DABNet: Deformable Contextual and Boundary-Weighted Network for Cloud Detection in Remote Sensing Images

Qibin He, *Graduate Student Member, IEEE*, Xian Sun, *Senior Member, IEEE*, Zhiyuan Yan, *Member, IEEE*, and Kun Fu, *Member, IEEE*

***Abstract*— In recent years, deep convolutional neural networks (DCNNs) have made significant progress in cloud detection tasks, and the detection accuracy has been greatly improved. However, most existing CNN-based models have high computational complexity, which limits their practical application, especially for spaceborne optical remote sensing. In addition, most of the methods cannot make adaptive adjustments based on the structural information of the clouds, and blurred boundaries often occur in the detection results. In order to address these problems, this article proposes a lightweight network (DABNet) to achieve high-accuracy detection of complex clouds, not only a clearer boundary but also lower false-alarm rate. Specifically, a deformable context feature pyramid module is proposed to improve the adaptive modeling capability of multiscale features. Besides, a boundary-weighted loss function is designed to direct the network to focus on cloud boundary information and optimize the relevant detection results. The proposed method has been validated on two data sets: the public GF-1 WFV benchmark and our self-built GF-2 cloud detection data set with higher spatial resolution. The experimental results exhibit that DABNet achieves state-of-the-art performance while only using 4.12M parameters and 8.29G multiadds.**

***Index Terms*— Cloud detection, deformable context, lightweight network, remote sensing images.**

I. Introduction

With the development of aerospace technology, the amount of remote sensing data has increased explosively. Since more than 66% earth surface is covered with clouds [1], most optical image data are inevitably contaminated and even partially become invalid data. Cloud detection is an effective solution to reduce these redundant invalid data. Especially, for spaceborne optical remote sensing, high-accuracy on-orbit cloud detection is conducive to speeding up subsequent data transmission and processing. Due to the limited resources of the spaceborne platform, it is of great significance to study efficient cloud detection methods.

Over the years, studies on cloud detection in remote sensing images have been widely carried out, and a large number of methods [2]–[9] have been proposed. These methods may be roughly separated into two categories: threshold-based method and statistical-based method. The former method sets thresholds according to information, such as spectral reflectance and brightness temperature to detect clouds. The threshold is not a universal value. Instead, it is usually set based on a specific satellite image (e.g., ISCCP [4], APOLLO [5], and MODIS [6]). The statistical-based method is to design handcrafted features based on physical attributes (i.e., texture, color, and geometry) and then use a statistical learning algorithm to classify pixels [7], [8]. Although the classification procedure is based on learning, the quality of the features still depends on manual design. The practical application scenarios of such methods are limited because of the diversity of the environment in remote sensing images.

Recently, deep convolutional neural networks (CNN) have made important progress in cloud detection with their powerful representation learning capabilities and have made significant progress [10]–[14]. By establishing a complex nonlinear mapping between input and output, the CNN-based method achieves automatic feature extraction and classification. Although the CNN-based method has implemented state-of-the-art results, it still faces some limitations in cloud detection.

First, a majority of existing CNN-based models enhance the performance by increasing the width or depth of the network. The models rely on high computational complexity to achieve high-accuracy detection results. As a result, the huge number of parameters of these complex models limits their applications on resource-constrained spaceborne platforms. In addition, the diversity of cloud shapes and sizes, as well as the complexity of the background, make cloud

Manuscript received November 1, 2020; revised November 28, 2020; accepted December 14, 2020. Date of publication January 5, 2021; date of current version December 3, 2021. This work was supported by the China High Resolution Major Scientific and Technological Special Project under Grant GFZX0404120201. *(Corresponding author: Xian Sun.)*

Qibin He, Xian Sun, and Kun Fu are with the Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China, also with the Key Laboratory of Network Information System Technology (NIST), Institute of Electronics, Chinese Academy of Sciences, Beijing 100190, China, also with the University of Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100190, China (e-mail: heqibin20@mails.ucas.ac.cn; sunxian@mail.ie.ac.cn; fukun@mail.ie.ac.cn).

Zhiyuan Yan is with the Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China, and also with the Key Laboratory of Network Information System Technology (NIST), Institute of Electronics, Chinese Academy of Sciences, Beijing 100190, China (e-mail: yanzy@aircas.ac.cn).

Digital Object Identifier 10.1109/TGRS.2020.3045474

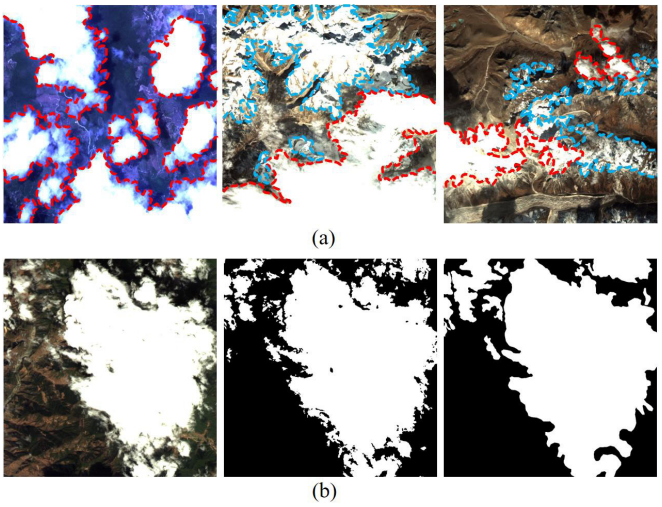

Fig. 1. (a) Illustration of the diverse cloud shapes and sizes, and complex background, with red and blue dashed lines depicting cloud and snow areas, respectively. (b) Illustration of the various geometric structure of the cloud boundary. (Left) Original remote sensing image. (Middle) ground truth. (Right) blurred boundary problem that often occurs in the detection results.

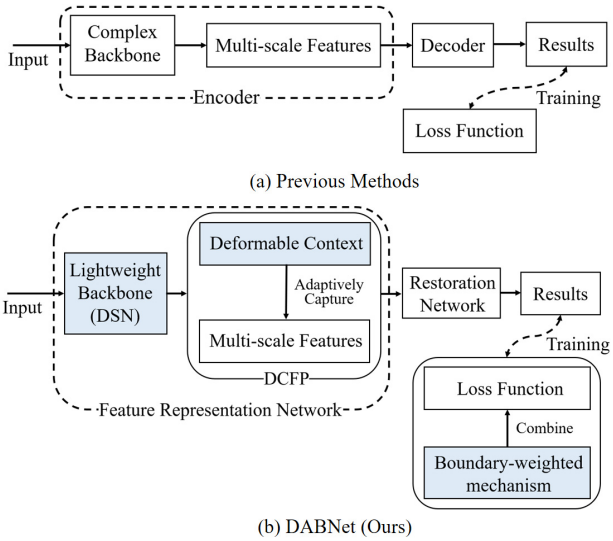

Fig. 2. Illustration of the cloud detection pipeline. (a) Previous methods usually use complex backbone networks and fixed receptive fields for feature coding. (b) DABNet (Ours) uses the lightweight backbone to minimize the amount of calculation and parameters and adaptively captures multiscale features by aggregating deformable context. In the training process, the boundary-weighted (BW) loss function is introduced to optimize the boundary detection results. Comparing previous methods, our DABNet is simple and effective.

detection tasks extremely challenging, as shown in Fig. 1(a). Existing CNN-based methods use fixed filter receiving fields to capture contextual features, which means that it is difficult to adaptively establish remote dependencies based on the physical characteristics of the cloud. Hence, even for complex models with huge parameters, visually similar snow pixels in the cloud–snow coexistence region often being misclassified as clouds. What is more, another challenge is the various geometric structure of the cloud boundary, as illustrated in Fig. 1(b). Affected by the imaging angle and daylight intensity, the boundary is usually rich in spatial detail information. Most previous methods only focus on regional accuracy but pay less attention to the boundary quality, leading to the blurred and noisy boundary in the detection results.

Therefore, it is very significant to design a high-accuracy but lightweight architecture to address the above problems. The most straightforward way to construct a lightweight network is to keep the number of model parameters and calculation operations (multiadds) as small as possible. On this basis, a novel cloud detection framework called deformable contextual and BW network (DABNet) is proposed. Different from previous methods, the proposed DABNet extracts features through an efficient feature representation network (FRN) and uses a BW loss function to optimize the boundary quality in the detection results, as demonstrated in Fig. 2. To build FRN, we design the lightweight backbone network Dilated ShuffleNet (DSN), inspired by compact model ShuffleNetV2 [45]. DSN not only guarantees a significant reduction in the amount of calculation and parameters but also optimizes the detection of small clouds by removing some pooling and introducing dilated convolutions. Besides, since the physical characteristics of the clouds are extremely complex, a deformable context feature pyramid (DCFP) module is introduced to enhance the category semantics of features. Compared with previous multiscale context capture approaches [52], [53], our DCFP can adaptively adjust the filter receptive fields based on the shapes and sizes of the clouds. Such a mechanism enables our model to construct long-range dependencies in multiple scale spaces. Moreover, we define a BW loss function that is a monotonically decreasing function about the boundary distance. By giving more weight to the boundary in network training instead of treating all pixels equally, the problem of the blurred boundary in cloud detection can be effectively improved.

Above all, the main contributions of this article are as follows.

1) Different from previous CNN-based methods, we present an efficient lightweight framework for cloud detection in remote sensing images, namely DABNet, which implements state-of-the-art detection accuracy with much fewer parameters and multiadds.
2) Considering the structural diversity of the clouds and the complexity of the background, the DCFP module is proposed to adaptively adjust the filter receptive field and capture long-range contextual features from different scale spaces.
3) For the blurred boundary of cloud detection, a novel BW loss function is designed, which makes the network training pay more attention to the pixels near the boundary and improves the clarity of the boundary.
4) To validate the effectiveness of our framework, we build a challenging cloud detection data set called AIR-CD, with higher spatial resolution and more representative land-cover types. The data set will be open to the community to provide support for research in the field of cloud detection (https://github.com/aicyberteam/air-cd).

The experimental results on our data set and another public benchmark [15] demonstrate that our DABNet implements a better detection performance than the prevalent methods, proving the superiority of our model.

The rest of this article is organized as follows. In Section II, we briefly review the related work of this method. Section III introduces the architecture of our model in detail. Section IV shows the data sets and implementation details, as well as ablation studies and experimental results. Finally, we summarize the method in Section V.

II. Related Works

Cloud detection has been widely researched for several years. In this section, we will briefly review some works related to our proposed method.

A. CNN-Based Cloud Detection Method

In recent years, CNN-based models [16]–[21] have made significant progress due to their automatic feature extraction capabilities. Xie *et al.* [22] first designed a cloud detection method according to shallow CNN, which only contained four convolution layers and achieved impressive performance. Specifically, the image was segmented into superpixels by linear iterative clustering method, and then, the multiscale features extracted by superpixels were classified by CNN. At the same time, Ye *et al.* [23] proposed to use the Fisher vector coding to aggregate spatial features and map high-dimensional features of the original deep convolution features and then obtained fine texture features and high-level semantic information through the hierarchical convolution layer. Shi *et al.* [25] introduced deep convolution activation feature to enrich texture information and then used SVM for feature classification. While using CNN to extract features could improve the accuracy, the simple model structure is still less practical in complex scenarios, especially for visually similar background noise, such as snow.

Zhan *et al.* [26] proposed to utilize fully convolutional network for cloud detection and discussed that the depth feature had richer discrimination information. Subsequently, more and more works [27]–[31] focused on improving by designing more complex end-to-end fully convolutional network architecture. For example, by combining context semantics with multilevel feature fusion mechanism, Yan *et al.* [27] proposed MFFSNet, which has more than 100 layers and can implement excellent cloud detection accuracy.

However, it is not free to use a more complex network structure to improve detection performance: it comes at the cost of increasing computing resources and time, which makes it hard to deploy to the resource-constrained spaceborne computing platform. Moreover, most of these methods only focus on regional accuracy, which easily leads to insufficient robustness in the detection performance of the cloud–snow coexistence area and blurred boundaries.

