LWCDnet: A Lightweight Network for Efficient Cloud Detection in Remote Sensing Images

Chen Luo, Shanshan Feng, Xiaofei Yang, Yunming Ye, Xutao Li, Baoquan Zhang, Zhihao Chen, and Yingling Quan

***Abstract*— Cloud detection is the task of detecting cloud areas in remote sensing images (RSIs), and it has attracted extensive research interest. Recently, deep learning (DL)-based methods have been proposed and achieved great performance for cloud detection. However, due to the satellite's limitation in storage and memory, existing DL approaches, which suffer from extensive computation and large model size, are almost impossible to be deployed on satellites. To fill this gap, we target at studying effective and efficient cloud detection solutions that are suitable for satellites. In this article, we develop a lightweight autoencoder-based cloud detection method, namely, lightweight encoder–decoder cloud detection network (LWCDnet). In the encoder part, the designed novel lightweight dual-branch block (LWDBB) in the backbone extracts spatial and contextual information concurrently. Moreover, a lightweight feature pyramid module (LWFPM) is proposed to capture high-level multiscale contextual information. In the decoder part, the lightweight feature fusion module (LWFFM) compensates for the missing spatial and detail information from the encoder to the high-level feature maps. We evaluate the proposed method on two public datasets: LandSat8 and Moderate-Resolution Imaging Spectroradiometer (MODIS). Extensive experiments demonstrate that the proposed LWCDnet achieves comparable accuracy as the state-of-the-art cloud detection methods and lightweight semantic segmentation algorithms. In the meantime, LWCDnet has much less computation burden with smaller model size.**

***Index Terms*— Cloud detection, deep learning (DL), LandSat8, lightweight network, Moderate-Resolution Imaging Spectroradiometer (MODIS).**

I. Introduction

SINCE almost two thirds of our Earth surface is covered by clouds [1], removing the cloud in satellite imagery is an important preprocessing step for various remote sensing applications, such as land cover classification [2], environment observation [3], and vegetation engineering [4]. With the cloud detection step, the quality of these applications can be improved. Therefore, the cloud detection method with high precision is in great demand for satellite imagery.

In recent years, with the development of modular satellites and commercial space launches, it has been a trend to deploy various tasks on space devices to reduce operational costs [5], especially by reducing communication costs or facilitating navigation. In 2020, Europa launched the satellite PhiSat-1, and one of its missions is saving downlink bandwidth by omitting to transfer lots of captured useless images, which are covered by clouds [6]. In this satellite, the equipped system HyperScout-2 can offer a powerful computational set of processors, including an FPGA, CPU, GPU, and a dedicated board for AI inference. They make the onboard data preprocessing and machine vision tasks possible [7]. Hence, the cloud detection method, which can achieve excellent performance and meanwhile can be implemented and executed on satellites, is highly desirable.

Traditional cloud detection methods, represented by threshold-based approaches and machine-learning-based methods, have low requirements to hardware and data volume. Although they can be easily deployed onboard satellite, the performance is not satisfactory enough because of the methods' limitations. In particular, threshold-based approaches identify cloud in pixel level for multispectral/hyperspectral imagery by utilizing the spectral and wavelength difference between cloud and other objects [8]. However, spatial information, such as texture, is rarely considered. In addition, it is also challenging for such methods to own reliable cloud detection results on multispectral remote sensing images (RSIs), which have only four bands [red, green, blue, and near infrared (NIR)] [55]. Furthermore, enormous calculation is inevitable on high-resolution RSIs. In contrast, machine-learning-based methods take the handcrafted features as the classifier's input and utilize the algorithms such as support vector machine (SVM) [10] or random forest (RF) [9] for cloud detection. However, under a complex environment, cloud characteristics are difficult to obtain. Also, the performance of such methods is highly influenced by the manual inputs, which usually contains insufficient distinguishable information.

In recent years, deep learning (DL) has made big progress in computer vision through its ability to extract and aggregate discriminative features from input images. Also, lots of remote sensing applications have also benefited from it. In cloud detection, some researchers have proposed models based on

Manuscript received June 24, 2021; revised October 30, 2021 and March 13, 2022; accepted May 2, 2022. Date of publication May 16, 2022; date of current version May 23, 2022. This work was supported by the Shenzhen Science and Technology Program under Grant JCYJ20210324120208022 and Grant JCYJ20200109113014456. *(Corresponding authors: Yunming Ye; Xutao Li.)*

Chen Luo, Shanshan Feng, Yunming Ye, Xutao Li, Baoquan Zhang, Zhihao Chen, and Yingling Quan are with the Department of Computer Science and the Shenzhen Key Laboratory of Internet Information Collaboration, Harbin Institute of Technology, Shenzhen 518055, China (e-mail: luochen@stu.hit.edu.cn; victor_fengss@foxmail.com; yeyunming@hit.edu.cn; lixutao@hit.edu.cn; zhangbaoquan@stu.hit.edu.cn; chenzhihao@stu.hit.edu.cn; ylquan@stu.hit.edu.cn).

Xiaofei Yang is with the Department of Computer and Information Science, University of Macau, Macau, China (e-mail: xiaofeiyang@um.edu.mo).

Digital Object Identifier 10.1109/TGRS.2022.3173661

convolutional neural networks (CNNs) in DL and achieved significant performance improvement [15], [56]. However, existing DL methods require a great number of parameters, which can be tens of millions. With DL-based models, extensive computation is unavoidable. Also, the satellite, as a space device, has normally limitations on computation, storage, and power resource. Therefore, deploying the existing DL-based cloud detection methods on satellites is impracticable. Another possibility is porting CNN models proposed for semantic segmentation to cloud detection problem [31] since these two tasks have the similar background and scenario. In the research task of semantic segmentation, a plenty of lightweight models have been proposed for application on mobile devices [48]. However, in comparison to natural scene images (NSIs), the RSIs have their particular characteristics. First, the object intraclass feature variance is larger in RSIs. In particular, in cloud detection scenarios, the spectral feature of thin cloud is heavily influenced by its background. Therefore, even though both thin cloud and cloud belong to the "cloud" class, they have different feature representations. Second, the object interclass feature variance is smaller in RSIs. In particular, in cloud detection scenarios, snow and ice cloud have quite similar features in most RSI bands [15]. These two difficulties make it unreliable to directly apply the methods that are designed for NSIs semantic segmentation on cloud detection tasks.

In this work, we make full use of multiscale context information to solve the difficulties mentioned above. First, when considering cloud is a continuous area influenced by background noise, by adding the short-range context information from surrounding cloud pixels to the thin-cloud pixel, the impact of background noise in thin cloud can be weakened. Second, the existence of ice/snow is closely related to the background scenes. The long-range semantic features and background scenarios have a positive impact on distinguishing between cloud and ice/snow by adding background area information. In this article, we propose a lightweight encoder–decoder cloud detection network (LWCDnet) to make full use of the semantic information in an effective way. In particular, we first propose a lightweight dual-branch block (LWDBB), which can extract spatial and short-range context information simultaneously. In the encoder, the LWDBB is stacked to build a ResNet backbone. The lightweight backbone can efficiently extract diverse short-range semantic features to avoid the impact of background noise. Then, we introduce a lightweight feature pyramid module (LWFPM). It contributes to the long-range multiscale and global contextual information fusion in high-level features to distinguish cloud from the cloud-like object with the help of global background scenes. In the decoder, the proposed lightweight feature fusion module (LWFFM) is exploited to compensate for the lost spatial information from low-level feature maps. By deploying proposed novel lightweight modules LWDBB and LWFPM, the model has only 1% of parameters of existing DL-based cloud detection methods. In addition, extensive experiments were conducted to compare LWCDnet with existing cloud detection methods and lightweight semantic segmentation methods in performance and efficiency, the performance degradation is under 2% in the worst case. To evaluate the performance of each introduced module, we conducted ablation experiments on them. The main contributions of this work can be summarized as follows.

1) We investigate a novel problem, which aims to effectively detect cloud in RSIs with limited computing resources. To the best of our knowledge, this is the first work to study lightweight DL networks for cloud detection tasks.
2) We develop a lightweight encoder–decoder method LWCDnet for cloud detection. In the encoder, LWCDnet employs a lightweight backbone built on LWDBBs and LWFPM to capture spatial and short-/long-range semantic features simultaneously. In the decoder, the low-level spatial feature is compensated via a lightweight decoder.
3) We conduct extensive experiments on LandSat8 and Moderate-Resolution Imaging Spectroradiometer (MODIS) datasets. The results demonstrate that LWCDnet can obtain the best tradeoff in terms of parameter size, floating-point operations per second (FLOPs), image processing spend, and accuracy in cloud detection.

