Research Article

Edge Prior Multilayer Segmentation Network Based on Bayesian Framework

Chu He¹,², Zishan Shi¹, Peizhang Fang¹, Dehui Xiong¹, Bokun He¹, and Mingsheng Liao²,³

¹Electronic and Information School, Wuhan University, Wuhan, China 430072
²State Key Laboratory for Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China 430079
³Collaborative Innovation Center for Geospatial Technology, 129 Luoyu Road, Wuhan 430079, China

Correspondence should be addressed to Chu He; chuhe@whu.edu.cn

Received 21 May 2019; Revised 11 January 2020; Accepted 27 January 2020; Published 10 February 2020

Academic Editor: Vincenzo Marletta

Copyright © 2020 Chu He et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent years, methods based on neural network have achieved excellent performance for image segmentation. However, segmentation around the edge area is still unsatisfactory when dealing with complex boundaries. This paper proposes an edge prior semantic segmentation architecture based on Bayesian framework. The entire framework is composed of three network structures, a likelihood network and an edge prior network at the front, followed by a constraint network. The likelihood network produces a rough segmentation result, which is later optimized by edge prior information, including the edge map and the edge distance. For the constraint network, the modified domain transform method is proposed, in which the diffusion direction is revised through the newly defined distance map and some added constraint conditions. Experiments about the proposed approach and several contrastive methods show that our proposed method had good performance and outperformed FCN in terms of average accuracy for 0.0209 on ESAR data set.

1. Introduction

1.1. Background. Semantic segmentation of images is a very important task in computer vision, which aims to classify each pixel in the image and can be applied to autopilot, 3D reconstruction, and other fields. Synthetic Aperture Radar (SAR) image semantic segmentation is widely utilized in military and civilian information applications because SAR can obtain images at any time of the day and night independently of the weather conditions. Traditional segmentation methods can be mainly divided into three steps: segmentation of super-pixel blocks, feature extraction, and classifier selection. Methods like Meanshift [1] and Watershed [2] are typical for extracting super-pixel blocks. For classification, Support Vector Machine [3], naive Bayes [4], and Maximum Likelihood method [5] are commonly used algorithms. Afterwards, Markov Random Field (MRF) [6] and Conditional Random Field (CRF) are also introduced to consider the information of surrounding pixels.

However, owing to the special imaging mechanism of SAR imagery, such as polarization characteristic expression [7], the modeling of multiplicative non-Gaussian noise, simply applying traditional strategies for optical images to SAR imagery is not suitable anymore. Many methods for SAR image segmentation have been proposed. Liu et al. [8] proposed a SAR image segmentation with reaction diffusion level set evolution equation in an active contour model. Zhang et al. [9] developed a semisupervised SAR image segmentation method based on a hierarchical CRF model. Liu et al. [10] proffered a new SAR image segmentation approach via a hierarchical visual semantic and adaptive neighborhood multinomial latent model.

Meanwhile, the color information of polarized SAR imagery has also begun to attract researcher’s attention. Through methods like Pauli decomposition [11] based on scattering matrix, the pseudo color and texture features in SAR imagery can be extracted, that many segmentation methods mainly for optical images can be utilised. Chen
et al. [12] proposed a multifeature segmentation for high-resolution polarimetric SAR image based on fractal net evolution approach. Lang et al. [13] developed the generalized mean shift algorithm for polarimetric SAR image segmentation. In recent years, since convolutional neural network (CNN) has achieved remarkable performance in image classification, various networks suitable for semantic segmentation, such as Fully Convolutional Networks (FCN) [14], DeepLab [15] (Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs), and CRF as Recurrent Neural Networks (CRFasRNN) [16], were proposed based on VGG-Net. CNN automatically learns multiple-level feature due to its multilayer structure; it can well judge what kind of objects are contained in an image and has achieved remarkable performance in image classification, but the accurate segmentation is difficult because of the loss of the object edge details. FCN modifies CNN to obtain the classification results of each pixel and implement semantic segmentation. In general, CRF usually serves as a postprocessing method of FCN [17, 18] to improve the segmentation results and DeepLab is a typical case. DeepLab performs semantic segmentation with atrous convolution, deep convolutional nets, and fully connected CRFs. The atrous convolution is used to achieve downsampling, and the fully connected CRF overcomes the negative impact on locating accuracy caused by the achievement of invariance. For better edge correction, Chen et al. [19] replaced the fully connected CRF with Domain Transform (DT) [20] to improve the efficiency. Though DT is traditionally used for common image processing, it still shows good effect in filtering rough segmentation results, making the object edges integrated precisely.

With complex boundaries in SAR imagery, segmentation near object edges usually has a high error rate. Considering the limitations of traditional classifiers applied to the image classification, FCN-8, the best in FCN networks, is chosen to improve the accuracy of the initial classification. FCN is responsible for convolutional feature extraction and works directly on RGB vectors of each single pixel. However, the broad receptive field in traditional FCN convolution kernel and the reduced dimension in pooling layer blur the output of a single pixel, limiting the segmentation accuracy. Therefore, DT and the fully connected CRF [21] are introduced, with the influence of surrounding pixels taken into consideration. Traditional edge feature extraction methods include Canny [22], LSD [23], and the ratio of exponentially weighted averages (ROEWA) edge extraction [24]. In these years, many new edge detection methods have been proposed. Zhang et al. [25] proposed a new edge detector using structured random forests as the classifier, which can make full use of RGB-D image information from Kinect. Tabb and Ahuja [26] describe an algorithm for image segmentation at multiple scales based on edge and region detection. Mandal et al. [27] presented a novel scheme for segmenting the image boundary in optoacoustic images. Marmanis et al. [28] proposed a boundary-aware semantic segmentation algorithm; this algorithm explicitly represents and extracts the boundaries between regions of different semantic classes. The class boundaries improve the deep convolutional neural network for semantic segmentation.

