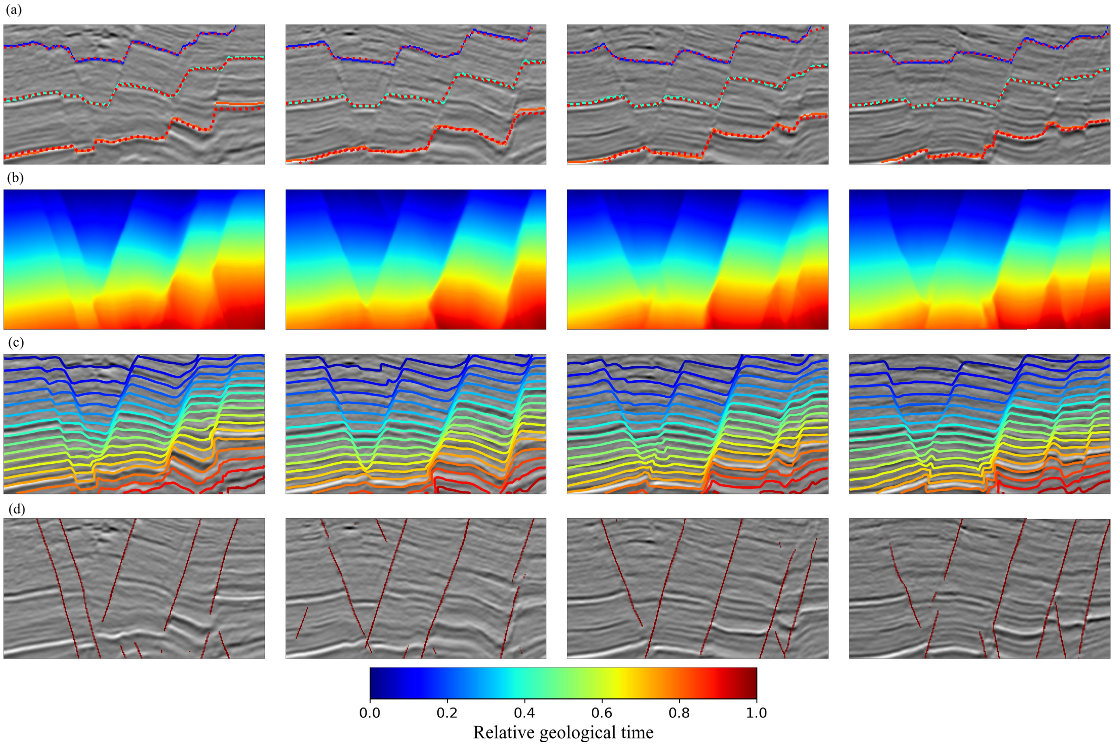

Fig. 15. By using our MTL network, we simultaneously compute reasonable predictions of RGT estimation and fault detection from the (a) 2-D seismic images and horizon constraints. In the (b) estimated RGT maps, we observe sharp edges or discontinuities, which correspond to dislocations in the (c) extracted horizons (contours) and (d) high fault probability in the fault images.

C. Simultaneous RGT and Fault Prediction

In this section, we use the same seismic image (Fig. 14) as in Figs. 9 and 10 to demonstrate the effectiveness of the simultaneous RGT and fault prediction by the complete MTL network with two branches. The inputs for the branch of fault detection consist of the second derivatives of the predicted