The rest of this article is organized as follows. In Section II, related works on cloud detection are briefly reviewed. In Section III, the proposed LWCDnet is introduced in detail. After that, the result of extensive experiments and ablation studies on LandSat8 and MODIS are presented and analyzed in Section IV. Finally, we conclude this work in Section V.

II. Related Work

A. Threshold-Based Methods

The threshold-based algorithms are built upon the different spectral characteristics of cloud and other objects in reflectance of bands [18]. Because of their simplicity, they are widely used in cloud detection. In [8], with Landsat Top of Atmosphere (TOA) reflectance and Brightness Temperature (BT) of Landsat images as inputs, Fmask used cloud physical properties and produced a probability mask to separate clouds over land and water. In addition, cloud shadows were detected with NIR Band. The improved version of Fmask [19] took advantage of a newly added cirrus band in LandSat8 and promoted a prototype algorithm for Sentinel 2 image. For mountain areas in LandSats 4–8 images, Qiu *et al.* [19] included TOA reflectance, BT, and digital elevation models (DEMs) for cloud and cloud shadow detection. Wei *et al.* [21] proposed an algorithm built on a database constructed using MODIS surface reflectance products to dynamically determine a proper threshold. In [22], Fisher worked on the problem that reflectance properties of clouds are very similar to common features on the Earth's surface by utilizing the watershed-from-markers transform. Li *et al.* [23] concentrated on cloud detection on polarization images to improve the accuracy of cloud detection in images with special scenes, such as bright land surface and severe haze. Even though the threshold-based methods are straightforward and simple, in some complex situations, such as thin clouds, clouds/ice coexistence, or cloud on bright surfaces, they are not robust.

B. Machine Learning Methods

To overcome the limitation of threshold-based algorithms, various researchers make full use of machine learning algorithms in cloud detection. Ishida *et al.* [25] applied the SVM, and empirical results showed that it was adjustable concerning clarified incorrect results. The comparison of two SVM-based cloud detection methods, i.e., CLAUDIA1–CAI and CLAUDIA3–CAI, was implemented in various land cover types in [26], Oishi *et al.* concluded that CLAUDIA3–CAI identifies bright surface and optically thin clouds. Other methods, such as [27], applied a Bayesian cloud detection scheme to 37 years of Advanced Very High-Resolution Radiometer (AVHRR) Global Area Coverage (GAC) data. In this method, features used in the observation vector and prior background knowledge are assumed independent. However, the relationship among features is ignored [24]. In conclusion, machine learning algorithms have always taken the manual-crafted features as input, and under complex scenarios, the discriminable features for cloud detection are difficult to obtain.

C. Deep Learning Methods

DL-based methods, especially the CNN, consider plenty of information extracted from images, including texture, color, shape, and correlation between spatial information. The combination makes the DL methods to achieve the state-of-the-art performance for image processing and remote sensing applications [28], [29]. Shi *et al.* [16] developed a model of CNNs, which can detect both thin cloud and thick cloud even in complex background scenes. In [30], multiscale features of presegmented superpixels are extracted via a two-branch deep CNN and employed for the superpixels classification to identify thick cloud, thin cloud, and noncloud. Motivated by the success of DL-based semantic segmentation models, such as UNet [31], fully convolutional network (FCN) [32], and ResNet [33], several algorithms were developed for cloud detection in RSIs. Drönner *et al.* [34] showed that their U-shape-like cloud segmentation CNN (CS-CNN) utilized cloud characteristics on cloud detection: cloud is spatially continuous entities. Zhan *et al.* [35] designed an FCN to learn deep patterns of cloud and snow in multispectrum satellite images. Yang *et al.* [15] proposed a cloud detection neural network (CDnet) with an encoder–decoder structure. The encoder part is a modified ResNet50 structure, which extracted features via deep-stacked CNN blocks. Mohajerani and Saeedi [13] trained an end-to-end FCN on multiple patches of LandSat8 images. However, existing DL cloud detection methods rarely consider the conflict between their big model size, computation complexity, and the resource limitation on satellites.

To speed up the computation and save memory cost, researchers have made numerous attempts in lightweight and real-time model development in semantic segmentation [37]–[39]. A real-time semantic segmentation CNN was designed for high-resolution image data [51]. Yang *et al.* [40] introduced a novel dense dual-path network (DDP-Net) for real-time semantic segmentation. The network was composed of a backbone with dense connectivity to facilitate feature reuse. In [11], an asymmetric encoder–decoder architecture was employed for the real-time semantic segmentation task. In the encoder, channel split and shuffle were adopted in residual blocks to reduce computation cost while maintaining segmentation accuracy. However, these lightweight segmentation models are designed for general images, which fails to well incorporate specific characteristics of RSIs.

D. Salient Object Detection

In some RSIs, the cloud can be treated as the salient object from the background. Also, the aim of salient object detection is to discover the salient objects, which are attractive and visual distinctive objects or regions [41]. However, it is worth mentioning that the salient object is different from cloud detection in the following aspects. First, salient object detection focuses on the most attractive objects in images, and the objects can be from different classes. Instead of intraclass features, spatial edge and boundary features are more important in salient object detection [42]. Second, depth information is used as comprehensive information in some salient object detection models [43]; however, such information is not available in cloud detection dataset. Third, in some of the cloud/thin-cloud, cloud/ice, and cloud/snow coexistence scenes, cloud is even not treated as the salient object [44].

III. Lightweight Cloud Detection Method

A. Overall Framework

The proposed framework exploits an encoder–decoder network structure, as shown in Fig. 1. In the encoder part, spatial and context features are captured simultaneously in a lightweight way in the designed backbone. A combination of short-/long-range semantic features helps feature alignment in RSIs. In addition, high-level multiscale features are fused with global context features to further add long-range semantic features and global background information to maximize the similarity of interclass features. In the decoder part, the lost low-level features are compensated for the output features. These low-level features come from outputs of two continuous encoder stages and are then fused in an efficient way for a better feature representation.

In particular, in the encoder part, the backbone is a ResNet-style module built on our proposed lightweight novel block LWDBB. The number of parameters and operation times is greatly reduced with LWDBB. Furthermore, by stacking LWDBBs properly, spatial information and different ranges of context information are extracted and fused simultaneously.

Given a multichannel RSI I_I as input, a downsampling unit is first adopted in the encoder. It concatenates outputs of convolution and max pooling. The downsampling unit helps reduce computation and meanwhile extract diverse information. The extracted feature map of the downsampling unit can be denoted as

$$F_{\text{downsampling}} = \text{concat}\big(H_{\text{conv}}(I_I), \text{MaxPool}(I_I)\big) \tag{1}$$

where $H_{\text{conv}}(\cdot)$ denotes the convolution operation, MaxPool is the max pooling, and concat means the concatenate operation.

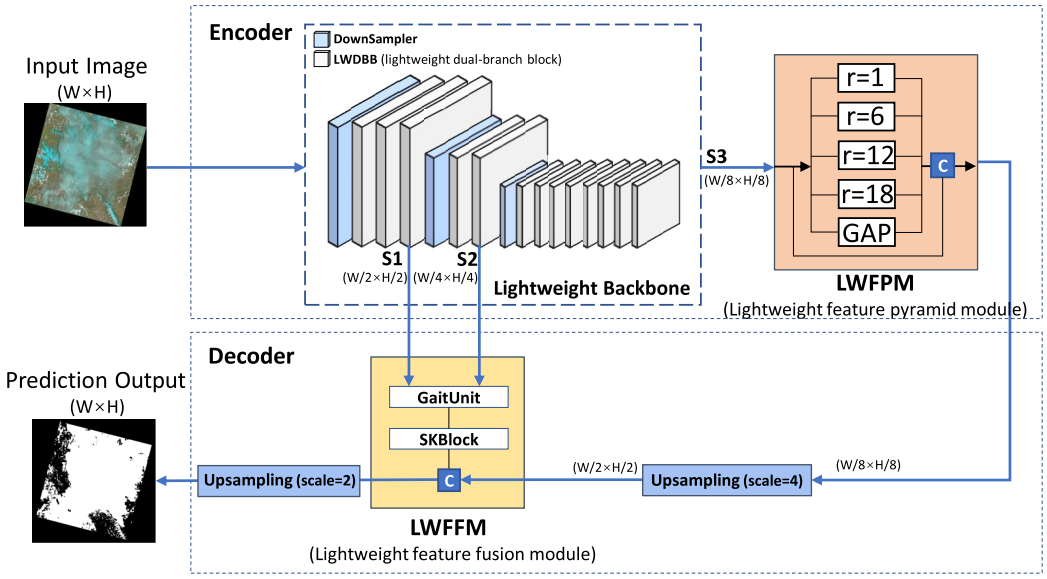


Fig. 1. Overall framework of the proposed methodology. The encoder is constructed with a lightweight backbone and an LWFPM. Here, the backbone is built on downsampler (light blue block) and the proposed lightweight dual-branch block (LWDBB). The decoder part is based on our proposed LWFFM.