DT can be seen as a recurrent neural network (RNN) [19]. The internal DT combining edge information and score map is applied for internal edge correction. The modified DT is utilized for external edge correction by combining the edge distance map and rough segmentation result. Meanwhile, holistically nested edge detection (HED) network [29] is employed to provide edge prior information to improve segmentation accuracy.

The contributions of the proposed approach are as follows:

(i) On the basis of Bayesian framework, this study presents a parallel network architecture, which is composed of two parallel networks called the likelihood network (FCN8) and the edge prior network (HED), respectively, and a constraint network (the directed DT) behind

(ii) To achieve accurate edge detection, edge map obtained from HED is utilized, serving as edge prior information to sharpen blurred edge-pixel categories

(iii) Considering that the edge distribution in traditional DT is not completely trusted and some edges may be lost, Directed Domain Transform (DDT) is proposed for image classification with complex edges. In this method, an edge distance map is defined to limit the diffusion direction, to avoid error label spreading. Besides, the gradient descent method is adopted for parameter determination, difference vectors of color and class probability about adjacent pixels are considered to deal with misclassification.

The remainder of this paper is organized as follows: Section 2 introduces the algorithms involved in each step, including HED, the fully connected CRF, internal fusion, and modified DT, and Section 3 shows the whole framework of our proposed method. In Section 4, the proposed method is tested, and its performance is compared with other semantic segmentation methods. Finally, this paper is concluded in Section 5.

2. Framework

The framework of our proposed method is shown in Figure 1; the upper part of the structure is the edge prior information, including edge map obtained by HED and the corresponding edge distance map. In the lower part of the structure, internal DT filters the rough FCN8 segmentation result by the fused edge prior information; the obtained FCN8+DT model is input into the fully connected CRF. Finally, the Directed DT combines the coarse segmentation result, the edge map, and the edge distance map to obtain the fine segmentation result.

The specific process of our proposed method mainly consists of two phases, training and testing. These two phases are described in the following.

Training: (1) based on the VGG-Net, training FCN8 with ESAR data. (2) Adding CRF as postprocessing of FCN8 to train the parameters of CRF. (3) Extracting the edge maps...
from internal FCN8 network and HED. (4) Generating the fused edge map—internal edge fusion. (5) Fine-tuning the entire model after adding DDT.

**Testing:** (1) the test image is input to FCN8 to obtain the score map, which has the same size as original image. At the same time, HED derives the corresponding reference edge map. (2) Fusing the internal edge map from FCN8 with the HED edge map. (3) Inputting the fused edge map into DT and the fully connected CRF. (4) Using DDT to filter the raw output, guided by the edge distance map.

### 3. Methodology

#### 3.1. HED

HED is an end-to-end edge detection algorithm based on the fully convolutional neural network and deeply supervised nets, referring to article [29, 30]. By relying on the weight update of convolution kernel, HED automatically learns the rich hierarchical features and determines whether the feature is important for solving the abstraction of object edges. The network architecture of HED is shown in Figure 2, five side-output layers of HED are inserted after the convolutional layer, respectively, and deep supervision is imposed at each side-output layer, so that the result is toward the edge detection. Finally, the output of HED is obtained through a weighted fusion layer from multiscale outputs. As can be seen from Figure 2, based on VGG-Net, HED incorporates two improvements. First, HED connects the side-output layer to the last convolutional layer in each stage, respectively, conv1\_2, conv2\_2, conv3\_3, conv4\_3, and conv5\_3, to obtain multiple prediction results and then fuse all the edge maps. The size of the receptive field of each convolutional layer is the same as the corresponding side-output layer, since the side-output layer is implemented as a convolution layer with 1 kernel and 1 output. Second, HED cuts the fifth pooling layer and all the fully connected layers of the VGG-Net to obtain meaningful side-output results and reduce the cost. The main goal of HED is to complete the overall image training and prediction and achieve nested multilevel feature learning.

HED is a network which can learn features to produce the edge maps approaching the ground truth. In the training process, all of the network parameters are denoted as \( W \). Suppose the network has \( M \) side-output layers, for each side-output layer, a classifier is connected with it. The weights of the corresponding classifiers are \( \omega = (\omega^{(1)}, \ldots, \omega^{(M)}) \). Then, the image-level loss function \( L_{\text{side}} \) is defined by this formulation:
the total loss function is as follows:

\[ L_{\text{side}}(W, w) = \sum_{m=1}^{M} \alpha_{m} \alpha_{\text{side}}(W, w^{(m)}). \]  

(1)

For normal images, the number of edge points is much smaller than nonedge points; only 10% of the ground truth consists of edge pixels; therefore, HED takes a simple strategy to solve the bias problem between the edge and nonedge pixels. Using \( I \) represents the original image, \( G \) represents the corresponding ground truth, and the edge and nonedge pixels are denoted as \( |G_{\text{e}}| \) and \( |G_{\text{n}}| \), respectively. The class-balanced cross-entropy loss function is defined as follows:

\[ \ell_{\text{side}}(W, w^{(m)}) = -\beta \sum_{j \in G_{\text{e}}} \log \Pr(g_j = 1 | I; W, w^{(m)}) - (1 - \beta) \sum_{j \in G_{\text{n}}} \log \Pr(g_j = 0 | I; W, w^{(m)}), \]

(2)

where \( \beta \) represents a class-balancing weight, and \( \beta = \frac{|G_{\text{e}}|}{|G_{\text{e}}| + |G_{\text{n}}|} \). \( \Pr(g_j = 1 | I; W, w^{(m)}) = \sigma(\alpha_{m}^{(m)} + w^{(m)} \cdot x_j) \) is calculated by the sigmoid function \( \sigma(\cdot) \) on the activation value at pixel \( j \).