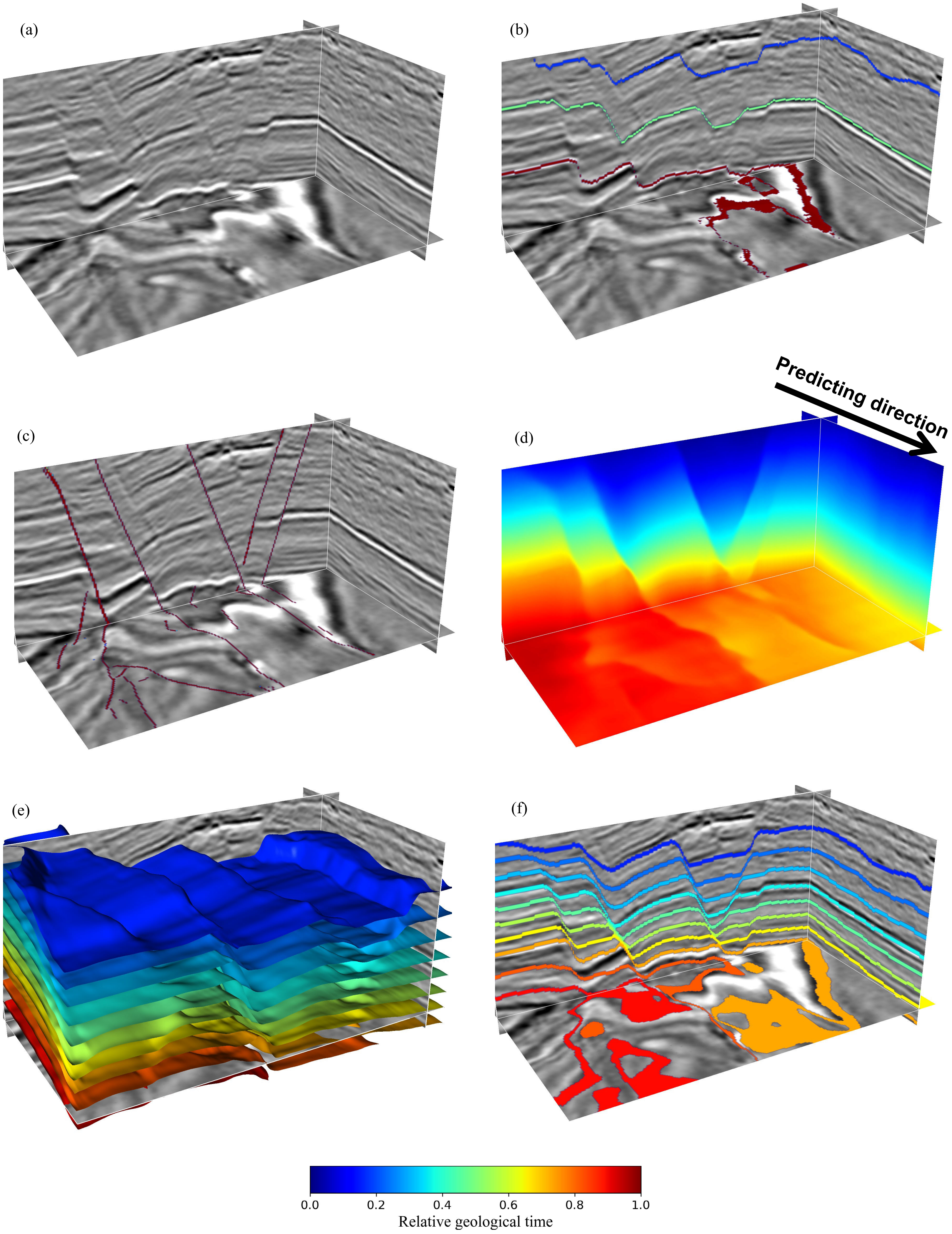

Fig. 16. With the (b) 3-D horizon constraints, we compute 2-D fault images and RGT maps (Fig. 15) slice-by-slice in a (a) 3-D seismic volume and directly assemble these predictions in the predicting direction to obtain 3-D volumes of (c) fault probabilities and (d) RGT values. Due to the constraints of 3-D horizons in the 2-D predictions, the (d) assembled 3-D RGT volume maintains lateral consistency in the predicting direction, and the iso-surfaces [(e) and (f)] extracted from the RGT volume fit seismic reflectors well. (f) Horizon dislocations correspond to the high fault probabilities in the (c) fault volume.

RGT map and features concatenated from the transformer encoder. Because the RGT map is a scalar field globally fitting the seismic structures, the RGT derivatives provide mostly detailed seismic structural features to help obtain a noise-free and continuous fault detection. The feature maps from the transformer encoder contain more detailed structure features to help detect small-scale faults. The fault detection branch network merges both the inputs and obtains a fault map with multi-scale faults as shown in Fig. 14(d), where the fault features are clean and continuously trackable. During the training process, the gradients of the fault detection loss function are backpropagated from the fault detection branch back to the RGT estimation branch, which helps the network to produce discontinuities across faults in the RGT result. As shown in the estimated RGT results [Fig. 14(b)] and their associated horizons or contours [Fig. 14(c)], we observe sharp RGT discontinuities and horizon dislocations across the detected fault positions.

