

SaltISCG: Interactive Salt Segmentation Method Based on CNN and Graph Cut

Hao Zhang[✉], Peimin Zhu, Zhiying Liao[✉], and Zewei Li

Abstract—Salt body extraction plays an important role in the analysis of salt structures and the exploration of oil and gas. Seismic attributes and edge detection algorithms, which require manual effort, are the conventional methods of extracting salt boundaries from seismic images. Convolutional neural networks (CNNs) have become the state-of-the-art automatic segmentation method for seismic interpretation. However, the fully automatic results of the extraction of salt boundaries may still need to be modified to become accurate and robust enough for practical production. We present a novel deep-learning-based interactive segmentation method for extracting salt boundaries. To incorporate the interaction points into our method, we transform positive and negative points into two Euclidean distance maps (EDMs), which are combined with seismic images to train our CNN model. The model is composed of a U-net and a pyramid pooling module (PPM), and it is trained on the Tomlinson Geophysical Services (TGS) Salt Identification Challenge dataset. Then, we use a graph cut algorithm to refine the likelihood maps predicted by our CNN model and, subsequently, update the salt boundaries. Some field examples show that the proposed method outperforms fully automatic CNN methods with a higher matching degree of the ground truth.

Index Terms—Deep learning, graph cut, interactive segmentation method, salt interpretation.

I. INTRODUCTION

GEOLOGICAL anomalies are very different from the surrounding environment in genesis, composition, and structure, and they possess specific spatial characteristics, such as faults, unconformities, and salt boundaries. The formation process and distribution of geological anomalies are related to geological conditions and time evolution. Thus, identifying geological anomalies from seismic images is important for the analysis of the geological structures and seismic interpretation. Salt bodies are important reservoir structures, because they contain resource information about oil and gas; hence, salt boundary detection is of great significance for the analysis of salt structures [1], building of velocity models [2], and the exploration of oil and gas [3]. However, salt boundaries are not only affected by seismic reflection but also closely related to the seismic facies of many different lithologic combinations, especially in the case of seismic images with

Manuscript received February 25, 2022; revised April 1, 2022; accepted April 19, 2022. Date of publication May 5, 2022; date of current version May 16, 2022. (*Corresponding author:* Peimin Zhu.)

The authors are with the Institute of Geophysics and Geomatics, China University of Geosciences (CUG), Wuhan 430074, China (e-mail: zhang_hao_igg@cug.edu.cn; zhupm@126.com; atliaoziying@outlook.com; zewei_li@cug.edu.cn).

Digital Object Identifier 10.1109/TGRS.2022.3172997

complex characteristics, where it is difficult to identify salt boundaries accurately and effectively.

Conventional methods of automatic salt boundary detection are based on seismic attributes, edge detection techniques, and image segmentation methods. The seismic attributes commonly applied for extracting salt boundaries are texture attributes [4]–[7], reflection dip attributes [8], normal vector fields [9], and salt likelihoods [10]. These seismic attributes are generally calculated and extended based on amplitude attributes, which can intuitively reflect special geological structure characteristics. Methods based on edge detection techniques, such as Sobel [11]–[13], Canny [14], and 3-D gradient [15], can highlight salt boundaries in seismic images after calculation with amplitude attribute, especially for data with strong seismic amplitude variations. Some authors [16]–[18] applied image segmentation methods to extract salt boundaries based on the texture of salt bodies in seismic images. Several researchers [19]–[21] combined multiple seismic attributes and edge detection techniques to extract salt boundaries from seismic images. However, due to poor seismic imaging, strong noise, and unclear events related to salt structures, these methods cannot completely and truly extract the complex features of seismic images, and multiple rounds of seismic interpretation are still required to obtain a perfect salt boundary [22]–[24].

Using a large number of hidden layers, deep learning can automatically extract seismic features that cannot be described by seismic attributes directly from seismic data. Some researchers proposed the use of automatic methods based on deep learning to segment salt bodies. Waldeland and Solberg [25] obtained a 2-D subpart of the 3-D seismic data as the training dataset and applied two convolutional neural networks (CNNs) of different architectures to automatically identify the top and bottom, respectively, of salt bodies. This automatic method can reduce the workload of extracting salt boundaries in extremely large regional seismic explorations. Wang *et al.* [26] directly applied synthetic pre-stack seismic data to train an improved full CNN, which was then used to predict the position and shape of salt bodies from the multiple seismic snapshots. Shi *et al.* [27] proposed a 3-D encoder-decoder network based on U-net [28] for salt boundary interpretation. This 3-D CNN model performed well in synthetic verification and on field datasets. However, to make the salt boundaries more continuous and smoother, the authors used postprocessing to remove false salt bodies and disperse noise, which often leads to the removal of the correct

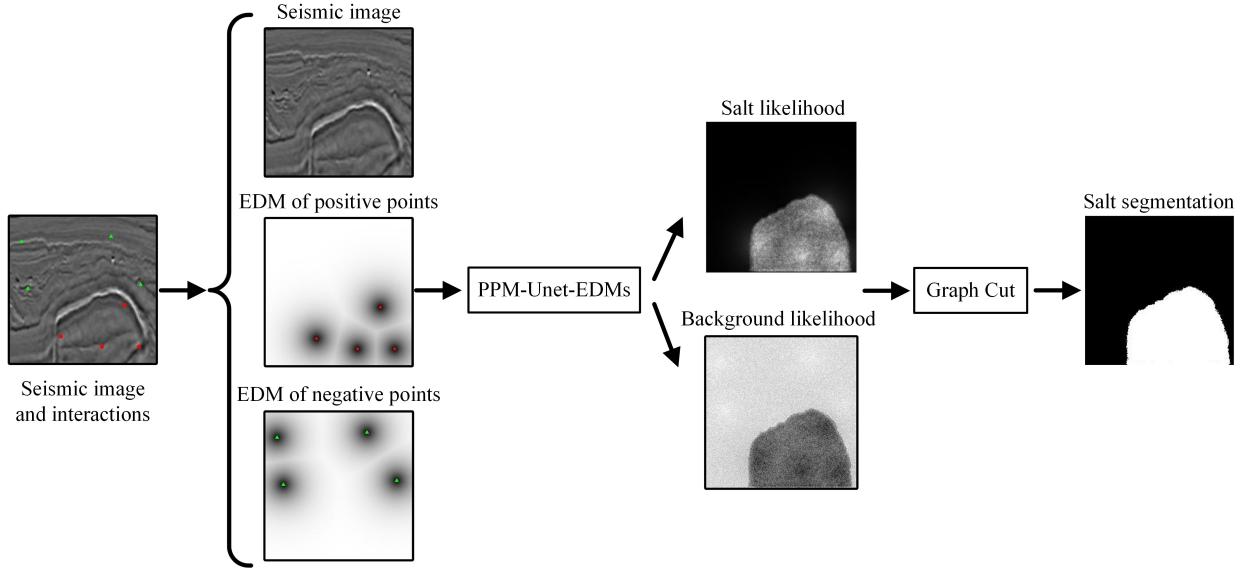


Fig. 1. Workflow of our proposed method. Human interactions are first transformed into EDMs of positive and negative points, respectively, and then concatenated with seismic images to be input into the PPM-Unet-EDMs model. The graph cut algorithm is combined with the likelihood maps predicted by the PPM-Unet-EDMs model to obtain the final segmentation results.

salt boundaries. To make the CNN model quickly suitable for new field data, Shi *et al.* [29] proposed an interactive method for extracting salt bodies and faults based on deep learning. This interactive method uses a modified U-net model to predict salt or fault likelihood and adopts the flood-filling network (FFN) algorithm to iteratively update the result of a relatively small field of view. However, when this method is extended to some field seismic images, salt bodies or faults may not be perfectly extracted. Guo *et al.* [30] applied a deep-supervised model to identify salt bodies and tested the model on the Tomlinson Geophysical Services (TGS) Salt Identification Challenge dataset [31]. This deep-supervised model consists of three branches, which output empty or nonempty, segment loss, and classifier loss, respectively, and guides feature learning through the segmentation results of an edge prediction model.

Although automatic methods based on seismic attributes and deep learning have been proposed for salt body interpretation, it is still a labor-intensive, time-consuming task. Due to low-quality imaging, discontinuous structures, and rapidly varying salt-body geometries, it may be difficult to extract salt boundaries accurately based on seismic attributes. Moreover, artificial neural network models used in deep-learning-based salt body interpretation exhibit poor generalization performance. For example, some scattered low probability noisy salt boundaries are often predicted from the test and field data. Eliminating such noise by setting thresholds or smoothing algorithms will make salt boundaries can be smoother and more continuous, but the accuracy of salt boundary prediction will be compromised.

To address the imperfection of fully automatic interpretation, we present a novel algorithm for interactive salt interpretation. To select the salt bodies in seismic images, we design a scheme that transforms into two types of Euclidean distance maps (EDMs) from positive and negative points provided

by users. Then, we concatenate these EDMs with the seismic image to compose three channels as the input data. We adopt a network architecture based on U-net to build a 2-D encoder-decoder network with skip connections [32] and a pyramid pooling module (PPM) [33]. We train the model on the TGS Salt Identification Challenge dataset [28] by randomly generating a large number of synthetic interactions and feeding seismic images and the corresponding synthetic interactions into the model. To obtain better salt boundaries compared with the U-net model, PPM-Unet model, and PPM-Unet-EDMs model, we use the graph cut algorithm [34] to optimize the thresholding results predicted by the PPM-Unet-EDMs model. We apply the method to test datasets and compare the predicted results with the ground truth. The proposed method performs high-precision salt segmentation on some field seismic images with simple guidance by humans.