A “weight-fusion” layer is introduced to connect all prediction results of side-output layers; the fusion weight is obtained by training process, denoted as \( h = (h_1, \ldots, h_M) \). Then, the loss function of the fusion layer is as follows:

\[ L_{\text{fuse}}(W, w, h) = \text{Dist}(G, \tilde{G}_{\text{fuse}}). \]

(3)

The activations of the side-output of layer \( m \) is \( \tilde{G}_{\text{fuse}} = \sigma(\sum_{m=1}^{M} h_{m} \tilde{A}_{\text{side}}^{(m)}) \), where \( \tilde{A}_{\text{side}}^{(m)} = \{a_j^{(m)}; j = 1, \ldots, |G| \} \). Then, the total loss function is as follows:

\[ L(W, w, h) = L_{\text{side}}(W, w) + L_{\text{fuse}}(W, w, h). \]

(4)

The greyscale edge map \( \tilde{G}_{\text{HED}} \) is obtained through HED and can be utilized as edge prior information.

3.2. Fully Connected CRFs. The probabilistic graphical model has been proven to effectively improve the image classification accuracy. CRF has a good performance in image classification and segmentation; it can extract fuzzy and insignificant pixel-level category annotations into sharp edge distributions and fine segmentation results. Therefore, CRF can be used to solve the classification error caused by fuzzy output of FCN.

Suppose \( X_i \) represents the category label of point \( i \), with \( L \) categories, as \( X_i \in L = \{l_1, l_2, \ldots, l_L\} \). The variable \( X \) is a random vector composed of \( X_1, X_2, \ldots, X_N \), where \( N \) represents the number of pixels of the image. The original image is denoted as \( I \); \( Z(I) \) represents a normalization function. CRF can be used to describe the above relationship based on Gibbs distribution:

\[ P(X = x | I) = \frac{1}{Z(I)} \exp (-E(x | I)). \]

(5)

In the fully connected CRF model, the energy distribution of \( x \) is expressed by the following formula:

\[ E(x) = \sum_{i} \psi_i^{(x)} + \sum_{ij} \varphi_{ij}(x_i, x_j), \]

(6)

where \( \psi_i^{(x)} \) is a unitary energy term representing the energy labeling the pixel \( i \) as \( X_i \); \( \varphi_{ij}^{(x)}(x_i, x_j) \) is the energy that simultaneously labeling the pixel \( i, j \) as \( X_i \) and \( X_j \). The value of unitary energy term is derived from FCN. FCN does not consider the smoothness and continuity of the point class assignment. The pairwise potential provides smooth rules based on image data and encourages the same category to be assigned to points of the same attribute. The pairwise potential is built into a model based on a Gaussian kernel as follows:

\[ \varphi_{ij}(x_i, x_j) = \mu(x_i, x_j) \sum_{m=1}^{M} w^{(m)} k_{G}(f_i, f_j). \]

(7)

In this formulation, if \( x_i \neq x_j \), \( \mu(x_i, x_j) = 1 \); otherwise, \( \mu(x_i, x_j) = 0 \). \( M \) is the number of Gaussian kernel used in the feature vector. \( f_i \) represents the feature vector of point \( i \) such as RGB color value or two-dimensional coordinates. Minimizing the energy function \( E(x) \) of CRF obtains the class label.

3.3. Internal DT

3.3.1. Internal Edge Fusion. The internal edge map is output by the internal edge net, which utilized the features obtained by the intermediate layer of FCN8. In the internal edge net, these feature layers obtained by FCN8 are filled to the same spatial scale before being stacked. A convolutional layer with \( 1 \times 1 \) convolution kernel and one output channel is introduced to predict edge strength. A RELU layer is added to limit the output to a range of 0 to 1. Since the ground truth does not contain reference edges, the implicitly generated edge map might be inaccurate. However, HED predicts object edge by training various contour types, instead of calculating gradients for edge discrimination. Thus, there is a direct connection between multilevel features. In the proposed approach, the edge map form HED is fused with the internal one, as expressed in Equation (8).

\[ g_i' = \alpha g_i + (1 - \alpha) g_{i}^{\text{HED}}, \]

(8)

where \( g_i \) is the internal edge map and \( g_{i}^{\text{HED}} \) represents another one from HED; \( g_i' \) denotes fused edge as new input of DT. \( \alpha \) means the weight with initial value of 0.5. Generally, \( \alpha \) is calculated using gradient descent algorithm. The contribution of \( \frac{\partial L}{\partial g_i} \) and \( \frac{\partial L}{\partial \alpha} \) in backpropagation is as follows:

\[ \frac{\partial L}{\partial g_i} = \frac{\partial L}{\partial g_i} \alpha, \]

(9)

\[ \left( \frac{\partial L}{\partial \alpha} \right)_n = \frac{\partial L}{\partial g_i} (g_i - g_i^{\text{HED}}) + \left( \frac{\partial L}{\partial \alpha} \right)_{n-1}, \]

(10)
where \( L \) is the loss function; \( \partial L / \partial a_i \) and \( \partial L / \partial y_j \) represent the current gradient and updated gradient, respectively. With \( g_i^{\text{HED}} \) fixed, \( a \) is updated through Equation (10). In practical application, a layer of convolutional neural network is defined to train parameters as shown in Equation (11). The network consists of a 1 × 1 convolution kernel with a single channel, an eltwise layer for add operation, and a ReLU layer. \( \alpha \) ranges from zero to infinity and is initialized to 1.