Then, three lightweight dual-branch blocks (LWDBBs) are stacked, and they have dilated rates separately (2, 3, and 5) to get rid of "gridding issue" [45]. The feature map of the first stage can be stated as

$$S_1 = H_{\text{LWDBB}}\Big(H_{\text{LWDBB}}\big(H_{\text{LWDBB}}\big(F_{\text{downsampling}}(I_1)\big)\big)\Big) \quad (2)$$

where $H_{\text{LWDBB}}(\cdot)$ denotes the proposed LWDBB and $F_{\text{downsampling}}$ is the downsampling unit. Here, S_1 is $1/2 \times 1/2$ size of the input image.

In the second stage, there are two LWDBBs with dilated rate (2 and 3) following a downsampling unit:

$$S_2 = H_{\text{LWDBB}}\Big(H_{\text{LWDBB}}\big(F_{\text{downsampling}}(S_1)\big)\Big). \quad (3)$$

Here, S_2 is $1/4 \times 1/4$ size of the input image. In the third stage, the dilated rates for eight stages are as in lightweight encoder–decoder network (LEDNet) [11] (1, 2, 5, 9, 2, 5, 9, and 17)

$$S_3 = H^8_{\text{LWDBB}}\big(F_{\text{downsampling}}(S_2)\big) \quad (4)$$

where H^8_{LWDBB} means that H_{LWDBB} is stacked eight times and S_3 is $1/8 \times 1/8$ size of the input image.

Following the backbone, the proposed lightweight atrous spatial pyramid pooling (ASPP) (LWFPM) extracts multiscale and global context information with fewer memory consumption. The output of encoder is

$$S_{\text{encoder}} = F_{\text{LWFPM}}(S_3). \quad (5)$$

In the decoder part, to further improve the segmentation accuracy, the lost detail spatial information during downsampling is compensated in the encoder result via the feature fusion module LWFFM. Instead of directly deploying the low-level feature, LWFFM takes S1 and S2 as input and incorporates the low-level feature to encoder output. The output feature map of LWFFM is $1/2$ size of the input image. The output feature maps of the encoder are upsampled four times and concatenated with the output of LWFFM. Then, the feature map is restored to the size of the input image. The decoder part of LWCDnet can be stated as

$$I_O = F_{\text{LWFPM}}(S_1, S_2, S_{\text{encoder}}) \quad (6)$$

where I_O is the predicted cloud detection result.

In Sections III-B–III-D, we will introduce the main parts: LWDBB, LWFPM, and LWFFM, respectively.

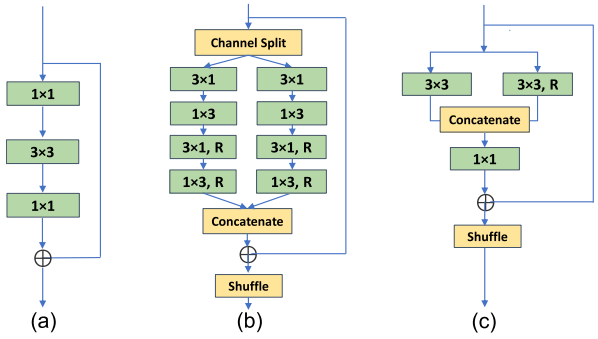


Fig. 2. Comparison of different blocks used in efficient CNNs. (a) Bottleneck in MobileNet [48]. (b) Nonbottleneck in LEDNet [11]. (c) Our proposed lightweight dual-branch block (LWDBB).

B. Lightweight Dual-Branch Block (LWDBB)

In recent years, multiple successful lightweight residual layers are proposed, e.g., bottleneck employed in MobileNet [48] Fig. 2(a) and nonbottleneck employed in LEDNet [11] Fig. 2(b). To reduce the number of parameters in the model, depthwise separable convolution is normally used to replace the normal convolution. The depthwise separable convolution is the combination of depthwise convolution and pointwise convolution (shown in Fig. 3). In depthwise convolution,

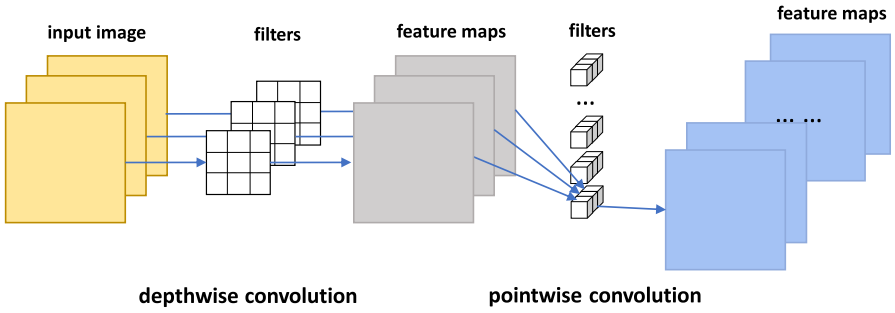


Fig. 3. Separable convolution, which consists of depthwise convolution and pointwise convolution.

each filter channel only works at one input channel, and the number of output features channels is exactly the same as that of input. In pointwise convolution, 1×1 kernel is used to fuse the output channel features from depthwise convolution and increase the image depth. For a convolution with 3×3 kernel size, taking the LandSat8 image as example, which is of size $[384 \times 384 \times 4]$, when the output channel is set to OC, in normal convolution, the number of parameters is $3 \times 3 \times 4 \times \mathrm{OC}$ and the multiple times is $3 \times 3 \times [384-3+1] \times [384-3+1] \times 3 \times \mathrm{OC}$. In depthwise separable convolution, the number of parameters is the sum of $3 \times 3 \times 4$ from depthwise convolution and $1 \times 1 \times 4 \times \mathrm{OC}$ from pointwise convolution. The multiple times is the sum of $3 \times 3 \times [384-3+1] \times [384-3+1] \times 3$ from depthwise convolution and $1 \times 1 \times 384 \times 384 \times 4 \times \mathrm{OC}$ from pointwise convolution. With depthwise separable convolution, both the number of parameters and calculation amount drop greatly. However, pointwise convolution accounts for lots of computational complexity. In nonbottleneck in LEDNet [11], the asymmetric convolutions are stacked four times to replace the depthwise separable convolution. However, from [12], the adoption of asymmetric convolutions in low levels is proven to decrease the accuracy. Therefore, the challenge of reducing computational complexity and keeping the performance at the same time still exists.

To reduce the number of parameters and computation introduced by pointwise convolution in the bottleneck model, we proposed the novel module LWDBB Fig. 2(c), which is a nonbottleneck dual-branch model. The LWDBB is designed to extract spatial and context information simultaneously to make full use of semantic information for feature alignment in RSIs. Also, in the nonbottleneck structure, the number of parameters and operation times is reduced by adopting convolution with kernel size 1×1 only once in feature fusion. The LWDBB is a dual-branch module. The left branch is a depthwise convolution for spatial information extraction, whereas the right branch is a dilated depthwise convolution with a rate r. Compared to normal convolution, it enlarges the receptive field with a square of exponentially increasing size and avoids computation explosion at the same time. Then, the outputs of two branches are concatenated and then fused with a 1×1 convolution. In addition, the residual part [33], which is widely used in deep CNNs to avoid gradients vanishing, is skip-connected in this block. It ensures that LWDBBs can be stacked deeply and more feature is extracted in the encoder [33]. Then, a channel shuffle operation is set after residual connection. In the channel shuffle step, information flows between groups [47], and network capacity is enlarged by the shuffled feature without significantly increasing in parameters and computation complexity. The LWDBB can be stated as

$$H_{\mathrm{LWDBB}} = \mathrm{shuffle} \times \left(H_{\mathrm{pointwise}} \left(\mathrm{concat} \left(H_{\mathrm{depthwise}}(I_I) H_{\mathrm{depthwise}-r}(I_I) \right) \right) + I_I \right) \tag{7}$$

where I_I is the input feature map for LWDBB, $H_{\mathrm{depthwise}}$ is the normal depthwise convolution, $H_{\mathrm{depthwise}-r}$ is the depthwise convolution with dilate rate r, $H_{\mathrm{pointwise}}$ is the pointwise convolution, and concat and shuffle denote concatenate and channel shuffle operation, respectively.