B. Compact Model

Commonly, complex networks have better performance, but the high computing resource and storage space consumption limit real-world applications. This results in a series of studies on compact model design, to achieve a balance between the model accuracy and efficiency in the case of limited calculations. Iandola *et al.* [32] used 1×1 convolution instead of 3×3 convolution to build an efficient neural network, which could achieve the same accuracy as AlexNet [38] with $50\times$ less parameters. Chollet [39] utilized depthwise convolution to make full use of model parameters, which increases network depth and improves model representation capability while reducing parameters. Howard *et al.* [41], [43] and Sandler *et al.* [42] proposed inverse residual block according to depthwise separable convolution and achieved better performance through AutoML technology. Zhang *et al.* [44] improved the exchange of information flow between channel groups by introducing channel shuffle operation. Ma *et al.* [45] further considered the actual environmental constraints on the target hardware for compact design, which enhances model representation capability while reducing parameters. Although the above methods are proposed for natural images, they are instructive for the research of lightweight architecture in remote sensing scenes. Compared with vision tasks, a compact model design in remote sensing is more challenging.

C. Multiscale Context

Multiscale context [46]–[52] is widely used in feature modeling, especially for objects with large-scale variation. Image pyramid [46] is a basic method to implement multiscale representation. Badrinarayanan *et al.* [47] and Ronneberger *et al.* [48] used the encoder–decoder architecture to aggregate high- and low-level features. Zhao *et al.* [52] and Chen *et al.* [53] designed pyramid pooling module (PPM) and atrous spatial pyramid pooling (ASPP) module to, respectively, encode multiscale contextual semantics. In recent years, more and more studies on multiscale context [56], [57] have emerged in the field of remote sensing. In terms of cloud detection task, Shao *et al.* [61] used parallel filters with various kernel sizes to enhance multiscale context, to find the optimal local sparse structure. Yang *et al.* [40] used dilated filter in each parallel filter path to further enlarge the receptive field. Li *et al.* [29] scaled and merged the feature maps of different scales, enriching the feature semantics. These methods are effective to some extent, but their filter receptive fields are fixed, not in an adaptive manner.

III. Methodology

DABNet is proposed to construct a lightweight framework for high-accuracy pixel-level cloud detection through feature extraction and training optimization. On the one hand, the multiscale feature adaptive representation is integrated into the feature extraction network to build long-range dependencies between relevant pixels. On the other hand, to optimize the boundary blur, a BW mechanism is introduced in the training procedure. The proposed DABNet framework is shown in Fig. 3, including three key components: 1) lightweight backbone network DSN; 2) DCFP module; and 3) BW loss function. Sections II-A–II-E start with the overview first and then the detailed implementations of the three key components in the framework.

A. Overview

To begin with, we describe the problem as follows. Given the remote sensing data $\boldsymbol{r}$ for cloud detection, the backbone

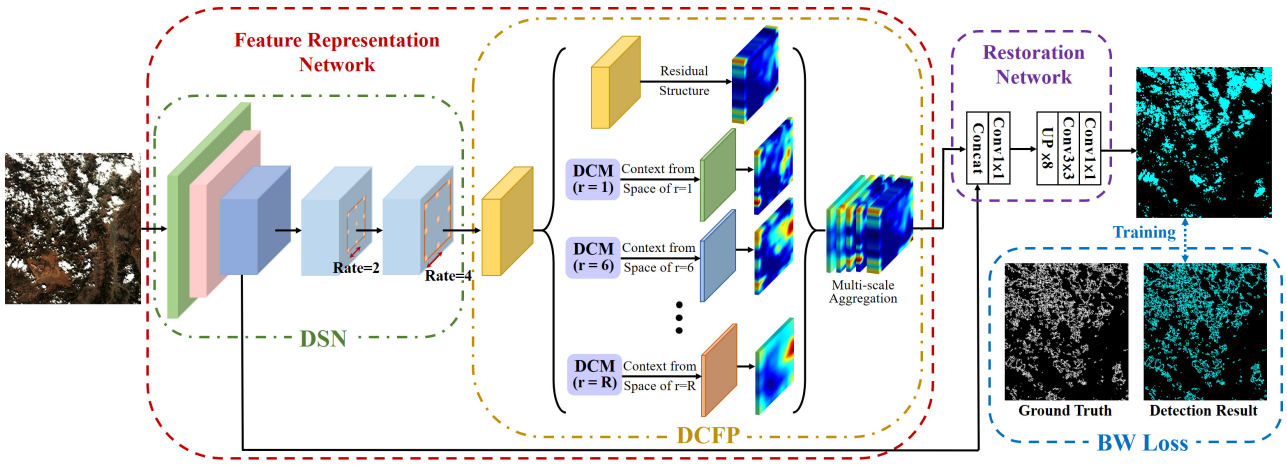

Fig. 3. Illustration of the overall framework of our proposed DABNet, including three key components: DSN, DCFP module, and BW loss function. The original remote sensing image is input into the DSN, where the convolutional features on clouds are extracted with a small amount of calculation and parameters. Then, the convolutional features are input into the DCFP module to output the long-range dependencies captured from different scale spaces. Finally, the detection result is inferred based on the multiscale context and trained with the BW loss function. BW loss makes network training more focus on the cloud boundary, improving the clarity of the boundary.

Algorithm 1 DABNet

Require: An initialized network $\mathcal{N}_{\mathrm{DAB}}$, a labeled cloud detection dataset D and test data T.

1: **Step 1: Train the network**
2: **repeat**
3: Randomly select a batch $\{\boldsymbol{r}_i^D\}_{i=1}^N$ from D;
4: Optimize $\mathcal{N}_{\mathrm{DAB}}$ and update the network parameters $\boldsymbol{\theta}$ following Eq. 3;
5: **until** convergence
6: **Step 2: Process the detection results**
7: Obtain the feature map $\boldsymbol{m}^T$ from T utilizing feature representation network $\mathcal{N}_{\mathrm{FR}}$ with Eq. 1;
8: Restore the resolution of $\boldsymbol{m}^T$ and generate corresponding detection results;

Ensure: Cloud detection network, and detection results.

network is used to calculate dense convolutional features. Usually, backbones with complex structures can output features with stronger semantics, but they will cause a substantial increase in the amount of calculations and parameters. A direct idea toward this problem is to design a lightweight backbone to complete the extraction of convolutional features, so as to achieve efficient computation. However, this idea ignores the need of features for category semantics and weakens the detection performance. To address this problem, contextual features can be used to boost detection accuracy. Although previous methods [11], [27], [28] have designed a variety of context capture modules, they basically cannot be adaptively adjusted, which limits the feature information. They also blindly use mature complex networks as backbones to improve performance.

As mentioned earlier, this article aims to build a lightweight high-accuracy cloud detection framework. To achieve this goal, we first design a lightweight backbone $\mathcal{N}_{\mathrm{DS}}$ to extract convolutional features. Then, we convert the features into a multiscale pyramid representation and adaptively build context tensor at each scale. In mathematics, we introduce

$$\mathcal{N}_{\mathrm{FR}} = \mathcal{N}_{\mathrm{DCFP}}(\mathcal{N}_{\mathrm{DS}}(\boldsymbol{r})) \tag{1}$$

to represent the FRN, where $\mathcal{N}_{\mathrm{DCFP}}$ denotes using context information to enhance category semantics. On this basis, the restoration subnetwork $\mathcal{N}_{\mathrm{R}}$ is utilized to extend $\mathcal{N}_{\mathrm{FR}}$ to form the inference architecture of the overall network $\mathcal{N}_{\mathrm{DAB}}$ by concatenation and fusion

$$\mathcal{N}_{\mathrm{DAB}} = \mathcal{N}_{\mathrm{FR}} \star \mathcal{N}_{\mathrm{R}} \tag{2}$$

which can effectively improve the regional accuracy with less computational cost. What is more, since the boundaries of the clouds are rich in tiny details, the problem of blurred boundaries is prone to appear in the detection results. Collegio *et al.* [33] indicate that people will more focus on the details of objects in the real world, which is beneficial for establishing visual cognition. Hence, if the model can pay more attention to cloud boundary details, such as people, the detection quality will be effectively boosted. Specifically, we define a novel BW loss function $\mathcal{L}_{\mathrm{bw}}$ and train the overall network to update parameters $\boldsymbol{\theta}$ with the following equation:

$$\mathcal{N}_{\mathrm{DAB}}^{\boldsymbol{\theta}} = \arg\min_{\mathcal{N}_{\mathrm{DAB}} \in \mathcal{N}} \max_{\boldsymbol{\theta}} \mathcal{L}_{\mathrm{bw}}(\mathcal{N}_{\mathrm{DAB}}, \boldsymbol{\theta}) \tag{3}$$

where $\mathcal{N}$ is the hypothesis space, similar to the approach proposed in [35]. $\mathcal{N}$ contains multiple model subsets composed of DCFP modules with different structures. The explicit structure information of DCFP is described in the following. In Algorithm 1, we present more details of our DABNet.

B. DSN

In order to boost the efficiency of the whole network as much as possible, we design the lightweight backbone, DSN, based on the compact model ShuffleNetV2 [45]. Too many downsampling operations in the original ShuffleNetV2 [45]

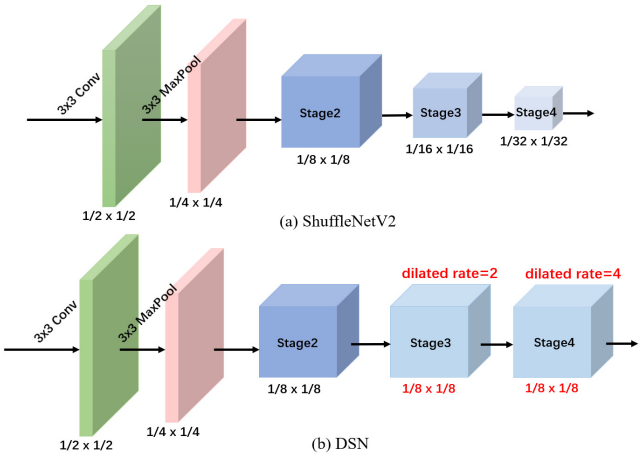

Fig. 4. Converting ShuffleNetV2 into DSN. (a) Original ShuffleNetV2. (b) Converted DSN. Downsampling in stage 3 and stage 4 is removed, resulting in the same resolution as stage 2 for all layers in stage 3 and stage 4. To compensate for the reduction of the receptive field, dilated convolutions with the rate of 2 and 4 are used in stage 3 and stage 4, respectively.

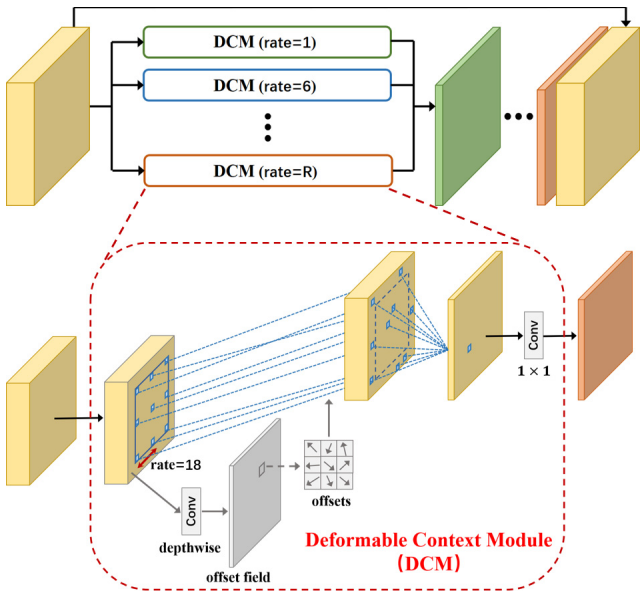

Fig. 5. Illustration of the DCFP structure, including four parallel DCMs with different DRs. Each DCM adaptively extracts context information of the specific scale by dilated convolution in deformable paradigm.

structure cause the loss of spatial details, which is not beneficial for cloud detection task since the clouds in remote sensing images are usually widely distributed, whose spatial attributes are very important for feature detection. For example, some small clouds are difficult to detect because they cover fewer pixels and are not spatially dominant. The signals of small clouds are easily suppressed by the background and may even be lost due to downsampling. However, if we keep high-resolution throughout the model and ensure that the output signal can cover the input domain densely, backpropagation can learn to save the important spatial information about the small cloud.