II. METHODS

To extract salt boundaries at a detailed scale, we propose a deep-learning-based algorithm with human interactions for interactive segmentation. The workflow of the interactive salt segmentation method based on CNN and graph cut (SaltISCG) is shown in Fig. 1. We first add positive points and negative points to the salt bodies and background, respectively, of a seismic image and transform these points into two EDMs. Then, the original seismic image and the EDMs of the positive points and negative points form a three-channel sample input into our model, which is composed of PPM [33] and U-net [28]. Based on the salt and background likelihood maps predicted by our model, we apply the graph cut algorithm to obtain more accurate salt boundaries from the seismic image.

A. Euclidean Distance Maps

In the process of our interactive method, we need to add some simple human interactions to extract salt boundaries

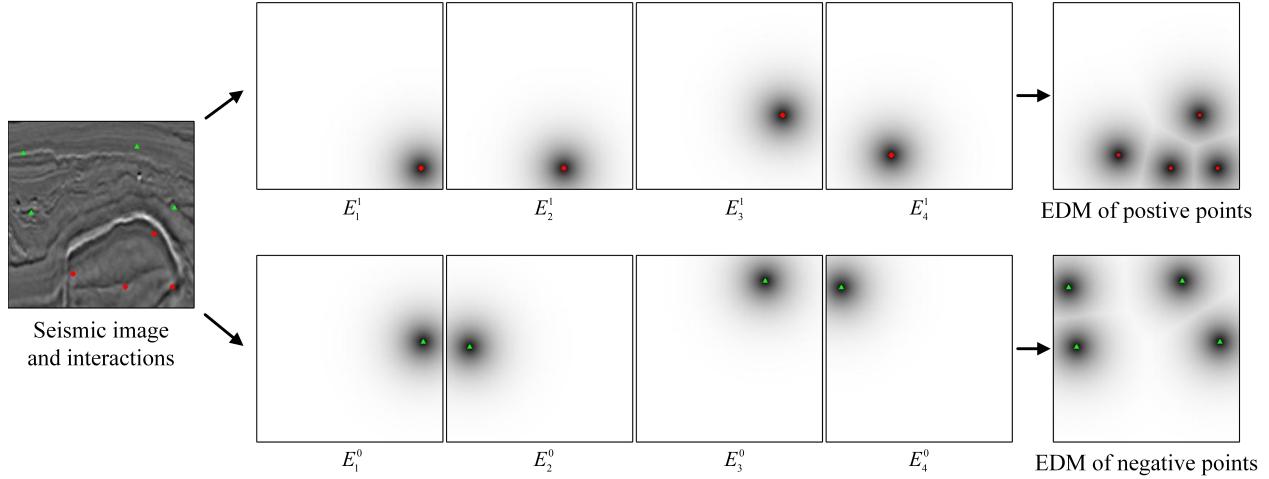


Fig. 2. Example of calculating EDMs of the positive and negative points. The human interactions include the red dots and green dots, which represent the positive points (salt) and negative points (background), respectively, in the seismic image. Two types of EDMs are obtained by calculating and maximizing the EDM of each corresponding point.

from seismic images. We use the set R to represent a series of human interactions that includes a positive point set R^1 (representing the salt bodies) and a negative point set R^0 (representing non-salt bodies or background). Both point sets are expressed as

$$R^c = \{r_1^c, r_2^c, r_3^c, \dots, r_k^c, \dots, r_N^c\} \quad (1)$$

where r_k^c represents the coordinate (m, n) of the k th point in the set, $c \in \{0, 1\}$, where 1 represents the salt and 0 represents the non-salt or background, and N is the number of elements in the set.

To apply the position information of the selected point r_k^c to extract salt boundaries, we need to calculate the EDM $E_{k,i,j}^c$ of the point r_k^c

$$E_{k,i,j}^c = \begin{cases} e^{-\sigma D^2}, & D < L \\ 0, & D \geq L \end{cases} \quad (2)$$

where (i, j) represents the coordinates of each pixel in the seismic image. D is the distance between the pixel (i, j) and the selected point (m, n) . L is the truncation coefficient, and σ is the expansion coefficient; both coefficients are used to regulate the range of the selected point r_k^c . The positive set R^1 is used as an example to illustrate the process of calculating the EDM. We need to calculate the EDM for each of the N positive points in the set R^1 and take the maximum value of the same pixel in all EDMs

$$E_{i,j}^1 = \max(E_{1,i,j}^1, E_{2,i,j}^1, E_{3,i,j}^1, \dots, E_{k,i,j}^1, \dots, E_{N,i,j}^1) \quad (3)$$

where $E_{i,j}^1$ represents the EDM of the positive set R^1 . Similarly, we can also obtain the EDM of the negative set R^0 through the steps mentioned earlier.

Fig. 2 shows the process of transforming interactions into EDMs; the red dots represent positive points (salt body), and the green dots represent negative points (background). There are four positive points and four negative points. We calculate the EDM of each point via (2) and maximize the two types of EDMs, respectively. The value ranges of the EDM and the

seismic image are $[0, 1]$ and $[0, 255]$, respectively. To improve the training efficiency of our model and maintain the data consistency of the input data, we normalize the seismic image data and combine the two EDMs as the input data.

Considering that different seismic interpreters often have different options for the same seismic data containing salt bodies during interactive salt interpretation, we need a large number of human interactive actions as input data for training our CNN model. However, it is difficult to collect such a quantity of interactive actions from seismic interpreters under realistic circumstances. Therefore, we use the random sampling method to generate a large number of synthetic interactive actions. The pixels representing the salt bodies and the background in the seismic image can be expressed as the sets S^1 and S^0 , respectively. We can obtain the positive set R^1 and the negative set R^0 by randomly selecting the corresponding points in the sets S^1 and S^0 , respectively. However, if the selected points are too close to the boundary and the influence range of the selected points is large, wrong interaction actions may be generated. Therefore, the randomly selected points must satisfy two conditions: 1) the distance between any two pixels is at least d_p pixels and 2) the distance between any pixel and the salt boundaries is at least d_{salt} pixels. Both the values of d_p and d_{salt} depend on the width and length of the input data. In addition, to ensure that the random positive points will not affect the background, we need to set the truncation coefficient L to the nearest distance between the point and the background when calculating the EDMs.

B. Network Architecture of Salt Segmentation

U-net is an improvement of the fully convolutional network (FCN) [35] model, which is widely used in biomedical image segmentation and has achieved good results [28]. Researchers have proposed some automatic seismic interpretation methods based on the U-net architecture, such as fault interpretation [36]–[38], horizon tracking [39], and seismic facies classification [40]–[42]. We design a modified U-net model

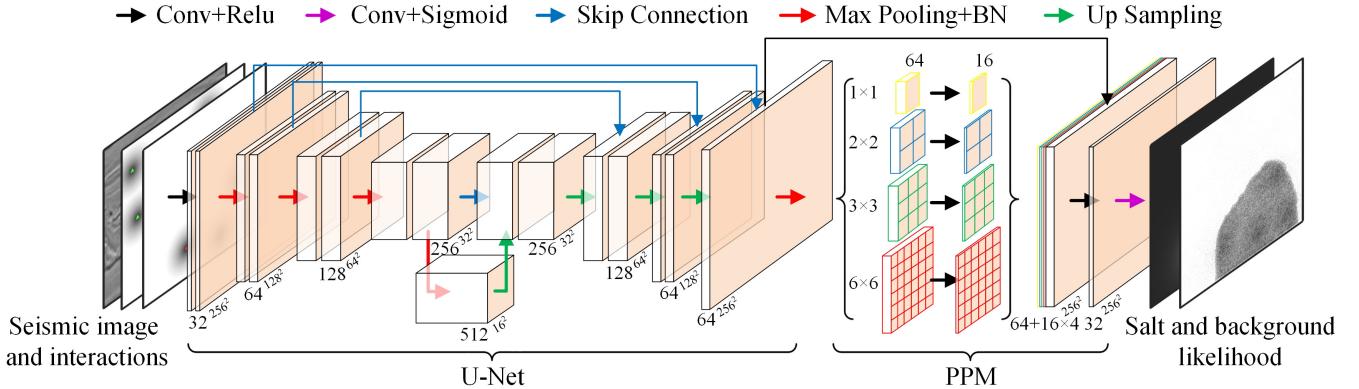


Fig. 3. Network architecture PPM-Unet-EDMs for salt interactive segmentation.

including the original U-net and PPM to predict the salt body and background likelihoods.

Our network architecture for salt interactive segmentation is shown in Fig. 3. The input data of our model include three channels: the seismic image and the two EDMs (positive points and negative points). The output data have two channels: the salt body and background likelihood maps, where the value 1 represents salt and 0 represents the background. Similar to U-net architecture, the front part of our model is composed of an encoder and a decoder. The encoder, which consists of four groups of convolutional layers and is similar to the typical FCN, is mainly used for extracting the salt features from the seismic image. Batch normalization is performed after each group [43], which includes two 3×3 convolutional layers followed by a rectified linear unit (ReLU) activation function and a 2×2 max pooling layer with stride 2 for downsampling. After the input data are calculated for each group of the encoder, the size of the output data will be halved, and the number of output channels will be doubled group by group until it reaches 512. Each group contains two 3×3 convolutional layers followed by an ReLU activation function and a 2×2 upsampling operation. Skip connection, where the layers of the corresponding encoder are copied and concatenated to the decoder, is executed to solve insufficient information caused by upsampling. In the U-net model, the size of the architecture's receptive field can affect the extraction of feature information. The size of the pooling layers of the original U-net architecture is fixed, so its receptive field will lead the model to extract only a single size of object features.

To effectively enhance the receptive field, Zhao *et al.* [33] proposed the use of a PPM to extend the capability of global context information with the relationship between different scales and different regions. Detecting salt bodies of various scales in seismic images requires receptive fields of different sizes, so we add a PPM to the single-pixel prediction framework based on U-net to improve the accuracy of salt body extraction. The PPM contains four different scales of max pooling layers of four scales: 1×1 , 2×2 , 3×3 , and 6×6 ; 1×1 is the coarsest level and denotes global pooling, whereas the other levels divide the feature map into different subregions with the pooled information of various locations.