The LWDBBs are stacked and structured in a hybrid dilated convolution (HDC) framework proposed in [45] to avoid the possible issues caused by dilated convolution, that is, first, in a single dilated convolutional layer, not all pixels are involved in the calculation and the involved pixels may not be continuous in position. Second, for big objects, convolutional kernels of big size can lead to high accuracy. Nevertheless, small objects could be skipped if kernels of big size were adopted. In the backbone, LWDBBs are stacked with repeated and a range of slow growth dilation rates. It alleviates the problems caused by uncontinuous pixels in dilated convolution operations. In addition, the dilation rates are relatively prime numbers, and it helps avoid the gridding effect caused by dilated convolution.

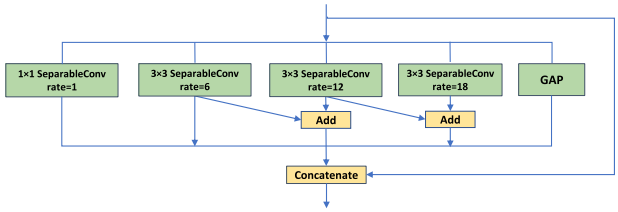


Fig. 4. LWFPM.

C. Lightweight Feature Pyramid Module

In RSIs, some interclass features are similar, and it challenges the robustness of cloud detection methods. As discussed

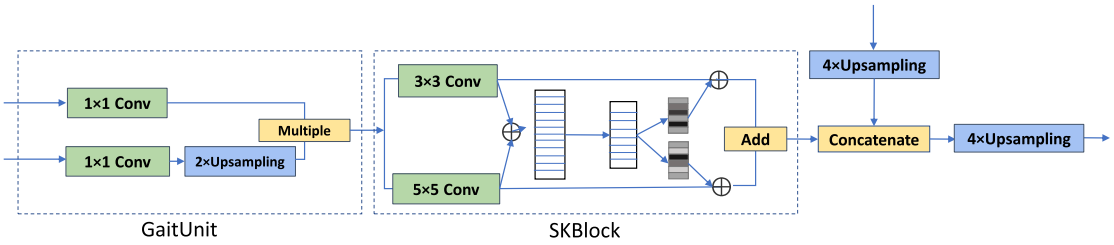


Fig. 5. LWFFM, which is built on a GaitUnit and an SEblock.

before, long-range context features and the background scenario can help in distinguishing cloud from other objects by utilizing the background area information of object. In our developed LWFPM, global and multiscale context information are extracted in an efficient way. Inspired by the ASPP model [49], we construct the LWFPM with multiple parallels dilated convolutional layers with different sampling rates, as shown in Fig. 4. To reduce the number of parameters, the normal convolution operations are replaced with depthwise separable convolutions. In order to capture multiscale context information, the receptive field is parallel enlarged with dilated rates of 1, 6, 12, and 18 [53]. In addition, a global average pooling [46] block for capturing global context information is added. To avoid the gradient vanishing problem and reuse features without increment in computation, a short connection is added. Then, all features are connected as the output of the feature pyramid module. The LWFPM module can be presented as

$$\begin{aligned} F_{\text{LWFPM}} &= H_{\text{conv}}\Big(\text{concate}\big(S3, H_{\text{conv}}(S3), H_{\text{separable}-r6}(S3) \\ &\qquad H_{\text{separable}-r12}(S3) + H_{\text{separable}-r6}(S3) \\ &\qquad H_{\text{separable}-r18}(S3) + H_{\text{separable}-r12}(S3)\big)\Big) \end{aligned} \tag{8}$$

where $H_{\text{separable}-r}$ is the separable convolution with dilate rate r.

D. Lightweight Feature Fusion Model

During downsampling in the encoder, the detail spatial information is lost. Restoring the lost detail spatial information in the decoder is essential for refining the final detection. Moreover, by reusing as many features as possible in the decoder, the number of parameters and the computation complexity of the network can be reduced. In the proposed LWFFM (see Fig. 5), the detail information is processed first to obtain diverse features and then concatenated to the output of the encoder. The inputs of LWFFM are output feature maps of S1 and S2 from the encoder. The feature map of S2 is $1/4 \times 1/4$ size of the input image, and in the gaitunit, it is restored to the size of the feature map of S1, which is $1/2 \times 1/2$ size of the input image. Then, they are multiplied. To enhance the significance of interdependent features by exploiting the channel relationships, we use an SKBlock [38]. In SKBlock, the input feature map is split into two branches with kernel sizes 3 and 5, branches with different receptive fields bring different information. Guided by fused information from branches, the channel attention weights are generated, and then, the channel correlated features are extracted. Finally, the output feature map of the encoder is restored to the same scale as that of SKBlock. The LWFFM takes two low-level feature maps S1 and S2 and the output of the encoder as input, and the fused feature map of them is

$$F_{\text{GaitUnit}} = H_{\text{conv}}(S_1) \otimes H_{\text{upsampling}}\big(H_{\text{conv}}(S_2)\big). \tag{9}$$

Here, the output feature map of gaitunit is denoted as F_{GaitUnit}. It is then forwarded to SKBlock and concatenated with unsampled encoder output as the network predict result

$$\begin{aligned} F_{\text{LWFFM}} = H_{\text{conv}}\Big(&\text{concat}\big(\text{SKBlock}(F_{\text{GaitUnit}})\big) \\ &H_{\text{upsampling}}(F_{\text{encoder}})\Big). \end{aligned} \tag{10}$$

Here, the fused feature map is processed via SKBlock and concatenated with upsampled encoder result with $H_{\text{upsampling}}$ function. F_{LWFFM} denotes the output feature map of LWFFM.

IV. Experimental Results

In this section, we evaluated the proposed LWCDnet by conducting experiments on two remote sensing datasets: LandSat8 and MODIS. We first introduce the dataset and experimental setup. Then, we present the performance of modules build on two backbones and state the comparison with state-of-the-art cloud detection and semantic segmentation approaches. Finally, we analyze the contribution of each proposed module by ablation study.

A. Dataset and Experimental Setup

1) LandSat8 Dataset: The LandSat8 is a remote sensing dataset for cloud detection, which is first released in [13] and originally named the 38-Cloud dataset. The dataset contains 18 LandSat8 images of size 1000×1000 for training and 20 same-size images for the test. The training set contains 8400 cropped patches with the size 384×384 as training samples, and there are 9201 patches with the same size in test data (see Table I). Each patch has four corresponding spectral channels: red (band 4), green (band 3), blue (band 2), and NIR (band 5). Fig. 6(a) and (b) shows two sample images in the LandSat8 dataset, and their background is easy to distinguish from the cloud. However, there are plenty of areas covered with thin cloud.

In addition, we pick up 484 patches from LandSat8 and set up a thin-cloud subdataset for the test. These patches are

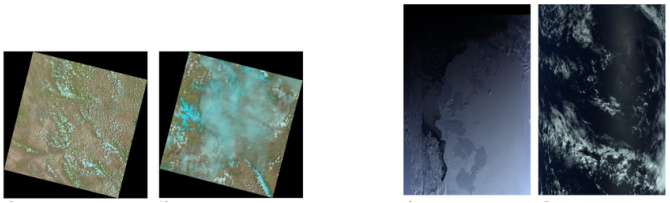


Fig. 6. Samples of several false-color images. (a) and (b) From LandSat8. (c) and (d) From MODIS.

TABLE I
STATISTICS OF DATASET LANDSAT8 AND MODIS

	LandSat8	**MODIS**
♯ Bands	4	10
♯ Training	7000	17880
♯ Valid	1400	1200
♯ Test	9201	2250
Patch Size	384×384	512×512

mostly covered by thin cloud and utilized to test the cloud detection performance on thin cloud.