The simplest way to enlarge the resolution of the network is to remove the downsampling directly from some inner layers. However, the removal of downsampling will reduce the receptive field of the subsequent layer, which is not conducive to the extraction of context information. Here, we use dilated convolution to enlarge the high-level receptive field. Specifically, we remove the downsampling in stage 2 and stage 3, and, respectively, replace some standard convolution with the dilated convolution with the rate of 2 and 4, so that it has the same size of receptive field as the corresponding unit in the original model. As illustrated in Fig. 4, different from the original ShuffleNetV2 [45] downsampling the input image by 32 times, the converted DSN only downsamples eight times. The number of layers and parameters has not changed, maintaining the characteristics of lightweight. The features extracted by DSN retain most of the information needed to analyze the input image at the pixel level, which is helpful to detect small clouds. Moreover, to accelerate the convergence of our network and avoid overfitting, we add the batch normalization layers [59] after some convolutional layers.

C. DCFP Module

We extend the above DSN architecture with a DCFP mechanism, which is learned in a supervised manner from the existing remote sensing data. Previous studies [34], [36] demonstrate that human visual cognition is guided by the structural factors of objects. By attaching the deformable context module (DCM), CNN is enforced to adaptively generate a more explicit spatial feature representation. This helps to disentangle the structural information between similar objects (i.e., cloud and snow) and enables the model to better capture long-range dependencies. In addition, the CNN architecture introduces a large number of parameters for modeling spatial patterns and requires a huge amount of labeled data for training. However, with the existing cloud detection data sets, the amount of training data is still insufficient, considering the high similarity between cloud and snow. The supervised deformable context mechanism can fully utilize the existing data to boost the generalization capability of DABNet. What is more, we utilize a feature pyramid to extract features from different scale spaces to enhance the representation information of clouds. DABNet, thus, can make more accurate predictions from a global perspective.

As demonstrated in Fig. 5, DCFP is mainly composed of multiple parallel DCMs with different dilated rates (DRs). In principle, the goal of DCFP is to calculate the context of each spatial location by using the structural information, such as the shape and size of the cloud, so as to adaptively capture of multiscale features and further enhance the category semantics. This process can be defined as

$$\mathcal{N}_{\mathrm{DCFP}} = \mathrm{Concat}(X, \mathcal{N}_{\mathrm{DCM}}(X, \boldsymbol{r}_1), \ldots, \mathcal{N}_{\mathrm{DCM}}(X, \boldsymbol{r}_n)) \quad (4)$$

where X represents the convolutional features extracted by DSN, and $\mathcal{N}_{\mathrm{DCM}}(\boldsymbol{x}, \boldsymbol{r})$ means to capture deformable context according to specific DR $\boldsymbol{r}$ for the input signal $\boldsymbol{x}$. In the experimental investigation, we find that the best performance can be obtained when the combination of DRs is set to $\{1, 6, 12, 18\}$. For details, please refer to Section IV.

In each DCM, we use the dilated convolution in a deformable paradigm with a specific DR to adaptively capture

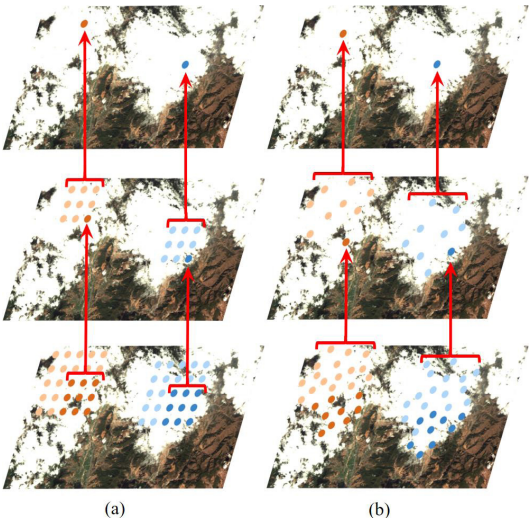

Fig. 6. Illustration of receptive field in (a) ordinary dilated convolution and (b) dilated convolution in deformable paradigm. Each is described using two layers, where the activation unit is highlighted. It is obvious that the receptive field of (a) is fixed and (b) is adaptive.

the context of the corresponding scale. The normal context capture method [53] is usually implemented by the ordinary dilated convolution. Dilated convolution uses regular grid $\boldsymbol{G}$ to sample the input feature map, and grid $\boldsymbol{G}$ defines the size and shape of the receptive field. Specifically

$$\boldsymbol{G}(\boldsymbol{r}) = \begin{Bmatrix} (-\boldsymbol{r}, \boldsymbol{r}) & (\boldsymbol{0}, \boldsymbol{r}) & (\boldsymbol{r}, \boldsymbol{r}) \\ (-\boldsymbol{r}, \boldsymbol{0}) & (\boldsymbol{0}, \boldsymbol{0}) & (\boldsymbol{r}, \boldsymbol{0}) \\ (-\boldsymbol{r}, -\boldsymbol{r}) & (\boldsymbol{0}, -\boldsymbol{r}) & (\boldsymbol{r}, -\boldsymbol{r}) \end{Bmatrix} \tag{5}$$

defines 3×3 kernel with DR $\boldsymbol{r}$. It is obvious that, on the one hand, the size of the dilated convolution receptive field is $(2r - 1)^2$, which can capture the context of different scales by setting different DRs; on the other hand, the shape and sampling location remain fixed, as shown in Fig. 6(a). However, the shapes of clouds in remote sensing image are various, so we introduce deformable paradigm to improve the capability of modeling geometric transformation. It adds 2-D direction offset to the standard grid sampling position in the dilated convolution, thereby achieving arbitrary deformation of the receptive field. Formally, for each location $\boldsymbol{l}$ in the output feature map $\boldsymbol{y}$, the dilated convolution in deformable paradigm of filter $\boldsymbol{w}$ given the input signal $\boldsymbol{x}$ and the sampling rate $\boldsymbol{r}$ is calculated as

$$\boldsymbol{y}(\boldsymbol{l}, \boldsymbol{x}, \boldsymbol{r}) = \sum_{\boldsymbol{g}_n \in \boldsymbol{G}(\boldsymbol{r})} \boldsymbol{x}(\boldsymbol{l} + \boldsymbol{g}_n + \Delta \boldsymbol{g}_n) \cdot \boldsymbol{w}(\boldsymbol{l}, \boldsymbol{g}_n) \tag{6}$$

where $\boldsymbol{g}_n$ enumerates the location in $\boldsymbol{G}(r)$, and $\Delta \boldsymbol{g}_n$ represents the offset. As shown in Fig. 5, the offset is learned from the input feature map by adding a convolutional layer. After determining the feasible variable context calculation for a single position, $\mathcal{N}_{\mathrm{DCM}}(\boldsymbol{x}, \boldsymbol{r})$ can be formulated as

$$\mathcal{N}_{\mathrm{DCM}}(\boldsymbol{x}, \boldsymbol{r}) = \{\boldsymbol{y} \mid \boldsymbol{y}(\boldsymbol{l}, \boldsymbol{x}, \boldsymbol{r}), \boldsymbol{l} \in L\} \tag{7}$$

in which L is the set of all pixel positions on output feature map.

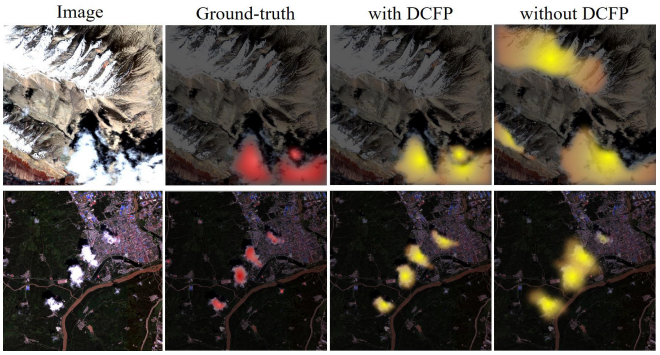

Fig. 7. Illustration of the attention maps predicted by the model with or without DCFP module on two remote sensing images. Best viewed in color.

Therefore, DCM can adjust adaptively according to the shape of the cloud, as shown in Fig. 6(b). Considering the requirement of lightweight, we calculate the convolution in DCM in terms of the depthwise mode, which can effectively reduce the calculations. Finally, the corresponding scale context is output after feature combination by a 1×1 convolution. In short, DCFP captures deformable contextual features by aggregating DCMs with different sampling rates, thereby achieving adaptively construct multiscale long-range dependencies. Multiple parallel DCM modules may ignore some useful information for learning an adjacent saliency representation between similar objects (i.e., cloud and snow), as they are more concerned with long-range contextual dependencies. For this, inspired by the recent advances of multiscale context mechanism and residual connection [54], [55], we introduce a residual structure in DCFP. Such design ensures that the original convolutional features and the enhanced features are further merged, improving the model's capability to discriminate in the cloud–snow coexistence area. Fig. 7 visualizes the attention map predicted by the model with or without the DCFP module on two images, showing that, using the deformable context mechanism, DABNet can capture the visually important areas in the complex remote sensing scene. Especially, for similar objects, such as snow, DCFP can assist the model to greatly reduce misjudgments. A more detailed quantitative analysis of the DCFP module is carried out in Section IV. Unlike previous multiscale feature extraction mechanism for cloud detection tasks, our DCFP module learns to adaptively build long-range dependencies based on the structural information of clouds.

D. Restoration Network

As shown in Fig. 3, our model directly processes the original remote sensing image to achieve efficient feature extraction. The final high-accuracy detection result is obtained by restoring the feature resolution through the restoration network (RN). Specifically, multiscale contextual features and low-dimensional features are concatenated in the input layer of the RN, further fusing category semantics and spatial details. Due to the high dimensionality of the input data, we use 1×1 convolution for dimensionality reduction to improve computational efficiency. Then, the feature map is reshaped into a high-level space by bilinear upsampling and continuous

convolution, where each channel represents the probability tensor of whether the corresponding pixel is the cloud.

E. Boundary-Weighted Loss Function

The purpose of cloud detection is to determine whether each pixel in the input image belongs to cloud, which is a pixel-level binary classification problem in essence. Based on this, we propose the concept of BW loss, which is used to optimize the boundary blur caused by downsampling in deep network. The loss in DABNet is formulated as a weighted cross-entropy loss function of each pixel. The BW loss can be defined as

$$\mathcal{L}_{\text{bw}} = -\frac{1}{\boldsymbol{p}}\sum_{i=1}^{\boldsymbol{p}} \boldsymbol{w}_i[(1-\boldsymbol{y}_i^*)\log(1-\boldsymbol{f}_i(\boldsymbol{y}_i^*|\boldsymbol{x}_i)) + \boldsymbol{y}_i^* \log \boldsymbol{f}_i(\boldsymbol{y}_i^*|\boldsymbol{x}_i)] + \lambda\|\boldsymbol{\theta}\|_2 \quad (8)$$

where $\boldsymbol{\theta}$ represents all parameters of the network, $\boldsymbol{p}$ is the number of pixels of the input image, $\boldsymbol{y}_i^*$ is the ground-truth category of pixel i, and $\boldsymbol{x}_i$ represents the unnormalized category vector output by the network of pixel i. $\boldsymbol{f}_i(\boldsymbol{y}_i^*|\boldsymbol{x}_i)$ denotes the model probability estimation that the pixel i is the ground-truth category, which is realized by the softmax unit

$$\boldsymbol{f}_i(j=\boldsymbol{y}_i^*|\boldsymbol{x}_i) = \frac{e^{\boldsymbol{x}_{ij}}}{\sum_{c\in C} e^{\boldsymbol{x}_{ic}}} \quad \forall i \in [1,\boldsymbol{p}], C=[0,1]. \quad (9)$$

To avoid overfitting, we add L_2-norm as regular term for $\mathcal{L}_{\text{bw}}$. $\boldsymbol{w}_i$ represents the weight mask of pixel i. If we set $\boldsymbol{w}_i$ to 1, the model will treat all pixels equally.