V. Application

In this section, we apply optimal schemes, based on the above-mentioned comparison experiments, to a 3-D seismic dataset with faults. First, we use the horizon segments and seismic images [Fig. 15(a)] as inputs to obtain RGT maps [Fig. 15(b)] and fault images [Fig. 15(c)]. The contours extracted from predicted RGT maps can fit interpreted horizons well [Fig. 15(a)], and these contours [Fig. 15(c)] accurately follow continuous seismic reflections and dislocations across faults. The fault detections in Fig. 15(d) are consistent with the sharp edges that appear in the predicted RGT maps, and we think it benefited from the MTL scheme of simultaneous RGT estimation and fault detection.

With the constraints of the three 3-D horizons, the independently computed 2-D predictions can be directly assembled in the predicting direction to obtain laterally consistent 3-D volumes (128 [vertical] × 256 [inline] × 150 [crossline] samples) of RGT values and fault probabilities. The 3-D RGT volume [Fig. 16(d)] and the corresponding iso-surfaces [Fig. 16(f)] accurately fit the reflections in the seismic volume [Fig. 16(a)]. The faults in the seismic volume are also accurately detected, as shown in Fig. 16(c). Finally, the iso-surfaces extracted from RGT volume [Fig. 16(d)] correspond to 3-D horizons, as shown in Fig. 16(e).

VI. Conclusion

We propose an MTL network for simultaneous RGT estimation and fault detection from a seismic image with the interpreted horizons as prior constraint information. The predicted RGT map can be further used to extract any horizon in the seismic image. The transformer encoder in the RGT estimation branch of the MTL network can capture global structural information from the inputs, and SFF can fuse the global and local features from the transformer encoder and CNN decoder. The simple end-to-end CNN uses the fault information from the second gradients of the predicted RGT and the feature maps from the transformer encoder to detect faults. By sharing feature maps and network parameters in the training, these two tasks could provide controls for each other. Two loss functions constrained by prior horizon information are used in the RGT estimation branch, and the loss function to fit the imbalanced distribution of the positive and negative samples is used in the fault detection branch. We carry out comprehensive comparison studies to optimally design our schemes, which include the architectures of networks, the addition of prior horizon information, predicted results from 2-D to 3-D, and MTL. In multiple filed applications, our method can robustly predict reasonable RGT maps even in seismic images containing complex fault systems, unconformities, and blurry reflections, and at the same time compute a fault image with clean and continuous fault features. Finally, this method significantly improves the efficiency and saves the memory of 3-D RGT estimation by computing and merging parallel 2-D results slice by slice with the constraints of 3-D horizons.

Some limitations still remain in our method. First, our network model was trained on training samples with a size of 128 × 256 pixels. In practical data applications, it is often necessary to reshape seismic data into the same size through interpolation or downsampling to obtain optimal results. When the actual data size is much larger than 128 × 256, we need to perform block-wise predictions and require additional post-processing to merge the block-wise results. Second, due to the large demand for memory and computational resources, we only tested the implementation of our method in two dimensions in this article. This 2-D implementation can be applied to process a 3-D seismic volume and obtain a reasonable result with lateral consistency by introducing constraints of 3-D horizons. However, we believe that the implementation of our method in three dimensions is still valuable for better analyzing and fitting 3-D structures. The extension of our method from two dimensions to three dimensions is straightforward in algorithm implementation, but optimization of the network's computational efficiency and memory requirements is needed given the current performance of GPUs.

References

[1] H. Posamentier, R. Davies, J. Cartwright, and L. Wood, "*Seismic Geomorphology—An Overview*, vol. 277, R. J. Davies, H. W. Posamentier, L. J. Wood, J. A. Cartwright, Eds. London, U.K.: Geological Society of London Special Publication, 2007, pp. 1–14.

[2] M. D. Patel and G. A. McMechan, "Building 2-D stratigraphic and structure models from well log data and control horizons," *Comput. Geosci.*, vol. 29, no. 5, pp. 557–567, Jun. 2003.