\[
    g'_i = \text{ReLU}(a g_i + g_i^{\text{HED}}). \tag{11}
\]

3.3.2. Standard DT. In the Domain Transform [19] (DT) network, the raw signal \( x \) and the positive “domain transform density” \( d \) are the inputs and the filtered signal \( y \) is the output. For 1-D signals, the output is calculated as follows:

\[
    y_i = (1 - \omega_i)x_i + \omega_i y_{i-1}. \tag{12}
\]

The weight \( \omega_i \) is related to density and defined as follows:

\[
    \omega_i = \exp\left(-\frac{\sqrt{2d_i}}{\sigma_k}\right) = \exp\left(-\frac{\sqrt{2}}{\sigma_k}\left(1 + g_i \frac{\sigma_j}{\sigma_r}\right)\right), \tag{13}
\]

where \( g_i \) is the edge map obtained by exploiting features from the intermediate network layers. \( \sigma_j \) and \( \sigma_r \) are the standard deviation of the convolution kernel over the spatial domain and the edge map, respectively. The details of \( \sigma_j, \sigma_r, \) and \( \sigma_k \) can be found in DeepLab DT [19].

Filtering from one side by Equation (12) is asymmetric. Therefore, DT filters from two sides for 1-D signals, left-to-right and right-to-left. For 2-D signals, DT employs 1-D filtering in each dimension, that is a horizontal pass (left-to-right and right-to-left) is performed along each row, followed by a vertical pass (top-to-bottom and bottom-to-top) along each column.

Actually, the edge map is implicitly generated by the DT network, not working well as the explicit network for complex boundaries. Besides, some reverse diffusion occurs in four-direction filtering, spreading error categories. Aimed at the unsatisfactory effect of DT network, the corresponding improvements in the proposed method are listed below:

1. Introduce the HED network to explicitly get the edge map, which truly reflects edge strength of different connection types

2. Define an edge distance map and obtain positive diffusion direction, that is, towards the nearest edge. Besides, the diffusion coefficients are adjusted by color vector and class probability vector, to deal with misclassification

3.4. Directed DT. To acquire a more precise edge map, an improved DT algorithm—DDT—is adopted in the presented approach, as shown in Figure 3. First, the edge distance map is defined to eliminate inappropriate diffusion. In four-direction filtering, the diffusion coefficient is adjusted by two parameters. For pixels that satisfy the diffusion conditions, the point is diffused in four directions until the iteration ends. Finally, an area-filling method is employed to fill holes.

In the distance map, the edge distance is measured by the minimum distance of each pixel to its nearest edge, as shown in formula (14).

\[
    D_{i,j} = \min \{\text{Dist}((i,j), (m,n))\}, \quad \text{for all } Y_{mn} \subset Y_+. \tag{14}
\]

where \( D_{i,j} \) is the minimum distance, \( Y_+ \) denotes all boundary pixels in the edge map, and \( (m, n) \) is the coordinate of pixel \((i, j)\).

In Figure 3(c), the dashed line means the actual edge and the solid one denotes the edge to be filtered. \( A \) and \( B \) are different class labels. The correct diffusion direction is from pixels far away to pixels near the actual edge.

\[
    \Delta D = D_{i,j} - D_{i,j+1}. \tag{15}
\]

As shown in Equation (15), \( \Delta D \) is the diffusion potential. For adjacent pixels with different classes, the diffusion condition is that \( \Delta D \) should be larger than 0.

Since a large color gradient of the same category and different classes with a small gradient may result in false edges and leaked edges, respectively, the weight \( \omega_i \) in Equation (12) is adapted as follows:

\[
    \omega_i = \exp\left\{-\frac{|I_{i+1} - I_{i-1}|^2}{\theta_1^2} - \frac{|P_{i+1} - P_{i-1}|^2}{\theta_2^2}\right\}. \tag{16}
\]

Weight \( \omega_i \) is determined by the color vector \( I \) and class probability vector \( P \). Two parameters \( \theta_1 \) and \( \theta_2 \) are used to balance the spreading strength and determined through
gradient descent method. Equation (12) is unrolled to compute derivatives with respect to \( y \) and \( \omega_i \).

\[
\frac{\partial L}{\partial \omega_i} = \begin{cases} 
0, & G(i) = 0, \\
\frac{\partial L}{\partial y_i} \sum_j (y_{i-1} - x_i), & G(i) = 1,
\end{cases}
\]  

(17)

\[
\left( \frac{\partial L}{\partial y_i} \right)_n = \begin{cases} 
\frac{\partial L}{\partial y_{i-1}}, & G(i) = 0, \\
\frac{\partial L}{\partial y_{i+1}} - \omega_{i+1}, & G(i) = 1,
\end{cases}
\]  

(18)

where \( G(i) = 1/0 \) represents whether area \( i \) has a diffusion or not. \( L \) is the loss function, \( (\partial L/\partial y_i)_{n-1} \) is the current gradient, while \( (\partial L/\partial y_i)_n \) denotes the updated gradient.

\[
\frac{\partial L}{\partial \theta_1} = 2 \sum_i \frac{\partial L}{\partial \omega_i} \frac{|I_{i+1} - I_{i-1}|^2}{\theta_i^1},
\]

\[
\frac{\partial L}{\partial \theta_2} = 2 \sum_i \frac{\partial L}{\partial \omega_i} \frac{|P_{i+1} - P_{i-1}|^2}{\theta_i^2}.
\]  

(19)

The segmentation error of DDT output \( y_i \) is backpropagated to \( \theta_1 \) and \( \theta_2 \).