In cloud detection, due to the downsampling, the boundary is often blurred. The main reason for this phenomenon is that the pixel information at the cloud boundary is easy to mix with the surrounding pixels. When it is far away from the boundary, the adjacent pixels are usually of the same category, there is no different information mixing, and the network detection effect is better. When it is close to the boundary, the pixel information becomes diversified, and the detection difficulty becomes larger. Therefore, we hope to be able to calculate a different weight mask $\boldsymbol{w}_i$ for each pixel i, which depends on the distance from the corresponding pixel to the boundary. Thus, the network pays more attention to the pixels near the boundary and improves the boundary blur. Specifically, $\boldsymbol{w}_i$ is calculated as follows. First, through the boundary detection of the ground truth, the boundary pixel set B is obtained. Then, according to the 2-D Euclidean distance $\boldsymbol{\rho}$ from each pixel to all boundary pixels, the softmax unit is used to calculate the corresponding weight mask. This process can be formulated as

$$\boldsymbol{\rho}_i = \sum_{j\in B}\sqrt{(\boldsymbol{x}_i-\boldsymbol{x}_j)^2+(\boldsymbol{y}_i-\boldsymbol{y}_j)^2} \quad (10)$$

$$\boldsymbol{w}_i = \frac{exp(-\boldsymbol{\rho}_i)}{\sum_{k=1}^{P} exp(-\boldsymbol{\rho}_k)} + 1 \quad (11)$$

where $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ represents the coordinates of pixel i.

As shown in Fig. 8, when the weight $\boldsymbol{w}_i$ is appended to the loss, the distance from the pixel to the boundary has a significant effect on the loss and gradient. Different from the traditional binary cross-entropy loss (BCE Loss) $\mathcal{L}_{\text{bce}}$, our loss $\mathcal{L}_{\text{bw}}$ has some additional items

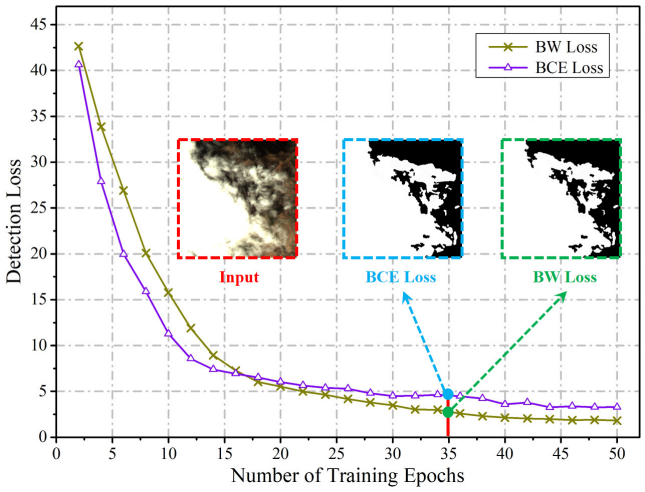

Fig. 8. Effectiveness of the proposed BW loss function, where the blue and green dashed lines, respectively, represent the output results of the network trained with the two different loss functions at epoch = 35. Compared with the BCE Loss, the output boundary of the network trained with BW Loss is clearer.

$$\begin{aligned}\Delta\mathcal{L} &= \mathcal{L}_{\text{bw}} - \mathcal{L}_{\text{bce}} \\ &= -\frac{1}{\boldsymbol{p}}\sum_{i=1}^{\boldsymbol{p}} \frac{\exp(-\boldsymbol{\rho}_i)}{\sum_{k=1}^{P}\exp(-\boldsymbol{\rho}_k)} \\ &\quad \times \left[\boldsymbol{y}_i^* \log \boldsymbol{f}_i\left(\boldsymbol{y}_i^*|\boldsymbol{x}_i\right) + \left(1-\boldsymbol{y}_i^*\right)\log\left(1-\boldsymbol{f}_i(\boldsymbol{y}_i^*|\boldsymbol{x}_i)\right)\right].\end{aligned} \quad (12)$$

$\Delta\mathcal{L}$ is related to the distance from the pixel to the boundary, and the pixel at the boundary has greater effects on the total loss. Meanwhile, the distance influences the gradient of training and further influences the detection of cloud boundary.

IV. Experimental Results

In this section, we present a comprehensive evaluation of the proposed DABNet on the AIR-CD and GF-1 WFV data sets. Specifically, we first give a brief introduction to the data sets and the implementation details. Then, we demonstrate ablation studies on the key components of DABNet. Finally, the overall performance of the model is analyzed quantitatively and qualitatively.

A. Experimental Data

To quantitatively evaluate the performance of DABNet, we conduct experiments on two typical optical remote sensing image cloud detection data sets. One is the public GF-1 WFV cloud and cloud shadow cover data set, and the other is the cloud detection data set that we collected and built from the GF-2 satellite data. Compared with the former, our data set is more challenging because of higher spatial resolution and more representative land-cover types. More details are given in the following.

1) GF-1 WFV: The GF-1 WFV cloud and cloud shadow cover data sets used in this article are created by SENDIMAGE lab [15]. This data set contains 108 full scenes of 2A level, in which all masks are labeled with clouds, cloud shadow,

clear-sky, and nonvalue pixels. The spatial resolution of the data set is 16 m, which consists of one near-infrared band and three visible bands. Since our research focuses on the detection of cloud pixels, the other classes in the mask of the data set are all set as background pixels (i.e., noncloud pixels). Specifically, the cloud shadow, clear-sky, and nonvalue pixels in the original mask are set to 0, and the cloud pixel is set to 1. In the experiment, we select 70% of the original images as the training set and 30% as the test set at random.

2) AIR-CD: In our study, we create a GF-2 high-resolution cloud detection data set and name it AIR-CD, which includes 34 full scenes collected by GF-2 satellite from different regions of China from February 2017 to November 2017. To the best of our knowledge, AIR-CD may be one of the earliest publicly available remote sensing image cloud detection data sets collected from GF-2 satellite. Moreover, compared with previous data sets, the scenes in AIR-CD are more complex and diverse, so it is more challenging to achieve high-accuracy cloud detection. As shown in Fig. 9, the image contains various land-cover types, including city, wasteland, snow, and forest. The data set consists of near-infrared and visible bands, with a spatial resolution of 4 m and a size of 7300 × 6908 pixels. Considering the radiation difference between PMS1 and PMS2 sensors in the GF-2 satellite imaging system, the scenes from both sensors are taken as experimental data to guarantee the generalization of the model. Besides, the reference cloud mask of the data set has been digitally annotated and available online (https://github.com/aicyberteam/air-cd). We believe that the open AIR-CD data set can help promote the research in cloud detection.

In the process of building the data set, experts manually labeled the position of the cloud in the image pixel by pixel and created a reference mask by, respectively, labeling the pixel values of the cloud and background with 1 and 0. Considering the high spatial resolution of the GF-2 satellite image and the relatively few cloud shadows, only cloud is labeled in the reference mask. To ensure the accuracy of the label, the reference mask has been iterative inspection and correction. In the experiment, we select 25 images from the data set as the training set and 9 as the test set at random.

In fact, a large amount of experimental data ensures that the network can be fully trained to avoid obvious bias. We also introduce L2 regulation [58] and batch normalization [59] to further reduce the deviation and improve the reliability of the results.

B. Implementation Details

1) Evaluation Metrics: To evaluate the performance of different methods in remote sensing image cloud detection, we use six widely used quantitative metrics, including the mean intersection over union (MIoU), overall accuracy (OA), F1 score, overall parameters of the model (Params), calculations (number of multiadd operations), and inference running time [27], [62]. It should be noted that MIoU, OA, and F1 are used as metrics of detection accuracy, and the greater the value, the higher the accuracy; Params, multiadds, and inference time are used as metrics of efficiency, generally the smaller the better.

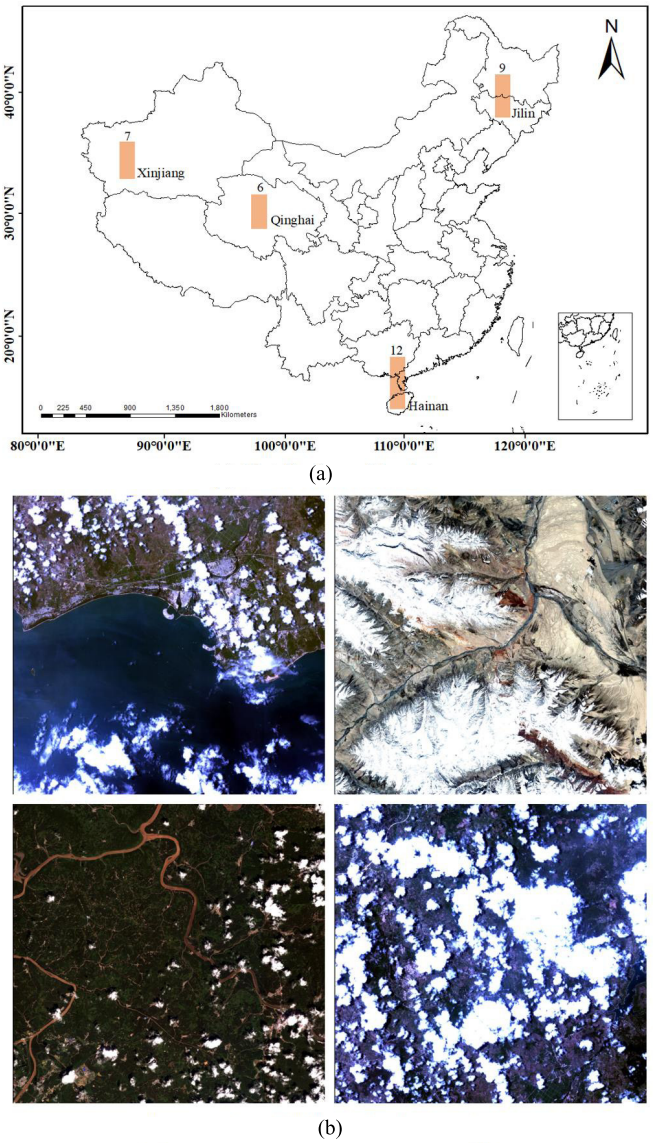

Fig. 9. Details of AIR-CD data set. These data come from four provinces in China. These provinces are distributed from 80°E to 120°E and contain a variety of land-cover types. (a) Distribution of the data. (b) Some data of different land-cover types.

2) Comparison Models: This article compares the proposed model with the representative CNN-based cloud detection methods, including MFFSNet [27], MSCFF [29], and CDNet [40]. MFFSNet [27] is a cloud detection network using multiscale fusion mechanism. MSCFF [29] effectively improves the cloud detection performance of bright surface coverage area. CDNet [40] designed a special subnetwork for multiscale features. Besides, because cloud detection is a semantic segmentation issue in the field of vision, we also compare three other typical segmentation networks: FCN [62], PSPNet [52], and DeepLabV3+ [60]. To be fair, ResNet-50 [37] is set as the backbone of all segmentation networks.

Since the lightweight backbone DSN is designed based on ShuffleNetV2 [45], we also analyze ShuffleNetV2 [45] in the ablation studies.