[3] P. R. Vail, R. G. Todd, and J. B. Sangree, "Seismic stratigraphy and global changes of sea level: Part 5. Chronostratigraphic significance of seismic reflections: Section 2. Application of seismic reflection configuration to stratigraphic interpretation," in *M 26: Seismic Stratigraphy–Applications to Hydrocarbon Exploration*, vol. 26. AAPG Memoirs, 1977, pp. 99–116.

[4] G. Caumon, P. Collon-Drouaillet, C. Le Carlier de Veslud, S. Viseur, and J. Sausse, "Surface-based 3D modeling of geological structures," *Math. Geosci.*, vol. 41, no. 8, pp. 927–945, 2009.

[5] J. C. Rivenæs, C. Otterlei, E. Zachariassen, C. Dart, and J. Sjøholm, "A 3D stochastic model integrating depth, fault and property uncertainty for planning robust wells, Njord field, offshore Norway," *Petroleum Geosci.*, vol. 11, no. 1, pp. 57–65, Jan. 2005.

[6] N. J. Kusznir and G. D. Karner, "Continental lithospheric thinning and breakup in response to upwelling divergent mantle flow: Application to the Woodlark, Newfoundland and Iberia margins," *Geological Soc., London, Special Publications*, vol. 282, no. 1, pp. 389–419, Jan. 2007.

[7] K. J. Marfurt, R. L. Kirlin, S. L. Farmer, and M. S. Bahorich, "3-D seismic attributes using a semblance-based coherency algorithm," *Geophysics*, vol. 63, no. 4, pp. 1150–1165, 1998.

[8] K. J. Marfurt, V. Sudhaker, A. Gersztenkorn, K. D. Crawford, and S. E. Nissen, "Coherency calculations in the presence of structural dip," *Geophysics*, vol. 64, pp. 104–111, Jan./Feb. 1999.

[9] Y. Lou, B. Zhang, R. Wang, T. Lin, and D. Cao, "Seismic fault attribute estimation using a local fault model," *Geophysics*, vol. 84, no. 4, pp. O73–O80, Jul. 2019.

[10] Y. Lou, B. Zhang, P. Yong, H. Fang, Y. Zhang, and D. Cao, "Semi-automatic fault-surface generation and interpretation using topological metrics," *Geophysics*, vol. 86, no. 3, pp. O13–O27, May 2021.

[11] P. P. Van Bemmel and R. E. Pepper, "Seismic signal processing method and apparatus for generating a cube of variance values," U.S. Patent 6 151 555, Nov. 21, 2000.

[12] S. Al-Dossary and K. J. Marfurt, "3D volumetric multispectral estimates of reflector curvature and rotation," *Geophysics*, vol. 71, no. 5, pp. P41–P51, Sep. 2006.

[13] D. Hale, "Methods to compute fault images, extract fault surfaces, and estimate fault throws from 3D seismic images," *Geophysics*, vol. 78, no. 2, pp. O33–O43, Mar. 2013.

[14] X. Wu and D. Hale, "3D seismic image processing for faults," *Geophysics*, vol. 81, no. 2, pp. IM1–IM11, Mar. 2016.

[15] S. I. Pedersen, T. Skov, A. Hetlelid, P. Fayemendy, T. Randen, and L. Sønneland, "New paradigm of fault interpretation," in *Proc. 73rd Annu. Int. Meeting, SEG, Expanded Abstr.*, 2003, pp. 350–353.

[16] X. Wu and S. Fomel, "Automatic fault interpretation with optimal surface voting," *Geophysics*, vol. 83, no. 5, pp. O67–O82, Sep. 2018, doi: 10.1190/geo2018-0115.1.

[17] H. Di, Z. Li, H. Maniar, and A. Abubakar, "Seismic stratigraphy interpretation by deep convolutional neural networks: A semisupervised workflow," *Geophysics*, vol. 85, no. 4, pp. WA77–WA86, Jul. 2020.

[18] T. J. Stark, "Unwrapping instantaneous phase to generate a relative geologic time volume," in *Proc. 73rd Annu. Int. Meeting, SEG, Expanded Abstr.*, 2003, pp. 1707–1710.