After the gradients of \( y_i \) and \( \omega_i \) are updated, the gradient of \( \theta \) can be further obtained.

4. Experiment

4.1. Experiment Data. The first experimental data in this paper is ESAR L-band PolInSAR image of the German Aerospace Center, with a dimension of 1,000 × 1,000 pixels. The spatial resolution of the SAR imagery is 3 m × 2.2 m. There are five classes, namely, building, forest, farmland, road, and others, as shown in Figure 4. The second experimental data in this paper is L-band fully polarized image of Foulum Area in Denmark, acquired on April 17, 1998. This data mainly consists of farmland, forest, residential area, higher crop, shrub, and untilled glebe, as shown in Figure 5.

For these two data sets, the entire imagery is divided into four subimages. Each time, one subimage is for testing and the rest three are sliding to 4 directions for training. After four cycles, four segmented images are spliced as the final result.

4.2. Experiment Result. Through HED, the edge map of ESAR data is obtained. The edge map and edge distance map are shown in Figure 6.

The proposed algorithm is conducted on the ESAR data with seven contrast methods: (1) semantic segmentation algorithm based on Meanshift and SVM, (2) CRF based on Meanshift and Potts, (3) FCN8, (4) DeepLab, (5) FCN8+DT+the fully connected CRF, (6) FCN8+HED+DT+the fully connected CRF, and (7) the proposed method. Traditional method first uses Meanshift to obtain coarsely divided pixel blocks and extracts intensity features, texture features, and polarization decomposition features for each block [31]; then, traditional method directly classifies these features with SVM [32] or introduces neighborhood information with
Potts prior to CRF [31]. These two methods mentioned above are the first and second methods of our comparative experiment. The third comparison method is FCN8, the most accurate net of FCN. The 4th comparative experiment is DeepLab, which proposed the fully connected CRF. Among the remaining three comparison methods, the last one is our proposed method, the 6th method deletes the DDT of our proposed method, and the 5th experiment inserts DT between FCN8 and the fully connected CRF. The results of seven methods are shown in Figure 7.

To further verify the effectiveness of our method, five methods related to neural networks from the above seven methods are applied on the second dataset. The results are shown in Figure 8.

4.2.1. Confusion Matrix. As an important measuring index, the average accuracy of different methods is assessed. Since objects of the same class may have quite different colors in Pauli SAR image, thus this color continuity-based method is largely limited. As presented in Table 1, seven algorithms

Figure 7: Segmentation results of data set 1. (a) GT. (b) Meanshift+CRF. (c) Meanshift+SVM. (d) FCN8. (e) DeepLab. (f) FCN8+DT+the fully connected CRF. (g) FCN8+HED+DT+the fully connected CRF. (h) The proposed method.

Figure 8: Segmentation results of data set 2. (a) GT. (b) FCN8. (c) DeepLab. (d) FCN8+DT+the fully connected CRF. (e) FCN8+HED+DT+the fully connected CRF. (f) The proposed method.
Based on deep learning, all exhibit better performance. The confusion matrix of five algorithms on data set 2 is shown in Table 2.

Comparing with traditional methods, FCN8 has greatly improved, mainly because FCN8 is suitable for images and high accuracy in image classification. DeepLab is equivalent to adding a fully connected CRF for postprocessing after FCN8. The classification information of surrounding pixels is introduced as a reference to further improve the classification accuracy. For data set 1, compared with DeepLab, the average accuracy of DeepLab DT only increases by 0.05%, revealing that DT with internal edge extraction has almost the same effect as postprocessing CRF. For the sixth method, the edge-weighted input obtained by the FCN-HED edge and the internal edge network is trained in the DT, but the correct rate is not greatly improved. These can be caused by two reasons: (1) DT has many unsolved problems; (2) the actual image category distribution is complex. For data set 1, the proposed approach achieves the highest performance with 81.39% accuracy, outperforming DeepLab DT by 0.49%, which is attributed to internal edge fusion, edge prior information, and directed DT. The accuracy of each class is improved, especially for nonbackground categories. For data set 2, the average accuracy of our proposed method is increased by 7.2% over FCN.

### 4.2.2. Edge Improvement

The results, including the segmentation images and some zoomed-in details, are shown in Table 1: Confusion matrix of data set 1.