To ensure that the PPM has feature maps with varied sizes, we place a 1×1 convolutional layer after all of the levels and reduce the number of channels from 64 to 16. Using the bilinear interpolation, the module extracts the different features with the same size as that of the original feature map, which concatenates the output of the U-net module as the final pyramid pooling global feature. The final layer of our model is a convolutional layer followed by a softmax activation function, and it is designed to predict the classification results of every pixel in the seismic image, which can output the salt and background likelihoods. Based on ensuring the prediction accuracy of the U-net model, our model extracts different levels of feature maps through PPM and aggregates the context information of the different subregions, thereby improving the extraction of global information.

Because the salt interpretation in a seismic image can be regarded as a binary classification problem, we use the cross-entropy loss function to train our model

$$L = -\frac{1}{m} \sum_{i=1}^m [y_i \log y'_i + (1 - y_i) \log(1 - y'_i)] \quad (4)$$

where m is the number of seismic image samples, y_i is the ground truth in the i th sample, and y'_i is the corresponding likelihood predicted by our model.

C. Graph Cut of Salt Segmentation

Graph cut is a very useful, well-known automatic generation algorithm for boundary optimization based on a directed graph; it associates the image segmentation problem with the minimum cut (min-cut) or maximum flow (max-flow) of a graph. This algorithm is widely used in computer vision fields, such as image segmentation [44]–[47], stereo vision [48], [49], and image matting [50], [51]. We assume that P represents all pixels of the whole seismic image, and that Q represents all pairs of adjacent points $\{p_{i,j}, p_{m,n}\}$, where $p_{i,j}, p_{m,n}$ are two adjacent pixels of the seismic image. We set the label of the seismic image as a binary vector $A = (A_1, A_2, \dots, A_{p_{i,j}}, \dots, A_P)$, where the element $A_{p_{i,j}}$ specifies assignment to the pixel $p_{i,j}$. The value of $A_{p_{i,j}}$ is 1 when the pixel $p_{i,j}$ is located in the salt body of the seismic

image; otherwise, it is 0. The cost function $C(A)$ of the vector A is defined as follows [34]:

$$C(A) = \lambda \cdot R(A) + B(A) \quad (5)$$

where λ is a coefficient that determines the importance of weights between the regional term $R(A)$ and the boundary term $B(A)$.

The term $R(A)$ reflects the penalty of pixels belonging to salt bodies or the background in the seismic image, which is defined as

$$R(A) = \sum_{p_{i,j} \in P} R_{p_{i,j}}(A_{p_{i,j}}) \quad (6)$$

where $R_{p_{i,j}}(A_{p_{i,j}})$ calculates the penalty for assigning pixel $p_{i,j}$ to label $A_{p_{i,j}}$. $R_{p_{i,j}}(A_{p_{i,j}})$ can be described by taking logarithm of the salt and background likelihoods predicted by our PPM-Unet-EDMs model

$$R_{p_{i,j}}(A_{p_{i,j}}) = \begin{cases} -\log(Po_{i,j}), & \text{if } A_{p_{i,j}} = \text{"salt"} \\ -\log(Pb_{i,j}), & \text{otherwise} \end{cases} \quad (7)$$

where $Po_{i,j}$ and $Pb_{i,j}$ represent the salt and the background likelihoods, respectively, of pixel $p_{i,j}$.

The term $B(A)$ reflects the boundary properties of the vector A , which is defined as

$$B(A) = \sum_{\{p_{i,j}, p_{m,n}\} \in Q} B_{\{p_{i,j}, p_{m,n}\}} \cdot \delta(A_{p_{i,j}}, A_{p_{m,n}}) \quad (8)$$

$$B_{\{p_{i,j}, p_{m,n}\}} \propto \exp\left(-\frac{(I_{p_{i,j}} - I_{p_{m,n}})^2}{2\varepsilon^2}\right) \cdot \frac{1}{\text{dist}(p_{i,j}, p_{m,n})} \quad (9)$$

$$\delta(A_{p_{i,j}}, A_{p_{m,n}}) = \begin{cases} 1, & \text{if } A_{p_{i,j}} \neq A_{p_{m,n}} \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

where $B_{\{p_{i,j}, p_{m,n}\}}$ represents the penalty of the discontinuity between pixels $p_{i,j}$ and $p_{m,n}$. $\text{dist}(p_{i,j}, p_{m,n})$ is the distance between pixels $p_{i,j}$ and $p_{m,n}$. $I_{p_{i,j}}$ and $I_{p_{m,n}}$ are the values of the pixels $p_{i,j}$ and $p_{m,n}$, respectively. ε is a coefficient that is used to control the value of $B_{\{p_{i,j}, p_{m,n}\}}$. The more similar (disimilar) the values of two pixels, the larger (smaller) the values of terms $B(A)$ and $B_{\{p_{i,j}, p_{m,n}\}}$.

There are some obvious defects in the outputs of our PPM-Unet-EDMs model, such as coarse salt segmentation and some scattered low probability noise, which hinder direct, proper reflection of salt bodies in seismic images. To enhance the salt segmentation results of our PPM-Unet-EDMs model, we use the max-flow/min-cut algorithm [52]–[54] to solve the minimization problem of the cost function $C(A)$. Fig. 4 shows an example of salt segmentation using the graph cut algorithm. A graph consists of two terminals: a set of nodes and a set of edges. These terminals include an object and a background and are marked as S and T , respectively. The size of the graph nodes is the same as that of the seismic image. The graph edges contain two type lines: the lines between nodes and terminals ($p_{i,j}, S$) or ($p_{i,j}, T$) are defined as t-links, and the lines of two adjacent nodes ($p_{i,j}, p_{m,n}$) are defined as n-links. The weights of edges can reflect the costs of edges, and the costs of t-links, which are indicated by regional term

$R(A)$, are estimated by (6), where the edges ($p_{i,j}, S$) and ($p_{i,j}, T$) represent the penalty of the salt and the background likelihoods, respectively. The costs of the n-links, which are indicated by the boundary term $B(A)$, are estimated by (8); these costs are affected by the distance, value, and category between two adjacent nodes. We define the min-cut as an edge set with the minimum weights. The min-cut can separate the graph into two parts (salt and background), as shown by the dotted line in Fig. 4. We set 1 and 0 as the value of the nodes assigned to the object and background terminals, respectively, to generate the final salt segmentation.

III. EXPERIMENTS

A. Training on the TGS Salt Dataset

After constructing the model and defining the loss function, we train the salt segmentation network models on the seismic data from the TGS Salt Identification Challenge on the Kaggle platform [31]. The salt segmentation dataset includes 4000 pairs of 101×101 seismic images and corresponding label images. Each label image is separated into two types: salt body (white) and background (black). To avoid overfitting during model training, we expand the size of the seismic images and corresponding labels to 256×256 by cubic spline interpolation.

To verify that the PPM module and EDMs can improve the accuracy of extracting salt bodies, we apply the U-net [28], PPM-Unet [33], PPM-Unet-EDMs, and SaltISCG to the TGS salt dataset, respectively. U-net and PPM-Unet are fully automatic segmentation methods based on CNNs, which do not need human interactions and are trained with only seismic data. The PPM-Unet-EDMs model needs to input three channels: the seismic data and two EDMs (positive points and negative points). Because the TGS salt dataset has no samples of human interactions, we simulate a large number of interactions to identify salt in the seismic images, including positive points and negative points. Here, we set the range of the number of points for generating interactions to be between 1 and 10. Then, we transform these points into EDMs and maximize EDMs of the same type. Finally, we combine the original seismic image with the two types of EDMs to form a new sample as input data of the PPM-Unet-EDMs model. The SaltISCG method combines the CNN module and the graph cut optimization to refine the salt segmentation, and its CNN module is the same as the PPM-Unet-EDMs model. Therefore, we preserve the training weights of the PPM-Unet-EDMs model as the CNN module of the SaltISCG method for application with the testing data and field seismic data.

To demonstrate that the PPM-Unet-EDMs model has a better performance than the U-net and the PPM-Unet in the case of insufficient samples, we also design two different splitting training/testing ratios: one is 2:8 that randomly selects 800 samples as the training dataset and 3200 samples as the testing dataset, and the other is 9:1 that randomly selects 3600 samples as the training dataset and 400 samples as the testing dataset. Due to the limited amounts of training samples, we use the $k\$$ -fold cross validation [55] to avoid

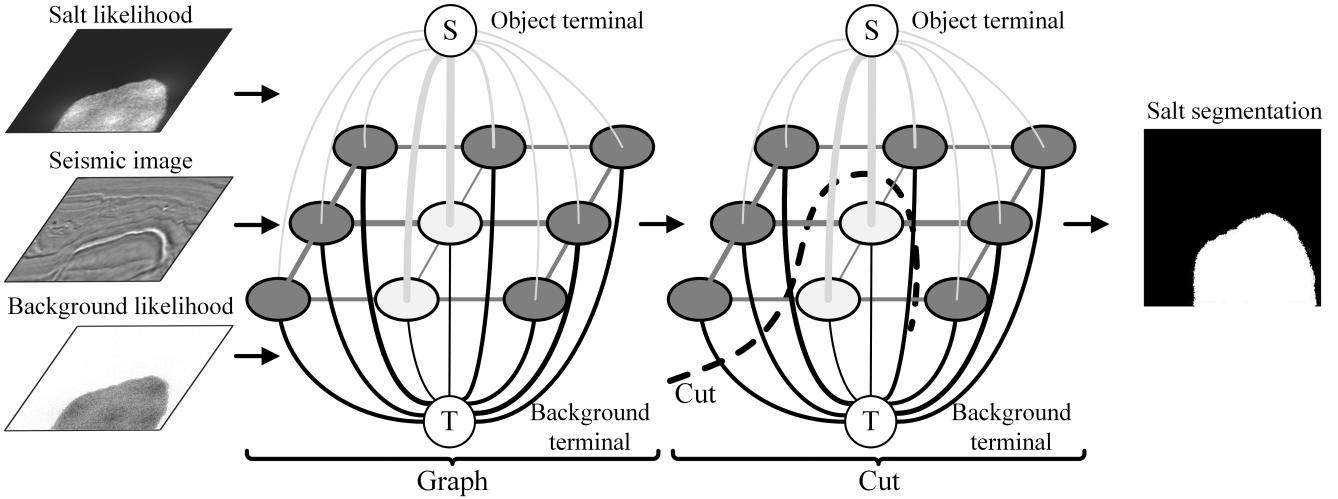


Fig. 4. Workflow of salt segmentation using graph cut algorithm.

the overfitting or underfitting in training. Here, we divide the training dataset into fivefold, where fourfold is used for training a model, and the remaining fold is used for validating the model by calculating a prediction error. For each method, we choose the model with the minimum error from fivefold for learning again on all training datasets and testing on the testing dataset.