[19] X. Wu and G. Zhong, "Generating a relative geologic time volume by 3D graph-cut phase unwrapping method with horizon and unconformity constraints," *Geophysics*, vol. 77, no. 4, pp. O21–O34, Jul. 2012.

[20] A. M. Figueiredo, M. Gattass, and F. Szenberg, "Seismic horizon mapping across faults with growing neural gas," in *Proc. 10th Int. Congr. Brazilian Geophys. Soc. EXPOGEF*, Rio de Janeiro, Brazil, Nov. 2007, pp. 1476–1481, doi: 10.1190/sbgf2007-288.

[21] A. M. Figueiredo, P. M. Silva, M. Gattass, F. B. Silva, and R. L. Milidiú, "A seismic faces analysis approach to map 3D seismic horizons," in *Proc. 84th Annu. Int. Meeting, SEG, Expanded Abstr.*, 2014, pp. 1501–1505, doi: 10.1190/segam2014-1382.1.

[22] A. Figueiredo, F. Silva, P. Silva, L. O. de Martins, R. L. Milidiú, and M. Gattass, "A clustering-based approach to map 3D seismic horizons," in *Proc. 14th Int. Congr. Brazilian Geophys. Soc. EXPOGEF*. Rio de Janeiro, Brazil: Brazilian Geophysical Society, Aug. 2015, pp. 1166–1170.

[23] P. Bakker, L. J. van Vliet, and P. W. Verbeek, "Edge preserving orientation adaptive filtering," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Jan. 1999, p. 540.

[24] X. Wu and X. Janson, "Directional structure tensors in estimating seismic structural and stratigraphic orientations," *Geophys. J. Int.*, vol. 210, no. 1, pp. 534–548, 2017.

[25] S. Fomel, "Applications of plane-wave destruction filters," *Geophysics*, vol. 67, no. 6, pp. 1946–1960, Nov. 2002.

[26] K. J. Marfurt, "Robust estimates of 3D reflector dip and azimuth," *Geophysics*, vol. 71, no. 4, pp. P29–P40, Jul. 2006.

[27] Y. Yu, C. Kelley, and I. Mardanova, "Volumetric seismic dip and azimuth estimation with 2D log-Gabor filter array," in *Proc. 83rd Annu. Int. Meeting, SEG, Expanded Abstr.*, 2013, pp. 1357–1362.

[28] E. Arias, "Estimating seismic reflection slopes," M.S thesis, Colorado School of Mines, Golden, CO, USA, 2016.

[29] X. Wu and S. Fomel, "Least-squares horizons with local slopes and multigrid correlations," *Geophysics*, vol. 83, no. 4, pp. IM29–IM40, Jul. 2018.

[30] Y. Lou, B. Zhang, H. Fang, D. Cao, K. Wang, and Z. Huo, "Simulating the procedure of manual seismic horizon picking," *Geophysics*, vol. 86, no. 1, pp. O1–O12, Jan. 2021.

[31] S. Luo and D. Hale, "Unfaulting and unfolding 3D seismic images," *Geophysics*, vol. 78, no. 4, pp. O45–O56, Jul. 2013.

[32] X. Wu, S. Luo, and D. Hale, "Moving faults while unfaulting 3D seismic images," *Geophysics*, vol. 81, no. 2, pp. IM25–IM33, Mar. 2016.

[33] X. Wu and D. Hale, "Horizon volumes with interpreted constraints," *Geophysics*, vol. 80, no. 2, pp. IM21–IM33, Mar. 2015.

[34] T. J. Stark, "Relative geologic time (age) volumes—Relating every seismic sample to a geologically reasonable horizon," *Lead. Edge*, vol. 23, no. 9, pp. 928–932, Sep. 2004.

[35] F. Qayyum, C. Betzler, and O. Catuneanu, "The wheeler diagram, flattening theory, and time," *Mar. Petroleum Geol.*, vol. 86, pp. 1417–1430, Sep. 2017.

[36] H. Zeng, X. Zhu, R. Zhu, and Q. Zhang, "Guidelines for seismic sedimentologic study in non-marine postrift basins," *Petroleum Explor. Develop.*, vol. 39, no. 3, pp. 295–304, Jun. 2012.