2) MODIS Dataset: MODIS is an important Earth observation system in NASA. The dataset used for performance evaluation contains the first five days images of each month in 2005 from the MODIS Level 1B product. Images in the MODIS dataset mainly contain four scenes: ocean, land, land and ocean, and polar glaciers. The coexistence of ice and clouds increases the difficulty of detection since their temperature characteristics are similar. In this dataset for cloud detection, some obviously problematic images have been removed. Also, channels in which cloud cannot be separated from other objects are ignored in the preprocessing step. Finally, 1422 RSIs with ten selected informative channels (bands 1, 3, 4, 18, 20, 23, 28, 29, 31, and 32) from the original 36 bands are picked. The cloud mask is from the MODIS Atmosphere L2 Cloud Mask Product [14]. They are separated into a training set with 1192 samples, a validation set with 80 samples, and a test set with 150 samples. Each image is cropped into overlapping patches with size 512×512, resulting in 17 880 images in the training set, 1200 images in the validation set, and 2250 images in the test set (see Table I). Fig. 6(c) and (d) shows the samples from the MODIS dataset. The scenes of polar [see Fig. 6(c)] and ocean [see Fig. 6(d)] account for a large part of them, which definitely makes cloud detection very difficult. MODIS cloud detection dataset is available at HTTPS://github.com/xiachangxue/MODIS-Dataset-for-Cloud-Detection.

3) Evaluation Metrics: For each pixel, the prediction result and ground truth have two classes: cloud and noncloud. The performance is measured via metrics widely used in semantic segmentation, including Jaccard index, precision, recall, mean intersection over union (MIoU), F1-score, and overall accuracy. These metrics are defined as follows:

$$\text{JaccardIndex} = \frac{\text{TP}}{\text{TP} + \text{FN} + \text{FP}} \tag{11}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{12}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{13}$$

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{14}$$

$$\text{Overall Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}} \tag{15}$$

$$\text{MIoU} = \frac{1}{k+1} \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{i=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}}. \tag{16}$$

Here, TP, TN, FP, and FN are the total number of true-positive, true-negative, false-positive, and false-negative pixels, respectively. The Jaccard index and MIoU are two widely adopted metrics for measuring the performance in image segmentation tasks. MIoU combines the accuracies of both cloud pixels and noncloud pixels in the calculation.

To measure the efficiency of models, in LandSat8, we collected the metrics, including giga FLOPs (GFLOPs), the number of model parameters, time spent for inference per image, model size, and memory occupied by model.

4) Parameter Settings: The evaluated networks equipped with our proposed modules are all implemented with the Pytorch framework and optimized by the Adam optimizer. The training step runs on Ubuntu 16.05 with two RTX 3090 GPUs in 100 epochs. The learning rate starts from 0.01 and decays with the policy that from epoch 36, the learning rate decays to 0.008; from epoch 65, it decays to 0.005; and from epoch 85, it decays to 0.003. All other methods are trained with the same configuration and settings without pretrain.

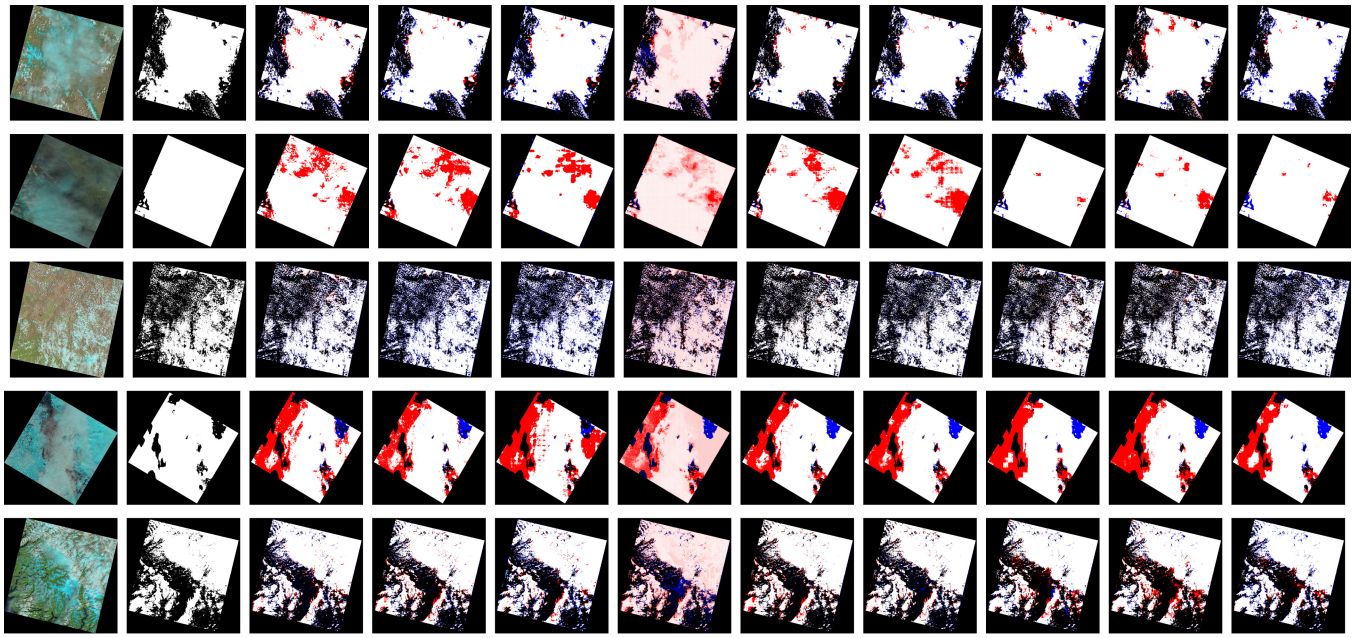
5) Baseline Approaches: The comparing baseline methods include classic cloud detection methods: FCN [32], U-Net [31], and Deeplab [53]. In addition, some state-of-the-art cloud detection methods are taken into comparison: CS-CNN [34], remote sensing network (RS-Net) [36], multiscale features-CNN (MF-CNN) [50], CDNet [15], and multiscale fusion gated network (MFGNet) [17]. To validate the cloud detection performance and efficiency meanwhile, the state-of-the-art real-time semantic segmentation methods, MobileNet [48], image cascade network (ICNet) [52], deep feature aggregation network (DFANet) [39], LEDNet [11], DDP-Net [40], and height-driven attention network (HANet) [20], are compared with our approaches.

B. Performance Comparison

1) Cloud Detection Results on LandSat8 Dataset: Table II reports the results of different cloud detection methods and lightweight semantic segmentation methods on the LandSat8 dataset. From the results, in comparison with cloud detection methods, our proposed LWCDnet outperforms most of them. The MF-CNN achieves slightly higher MIoU, Jaccard, and overall scores, while MFGNet obtains a marginally better recall score than our model. Even though FCN has the highest precision value, its recall value is much worse than that of our methods. The results show that the performance of our proposed LWCDnet is very close to the best metric scores of state-of-the-art cloud detection methods. In comparison with

TABLE II
COMPARISON WITH CLOUD DETECTION METHODS AND SEMANTIC SEGMENTATION METHODS ON THE LANDSAT8 DATASET

Category	Methods	Jaccard	Precision	Recall	MIoU	Overall	F1-Score
Cloud Detection Methods	FCN [32]	83.2	**95.3**	86.7	87.7	94.4	90.8
	U-Net [31]	85	93.2	90.6	88.9	94.9	91.9
	Deeplab [53]	84.7	92.5	90.8	88.6	94.7	91.6
	CS-CNN [34]	73.6	87.9	91.2	81.0	95.3	89.5
	RS-Net [36]	85.8	93.4	91.4	89.5	95.2	92.4
	MF-CNN [50]	87	95.1	91	90.4	95.6	93.0
	CDNet [15]	85.9	94.3	94.7	89.7	95.4	94.5
	MFGNet [17]	85.2	93.3	**95.6**	89.3	95.3	94.4
Semantic Segmentation Methods	MobileNet [48]	85.5	93.5	95.4	89.5	95.3	94.4
	ICNet [52]	85.4	93.0	95.5	89.4	95.3	94.2
	DFAnet [39]	85.4	92.1	93.1	86.3	93.7	92.6
	LEDNet [11]	85.4	93.5	95.4	89.5	95.3	94.4
	HANet [20]	**87.7**	94	96.3	**91.2**	**96.1**	95.1
	DDP-Net [40]	**87.7**	95.0	95.5	91.0	96.0	**95.3**
	LWCDnet (Ours)	86.2	95.1	94.3	89.9	95.3	94.7



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k)

Fig. 7. Visual comparisons of different cloud detection methods in the scene of five examples from LandSat8 dataset. White area represents cloud and black area represents noncloud. In addition, red area represents false-positive detection and blue area represents true-negative detection. (a) False-color RSI. (b) Ground truth. (c) Result of FCN [32]. (d) Result of U-Net [31]. (e) Result of DeepLab [53]. (f) Result of CS-CNN [34]. (g) Result of RS-Net [36]. (h) Result of MF-CNN [50]. (i) Result of CDNet [15]. (j) Result of MFGNet [17]. (k) Result of LWCDnet (ours).

lightweight semantic segmentation methods, our LWCDnet can achieve satisfactory performance in metrics, only inferior to HANet and DDP-Net.