| Method                          | Category | Building | Forest | Farmland | Road | Others | Average acc. |
|---------------------------------|----------|----------|--------|----------|------|--------|--------------|
| Meanshift+CRF                   | Buildings| 75.20    | 8.54   | 0.43     | 8.34 | 7.49   |              |
|                                 | Forest   | 10.20    | 72.97  | 2.11     | 6.48 | 8.24   |              |
|                                 | Farmland | 0.92     | 24.02  | 26.36    | 0.32 | 48.38  | 70.33        |
|                                 | Road     | 6.97     | 4.69   | 1.10     | 58.46| 28.78  |              |
|                                 | Others   | 3.18     | 2.35   | 5.44     | 11.62| 77.42  |              |
| Meanshift+SVM                   | Buildings| 74.34    | 8.15   | 1.14     | 5.89 | 10.47  |              |
|                                 | Forest   | 10.79    | 64.69  | 13.53    | 2.73 | 8.26   |              |
|                                 | Farmland | 1.15     | 27.74  | 26.04    | 2.47 | 42.60  | 66.32        |
|                                 | Road     | 8.59     | 5.04   | 4.74     | 57.12| 24.51  |              |
|                                 | Others   | 5.22     | 3.12   | 8.16     | 10.80| 72.69  |              |
| FCN8                            | Buildings| 94.06    | 1.24   | 0.2      | 1.55 | 2.96   |              |
|                                 | Forest   | 18.12    | 71.47  | 1.26     | 2.77 | 6.37   |              |
|                                 | Farmland | 3.50     | 12.39  | 51.34    | 7.16 | 25.61  | 79.30        |
|                                 | Road     | 10.26    | 2.75   | 0.6      | 61.11| 25.29  |              |
|                                 | Others   | 5.18     | 1.51   | 0.68     | 6.17 | 86.45  |              |
| DeepLab                         | Buildings| 95.38    | 0.67   | 0.21     | 0.77 | 2.97   |              |
|                                 | Forest   | 18.33    | 74.57  | 0.25     | 1.23 | 5.61   |              |
|                                 | Farmland | 2.29     | 16.42  | 48.44    | 3.02 | 29.83  | 80.85        |
|                                 | Road     | 11.57    | 3.43   | 0.44     | 53.26| 31.30  |              |
|                                 | Others   | 5.20     | 1.17   | 0.16     | 2.68 | 90.79  |              |
| FCN8+DT+the fully connected CRF | Buildings| 95.32    | 0.70   | 0.28     | 0.84 | 2.86   |              |
|                                 | Forest   | 18.25    | 74.85  | 0.20     | 1.31 | 5.40   |              |
|                                 | Farmland | 2.24     | 17.33  | 47.92    | 3.12 | 29.39  | 80.90        |
|                                 | Road     | 11.57    | 3.50   | 0.44     | 55.70| 29.42  |              |
|                                 | Others   | 5.28     | 1.23   | 0.16     | 2.91 | 90.42  |              |
| FCN8+HED+DT+the fully connected CRF | Buildings| 95.41    | 0.70   | 0.19     | 0.84 | 2.86   |              |
|                                 | Forest   | 18.25    | 74.85  | 0.20     | 1.31 | 5.40   |              |
|                                 | Farmland | 2.24     | 17.33  | 47.92    | 3.12 | 29.39  | 80.94        |
|                                 | Road     | 11.57    | 3.50   | 0.44     | 55.07| 29.42  |              |
|                                 | Others   | 5.28     | 1.23   | 0.16     | 2.91 | 90.42  |              |
| Pro. approach                   | Buildings| **95.94**| 0.77   | 0.18     | 0.90 | 2.21   |              |
|                                 | Forest   | 18.03    | **75.46**| 0.23   | 1.20 | 5.09   |              |
|                                 | Farmland | 2.21     | 16.84  | **49.33**| 3.53 | 28.08  | **81.39**    |
|                                 | Road     | 11.23    | 3.45   | 0.51     | **56.78**| 28.02  |              |
|                                 | Others   | 4.98     | 1.14   | 0.16     | 3.46 | **90.27**|              |
Figure 9, respectively. It is clear that the upper right area is labeled as building (water blue in Figure 4(b)), which corresponds to uneven color in the Pauli image. Besides, the road (pink in ground truth) shows two different colors in the original image (black and dark green in Figure 4(a)). Since the rest of comparative methods and the proposed one are network based, they achieve better performance than the traditional algorithm. To show the improvement of DDT, some details of segmentation result are magnified and presented in Figure 9; the segmentation images by FCN8+DT+the fully connected CRF and the presented one are quite similar. From the four zoomed-in areas, it is obvious that the result by the proposed approach matches the ground truth more closely.

The average accuracy, which reflects improvement in the whole image (edge area and nonedge area), is insufficient to measure the edge enhancement. To verify the contributions of the proposed approach, all improved pixels in the marginal area are counted in Table 3 and Table 4. The number of boundary points is rather smaller than that of the total pixels. Thus, the contribution is evaluated by the ratio of edge-improved points to all boundary pixels. With DeepLab as a comparison basis, the edge accuracy of FCN8+DT+the fully connected CRF and the proposed algorithm increases by 0.9% and 4.4% for data set 1, respectively. For data set 2, the edge accuracy increases for these two methods are 0.2% and 2.2%, respectively. It is indicated that DDT guided by prior information does have a good effect on edge segmentation.

### 4.3. Discussion

There are three reasons for the improvements of our proposed method. (1) Based on Bayesian framework, HED is introduced to provide the edge prior information. In the segmentation process, the edge map and the edge distance map play an important role in improving the accuracy of segmentation. (2) FCN8 is chosen to improve the initial classification accuracy replacing traditional classifiers, and the fully connected CRF is employed to improve the classification accuracy of the points since it introduces the prior information of the surrounding pixel classification categories. (3) Internal DT and modified DT (DDT) contribute