[37] L. Grose, G. Laurent, L. Aillères, R. Armit, M. Jessell, and G. Caumon, "Structural data constraints for implicit modeling of folds," *J. Structural Geol.*, vol. 104, pp. 80–92, Nov. 2017.

[38] S. Bader, K. Spikes, and S. Fomel, "Missing well-log data prediction using Bayesian approach in the relative-geologic time domain," in *SEG Technical Program Expanded Abstracts 2018*. Society of Exploration Geophysicists, 2018, pp. 804–808.

[39] H. Zeng, M. M. Backus, K. T. Barrow, and N. Tyler, "Stratal slicing—Part 1: Realistic 3-D seismic model," *Geophysics*, vol. 63, no. 2, pp. 502–513, Mar. 1998.

[40] G. de Bruin, N. Hemstra, and A. Pouwel, "Stratigraphic surfaces in the depositional and chronostratigraphic (wheeler-transformed) domain," *Lead. Edge*, vol. 26, no. 7, pp. 883–886, Jul. 2007.

[41] S. Fomel, "Predictive painting of 3D seismic volumes," *Geophysics*, vol. 75, no. 4, pp. A25–A30, Jul. 2010.

[42] S. Yu, J. Ma, and W. Wang, "Deep learning for denoising," 2018, *arXiv:1810.11614*.

[43] S. Zu, J. Cao, S. Qu, and Y. Chen, "Iterative deblending for simultaneous source data using the deep neural network," *Geophysics*, vol. 85, no. 2, pp. V131–V141, Mar. 2020.

[44] X. Wu, L. Liang, Y. Shi, and S. Fomel, "FaultSeg3D: Using synthetic datasets to train an end-to-end convolutional neural network for 3D seismic fault segmentation," *Geophysics*, vol. 84, pp. IM35–IM45, 2019.

[45] H. Gao, X. Wu, and G. Liu, "ChannelSeg3D: Channel simulation and deep learning for channel interpretation in 3D seismic images," *Geophysics*, vol. 86, no. 4, pp. IM73–IM83, Jul. 2021.

[46] Y. Shi, X. Wu, and S. Fomel, "SaltSeg: Automatic 3D salt segmentation using a deep convolutional neural network," *Interpretation*, vol. 7, no. 3, pp. SE113–SE122, Aug. 2019.

[47] F. Yang and J. Ma, "Deep-learning inversion: A next-generation seismic velocity model building method," *Geophysics*, vol. 84, no. 4, pp. R583–R599, Jul. 2019, doi: 10.1190/geo2018-0249.1.

[48] Z. Geng, Z. Zhao, Y. Shi, X. Wu, S. Fomel, and M. Sen, "Deep learning for velocity model building with common-image gather volumes," *Geophys. J. Int.*, vol. 228, no. 2, pp. 1054–1070, 2021.

[49] Z. Geng, X. Wu, Y. Shi, and S. Fomel, "Deep learning for relative geologic time and seismic horizons," *Geophysics*, vol. 85, no. 4, pp. WA87–WA100, Jul. 2020.

[50] Z. Bi, X. Wu, Z. Geng, and H. Li, "Deep relative geologic time: A deep learning method for simultaneously interpreting 3-D seismic horizons and faults," *J. Geophys. Res., Solid Earth*, vol. 126, no. 9, Sep. 2021, Art. no. e2021JB021882.

[51] A. Vaswani et al., "Attention is all you need," *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[52] S. M. Mousavi, W. L. Ellsworth, W. Zhu, L. Y. Chuang, and G. C. Beroza, "Earthquake transformer—An attentive deep-learning model for simultaneous earthquake detection and phase picking," *Nature Commun.*, vol. 11, no. 1, p. 3952, Aug. 2020.

[53] O. M. Saad, Y. Chen, A. Savvaidis, S. Fomel, and Y. Chen, "Real-time earthquake detection and magnitude estimation using vision transformer," *J. Geophys. Res., Solid Earth*, vol. 127, no. 5, May 2022, Art. no. e2021JB023657.