TABLE I

EFFECTS OF DIFFERENT BACKBONE NETWORKS ON THE AIR-CD DATA SET

Method	MIoU(%)	OA(%)	F1(%)
ShuffleNetV2 [45] + RN	82.16	87.84	85.92
DSN + RN	**84.92**	**90.03**	**88.17**

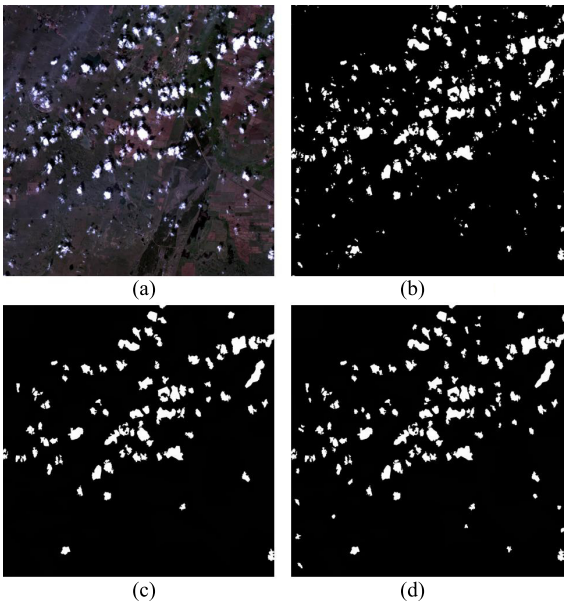

Fig. 10. Qualitative detection results with DSN compared to ShuffleNetV2. With DSN, more space information can be reserved, and more small clouds can be detected effectively. (a) Image. (b) Ground truth. (c) ShuffleNetV2 + RN. (d) DSN + RN.

3) Experimental Setting: All models in the experiment are trained on the training set and tested on the corresponding test set. All models use RGB images as input and only detect clouds, excluding cloud shadows. In the training procedure, we cut the original image into 321 × 321 pixel slices and randomly select 16 slices in each small batch as the input. We enhance the training image by flipping horizontally and rotating at a specific angle. The model is optimized by the adaptive motion estimation (Adam) algorithm. In particular, we set the hyperparameters to $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 10^{-8}$. The initial learning rate is set to $1e^{-6}$, and then, every 10 epochs are reduced by half, for a total of 50 epochs. All the experiments are implemented with a piece of NVIDIA RTX 2080ti GPU under the PyTorch framework.

C. Ablation Studies

To better verify the superiority of the proposed method, we have carried out ablation studies from the following aspects: DSN, DCFP module, and BW loss.

1) Effect of DSN: Fig. 4 shows two different backbone network structures: the DSN that we designed and the original ShuffleNetV2 [45]. We conduct ablation experiments on these backbones, and the corresponding results are illustrated in Table I and Fig. 10. In Table I, "ShufflenetV2 [45] + RN" and "DSN + RN" indicate that the backbone network is directly connected with the RN to analyze its feature extraction performance. It can be seen that using DSN as the backbone implements better detection accuracy, especially for small clouds. This is because DSN guarantees that the feature map always maintains a higher resolution, and more spatial detail information is preserved.

TABLE II

EFFECTS OF THE DCFP MODULE WITH DIFFERENT COMBINATION OF DRS ON THE AIR-CD DATA SET

Method	DR	MIoU(%)	OA(%)	F1(%)
Baseline	None	84.92	90.03	88.17
+DCFP	{1}	86.83	91.95	90.13
+DCFP	{1,6}	87.96	93.29	91.58
+DCFP	{1,6,12}	88.68	94.37	93.02
+DCFP	{1,6,12,18}	**89.75**	**95.98**	**94.83**
+DCFP	{1,6,12,18,24}	88.94	94.82	93.85

TABLE III

EFFECTS OF DIFFERENT MULTISCALE CONTEXTUAL FEATURE EXTRACTION MODULES ON THE AIR-CD DATA SET

Method	MIoU(%)	OA(%)	F1(%)
Baseline	84.92	90.03	88.17
+ PPM [52]	87.16	92.68	91.24
+ ASPP [53]	88.39	94.13	92.05
+ DCFP	**89.75**	**95.98**	**94.83**

2) Effect of DCFP Module: In order to evaluate the performance of the DCFP module, we take "DSN+RN" as the baseline so that the whole network does not consider the feature pyramid mechanism. We actually conduct a detailed investigation of the performance of the DCFP with different combinations of DRs. Specifically, inspired by Chen *et al.* [60], we design multiple combinations of DRs for experiments, as shown in Table II. Compared with the original "DSN+RN," the performance of the DCFP module with any combination of DRs will be greatly improved. The DRs of {1, 6, 12, 18} achieve the best results and increased by 4.83% (from 84.92% to 89.75%) over the baseline on MIoU. We can infer that the appropriate combination of DRs can effectively capture multiscale contextual features and construct long-range dependencies. In the subsequent experiments, we all adopted the DRs of {1, 6, 12, 18}.

As demonstrated in Table III, compared with traditional multiscale feature extraction methods, such as PPM [52] and ASPP [53], DCFP can adaptively adjust according to the shape of the cloud to obtain more context information. Fig. 11 illustrates the detection results in the relatively difficult cloud–snow coexistence area. It is obvious that fewer pixels are misclassified using DCFP. This is because DCFP can capture more abundant discrimination information.

3) Effect of Boundary-Weighted Loss: To validate the effect of the loss function, we train the network in two ways. Formally, the first way is "BCE Loss" (i.e., training with BCE Loss), and the other way is "BW Loss" (i.e., training with BW loss). As shown in Fig. 12, BW loss results in

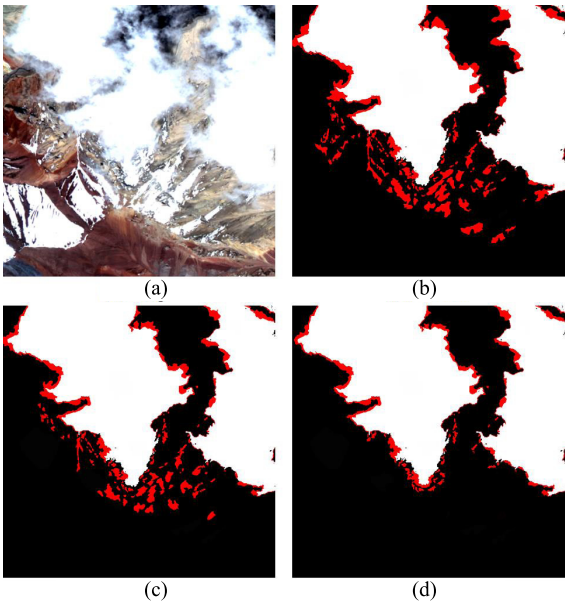

Fig. 11. Qualitative detection results using different multiscale contextual feature extraction methods in the cloud–snow coexistence area. Correctly detected cloud pixels are marked with white, and noncloud pixels are marked with black, where red marks are misclassified pixels. (a) Image. (b) +PPM (c) +ASPP. (d) +DCSP.

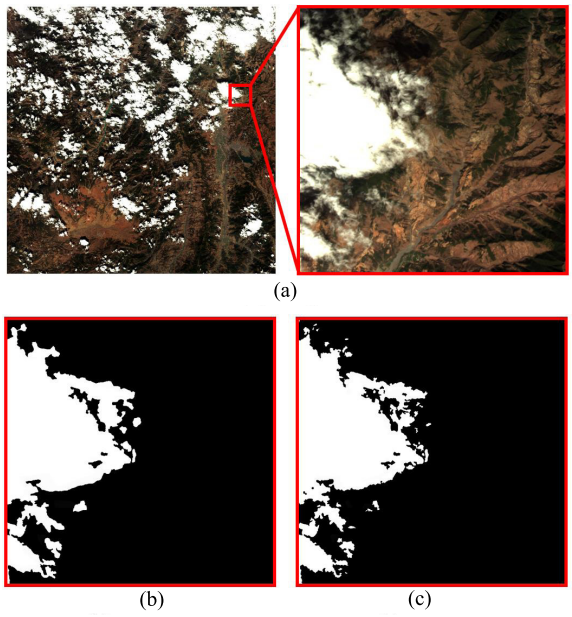

Fig. 12. Qualitative detection results of models trained with different loss functions at the cloud boundary. Compared with BCE Loss, the detection result of the model trained with BW Loss has richer spatial details of the boundary. (a) Image. (b) BCE Loss. (c) BW Loss.

clearer and more detailed cloud boundary detection results. The corresponding quantitative results are shown in Table IV. BW loss has achieved good results in three accuracy indexes. The results show that BW loss can direct the network to learn the strong semantics, which is helpful to optimize the boundary detection of the cloud.

We also conduct some experiments on the GF-1 WFV data set. As shown in Table V, the experimental results once again demonstrate that our method can effectively boost cloud detection accuracy.

TABLE IV
EFFECTS OF DIFFERENT LOSS FUNCTIONS ON THE AIR-CD DATA SET

Loss Func.	MIoU(%)	OA(%)	F1(%)
BCE Loss	91.26	95.07	94.39
BW Loss	**92.08**	**97.69**	**96.95**

TABLE V
EFFECT OF EACH COMPONENT ON THE GF-1 WFV DATA SET

Method	MIoU(%)	OA(%)	F1(%)
Baseline	83.26	87.16	85.27
+ DCFP	87.96	93.28	91.43
+ BW Loss	85.68	89.84	88.15
+ DCFP + BW Loss	**90.68**	**96.52**	**94.85**

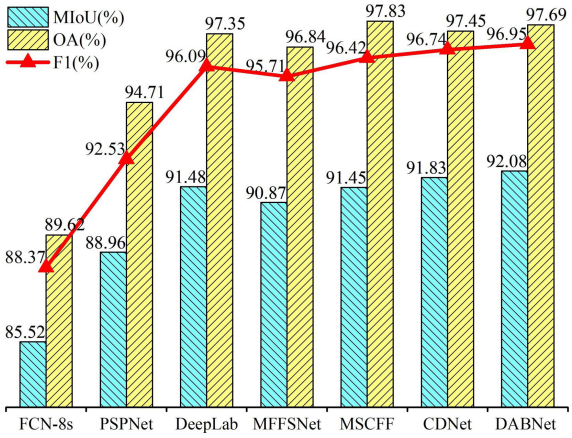

Fig. 13. Comparison of the detection accuracy of different methods on the AIR-CD data set.

D. Comparison With State-of-the-Art Methods

1) Quantitative Analysis: Table VI and Fig. 13 demonstrate the performance of our DABNet and other state-of-the-art methods on the AIR-CD data set. Considering parameters and multiadds, DABNet has a great advantage over the other models in terms of efficiency. Our method can produce 92.08% MIoU without bells and whistles, which achieves the accuracy of complex cloud detection models, such as MFFSNet [27], MSCFF [29], and CDNet [40]. However, the number of parameters of our method can be reduced by about 100 times, and the calculations are even reduced by 200 times, as illustrated in Fig. 14. Compared with the semantic segmentation model [52], [60], [62], the existing cloud detection model usually designs dilated convolution structure in the backbone, so the calculation amount is generally larger, and the accuracy is significantly higher. However, even if our method uses the dilated convolution mechanism, it can still keep a very small amount of calculation. This is due to the lightweight characteristics of the overall structure of DABNet. Table VI and Fig. 15 show the performance of our method and other typical models on the GF-1 WFV data set. Similar to the AIR-CD

TABLE VI
COMPARISON OF THE PERFORMANCE OF DIFFERENT METHODS FOR CLOUD DETECTION

Method	Ref.	Year	Publication	Accuracy						Efficiency	
				AIR-CD			GF-1 WFV			Params(M)	Multi-adds(G)
				MIoU(%)	OA(%)	F1(%)	MIoU(%)	OA(%)	F1(%)		
FCN-8s	[62]	2015	IEEE CVPR	85.52	89.62	88.37	84.19	89.17	87.96	67.57	49.41
PSPNet	[52]	2017	IEEE CVPR	88.96	94.71	92.53	86.32	93.24	91.03	65.57	19.86
DeepLabV3+	[60]	2018	arXiv	91.48	97.35	96.09	89.27	95.07	94.68	59.34	27.90
MFFSNet	[27]	2018	IEEE GRSL	90.87	96.84	95.71	89.08	95.83	94.59	73.32	116.91
MSCFF	[29]	2019	ISPRS JPRS	91.45	**97.83**	96.42	90.35	**96.74**	95.27	49.46	188.80
CDNet	[40]	2019	IEEE TGRS	91.83	97.45	96.74	89.97	96.46	**95.73**	85.52	227.97
DABNet	-	-	-	**92.08**	97.69	**96.95**	**90.68**	96.52	94.85	**4.12**	**8.29**

[1] 1 M = 1×10^6
[2] 1 G = 1×10^9

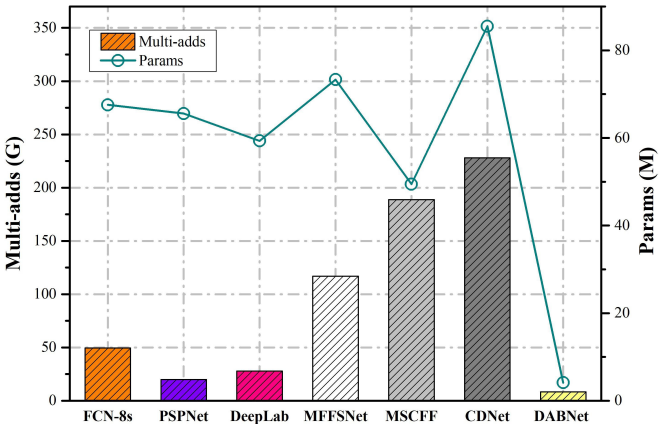

Fig. 14. Comparison of the detection efficiency of different methods for cloud detection.