| Method                          | Category   | Others | Shrub | Higher crop | Untilled glebe | Building | Forest | Average acc. |
|---------------------------------|------------|--------|-------|-------------|----------------|----------|--------|--------------|
| FCN8                            | Others     | 81.62  | 1.86  | 4.76        | 6.82           | 2.10     | 2.84   |              |
|                                 | Shrub      | 33.45  | 65.23 | 0.87        | 0.12           | 0.19     | 0.14   |              |
|                                 | Higher crop| 47.69  | 1.17  | 43.98       | 4.85           | 1.42     | 0.90   | 75.34        |
|                                 | Untilled glebe | 11.88 | 0.09  | 2.72        | 82.89          | 1.45     | 0.97   |              |
|                                 | Building   | 34.69  | 0.36  | 1.54        | 1.80           | 44.92    | 16.68  |              |
|                                 | Forest     | 9.06   | 0.75  | 0.43        | 0.73           | 7.74     | 81.29  |              |
| DeepLab                         | Others     | 73.29  | 0.76  | 25.85       | 0.02           | 0.06     | 0.01   | 78.83        |
|                                 | Shrub      | 1.29   | 49.23 | 41.26       | 2.15           | 5.07     | 1.01   |              |
|                                 | Higher crop| 2.33   | 3.92  | 81.26       | 2.57           | 7.49     | 2.44   |              |
|                                 | Untilled glebe | 0.03 | 0.72  | 19.81       | 69.94          | 1.26     | 8.23   |              |
|                                 | Building   | 0.15   | 1.16  | 10.73       | 2.34           | 84.60    | 1.02   |              |
|                                 | Forest     | 0.07   | 0.22  | 4.75        | 5.33           | 0.62     | 89.00  |              |
| FCN8+DT+the fully connected CRF | Others     | 82.77  | 1.49  | 2.76        | 7.28           | 3.23     | 2.47   | 79.48        |
|                                 | Shrub      | 25.03  | 74.71 | 0.14        | 0.06           | 0.03     | 0.02   |              |
|                                 | Higher crop| 39.64  | 0.94  | 52.88       | 3.29           | 2.23     | 1.03   |              |
|                                 | Untilled glebe | 13.56 | 0.13  | 0.80        | 82.37          | 2.11     | 1.01   |              |
|                                 | Building   | 26.19  | 0.31  | 0.14        | 0.48           | 69.91    | 2.97   |              |
|                                 | Forest     | 10.04  | 0.08  | 0.04        | 0.27           | 3.56     | 86.01  |              |
| FCN8+HED+DT+the fully connected CRF | Others | 84.56  | 0.75  | 2.40        | 6.45           | 2.85     | 2.99   | 80.73        |
|                                 | Shrub      | 24.68  | 74.72 | 0.22        | 0.26           | 0.09     | 0.03   |              |
|                                 | Higher crop| 39.16  | 0.55  | 53.20       | 3.88           | 2.21     | 1.00   |              |
|                                 | Untilled glebe | 14.01 | 0.14  | 0.75        | 81.93          | 2.19     | 0.98   |              |
|                                 | Building   | 23.47  | 0.16  | 0.20        | 0.38           | 73.23    | 2.57   |              |
|                                 | Forest     | 9.58   | 0.09  | 0.06        | 0.26           | 3.58     | 86.43  |              |
| Pro. approach                   | Others     | **86.90** | 0.39 | 1.63        | 6.96           | 2.21     | 1.90   | 82.54        |
|                                 | Shrub      | **30.82** | **69.00** | 0.17 | 0             | 0     | 0     |              |
|                                 | Higher crop| 40.44  | 0.11  | **53.11**   | 3.37           | 2.25     | 0.71   |              |
|                                 | Untilled glebe | 10.08 | 0.07  | 0.65        | **85.92**      | 2.48     | 0.80   |              |
|                                 | Building   | 16.63  | 0.03  | 0.10        | 0.47           | **80.68**| 2.09   |              |
|                                 | Forest     | 4.97   | 0.08  | 0.03        | 0.29           | 8.04     | **86.58**|              |
to improving segmentation accuracy. As a constraint network, DDT analyzes the core problem of original DT and improves it. DDT combines the improved diffusion method and the hole-filling method for external fusion, which enhances the correctness of the edge distribution, improves the classification accuracy of points near the edge, and further improves the overall accuracy.

5. Conclusions

An edge prior Bayesian semantic segmentation network for SAR image is proposed in this study. The raw segmentation result is firstly derived from the likelihood network (FCN8). Afterwards, the edge map from HED network is fused with the map from intermediate FNC8 layers, to obtain more accurate boundaries as prior information. In the last stage, the newly defined edge distance is utilised for eliminating inappropriate diffusion directions. Besides, the DDT method, which redefines the domain transform density, is proposed to improve the segmentation performance. Experiments about the proposed approach and six comparative methods are conducted on the two data sets. The experimental results demonstrate that the proposed method has a good effect on edge correction, improving the overall accuracy as well.

However, Pauli SAR image is segmented directly without preprocessing in this paper. In the future work, we will focus on preprocessing the SAR image before segmentation to improve the accuracy.

Data Availability

The image data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare no conflict of interest.

Authors’ Contributions

Chu He and Zishan Shi conceived and designed the experiments. Dehui Xiong and Zishan Shi performed the experiments and analyzed the results. Chu He and Peizhang Fang wrote the paper. Bokun He and Mingsheng Liao revised the paper.

Acknowledgments

The authors would like to thank the National Key Research and Development Program of China (grant number 2016YFC0803000), the National Natural Science Foundation of China (grant numbers 41371342 and 61331016), and the Hubei Innovation Group (grant number 2018CFA006).

References

[1] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, 2002no. 5, pp. 603–619, 2002.
[2] N. Amoda and R. K. Kulkarni, "Image segmentation and detection using watershed transform and region based image retrieval," International Journal of Emerging Trends and Technology in Computer Science, vol. 2, pp. 89–94, 2013.
[3] M. Salehi, M. R. Sahebi, and Y. Maghsoudi, “Improving the accuracy of urban land cover classification using Radarsat-2 PolSAR data,” IEEE Journal of selected topics in applied Earth Observations and Remote Sensing, vol. 7, no. 4, pp. 1394–1401, 2013.

[4] A. Saepuloh, K. Koike, and M. Omura, “Applying Bayesian decision classification to Pi-SAR polarimetric data for detailed extraction of the geomorphologic and structural features of an active volcano,” IEEE Geoscience and Remote Sensing Letters, vol. 9, no. 4, pp. 554–558, 2012.

[5] J. S. Lee, M. R. Grunes, and E. Pottier, “Quantitative comparison of classification capability: Fully polarimetric versus dual and single-polarization SAR,” IEEE Transactions on Geoscience and Remote Sensing, vol. 39 no. 11, pp. 2343–2351, 2001.