In our experiments, the loss function L is minimized, and the model parameters are optimized by the Adam optimizer [56]. To better improve the convergence speed of our model, we set the initial learning rate to 1×10^{-4} and use a discounting factor of 0.9 to adjust the learning rate in each epoch. We train the U-net, PPM-Unet, and PPM-Unet-EDMs with 100 epochs on one GPU (Tesla V100), and the batch size is set to 64. As the training dataset is divided into fivefold, we can obtain five prediction errors for each method, and then, we choose the model with the minimum error for testing. As shown in Fig. 5, the loss curves of different models with the minimum error at the different training/testing ratios gradually decrease with the increase in training epochs. From the loss curves on the ratio of 1:4 [Fig. 5(a)], we can observe that the training losses of U-net (green solid line), PPM-Unet (blue solid line), and PPM-Unet-EDMs (red solid line) gradually converge to 0.101, 0.089, and 0.078, respectively, whereas the testing losses of U-net (green dotted line) and PPM-Unet (blue dotted line) are only slightly decreasing trends except PPM-Unet-EDMs (red dotted line). When the training samples are sufficient enough, the training and testing loss curves of all models gradually converge to a stable state, as shown in Fig. 5(b). The training loss of the PPM-Unet is lower than the U-net model, but it is higher than the PPM-Unet-EDMs model. The testing loss curves in Fig. 5(b) provide similar results of the models. It shows that the PPM-Unet-EDMs model still has a good performance, even though there are a few data in the training dataset.

B. Comparison and Evaluation

To quantitatively analyze the performances of different methods, the mean average precision (mAP) is used as

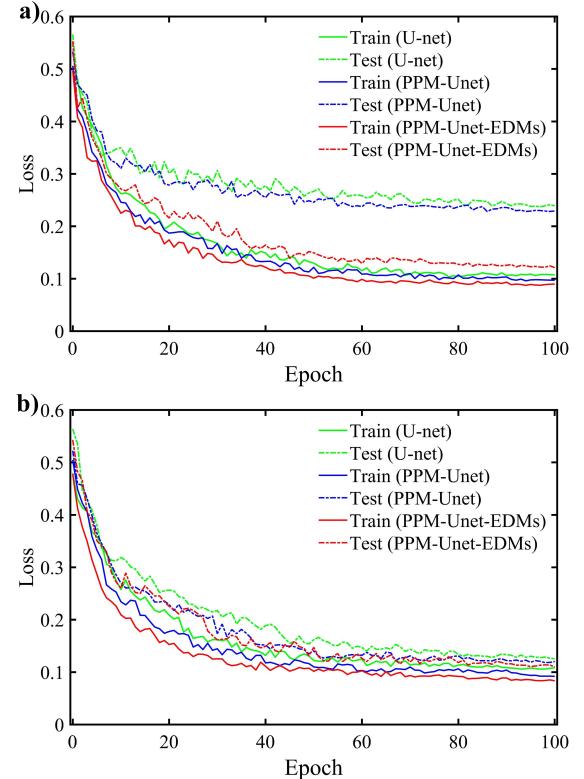


Fig. 5. Training and testing loss curves of different methods at the training/testing ratios of (a) 2:8 and (b) 9:1.

evaluation metric [31], which is calculated by different thresholds of intersection over union (IoU). IoU is defined as

$$\text{IoU}(A, B) = \frac{A \cap B}{A \cup B} \quad (11)$$

where A represents a set of predicted salt pixels, and B represents a set of true salt pixels.

We set the range of the threshold values from 0.5 to 0.95 with the step size of 0.05. For calculating the AP of a single image, we use the number of true positives (TPs),

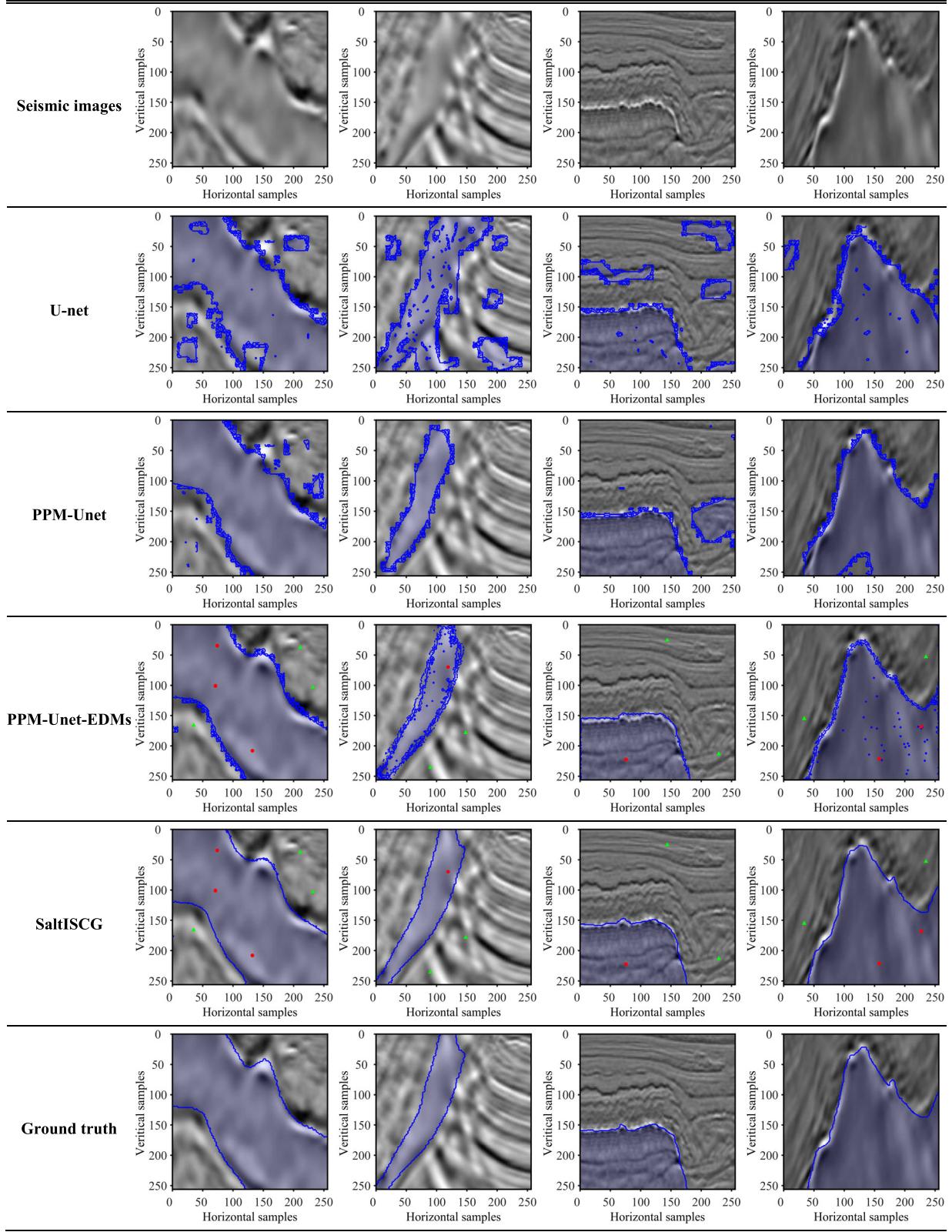


Fig. 6. Salt segmentation results obtained by different algorithms on the testing dataset.

false negatives (FNs), and false positives (FPs) generated by comparing the predicted results with the ground truth

$$AP = \frac{1}{|\text{thresholds}|} \sum_t \frac{TP(t)}{TP(t) + FP(t) + FN(t)} \quad (12)$$

where a TP is calculated when the IoU between a single predicted and the ground truth is higher than the threshold value t . An FP means that a negative category is predicted to be the ground truth. An FN indicates that the ground truth is predicted to be a negative category. The final evaluation

TABLE I
EVALUATION OF DIFFERENT METHODS ON THE TGS SALT DATASET

training/testing ratios	Method	mAP(%)
2:8	U-net	64.90
	PPM-Unet	69.25
	PPM-Unet-EDMs	84.47
	SaltISCG	86.94
9:1	U-net	76.68
	PPM-Unet	80.82
	PPM-Unet-EDMs	88.31
	SaltISCG	91.59

metric (mAP) is obtained as the mean of the individual AP of each image in the test dataset.

We compare the SaltISCG method to the fully automatic segmentation methods on the TGS salt dataset. For each method, we take the evaluation metric of the best model as the quantitative result. Table I shows the quantitative results of the best models for different methods. The SaltISCG method gives two mAPs of 86.94% and 91.59% on the 2:8 and 9:1 training/testing ratios, respectively, which is higher than other methods. On the two different training/testing ratios, the PPM module, EDMs, and graph cut optimization improve the mAP of extracting salt. Especially, when the ratio training/testing is 2:8 that indicates the training data is insufficient, PPM-Unet-EDMs and SaltISCG have good evaluation scores, even higher than the mAPs of U-net and PPM-Unet when using 80% of the data for training. It shows that our method can greatly improve the accuracy of extracting salt bodies through human interaction when the training dataset is insufficient, which is of great significance in practical application.