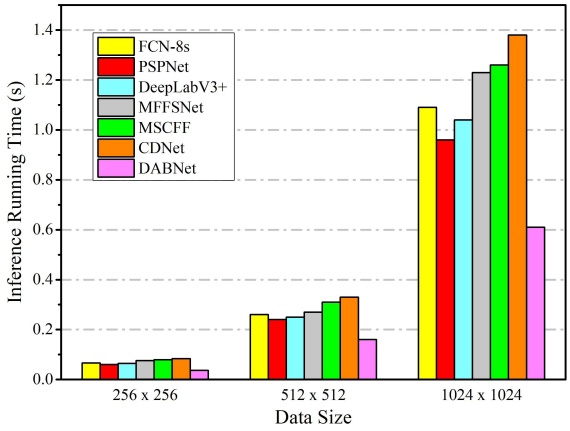

Fig. 16. Comparison of the inference running time of different methods for cloud detection.

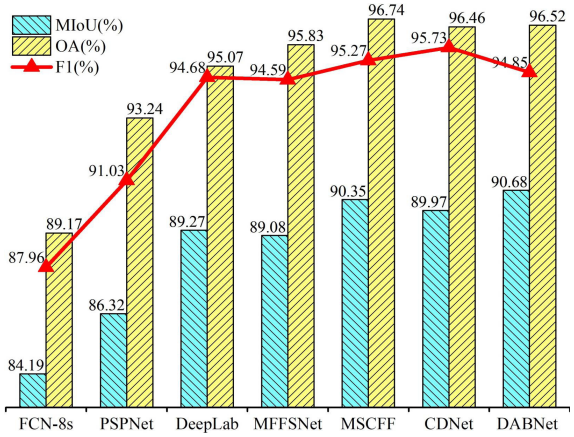

Fig. 15. Comparison of the detection accuracy of different methods on the GF-1 WFV data set.

TABLE VII
COMPARISON OF THE INFERENCE RUNNING TIME (s) OF DIFFERENT METHODS FOR CLOUD DETECTION

Method \ Data Size	256×256	512×512	1024×1024
FCN-8s [62]	0.066	0.26	1.09
PSPNet [52]	0.061	0.24	0.96
DeepLabV3+ [60]	0.065	0.25	1.04
MFFSNet [27]	0.076	0.27	1.23
MSCFF [29]	0.079	0.31	1.26
CDNet [40]	0.084	0.33	1.38
DABNet	**0.037**	**0.16**	**0.62**

data set, our method implements state-of-the-art performance in efficiency, and the accuracy has basically reached the highest level of the current cloud detection model.

In Table VII, we evaluate the inference running time of these networks. The running time is computed based on input data of sizes 256×256, 512×512, and 1024×1024 pixels, respectively. As illustrated in Fig. 16, the running time of DABNet is much lower than other methods; especially, for input data of large size, the advantages are more obvious. Running time can be used to intuitively evaluate model efficiency (i.e., complexity), which is mainly determined by the model's parameters (Params) and calculations (multiadds). The smaller the Params and multiadds, the higher the efficiency of the model, and the shorter the inference running time. Since

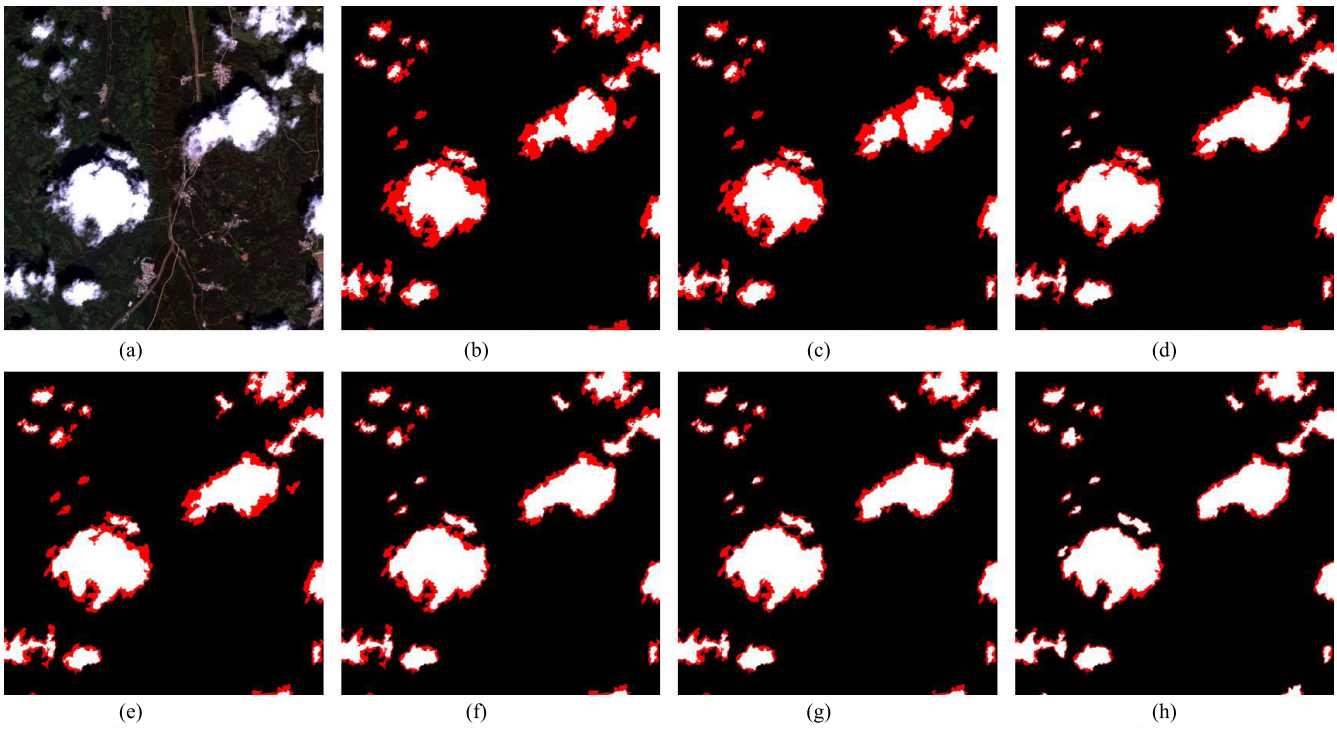

Fig. 17. Qualitative comparison of different cloud detection methods on the typical partial scene of the AIR-CD data set. (a) Image. (b) FCN-8s. (c) PSPNet. (d) DeepLabv3+. (e) MFFSNet. (f) MSCFF. (g) CDNet. (h) DABNet.

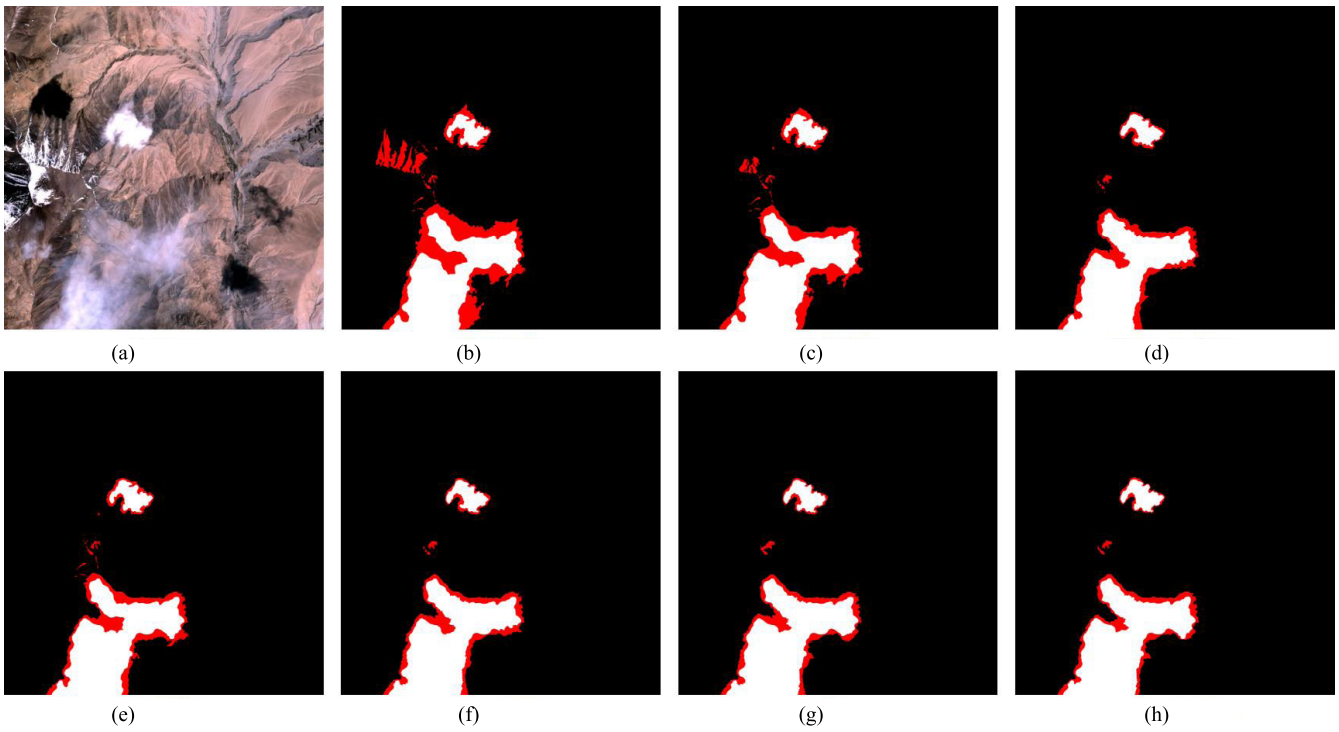

Fig. 18. Qualitative comparison of different cloud detection methods on the typical partial scene of the AIR-CD data set. (a) Image. (b) FCN-8s. (c) PSPNet. (d) DeepLabv3+. (e) MFFSNet. (f) MSCFF. (g) CDNet. (h) DABNet.

the running time also depends on the memory access [45], some comparison methods seem to be very close on this indicator. In the following work, we will optimize the model structure of DABNet to further curtail the inference running time.