Fig. 7 shows the visual comparison of cloud segmentation methods on five examples from the LandSat8 dataset. The examples include a diverse backgrounds, e.g., thin-cloud and cloud-ice coexisting cases. From the results, it is significant that LWCDnet obtains comparable performance. In addition, it performs even better in some complicated situations like thin cloud. Fig. 8 shows the visual results of cloud segmentation methods, and our proposed model can accurately identify the cloud in diverse backgrounds. The proposed methodology can achieve better or comparable performance in cloud detection. What should be noted is that the last two samples are of scenarios ice/snow and cloud coexistence. From the visual results, our proposed LWCDnet has less false-positive detection (red area) and is better at distinguishing cloud from ice/snow. However, when the cloud overlaps with the ice/snow, our LWCDNet has more true-negative detections because of the introduction of context information.

2) Cloud Detection Results on MODIS Dataset: Table III shows the results of different cloud detection and semantic segmentation methods on the MODIS dataset data. Compared to the cloud detection methods, our LWCDnet achieves better or similar performance. Compared to MF-CNN which has the highest value in MIoU, our method LWCDnet has a 1.6% decrease in MIoU. LWCDnet performs best in all metrics in comparison with other semantic segmentation methods. Different from the LandSat8 dataset, the special scenes, such as oceans and polar ice regions in the MODIS dataset, have much more requirements for cloud boundary processing in

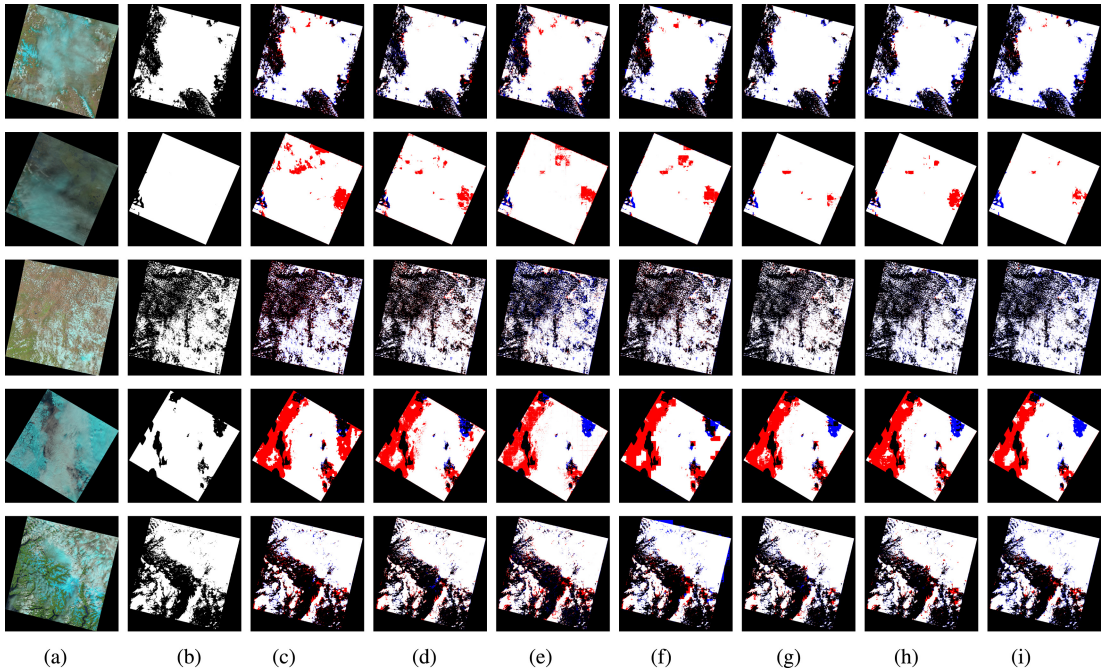


Fig. 8. Visual comparisons of different lightweight semantic segmentation methods in the scene of five examples from LandSat8 dataset. White area represents cloud and black area represents noncloud. In addition, red area represents false-positive detection and blue area represents true-negative detection. (a) False-color RSI. (b) Ground truth. (c) Result of MobileNet [48]. (d) Result of ICNet [52]. (e) Result of DFAnet [39]. (f) Result of LEDNet [11]. (g) Result of HANet [20]. (h) Result of DDP-Net [40]. (i) Result of LWCDnet (ours).

TABLE III
COMPARISON WITH CLOUD DETECTION AND LIGHTWEIGHT SEMANTIC SEGMENTATION METHODS ON THE MODIS DATASET

Category	Methods	Jaccard	Precision	Recall	MIoU	Overall	F1-Score
Cloud Detection Methods	FCN [32]	89.4	92.3	91.6	86.3	93.1	92.0
	UNet [31]	89.7	92.6	91.5	86.8	93.2	92.1
	Deeplab [53]	86.9	91.0	91.9	82.8	91.1	91.5
	CS-CNN [34]	87.2	91.2	90.7	83.4	91.5	82.3
	RS-Net [36]	89.7	**95.2**	92.5	86.6	93.3	**93.8**
	MF-CNN [50]	**90.0**	92.4	**93.5**	**86.9**	**93.4**	93.0
	CDNet [15]	89.0	92.0	92.4	85.8	92.8	92.0
	MFGNet [17]	87.7	90.5	**93.5**	83.5	91.6	92.0
Semantic Segmentation Methods	MobileNet [48]	82.2	87.3	86.5	76.9	87.7	81.4
	ICNet [52]	85.2	87.5	88.3	79.8	89.6	87.9
	DFAnet [39]	81.3	85.9	84.6	74.5	86.5	85.3
	LEDNet [11]	85.8	89.8	88.7	80.6	90.1	89.3
	HANet [20]	87.8	91.5	91.4	84.3	92.0	87.7
	DDP-Net [40]	88.6	92.1	91.8	85.2	92.4	91.9
	LWCDnet (Ours)	89.0	92.6	91.7	85.5	92.7	92.1

cloud detection methods. However, to keep the parameter and size of the model at a low level, the lightweight model usually adopted the dilate revolution or multiresolution; they all miss some detail information in the process. The best available tradeoff between accuracy and efficiency is the target in the lightweight model.

Fig. 9 shows the visual comparison of cloud detection methods with our LWCDnet on three examples from the MODIS dataset. The false-color images show that the cloud in imagery in MODIS has more detail information than that in LandSat8. However, in such a situation, our LWCDnet has comparable performance. Fig. 10 shows the visual results of semantic segmentation methods, our proposed model has much better comparable results in scenes.

3) Cloud Detection Results on the Subdataset of Thin Cloud: Table IV reports the test results of different cloud detection and lightweight semantic segmentation methods on the thin-cloud subdataset from the LandSat8 Dataset. In comparison with the stated methods, our proposed LWCDnet has an inferior performance to CDNet on metrics besides precision and has the second-best performance over the other methods.