To further compare other methods with the SaltISCG method, we select four samples from the testing dataset, as shown in Fig. 6. The first row shows four seismic images, and the bottommost row shows the ground truths. From the second to the third row shows the threshold results of the U-net and PPM-Unet, respectively. The fourth row shows the threshold results of the PPM-Unet-EDMs model with human interactions (red and green points). Red dots and green dots represent the positive points and negative points, respectively. These methods output the salt and background likelihood maps, but we set a 0.5 thresholding to classify them as salt (salt likelihood > 0.5) or background (otherwise) for intuitive comparisons with the SaltISCG method and performance analysis. The fifth row shows the results predicted by the SaltISCG method that combines the PPM-Unet-EDMs model and the graph cut algorithm. We can observe that the binarization results of other methods exhibit some scattered noisy salt boundaries, and the matching of the predicted salts of our method with the labels of the salts (the bottommost row) is better than that of the PPM-Unet-EDMs model (the fourth row).

As the interactions of the SaltISCG method is not human picked, we study the variation between the mAP and the number of points. The points of a sample are automatically

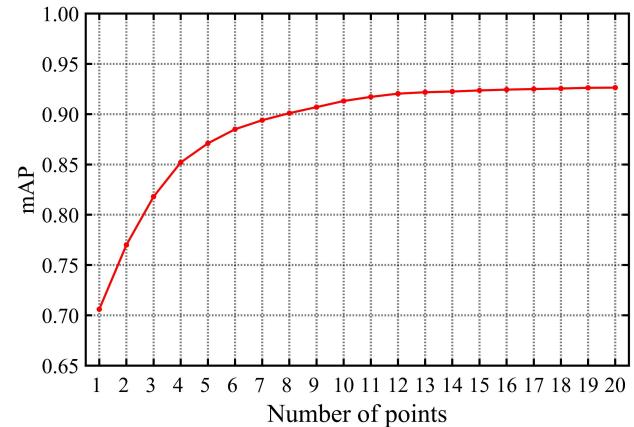


Fig. 7. Testing (80% of the TGS salt dataset) mAP of the SaltISCG method increases with the number of points.

generated using the random sampling method described in Section II-A. We use the CNN weights trained with the ratio of 2:8 to predict salt bodies and record the mAP change of the SaltISCG method over the first 20 points on the testing dataset. From the mAP curve (Fig. 7), we can observe that the growth rate of the mAP is fast, first, and then slow as the number of points increases. For example, the mAP increases rapidly from 70.6% to 91.3% when the number of the points is from 1 to 10, and slowly from 91.7% to 92.6% when the number of the points is from 11 to 20. When the number of points is 20, the SaltISCG method still achieves the highest mAP. We also show some qualitative results by the SaltISCG method at different interactive points (Fig. 8). It demonstrates that our method improves the mAP of segmentation results when adding some new points.

C. Several Factors Affecting the Performance of the SaltISCG

To select the best network parameters of the SaltISCG method, we separately experiment the effect of the depth of the CNN module, the type of pooling operation, and the size of the convolution kernel on the evaluation metrics in extracting salt from the TGS dataset.

The size and the depth of CNN are the important factors that determine its representation capacity. As the CNN's depth and size increase, so does the number of convolutional layers, pooling layers, and the trainable parameters. The increase in trainable parameters is conducive to enhance the feature extraction and representation capacity of CNN. For the U-Net architecture in Fig. 3, the size and the depth of U-Net are determined by the number of pooling or upsampling operations. Meanwhile, to avoid the impacting of the type of pooling layer and the size of convolution kernel, we uniformly use the 2×2 max pooling layer with stride 2 and the convolutional kernel of 3×3 for the CNN of different depths. Fig. 9 shows the variation of the evaluation metrics with different numbers of pooling or upsampling operations. It suggests that the performances of the SaltISCG improve with the increase in the depth of the CNN module. We also observe that the performance of the SaltISCG does not improve when the

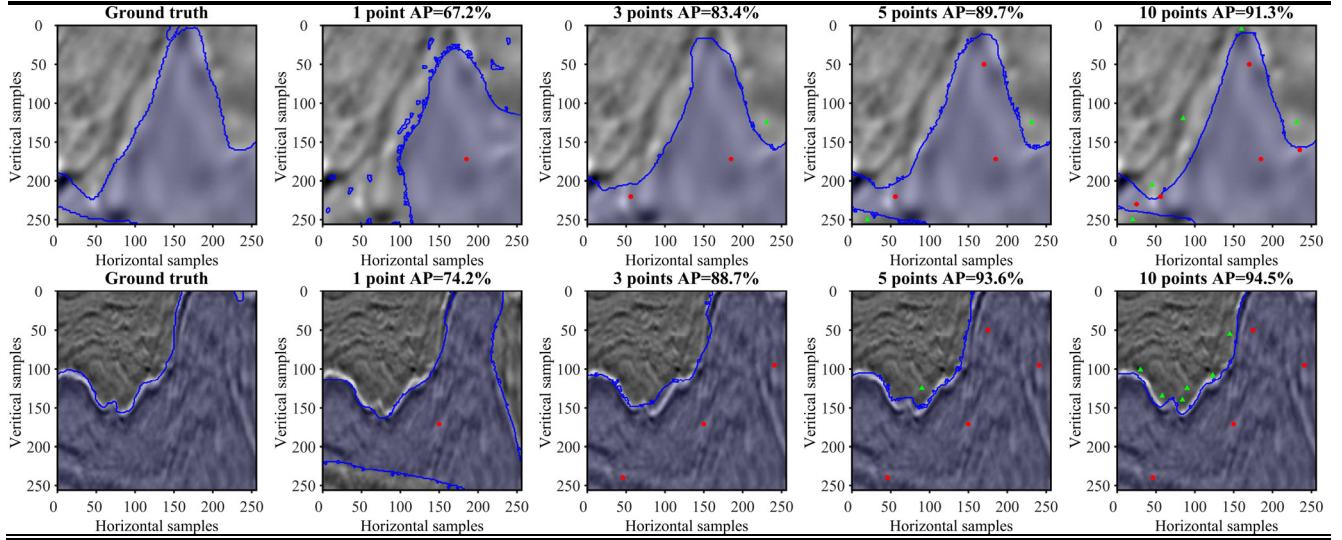


Fig. 8. Qualitative results of the SaltISCG method on the TGS testing dataset.

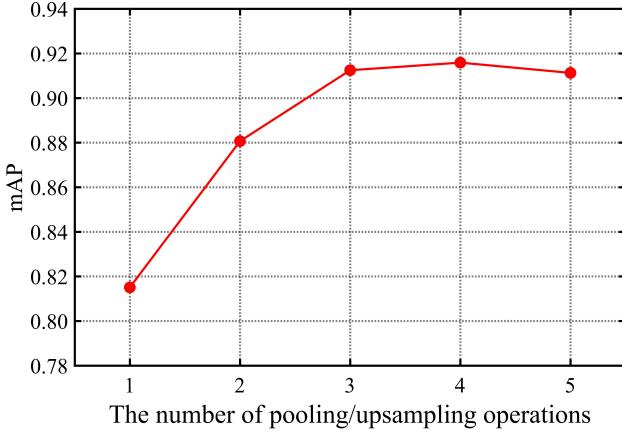


Fig. 9. Variation of the evaluation metrics with the number of pooling/upsampling operations when constructing the CNN module of SaltISCG.

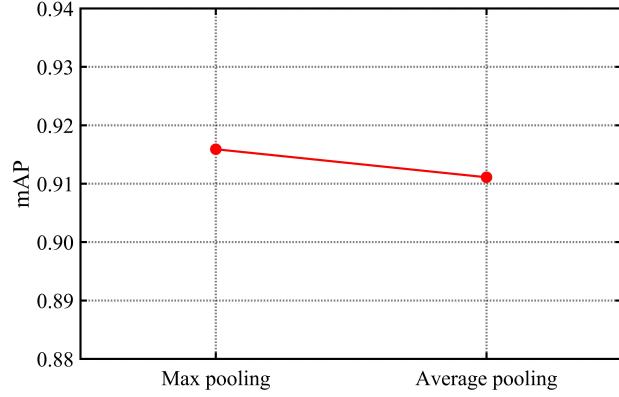


Fig. 10. Evaluation metrics with different pooling functions are used in the SaltISCG.

number of pooling or upsampling operations is increased to 5. As the deeper CNN module also increases the trainable parameters, choosing a reasonable depth allows our method to

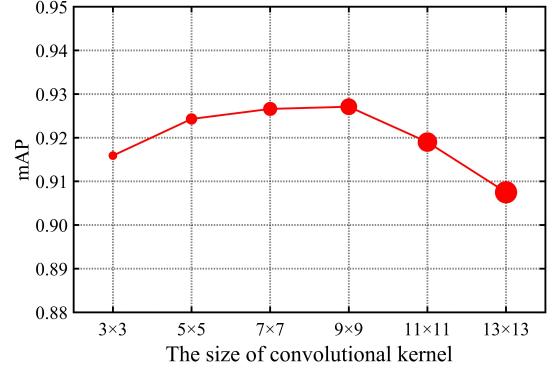


Fig. 11. Evaluation metrics with different kernel sizes are used in the SaltISCG.

achieve better performance while saving training time. Therefore, we ultimately set the number of pooling or upsampling operations to 4.

The pooling layer not only reduces the number of trainable parameters and the computation cost in the network, but also summarizes the features from the feature map generated by a convolutional layer and makes the model invariant to the distortion of feature map. The common types of pooling functions in the deep learning are max pooling and average pooling. The max pooling function captures the maximum value of each kernel in each feature map, while the average pooling function calculates the mean value of each kernel. To determine which pooling function is more suitable for extracting salt, we evaluate the performance of the two trained CNN modules with max and average pooling functions. Fig. 10 shows that the evaluation metric using the max pooling function is better than the average pooling function.