2) Qualitative Analysis: Figs. 17 and 18 illustrate the comparison of different methods on the AIR-CD data set. We selected some representative image visualization results, including the small cloud and cloud–snow coexistence area. For the convenience of observation, we mark correctly

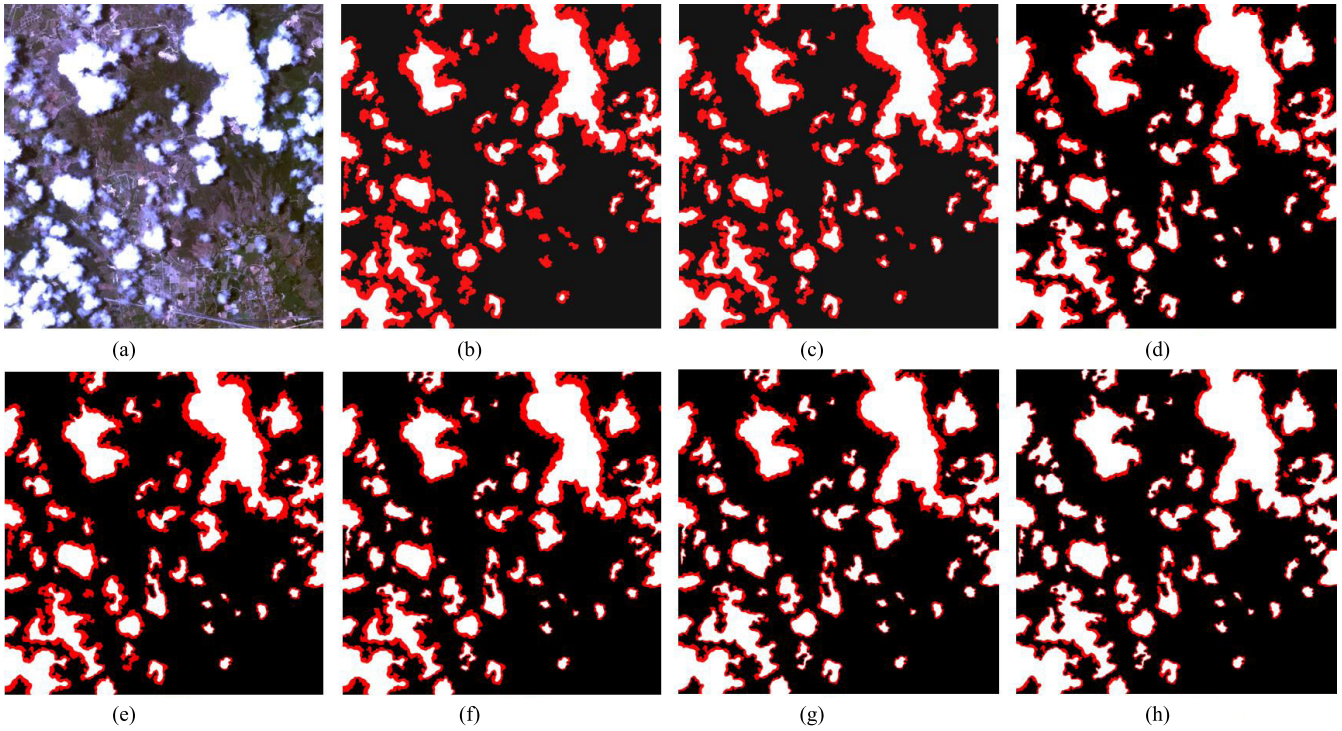

Fig. 19. Qualitative comparison of different cloud detection methods on the typical partial scene of the GF-1 WFV data set. (a) Image. (b) FCN-8s. (c) PSPNet. (d) DeepLabv3+. (e) MFFSNet. (f) MSCFF. (g) CDNet. (h) DABNet.

detected cloud pixels with white and noncloud pixels with black, in which red marks pixels with misclassification. The results show that our method has obvious advantages. There are relatively fewer misclassified pixels marked in red. Specifically, FCN [62] produces more misclassification results in the snow area. PSPNet [52] and DeepLabV3+ [60] are significantly improved because of the introduction of multiscale feature extraction mechanism. Compared with the semantic segmentation model, the performance of the models [27], [29], [40] designed for cloud detection task has been further improved, especially for small cloud and snow area. Different from the above six methods, DABNet has a lightweight architecture, and the total amount of parameters and calculation is significantly reduced. By introducing the DCFP module and BW loss, it is possible to effectively separate cloud and snow pixels with similar colors, and the result of cloud boundary detection is more detailed. Fig. 19 demonstrates the comparison results of different methods on the GF-1 WFV data set. Therefore, it can be proved that our DABNet has good generalization capability for the cloud detection task of optical remote sensing images.

E. Discussion

1) Slice Size: Since clouds are targets of different sizes, the information presented in slices of different sizes is also different. As shown in Fig. 20, we can observe structural features, such as cloud boundaries in a 1024 × 1024 pixels slice, but, basically, only white pixels can be seen in the 256 × 256 pixels slice corresponding to the center position. Without prior knowledge, the clouds in the latter will be difficult to identify. We guess that using slices of different sizes for cloud detection may have an impact on model accuracy. Therefore, we, respectively, cut the data in AIR-CD into 256 × 256, 512 × 512, and 1024 × 1024 pixels slices for training and testing. The experimental results in Table VIII show that using large-size slices as input helps improve the detection accuracy to a certain extent, but the effect is not obvious. This shows that the slice size has little effect on the accuracy of our model, possibly because DABNet is rich in expert knowledge required for cloud detection tasks. DABNet is robust to slice size

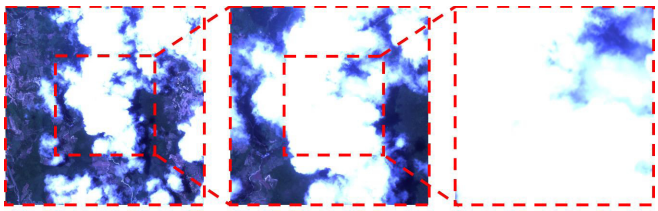

Fig. 20. Illustration of slices of different sizes. (a) 1024 × 1024. (b) 512 × 512. (c) 256 × 256.

TABLE VIII
EFFECTS OF THE SLICE SIZE FOR DABNET PERFORMANCE ON THE AIR-CD DATA SET

Slice Size	MIoU(%)	OA(%)	F1(%)
256 × 256	91.27	97.15	96.37
512 × 512	92.18	97.52	97.04
1024 × 1024	**92.53**	**97.98**	**97.61**

TABLE IX

COMPARISON OF THE PERFORMANCE OF DIFFERENT METHODS FOR AUTOMATIC WATER-BODY EXTRACTION

Method	MIoU(%)	OA(%)	F1(%)
PSPNet [52]	83.94	87.82	86.35
DeepLabV3+ [60]	86.21	90.96	89.74
DABNet	**88.37**	**92.03**	**91.26**

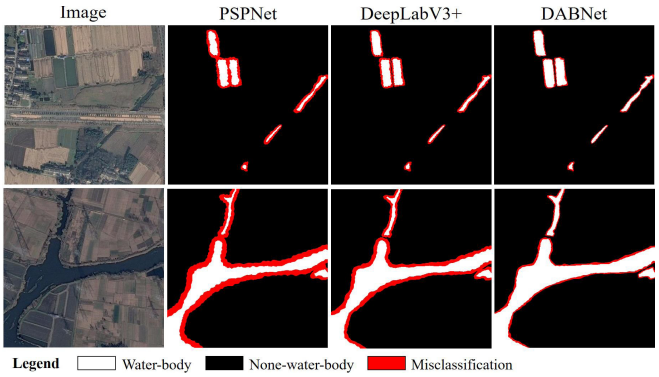

Fig. 21. Qualitative comparison of different methods on automatic water-body extraction.

changes, which is another advantage of the model. Considering the inference running time analyzed above, it is necessary to select a moderate slice size as input because the slice size has a more obvious influence on the inference speed. Based on the analysis, we recommend using slices of about 512×512 pixels slices as the model input.

2) Universality: To verify the universality of the method, we tried to use DABNet for automatic water-body extraction. This task is chosen because it has many similarities with cloud detection, such as the diversity of object shapes and sizes. The data set that we selected comes from the open data of the automatic water-body extraction competition, i.e., the 2020 Gaofen Challenge on Automated High-Resolution Earth Observation Image Interpretation (http://sw.chreos.org/). The data set is collected from Gaofen-2 satellite, with a total of 1000 492×492 pixels images. In the training procedure, we randomly select 16 images in each small batch as the input. The other settings are the same as the cloud detection experiment. Table IX demonstrates quantitative results. Our DABNet can achieve 88.37% MIoU, better than the other two classic segmentation models [52], [60]. Some representative image visualization results are illustrated in Fig. 21, where part of the water-body is visually similar to the surrounding farmland. DABNet obviously produces fewer misclassified pixels in this area. We believe that the main reason why DABNet has obvious advantages for such a difficult pixel classification problem lies in the DCFP module. This module can assist the model to adaptively extract multiscale context and build long-range dependencies to enhance feature discrimination information. Through the experiment of automatic water-body extraction, we can know that DABNet has considerable universality, which can also be applied to other remote sensing image interpretation tasks.

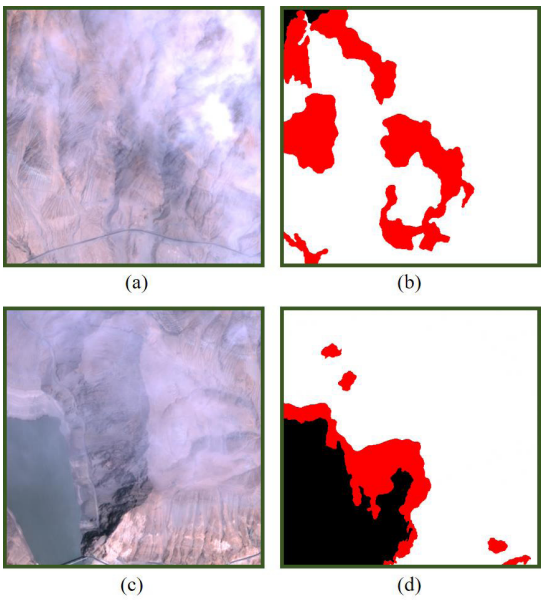

Fig. 22. Two typical cases of detection errors in thin cloud regions. (b) and (d) Detection results of (a) and (c), respectively.

3) Limitations: DABNet can achieve nice cloud detection performance in a series of tough situations, but there are still some obvious errors in thin cloud detection. Thin clouds are usually visually recognizable, and the surface below is vaguely visible, which means that most of the thin clouds are almost the same as underground objects. Therefore, the problem of omission of thin clouds is prone to appear in the detection results. Especially, at the boundary of thin clouds, this problem is more serious because of more tiny details. Fig. 22 illustrates two typical cases of detection errors in thin cloud regions, and there are many omission or misclassified pixels in the results. Later, we will optimize the model to further improve the robustness of thin cloud detection.

V. CONCLUSION

This article proposes an efficient method (DABNet) for cloud detection in optical remote sensing images. Compared with the cloud detection model published at present, the parameters and calculations of DABNet are significantly reduced, but the detection accuracy can reach the same level or even better. To be specific, our method uses the DCFP module to extract multiscale features adaptively according to the characteristics of cloud shape and size. A novel BW loss function is designed, which can direct the network to more focus on the pixels near the boundary. Experiments based on the AIR-CD and GF-1 WFV data sets show that our method is superior to the baseline method and achieves better efficiency than prevalent algorithms. In the future, we will attempt to boost the performance of thin cloud detection by optimizing the model structure. In addition, we will explore model designing based on specific spaceborne computing platforms, such as FPGA and ASIC, to further improve running efficiency. What is more, we will also design a lightweight general model based on DABNet, which can complete multiple remote sensing image interpretation tasks at the same time (such as simultaneous cloud detection and automatic water-body extraction).

REFERENCES

[1] Y. Zhang, "Calculation of radiative fluxes from the surface to top of atmosphere based on ISCCP and other global data sets: Refinements of the radiative transfer model and the input data," *J. Geophys. Res.*, vol. 109, no. D19, pp. 1–27, 2004.

[2] X.-Y. Zhuge, X. Zou, and Y. Wang, "A fast cloud detection algorithm applicable to monitoring and nowcasting of daytime cloud systems," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6111–6119, Nov. 2017.

[3] H. Ishida, Y. Oishi, K. Morita, K. Moriwaki, and T. Y. Nakajima, "Development of a support vector machine based cloud detection method for MODIS with the adjustability to various conditions," *Remote Sens. Environ.*, vol. 205, pp. 390–407, Feb. 2018.