For visual comparison, we picked three samples from thin-cloud zones. These zones are included in the thin-cloud subdataset and are almost totally covered by thin cloud from the ground truths. Fig. 11 shows the visual comparison of cloud detection methods with our LWCDnet on them. From the visual results, our proposed LWCDnet and CDNet have much better performance than other methods. The false-positive

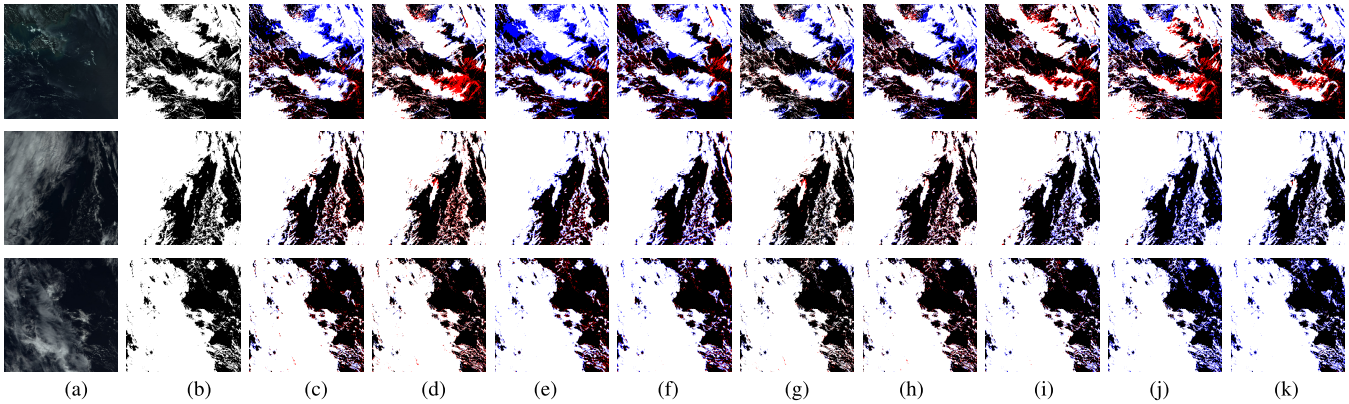


Fig. 9. Visual comparisons of different cloud detection methods in the scene of three examples from the MODIS dataset. White area represents cloud and black area represents noncloud. In addition, red area represents false-positive detection and blue area represents true-negative detection. (a) False-color RSI. (b) Ground truth. (c) Result of FCN [32]. (d) Result of U-Net [31]. (e) Result of DeepLab [53]. (f) Result of CS-CNN [34]. (g) Result of RS-Net [36]. (h) Result of MF-CNN [50]. (i) Result of CDNet [15]. (j) Result of MFGNet [17]. (k) Result of LWCDnet (ours).

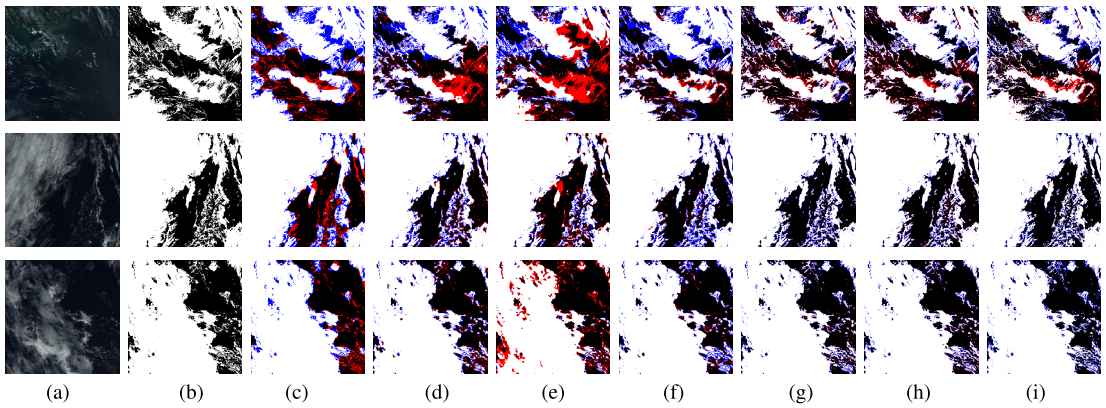


Fig. 10. Visual comparisons of different lightweight semantic segmentation methods in the scene of three examples from MODIS dataset. White area represents cloud and black area represents noncloud. In addition, red area represents false-positive detection and blue area represents true-negative detection. (a) False-color RSI. (b) Ground truth. (c) Result of MobileNet [48]. (d) Result of ICNet [52]. (e) Result of DFAnet [39]. (f) Result of LEDNet [11]. (g) Result of HANet [20]. (h) Result of DDP-Net [40]. (i) Result of LWCDnet (ours).

TABLE IV

COMPARISON WITH CLOUD DETECTION METHODS AND LIGHTWEIGHT SEMANTIC SEGMENTATION METHODS ON THE THIN-CLOUD SUBDATASET FROM THE LANDSAT8 DATASET

Category	Methods	Jaccard	Precision	Recall	MIoU	Overall	F1-Score
Cloud Detection Methods	FCN [32]	75.2	85.0	81.6	69.7	82.9	83.3
	U-Net [31]	71.0	77.4	82.1	64.2	80.1	79.7
	Deeplab [53]	77.7	81.5	81.9	69.4	83.5	81.7
	CS-CNN [34]	61.9	78.8	76.0	58.3	73.9	77.4
	RS-Net [36]	70.0	78.2	81.4	64.4	79.0	79.8
	MF-CNN [50]	72.8	79.6	82.8	66.9	80.9	81.2
	CDNet [15]	**83.0**	86.1	**86.2**	**76.0**	**87.6**	**86.1**
	MFGNet [17]	70.5	82.8	79.4	65.5	79.7	81.1
Semantic Segmentation Methods	MobileNet [48]	78.2	87.0	83.5	73.0	85.0	85.2
	ICNet [52]	75.8	86.2	82.5	70.8	83.5	84.3
	DFAnet [39]	68.6	80.5	77.3	63.0	77.9	78.9
	LEDNet [11]	75.2	85.0	81.6	69.7	82.9	83.3
	HANet [20]	80.9	**86.8**	84.6	74.8	86.5	85.7
	DDP-Net [40]	78.9	84.1	82.8	71.7	84.8	83.4
	LWCDnet (Ours)	**82.0**	**86.7**	**85.3**	**75.6**	**87.2**	**86.0**

detection of thin cloud (red area) by LWCDnet and CDNet is much smaller than by others. Fig. 12 presents the visual results of lightweight semantic segmentation methods and our LWCDnet on thin-cloud zones. In summary, our proposed LWCDnet has much smaller false-positive detection areas of thin cloud (red area) and works great in thin-cloud detection.

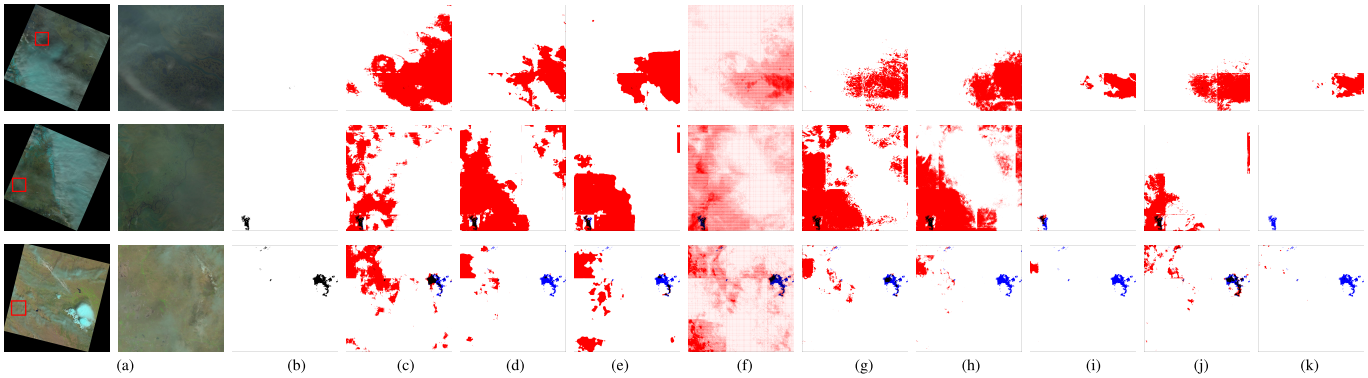


Fig. 11. Visual comparisons of different cloud detection methods in the scene of three examples from the thin-cloud subdataset of LandSat8. White area represents cloud and black area represents noncloud. In addition, red area represents false-positive detection and blue area represents true-negative detection. (a) False-color RSI. (b) Ground truth. (c) Result of FCN [32]. (d) Result of U-Net [31]. (e) Result of DeepLab [53]. (f) Result of CS-CNN [34]. (g) Result of RS-Net [36]. (h) Result of MF-CNN [50]. (i) Result of CDNet [15]. (j) Result of MFGnet [17]. (k) Result of LWCDnet (ours).