The size of convolutional kernel is one of the important factors affecting the feature extraction of CNN. Each convolutional kernel is multiplied by a certain part of the input image and moves by a defined stride. The certain part

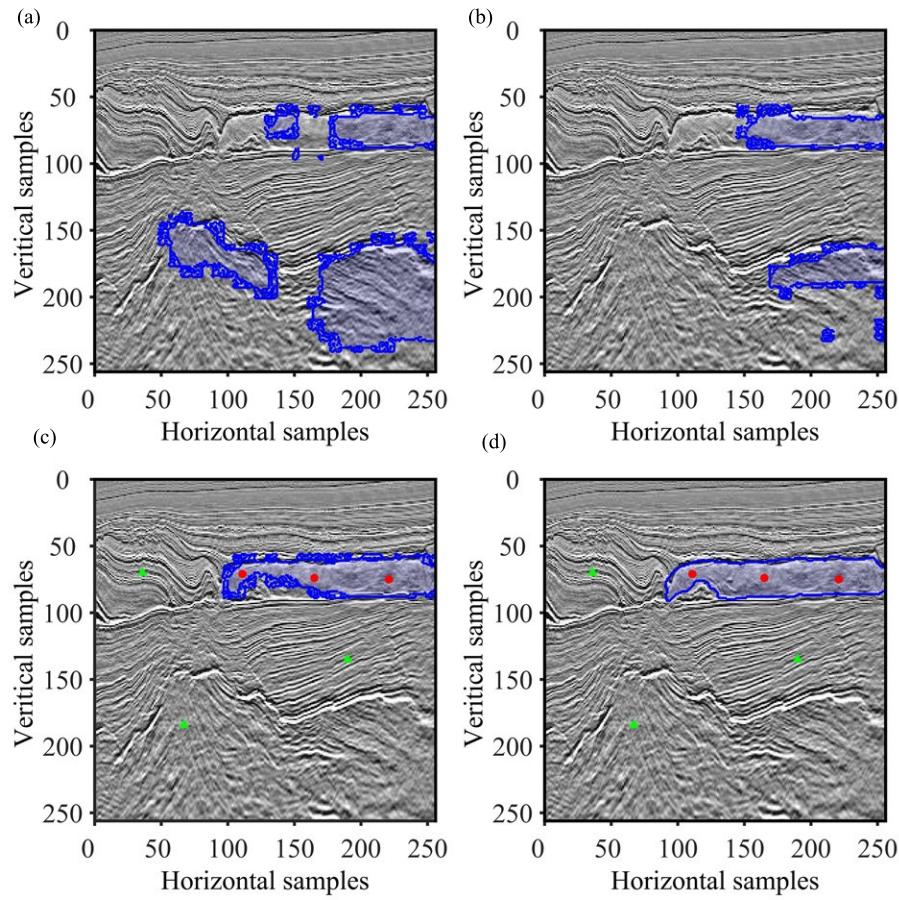


Fig. 12. 2-D seismic image is displayed with the salt segmentation results predicted by (a) U-net, (b) PPM-Unet, (c) PPM-Unet -EDMs, and (d) SaltISCG method. Our method achieves the best performance in extracting complete and accurate salt bodies.

scanned by the kernel is called the receptive field, so the receptive field increases with the increase in the size of the kernel. A larger convolutional kernel size is conducive to extracting higher resolution features and can also improve the performance of the network. However, it also brings a large number of trainable parameters and computational costs. Therefore, we investigate the effect of different kernel sizes on extracting salt using the SaltISCG. Fig. 11 shows the relationship between the evaluation metrics and the sizes of convolution kernel, where each point represents a kernel variant of the SaltISCG, and the kernel sizes are replaced by point sizes. As expected, the overall trend is that larger kernel sizes can improve the performance of the SaltISCG. However, the evaluation metric decreases quickly when the kernel size is greater than 11×11 . This is because larger kernel sizes tend to ignore the small-scale and low-resolution features. Therefore, choosing an appropriate kernel size can enable our method to achieve better performance and efficiency.

D. Evaluation of Field Seismic Data

To further verify the performance of the proposed approach, we use it to detect the salt bodies in some 2-D field seismic data. Fig. 12(a)–(d) shows four salt segmentation results (the blue region) that are, respectively, predicted using the U-net

model, PPM-Unet model, PPM-Unet-EDMs method, and our method. The first two methods [Fig. 12(a) and (b)], based on the fully CNN model, can extract some parts of the salt bodies and also highlight some scattered noise and false salt boundaries. This is because this 2-D field seismic image is different from the salt features of the training dataset, and the trained samples of the CNN models are insufficient. This indicates that the fully automatic salt segmentation method based on the CNN model has some shortcomings in field seismic data. Compared with the first two methods, the PPM-Unet-EDMs method [Fig. 12(c)] combined with human interactions provides better salt segmentation and reduces some false salt bodies, where the red dot represents the positive point (salt), and the green triangles represent the negative points (background). However, the salt segmentation results still exist some little noisy features around the salt bodies. We use the graph cut algorithm to guide the PPM-Unet-EDMs method for finely detecting salt from the complex field seismic data, as shown in Fig. 12(d). Combined with the graph cut algorithm, our method not only removes the scattered noise and false salt bodies from the predicted results of the PPM-Unet-EDMs method, but also supplements the undetected salt boundaries, thereby enhancing the accuracy of extracting salt bodies.

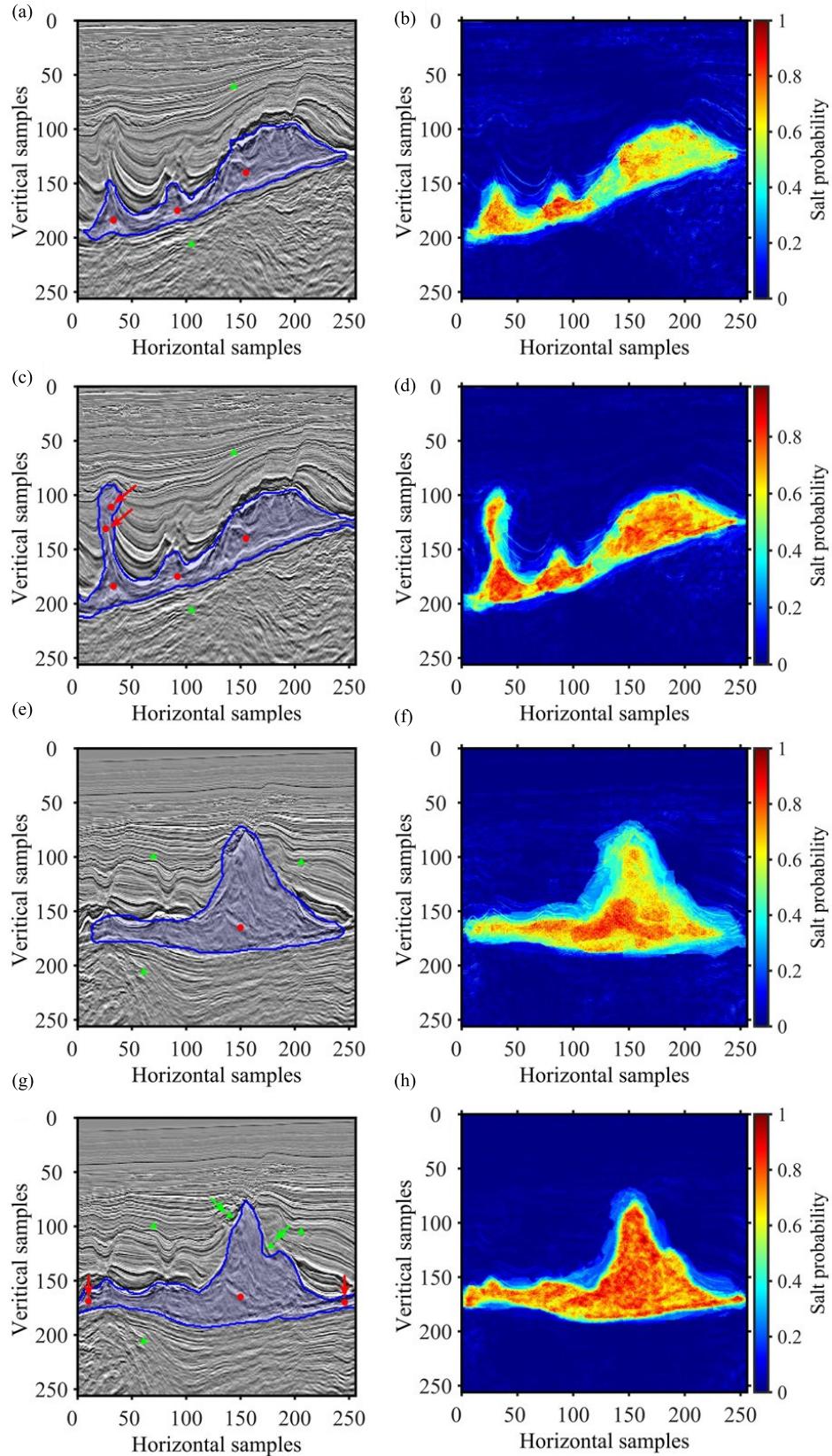


Fig. 13. Two field seismic images are overlaid with the salt segmentation results by our method. (a) and (e) Defective salt segmentation and (b) and (f) salt probability from first use of our method. We then add some interaction points (red and green arrows) to the missing and wrong regions. Better extraction of (c) and (g) complete salt segmentation and (d) and (h) salt probability from the second use of our method.