[4] W. B. Rossow and L. C. Garder, "Cloud detection using satellite measurements of infrared and visible radiances for ISCCP," *J. Climate*, vol. 6, no. 12, pp. 2341–2369, Dec. 1993.

[5] G. Gesell, "An algorithm for snow and ice detection using AVHRR data an extension to the APOLLO software package," *Int. J. Remote Sens.*, vol. 10, nos. 4–5, pp. 897–905, Apr. 2007.

[6] S. A. Ackerman, K. I. Strabala, W. P. Menzel, R. A. Frey, C. C. Moeller, and L. E. Gumley, "Discriminating clear sky from clouds with MODIS," *J. Geophys. Res., Atmos.*, vol. 103, no. D24, pp. 32141–32157, Dec. 1998.

[7] C.-H. Lin, B.-Y. Lin, K.-Y. Lee, and Y.-C. Chen, "Radiometric normalization and cloud detection of optical satellite images using invariant pixels," *ISPRS J. Photogramm. Remote Sens.*, vol. 106, pp. 107–117, Aug. 2015.

[8] B. Zhong, W. Chen, S. Wu, L. Hu, X. Luo, and Q. Liu, "A cloud detection method based on relationship between objects of cloud and cloud-shadow for chinese moderate to high resolution satellite imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 11, pp. 4898–4908, Nov. 2017.

[9] T. Wu, X. Hu, Y. Zhang, L. Zhang, P. Tao, and L. Lu, "Automatic cloud detection for high resolution satellite stereo images and its application in terrain extraction," *ISPRS J. Photogramm. Remote Sens.*, vol. 121, pp. 143–156, Nov. 2016.

[10] M. Wieland, Y. Li, and S. Martinis, "Multi-sensor cloud and cloud shadow segmentation with a convolutional neural network," *Remote Sens. Environ.*, vol. 230, Sep. 2019, Art. no. 111203.

[11] M. Tian, H. Chen, and G. Liu, "Cloud detection and classification for S-NPP FSR CRIS data using supervised machine learning," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, Jul. 2019, pp. 9827–9830.

[12] S. Ji, P. Dai, M. Lu, and Y. Zhang, "Simultaneous cloud detection and removal from bitemporal remote sensing images using cascade convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, early access, May 22, 2020, doi: 10.1109/TGRS.2020.2994349.

[13] J. H. Jeppesen, R. H. Jacobsen, F. Inceoglu, and T. S. Toftegaard, "A cloud detection algorithm for satellite imagery based on deep learning," *Remote Sens. Environ.*, vol. 229, pp. 247–259, May 2019.

[14] G. Mateo-Garcıa, J. E. Adsuara, A. Perez-Suay, and L. Gomez-Chova, "Concolutional long short-term memory network for multitemporal cloud detection over landmarks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2019, pp. 210–213.

[15] Z. Li, H. Shen, H. Li, G. Xia, P. Gamba, and L. Zhang, "Multi-feature combined cloud and cloud shadow detection in GaoFen-1 wide field of view imagery," *Remote Sens. Environ.*, vol. 191, pp. 342–358, Mar. 2017.

[16] L. Sun *et al.*, "A cloud detection algorithm-generating method for remote sensing data at visible to short-wave infrared wavelengths," *ISPRS J. Photogramm. Remote Sens.*, vol. 124, pp. 70–88, Feb. 2017.

[17] L. Ye, Z. Cao, Y. Xiao, and Z. Yang, "Supervised fine-grained cloud detection and recognition in whole-sky images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7972–7985, Oct. 2019.

[18] P. Zhou, J. Han, G. Cheng, and B. Zhang, "Learning compact and discriminative stacked autoencoder for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 4823–4833, Jul. 2019.

[19] P. Bo, S. Fenzhen, and M. Yunshan, "A cloud and cloud shadow detection method based on fuzzy c-Means algorithm," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1714–1727, May 2020.

[20] S. Skakun, E. F. Vermote, J.-C. Roger, C. O. Justice, and J. G. Masek, "Validation of the LaSRC cloud detection algorithm for landsat 8 images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 7, pp. 2439–2446, Jul. 2019.

[21] S. Mohajerani and P. Saeedi, "Cloud-Net: An end-to-end cloud detection algorithm for landsat 8 imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, Jul. 2019, pp. 1029–1032.

[22] F. Xie, M. Shi, Z. Shi, J. Yin, and D. Zhao, "Multilevel cloud detection in remote sensing images based on deep learning," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 8, pp. 3631–3640, Aug. 2017.

[23] L. Ye, Z. Cao, and Y. Xiao, "DeepCloud: Ground-based cloud image categorization using deep convolutional features," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 10, pp. 5729–5740, Oct. 2017.

[24] H. Zhai, H. Zhang, L. Zhang, and P. Li, "Cloud/shadow detection based on spectral indices for multi/hyperspectral optical remote sensing imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 144, pp. 235–253, Oct. 2018.

[25] C. Shi, C. Wang, Y. Wang, and B. Xiao, "Deep convolutional activations-based features for ground-based cloud classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 6, pp. 816–820, Jun. 2017.

[26] Y. Zhan, J. Wang, J. Shi, G. Cheng, L. Yao, and W. Sun, "Distinguishing cloud and snow in satellite images via deep convolutional network," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1785–1789, Oct. 2017.

[27] Z. Yan *et al.*, "Cloud and cloud shadow detection using multilevel feature fused segmentation network," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 10, pp. 1600–1604, Oct. 2018.

[28] D. Chai, S. Newsam, H. K. Zhang, Y. Qiu, and J. Huang, "Cloud and cloud shadow detection in landsat imagery based on deep convolutional neural networks," *Remote Sens. Environ.*, vol. 225, pp. 307–316, May 2019.

[29] Z. Li, H. Shen, Q. Cheng, Y. Liu, S. You, and Z. He, "Deep learning based cloud detection for medium and high resolution remote sensing images of different sensors," *ISPRS J. Photogramm. Remote Sens.*, vol. 150, pp. 197–212, Apr. 2019.

[30] J. Guo, J. Yang, H. Yue, H. Tan, C. Hou, and K. Li, "CDnetV2: CNN-based cloud detection for remote sensing imagery with cloud-snow coexistence," *IEEE Trans. Geosci. Remote Sens.*, early access, May 18, 2020, doi: 10.1109/TGRS.2020.2991398.

[31] X. Zhang, T. Wang, G. Chen, X. Tan, and K. Zhu, "Convective clouds extraction from himawari-8 satellite images based on double-stream fully convolutional networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 4, pp. 1–5, Apr. 2020.

[32] F. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 5987–5995.

[33] A. J. Collegio, J. C. Nah, P. S. Scotti, and S. Shomstein, "Attention scales according to inferred real-world object size," *Nature Hum. Behav.*, vol. 3, no. 1, pp. 40–47, Jan. 2019.

[34] L. Itti, "Quantifying the contribution of low-level saliency to human eye movements in dynamic scenes," *Vis. Cognition*, vol. 12, no. 6, pp. 1093–1123, Aug. 2005.

[35] R. Wang and K. Tang, "Minimax classifier for uncertain costs," 2012, *arXiv:1205.0406*. [Online]. Available: http://arxiv.org/abs/1205.0406

[36] L. W. Renninger, P. Verghese, and J. Coughlan, "Where to look next? Eye movements reduce local uncertainty," *J. Vis.*, vol. 7, no. 3, p. 6, Feb. 2007.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105.

[39] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1800–1807.

[40] J. Yang, J. Guo, H. Yue, Z. Liu, H. Hu, and K. Li, "CDNet: CNN-based cloud detection for remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 6195–6211, Aug. 2019.

[41] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861

[42] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[43] A. Howard *et al.*, "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Nov. 2019, pp. 1314–1324.

[44] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.
[45] N. Ma, X. Zhang, H. Zheng, and J. Sun, "Shufflenet V2: Practical guidelines for efficient CNN architecture design," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 122–138.
[46] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, Aug. 2013.
[47] V. Badrinarayanan, A. Handa, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling," 2015, *arXiv:1505.07293*. [Online]. Available: http://arxiv.org/abs/1505.07293
[48] O. Ronneberger, P. Fischer, and T. Brox, "UNet: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.
[49] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 764–773.
[50] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable convnets V2: More deformable, better results," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 9308–9316.
[51] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.
[52] 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. 2881–2890.
[53] 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," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.
[54] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, "Object detection in optical remote sensing images: A survey and a new benchmark," *ISPRS J. Photogram. Remote Sens.*, vol. 159, pp. 296–307, Dec. 2019.
[55] X. Sun, Y. Liu, Z. Yan, P. Wang, W. Diao, and K. Fu, "SRAF-Net: Shape robust anchor-free network for garbage dumps in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, early access, Oct. 5, 2020, doi: 10.1109/TGRS.2020.3023928.
[56] F. Zhou, R. Hang, and Q. Liu, "Class-guided feature decoupling network for airborne image segmentation," *IEEE Trans. Geosci. Remote Sens.*, early access, Jul. 15, 2020, doi: 10.1109/TGRS.2020.3006872.
[57] X. Sun, A. Shi, H. Huang, and H. Mayer, "BAS^{4}Net: Boundary-aware semi-supervised semantic segmentation network for very high resolution remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 5398–5413, 2020.
[58] S. Theodoridis, "Neural networks and deep learning," in *Machine Learning: A Bayesian and Optimization Perspective*. 2015, ch. 18, pp. 875–936, doi: 10.1016/B978-0-12-801522-3.00018-5.
[59] 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.
[60] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," 2018, *arXiv:1802.02611*. [Online]. Available: http://arxiv.org/abs/1802.02611
[61] Z. Shao, Y. Pan, C. Diao, and J. Cai, "Cloud detection in remote sensing images based on multiscale features-convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 6, pp. 4062–4076, Jun. 2019.
[62] 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.

Qibin He (Graduate Student Member, IEEE) received the B.Sc. degree from the Beijing Institute of Technology, Beijing, China, in 2020. He is currently pursuing the Ph.D. degree with the University of Chinese Academy of Sciences, Beijing, and the Aerospace Information Research Institute, Chinese Academy of Sciences (CAS), Beijing.

His research interests include computer vision and remote sensing image understanding.

Xian Sun (Senior Member, IEEE) received the B.Sc. degree from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 2004, and the M.Sc. and Ph.D. degrees from the Institute of Electronics, Chinese Academy of Sciences (CAS), Beijing, in 2009.

He was a Visiting Scholar, Karlsruher Institut für Technologie, Karlsruhe, Germany, in 2013. He is currently a Professor with the Aerospace Information Research Institute, CAS. His research interests include computer vision, geospatial data mining, and remote sensing image understanding.

Dr. Sun was a recipient of the Outstanding Science and Technology Achievement Prize of the CAS in 2016 and the First Prize for The State Scientific and Technological Progress of China in 2019. He also serves as an Associate Editor for IEEE ACCESS and a Guest Editor for the special issue of the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING (JSTARS) and other journals.

Zhiyuan Yan (Member, IEEE) received the B.Sc. degree from Xiamen University, Xiamen, China, in 2016, and the M.Sc. degree from the Aerospace Information Research Institute, Chinese Academy of Sciences (CAS), Beijing, China, in 2019.

She is currently an Assistant Engineer with the Aerospace Information Research Institute. Her research interests include computer vision and remote sensing image analysis.

Kun Fu (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees from the National University of Defense Technology, Changsha, China, in 1995, 1999, and 2002, respectively.

He is currently a Professor with the Aerospace Information Research Institute, Chinese Academy of Sciences (CAS), Beijing, China. His research interests include computer vision, remote sensing image understanding, and geospatial data mining and visualization.

Dr. Fu was a recipient of the First Prizes for The State Scientific and Technological Progress of China in 2015 and 2019, the Outstanding Science and Technology Achievement Prize of the CAS in 2016, the Scientific and Technological Innovation Leading Talent by the National High-Level Talents Special Support Plan in 2017, and the Distinguished Young Scholars from the National Natural Science Foundation of China in 2017.