[54] S. Zu, C. Ke, C. Hou, J. Cao, and H. Zhang, "End-to-end deblending of simultaneous source data using transformer," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[55] L. Yang et al., "Porosity and permeability prediction using a transformer and periodic long short-term network," *Geophysics*, vol. 88, no. 1, pp. WA293–WA308, Jan. 2023.

[56] D. Kim, W. Ka, P. Ahn, D. Joo, S. Chun, and J. Kim, "Global–local path networks for monocular depth estimation with vertical CutDepth," 2022, *arXiv:2201.07436*.

[57] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. 37th Asilomar Conf. Signals, Syst. Comput.*, vol. 2, Jul. 2003, pp. 1398–1402.

[58] J. Li, X. Wu, and Z. Hu, "Deep learning for simultaneous seismic image super-resolution and denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–11, 2021.

[59] I. Vasiljevic et al., "DIODE: A dense indoor and outdoor depth dataset," 2019, *arXiv:1908.00463*.

[60] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 47–57, Mar. 2017.

[61] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 565–571.

[62] X. Wu, Z. Geng, Y. Shi, N. Pham, S. Fomel, and G. Caumon, "Building realistic structure models to train convolutional neural networks for seismic structural interpretation," *Geophysics*, vol. 85, no. 4, pp. WA27–WA39, Jul. 2020.

[63] Z. Bi, X. Wu, Z. Li, D. Chang, and X. Yong, "DeepISMNet: Three-dimensional implicit structural modeling with convolutional neural network," *Geosci. Model Develop. Discuss.*, vol. 15, no. 17, pp. 1–28, 2022.

[64] Z. Geng, X. Wu, Y. Shi, and S. Fomel, "Relative geologic time estimation using a deep convolutional neural network," in *Proc. 88th Annu. Int. Meeting, SEG, Expanded Abstr.*, 2019, pp. 2238–2242.

[65] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 1–9.

[66] X. Wu et al., "Sensing prior constraints in deep neural networks for solving exploration geophysical problems," *Proc. Nat. Acad. Sci. USA*, 2023.

Jiarun Yang received the B.S. degree in geophysics from the Chengdu University of Technology, Chengdu, China, in 2018, and the master's degree in geophysics from the China University of Petroleum (Beijing), Beijing, China, in 2021. He is currently pursuing the Ph.D. degree in geophysics from the University of Science and Technology of China, Hefei, China.

His research interests include deep-learning applications on seismic interpretation.

Xinming Wu (Associate Member, IEEE) received the Ph.D. degree in geophysics from the Colorado School of Mines, Golden, Colorado, in 2016.

He is a Professor of Geophysics at the University of Science and Technology of China, where he is Leading the Computational Interpretation Group (http://cig.ustc.edu.cn/). He worked with the Colorado School of Mines and Dr. Dave Hale at the Center for Wave Phenomena. He worked as a Post-Doctoral from 2016 to 2019 with Dr. Sergey Fomel at the Bureau of Economic Geology, UT Austin.

Dr. Wu is a member of the SEG Distinguished Lecture Committee (2020-present), and a session chair of the SEG Annual Meetings (2016-present). He received the SEG's awards for the J. Clarence Karcher Award in 2020, a Honorary Lecturer, South and East Asia in 2020, Best Paper (Honorable Mention) in 2018 SEG Annual meeting, and Best Paper of Geophysics in 2016. He actively contributes to the geophysics community by serving as an Associate Editor for the journals of Geophysics (2020-present) and Computers and Geosciences (2022-present).

Zhengfa Bi received the B.S. degree in geology from the China University of Geosciences Beijing, Beijing, China, in 2017. He is currently pursuing the Ph.D. degree in geophysics with the University of Science and Technology of China, Hefei, China.

His research interests include imaging processing, numerical geologic modeling, geophysical computing, and machine learning algorithms, with an emphasis on problems involving seismology and marine data processing.

Zhicheng Geng received the B.S. degree in geophysics from Tongji University, Shanghai, China, in 2017, and the Ph.D. degree in geophysics from The University of Texas at Austin, Austin, TX, USA, in 2022, under the supervision of Dr. Sergey Fomel.

He is currently an Applied Scientist with the Amazon Machine Learning Solutions Laboratory, Amazon.com, Inc., Austin. His research interests include deep-learning applications to solve geophysical problems.