The predicted results from the first use of our method still show some noise and incomplete and wrong salt bodies with the application of added field data. Fig. 13(a) and (e) shows two salt segmentation results from the first use of our method, which are not very perfect and show a small amount of noise. We use the CNN module of our method to output the salt probabilities, and those not detected salt bodies have low probabilities, as shown in Fig. 13(b) and (f). This is because the variations of the event of salt boundaries are weak in the field seismic images, and the training dataset is insufficient. For these small errors, we add some points (red and green points) to the seismic images, and the new and complete salt segmentation results are extracted, as shown in Fig. 13(c) and (g); the red arrows indicate that some missing parts of salt bodies are detected, and the green arrows indicate that the unexpected and noise regions are removed. As shown in Fig. 13(d) and (h), we can observe that the salt probabilities increase at the corresponding position where the red points are added, and the probabilities reduce where the green points are added. Our method transforms these points into new EDMs that are input into the CNN module, which guides the network to extract interactive information. It means that using new EDMs can increase or decrease the weights of the region around the interaction points becoming the salt and improve the accuracy of our method in extracting salt bodies from seismic images. The blue regions in Fig. 13(c) and (g) show the more complete salt bodies detected by the second use of our method. This shows that the detailed salt boundaries are extracted by adding more interaction points and using our method repeatedly.

IV. DISCUSSION

Some field examples show that our interactive method, which is based on deep learning and the graph cut algorithm, can precisely detect the salt bodies from seismic images. Compared with the U-net, PPM-Unet, and PPM-Unet-EDMs, our method achieves the best performance on the TGS Salt Identification Challenge. When noise, faults, or discontinuities appear in field seismic images, the predicted results using other methods often show some scattered salt noise and strong false salt bodies. We can add some interaction points and use the graph cut algorithm to solve these problems. To remove small false salt bodies or scattered noise, we can also regulate the expansion coefficient σ and the truncation coefficient L to control the influence range of the EDMs. In addition, our proposed method only requires simple manual guidance to quickly mark the locations and shapes of salt bodies. This process will provide a large number of salt labels that are used for training the CNN module of our method, thus further improving the performance of our method.

Our method is still not a fully automatic method, and it requires adding some interaction points to the complex seismic images. Consequently, the number of interaction points and the locations of these points in the seismic images have a large impact on the salt prediction performance of our method. For example, the more points we add or the more accurate the positions of these points, the more detailed the salt boundaries result. However, it also increases the time

and workload of seismic interpretation, so we need to find a compromise between the quality of the predicted results and the workload. It is also an interesting research direction to use fewer interaction points to detect salt bodies with high precision in seismic images.

When our proposed method is extended to a new field seismic image, the predicted results of the CNN module in the method are often very unsatisfactory because of the lack of adequate data for pre-training the CNN module. To obtain satisfactory results, we add some interaction points and integrate the graph cut algorithm. After multiple interactive salt interpretations using our method, the prediction accuracy of salt boundaries is improved. However, if the features of new field data are quite different from those of the TGS dataset, we need more interaction points and interaction times to extract salts, which also reduces the efficiency of our method with salt interpretation. Another solution based on transfer learning can improve the generalization of the pre-trained model. Therefore, future research will focus on how to leverage transfer learning to improve the performance of our method on the new field data. Our method also has great application potential in 3-D salt boundary detection and can be extended to fault interpretation, horizon interpretation, and seismic facies classification.

V. CONCLUSION

We present an interactive salt segmentation algorithm by combining human interactions with a deep CNN model from seismic images. Our CNN model with PPM based on the U-net model improves the receptive field and extends the capability of extracting important features with relationships between different scales and regions from the training dataset. We transform interaction points into EDMs and combine them with seismic images for pre-training the PPM-Unet-EDMs model on the TGS dataset. To remove scattered noise and false salts in the predicted results using the pre-trained model, we apply the graph cut algorithm to optimize the salt likelihood output by the PPM-Unet-EDMs model, which can obtain better results of salt boundaries. Experimental results demonstrate that the proposed method can improve the performance with the increase in interaction times, thereby offering potential for seismic interpretation in practical applications.

REFERENCES

- [1] J. P. Leveille, I. F. Jones, Z.-Z. Zhou, B. Wang, and F. Liu, "Subsalt imaging for exploration, production, and development: A review," *Geophysics*, vol. 76, no. 5, pp. WB3–WB20, Sep. 2011, doi: [10.1190/geo2011-0156.1](https://doi.org/10.1190/geo2011-0156.1).
- [2] G. Chen, W. Yang, S. Chen, Y. Liu, and Z. Gu, "Application of envelope in salt structure velocity building: From objective function construction to the full-band seismic data reconstruction," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 9, pp. 6594–6608, Sep. 2020, doi: [10.1109/TGRS.2020.2978125](https://doi.org/10.1109/TGRS.2020.2978125).
- [3] M. R. Hudec and M. P. A. Jackson, "Terra infirma: Understanding salt tectonics," *Earth-Sci. Rev.*, vol. 82, nos. 1–2, pp. 1–28, May 2007, doi: [10.1016/j.earscirev.2007.01.001](https://doi.org/10.1016/j.earscirev.2007.01.001).
- [4] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973, doi: [10.1109/TSMC.1973.4309314](https://doi.org/10.1109/TSMC.1973.4309314).
- [5] D. Gao, "Volume texture extraction for 3D seismic visualization and interpretation," *Geophysics*, vol. 68, no. 4, pp. 1294–1302, Jul. 2003, doi: [10.1190/1.1598122](https://doi.org/10.1190/1.1598122).