[6] X. He, R. S. Zemel, and M. A. Carreira-Perpinan, “Multiscale conditional random fields for image labeling,” in Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004, vol. 2, Washington, DC, USA, 2004.

[7] C. He, S. Li, Z. Liao, and M. Liao, “Texture classification of PolSAR data based on sparse coding of wavelet polarized textons,” IEEE Transactions on Geoscience and Remote Sensing, vol. 51, no. 8, pp. 4576–4590, 2013.

[8] J. Liu, X. Wen, Q. Meng, H. Xu, and L. Yuan, “Synthetic aperture radar image segmentation with reaction diffusion level set evolution equation in an active contour model,” Remote Sensing, vol. 10, no. 6, p. 906, 2018.

[9] P. Zhang, M. Li, Y. Wu, and H. Li, “Hierarchical conditional random fields model for semisupervised SAR image segmentation,” IEEE Transactions on Geoscience and Remote Sensing, vol. 53, no. 9, pp. 4933–4951, 2015.

[10] F. Liu, Y. Duan, L. Li et al., “SAR image segmentation based on hierarchical visual semantic and adaptive neighborhood multinomial latent model,” in IEEE Transactions on Geoscience and Remote Sensing, vol. 54 no. 7, pp. 4287–4301, 2016.

[11] S. Cloude, “Group theory and polarisation algebra,” Optik, vol. 75, no. 1, pp. 26–36, 1986.

[12] Q. Chen, L. Li, Q. Xu, S. Yang, X. Shi, and X. Liu, “Multi-feature segmentation for high-resolution polarimetric SAR data based on fractal net evolution approach,” Remote Sensing, vol. 9, no. 6, p. 570, 2017.

[13] F. Lang, J. Yang, S. Yan, and F. Qin, “Superpixel segmentation of polarimetric synthetic aperture radar (SAR) images based on generalized mean shift,” Remote Sensing, vol. 10, no. 10, p. 1592, 2018.

[14] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440, 2015.

[15] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,” in IEEE transactions on pattern analysis and machine intelligence, vol. 40 no. 4, pp. 834–848, 2017.

[16] S. Zheng, S. Jayasumana, B. Romera-Paredes et al., “Conditional random fields as recurrent neural networks,” in Proceedings of the IEEE international conference on computer vision, pp. 1529–1537, 2015.

[17] R. Guo, J. Liu, N. Li et al., “Pixelwise classification method for high resolution remote sensing imagery using deep neural networks,” ISPRS International Journal of Geo-Information, vol. 7, no. 3, p. 110, 2018.

[18] Y. Liu, Q. Ren, J. Geng, M. Ding, and J. Li, “Efficient patch-wise semantic segmentation for large-scale remote sensing images,” Sensors, vol. 18, no. 10, p. 3232, 2018.

[19] L. C. Chen, J. T. Barron, G. Papandreou, K. Murphy, and A. L. Yuille, “Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4545–4554, 2016.

[20] E. S. L. Gastal and M. M. Oliveira, “Domain transform for edge-aware image and video processing,” ACM Transactions on Graphics, vol. 30, no. 4, p. 1, 2011.

[21] P. Kränhenbühl and V. Koltun, “Efficient inference in fully connected crfs with gaussian edge potentials,” in Advances in neural information processing systems, pp. 109–117, 2011.

[22] J. Canny, “A computational approach to edge detection,” in IEEE Transactions on pattern analysis and machine intelligence, no. 6, pp. 679–698, 1986.

[23] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, “LSD: a line segment detector,” Image Processing On Line, vol. 2, no. 4, pp. 35–55, 2012.

[24] R. Fjortoft, A. Lopes, P. Marthon, and E. Cubero-Castan, “An optimal multiedge detector for SAR image segmentation,” IEEE Transactions on Geoscience and Remote Sensing, vol. 36, no. 3, pp. 793–802, 1998.

[25] H. Zhang, Z. Wen, Y. Liu, and G. Xu, “Edge detection from RGB-D image based on structured forests,” Journal of Sensors, vol. 2016, 10 pages, 2016.

[26] M. Tabb and N. Ahuja, “Multiscale image segmentation by integrated edge and region detection,” IEEE Transactions on Image Processing, vol. 6, no. 5, pp. 642–655, 1997.

[27] S. Mandal, P. Viswanath, N. Yeshaswini, X. L. Dean-Ben, and D. Razansky, “Multiscale edge detection and parametric shape modeling for boundary delineation in optoacoustic images,” in 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 707–710, Milan, Italy, 2015.

[28] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, and U. Stilla, “Classification with an edge: improving semantic image segmentation with boundary detection,” ISPRS Journal of Photogrammetry and Remote Sensing, vol. 135, pp. 158–172, 2018.

[29] S. Xie and Z. Tu, “Holistically-nested edge detection,” in Proceedings of the IEEE international conference on computer vision, pp. 1395–1403, 2015.

[30] C. He, P. Fang, D. Xiong, W. Wang, and M. Liao, “A point pattern chamfer registration of optical and SAR images based on mesh grids,” Remote Sensing, vol. 10, no. 11, p. 1837, 2018.

[31] X. Su, C. He, Q. Feng, X. Deng, and H. Sun, “A supervised classification method based on conditional random fields with multiscale region connection calculus model for SAR image,” IEEE Geoscience and Remote Sensing Letters, vol. 8, no. 3, pp. 497–501, 2011.

[32] Y. Maghsoudi, M. J. Collins, and D. G. Leckie, “Radarsat-2 polarimetric SAR data for boreal forest classification using SVM and a wrapper feature selector,” in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 6, no. 3, pp. 1531–1538, 2013.