- [6] A. Berthelot, A. H. S. Solberg, and L.-J. Gelius, "Texture attributes for detection of salt," *J. Appl. Geophys.*, vol. 88, pp. 52–69, Jan. 2013, doi: [10.1016/j.jappgeo.2012.09.006](https://doi.org/10.1016/j.jappgeo.2012.09.006).
- [7] Z. Wang, T. Hegazy, Z. Long, and G. AlRegib, "Noise-robust detection and tracking of salt domes in postmigrated volumes using texture, tensors, and subspace learning," *Geophysics*, vol. 80, no. 6, pp. WD101–WD116, Nov. 2015, doi: [10.1190/geo2015-0116.1](https://doi.org/10.1190/geo2015-0116.1).
- [8] A. Halpert and R. G. Clapp, "Salt body segmentation with dip and frequency attributes," *Stanford Explor. Project*, vol. 136, nos. 1–12, p. 2, 2008.
- [9] J. Haukås, O. R. Ravndal, B. H. Fotland, A. Bounaim, and L. Sonneland, "Automated salt body extraction from seismic data using the level set method," *1st Break*, vol. 31, no. 4, pp. 35–42, Apr. 2013, doi: [10.3997/1365-2397.2013009](https://doi.org/10.3997/1365-2397.2013009).
- [10] X. Wu, "Methods to compute salt likelihoods and extract salt boundaries from 3D seismic images," *Geophysics*, vol. 81, no. 6, pp. IM119–IM126, Nov. 2016, doi: [10.1190/geo2016-0250.1](https://doi.org/10.1190/geo2016-0250.1).
- [11] Z. Jing, Z. Yanqing, C. Zhigang, and L. Jianhua, "Detecting boundary of salt dome in seismic data with edge-detection technique," in *Proc. 77th Annu. Int. Meeting, SEG Expanded Abstr.*, 2007, pp. 1392–1396.
- [12] A. A. Aqrabi, T. H. Boe, and S. Barros, "Detecting salt domes using a dip guided 3D Sobel seismic attribute," in *Proc. 81st Annu. Int. Meeting SEG Tech. Program Expanded Abstr.*, Jan. 2011, pp. 1014–1018.
- [13] H. Ashraf, W. A. Mousa, and S. Al Dossary, "Sobel filter for edge detection of hexagonally sampled 3D seismic data," *Geophysics*, vol. 81, no. 6, pp. N41–N51, Nov. 2016, doi: [10.1190/geo2015-0495.1](https://doi.org/10.1190/geo2015-0495.1).
- [14] H. B. Di and D. L. Gao, "Gray-level transformation and Canny edge detection for 3D seismic discontinuity enhancement," *Comput. Geosci.*, vol. 72, no. 1, pp. 192–200, Nov. 2014, doi: [10.1016/j.cageo.2014.07.011](https://doi.org/10.1016/j.cageo.2014.07.011).
- [15] M. A. Shafiq, Z. Wang, G. AlRegib, A. Amin, and M. Deriche, "A texture-based interpretation workflow with application to delineating salt domes," *Interpretation*, vol. 5, no. 3, pp. SJ1–SJ19, Aug. 2017, doi: [10.1190/INT-2016-0043.1](https://doi.org/10.1190/INT-2016-0043.1).
- [16] J. Lomask, B. Biondi, and J. Shragge, "Image segmentation for tracking salt boundaries," in *Proc. 84th Annu. Int. Meeting SEG Tech. Program Expanded Abstr.*, Jan. 2004, pp. 1–4.
- [17] Y. Zhang and A. Halpert, "Enhanced interpreter-aided salt boundary extraction using shape deformation," in *Proc. 82nd Annu. Int. Meeting, SEG, Expanded Abstr.*, 2012, pp. 1–5.
- [18] M. A. Shafiq, Z. Wang, and G. AlRegib, "Seismic interpretation of migrated data using edge-based geodesic active contours," in *Proc. IEEE GlobalSIP*, Dec. 2015, pp. 596–600.
- [19] A. D. Halpert, R. G. Clapp, and B. Biondi, "Salt delineation via interpreter-guided 3D seismic image segmentation," *Interpretation*, vol. 2, no. 2, pp. T79–T88, May 2014, doi: [10.1190/INT-2013-0159.1](https://doi.org/10.1190/INT-2013-0159.1).
- [20] A. Amin and M. Deriche, "A hybrid approach for salt dome detection in 2D and 3D seismic data," in *Proc. IEEE ICIP*, Sep. 2015, pp. 2537–2541.
- [21] A. Amin and M. Deriche, "Salt-dome detection using a codebook-based learning model," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 11, pp. 1636–1640, Nov. 2016, doi: [10.1109/LGRS.2016.2599435](https://doi.org/10.1109/LGRS.2016.2599435).
- [22] K. J. Marfurt and T. M. Alves, "Pitfalls and limitations in seismic attribute interpretation of tectonic features," *Interpretation*, vol. 3, no. 1, pp. SB5–SB15, Feb. 2015, doi: [10.1190/INT-2014-0122.1](https://doi.org/10.1190/INT-2014-0122.1).
- [23] C. Ramirez, G. Larrazabal, and G. Gonzalez, "Salt body detection from seismic data via sparse representation," *Geophys. Prospecting*, vol. 64, no. 2, pp. 335–347, 2016, doi: [10.1111/1365-2478.12261](https://doi.org/10.1111/1365-2478.12261).
- [24] X. Wu, S. Fomel, and M. Hudec, "Fast salt boundary interpretation with optimal path picking," *Geophysics*, vol. 83, no. 3, pp. O45–O53, May 2018, doi: [10.1190/geo2017-0481.1](https://doi.org/10.1190/geo2017-0481.1).
- [25] A. Waldehand and A. Solberg, "Salt classification using deep learning," in *Proc. 79th Annu. Int. Conf. Exhib., Extended Abstr. (EAGE)*, 2017, pp. 1–5, doi: [10.3997/2214-4609.201700918](https://doi.org/10.3997/2214-4609.201700918).
- [26] W. Wang, F. Yang, and J. Ma, "Automatic salt detection with machine learning," in *Proc. 80th Annu. Int. Conf. Exhib., Extended Abstr. (EAGE)*, 2018, pp. 1–5, doi: [10.3997/2214-4609.201800917](https://doi.org/10.3997/2214-4609.201800917).
- [27] 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, doi: [10.1190/INT-2018-0235.1](https://doi.org/10.1190/INT-2018-0235.1).
- [28] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Imag. Comput. Comput.-Assist. Intervent. (MICCAI)*, 2015, pp. 234–241.
- [29] Y. Shi, X. Wu, and S. Fomel, "Interactively tracking seismic geobodies with a deep-learning flood-filling network," *Geophysics*, vol. 86, no. 1, pp. A1–A5, Jan. 2021, doi: [10.1190/geo2020-0042.1](https://doi.org/10.1190/geo2020-0042.1).
- [30] J. Guo, L. Xu, J. Ding, B. He, S. Dai, and F. Liu, "A deep supervised edge optimization algorithm for salt body segmentation," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 10, pp. 1746–1750, Oct. 2021, doi: [10.1109/LGRS.2020.3007258](https://doi.org/10.1109/LGRS.2020.3007258).
- [31] Kaggle: *TGS Salt Identification Challenge*. Accessed: Oct. 20, 2018. [Online]. Available: <https://www.kaggle.com/c/tgs-salt-identificationchallenge>
- [32] A. E. Orhan and X. Pitkow, "Skip connections eliminate singularities," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–22.
- [33] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6230–6239, doi: [10.1109/CVPR.2017.660](https://doi.org/10.1109/CVPR.2017.660).
- [34] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, Jul. 2001, pp. 105–112.
- [35] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
- [36] W. Xiong et al., "Seismic fault detection with convolutional neural network," *Geophysics*, vol. 83, no. 5, pp. O97–O103, Sep. 2018, doi: [10.1190/geo2017-0666.1](https://doi.org/10.1190/geo2017-0666.1).
- [37] X. Wu, L. Liang, Y. Shi, and S. Fomel, "FaultSeg3D: Using synthetic data sets to train an end-to-end convolutional neural network for 3D seismic fault segmentation," *Geophysics*, vol. 84, no. 3, pp. IM35–IM45, May 2019, doi: [10.1190/geo2018-0646.1](https://doi.org/10.1190/geo2018-0646.1).
- [38] X. Wu, Y. Shi, S. Fomel, L. Liang, Q. Zhang, and A. Z. Yusifov, "FaultNet3D: Predicting fault probabilities, strikes, and dips with a single convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9138–9155, Nov. 2019, doi: [10.1109/TGRS.2019.2925003](https://doi.org/10.1109/TGRS.2019.2925003).
- [39] V. Tschanne, M. Delescluse, N. Ettrich, and J. Keuper, "Extracting horizon surfaces from 3D seismic data using deep learning," *Geophysics*, vol. 85, no. 3, pp. N17–N26, May 2020, doi: [10.1190/geo2019-0569.1](https://doi.org/10.1190/geo2019-0569.1).
- [40] T. Zhao, "Seismic facies classification using different deep convolutional neural networks," in *Proc. 88th Annu. Int. Meeting, SEG Tech. Program Expanded Abstr.*, Aug. 2018, pp. 2046–2050.
- [41] Y. Zhang, Y. Liu, H. Zhang, and H. Xue, "Seismic facies analysis based on deep learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 7, pp. 1119–1123, Jul. 2020, doi: [10.1109/LGRS.2019.2941166](https://doi.org/10.1109/LGRS.2019.2941166).
- [42] F. Li, H. Zhou, Z. Wang, and X. Wu, "ADDCNN: An attention-based deep dilated convolutional neural network for seismic facies analysis with interpretable spatial-spectral maps," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 2, pp. 1733–1744, Feb. 2021, doi: [10.1109/TGRS.2020.2999365](https://doi.org/10.1109/TGRS.2020.2999365).
- [43] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
- [44] J. Bai, S. Xiang, and C. Pan, "A graph-based classification method for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 2, pp. 803–817, Feb. 2013, doi: [10.1109/TGRS.2012.2205002](https://doi.org/10.1109/TGRS.2012.2205002).
- [45] Y. Peng, L. Chen, F. Ou-Yang, W. Chen, and J. Yong, "JF-Cut: A parallel graph cut approach for large-scale image and video," *IEEE Trans. Geosci. Remote Sens.*, vol. 24, no. 2, pp. 655–666, Feb. 2015, doi: [10.1109/TIP.2014.2378060](https://doi.org/10.1109/TIP.2014.2378060).
- [46] P. Salembier and S. Foucher, "Optimum graph cuts for pruning binary partition trees of polarimetric SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 9, pp. 5493–5502, Sep. 2016, doi: [10.1109/TGRS.2016.2566581](https://doi.org/10.1109/TGRS.2016.2566581).
- [47] J. Williams et al., "3D segmentation of trees through a flexible multiclass graph cut algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 2, pp. 754–776, Feb. 2020, doi: [10.1109/TGRS.2019.2940146](https://doi.org/10.1109/TGRS.2019.2940146).
- [48] G. Vogiatzis, C. H. Esteban, P. H. S. Torr, and R. Cipolla, "Multiview stereo via volumetric graph-cuts and occlusion robust photo-consistency," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2241–2246, Dec. 2007, doi: [10.1109/TPAMI.2007.70712](https://doi.org/10.1109/TPAMI.2007.70712).
- [49] Y. Deng, Q. Yang, X. Lin, and X. Tang, "Stereo correspondence with occlusion handling in a symmetric patch-based graph-cuts model," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 1068–1079, Jun. 2007, doi: [10.1109/TPAMI.2007.1043](https://doi.org/10.1109/TPAMI.2007.1043).
- [50] J. Xiao and M. Shah, "Accurate motion layer segmentation and matting," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2005, pp. 698–703, doi: [10.1109/CVPR.2005.57](https://doi.org/10.1109/CVPR.2005.57).

- [51] R. Guo, Q. Dai, and D. Hoiem, "Paired regions for shadow detection and removal," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2956–2967, Dec. 2013, doi: [10.1109/TPAMI.2012.214](https://doi.org/10.1109/TPAMI.2012.214).
- [52] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001, doi: [10.1109/34.969114](https://doi.org/10.1109/34.969114).
- [53] Y. Boykov and V. Kolmogorov, "An experimental comparison of mincut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004, doi: [10.1109/TPAMI.2004.60](https://doi.org/10.1109/TPAMI.2004.2004.60).
- [54] V. Kolmogorov and R. Zabin, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, Feb. 2004, doi: [10.1109/TPAMI.2004.1262177](https://doi.org/10.1109/TPAMI.2004.1262177).
- [55] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *J. Roy. Stat. Soc. B, Methodol.*, vol. 36, no. 2, pp. 111–147, 1974.
- [56] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.



Hao Zhang received the M.Sc. degree in geophysics from the China University of Geosciences (CUG), Wuhan, China, in 2019, where he is currently pursuing the Ph.D. degree.

His research interests include deep learning and seismic interpretation topics, such as image processing, salt body detection, and seismic facies classification.



Peimin Zhu received the M.Sc. and Ph.D. degrees in geophysics from the China University of Geosciences, Wuhan, China, in 1993 and 2000, respectively.

He is currently a Professor in seismic and planetary science with the China University of Geosciences. He is also a Scientific Consultant with China Huadian Corporation Ltd., Beijing, China, and a Chief Scientist with GeoCMW Ltd., Beijing. He has authored over 70 publications and supervised 12 Ph.D. theses. His research interests include

the study of seismic imaging for Earth and Moon, and of radar imaging for interiors of Moon, Mars and asteroid.



Zhiying Liao received the B.S. degree in geophysics from Yangtze University, Wuhan, China, in 2017. He is currently pursuing the Ph.D. degree with the China University of Geosciences (CUG), Wuhan.

His research interests include machine learning and seismic interpretation topics, such as velocity modeling, horizon tracking, and seismic facies classification.



Zewei Li received the M.Sc. degree in geophysics from Southwest Petroleum University, Chengdu, China, in 2018. He is currently pursuing the M.Sc. degree with the China University of Geosciences (CUG), Wuhan.

His research interests include deep learning and seismic interpretation topics, such as fault detection, horizon extraction, and seismic facies classification.