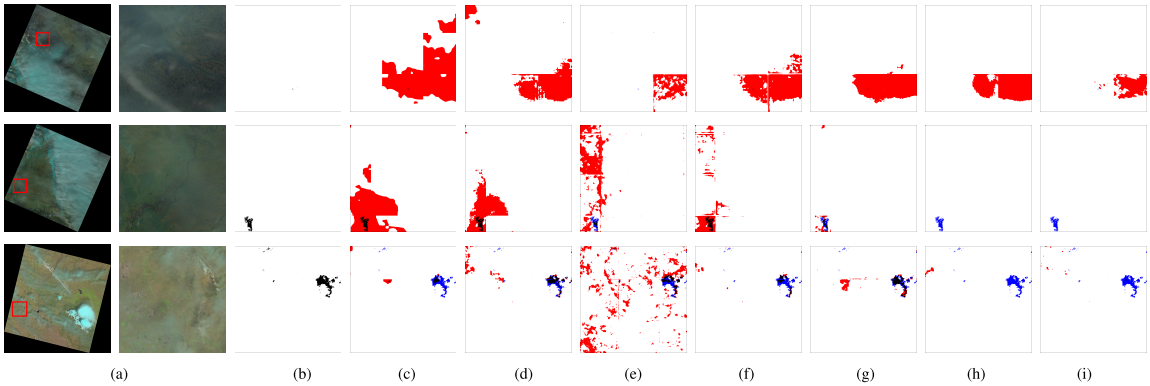


Fig. 12. Visual comparisons of different lightweight semantic segmentation methods in the scene of three examples from the thin-cloud subdataset of LandSat8. White area represents cloud and black area represents noncloud. In addition, red area represents false-positive detection and blue area represents true-negative detection. (a) False-color RSI. (b) Ground truth. (c) Result of MobileNet [48]. (d) Result of ICNet [52]. (e) Result of DFAnet [39]. (f) Result of LEDNet [11]. (g) Result of HANet [20]. (h) Result of DDP-Net [40]. (i) Result of LWCDnet (ours).

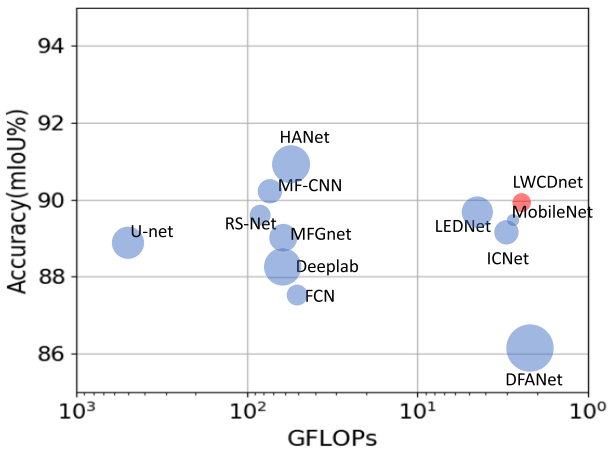


Fig. 13. GFLOPs, MIoU performance, and inference speed on the LandSat8 dataset. x-axis is the GFLOPs, y-axis is the accuracy presented by MIoU, and the bubble size represents the inference time per image. The smaller bubble, the faster inference speed. Here, our method is compared with cloud detection methods (FCN [32], U-net [31], Deeplab [53], RS-Net [36], MF-CNN [50], and MFGNet [17]) and real-time semantic segmentation methods (MobileNet [48], ICNet [52], LEDNet [11], DFANet [39], and HANet [20]).

C. Efficiency Evaluation

Since the efficiency evaluation results increase proportionally to image size, to evaluate the model efficiency, we summarize the efficiency-related metrics only on LandSat8 and state them in Table V and Fig. 13. As shown, our LWCDnet is the smallest model and has the fewest parameters among all the models. These metrics are crucial for cloud detection on satellite, which has limited storage. Here, the time/image is an average time spent by cloud detection per frame with size 384 × 384. Data and parameters load time is not considered, and the employed GPU is NVIDIA Corporation Device 2204 with 24-G storage. The time spent per image of our LWCDnet is less than other compared cloud detection and semantic segmentation methods, except the MobileNet. Note that our LWCDnet has a smaller memory occupation and, as a result, can run with a larger batch size in the same storage. Each batch load has extra time consumption, and hence, our model has higher speed in situations where the batch size is much larger. In conclusion, our method achieves the state-of-the-art performance in cloud detection and meanwhile is much efficient than methods with comparable accuracy.

D. Ablation Study

To evaluate the proposed key components in LWCDnet, we conducted ablation studies on LandSat8 and MODIS datasets. These models are trained on their training set and tested on the test set. The comparison metrics include the

TABLE V
EFFICIENCY COMPARISON ON THE LANDSAT8 DATASET

Category	Methods	FLOPS (GFLOPs)	Params (10^6)	Time/image(ms)	Model size (M)	Occupied Memory (M)
Cloud Detection Methods	FCN [32]	45.30	18.64	22.8	143	409.78
	UNet [31]	147.50	34.53	27.5	102	1163.25
	Deeplab [53]	50.00	59.34	38.0	454	607.02
	CS-CNN [34]	11.37	8.63	**7.2**	104	97.45
	RS-Net [36]	92.40	7.85	25.0	60	498.90
	MF-CNN [50]	70.57	17.41	29.1	133	412.85
	CDNet [15]	435.20	81.31	79.1	976	1688.64
	MFGNet [17]	68.65	7.83	25.1	90	1613.01
Semantic Segmentation Methods	MobileNet [48]	1.72	4.5	8.4	38	169.6
	ICNet [52]	1.88	4.98	20.9	58	**79.57**
	DFAnet [39]	1.01	2.17	43.1	26	124.45
	LEDNet [11]	3.56	0.92	30.0	11	213.17
	HANet [20]	39.13	40.96	28.1	470	436.53
	DDP-Net [40]	3.31	3.64	31.3	44	752.61
	LWCDnet (Ours)	**1.41**	**0.32**	15.2	**4.2**	243.70

TABLE VI
ABLATION STUDY OF LWFPM AND LWFFM BASED ON LWCDNET

Dataset	Methods	Jaccard	Precision	Recall	MIoU	Overall	F1-Score
LandSat8	Backbone	85.0	94.1	94.2	89.0	95.0	94.1
	Backbone+LWFPM	85.6	94.5	94.2	89.4	95.2	94.3
	Backbone+LWFFM	85.2	94.4	94.1	89.2	95.1	94.2
	Backbone+LWFPM+LWFFM	**86.2**	**95.1**	**94.3**	**89.9**	**95.3**	**94.7**
MODIS	Backbone	87.4	91.5	90.3	83.3	91.5	90.9
	Backbone+LWFPM	87.8	91.9	90.6	83.8	91.8	91.2
	Backbone+LWFFM	89.0	92.9	91.3	85.3	92.6	92.1
	Backbone+LWFPM+LWFFM	**89.0**	**92.6**	**91.7**	**85.5**	**92.7**	**92.1**

performance metrics and two of representative efficiency metrics: GFLOPS and number of parameters.

1) Key Components: The proposed novel key components in LWCDnet are the lightweight backbone, LWFPM, and LWFFM. Models on backbone, backbone + LWFPM, and backbone + LWFFM are trained. Then, each model is evaluated to validate the effectiveness of the developed components. In particular, in backbone + LWFPM, the feature map generated in LWFPM is upsampled directly to the same size as the input image. Also, the ablation experiment backbone + LWFFM is conducted by replacing LWFPM with a normal convolution with kernel size 1. Table VI shows the ablation experiments results on LandSat8 and MODIS. In both datasets, the best performance is obtained when all components are utilized, i.e., backbone + LWFPM + LWFFM. Without using LWFFM, the performance decreases in terms of MIoU. A similar result can be observed when LWFPM is removed. The results demonstrate that both the LWFPM and LWFFM can contribute to the cloud detection accuracy.

2) Lightweight Dual-Branch Blocks (LWDBB): In the encoder of LWCDnet, we proposed the lightweight block LWDBB for spatial and context feature extraction. We compared the LWDBB with two existing lightweight feature extraction blocks: bottleneck adopted in MobileNet [48] and nonbottleneck adopted in LEDNet [11]. The results are stated in Table VII. In LandSat8, the model built on LWDBB has a 1% decrease in MIoU in comparison with the model built on the bottleneck. However, its GFLOPS is only 25% of the model equipped with the bottleneck, and the number of parameters is 22% of that. With LWDBB, even though the performance degrades slightly, the calculation complexity is reduced greatly. By comparing LWDBB with the nonbottleneck, they have almost the same performance in cloud detection. Also, notably, the model built on LWDBB only has half of the complexity of the model built on the nonbottleneck.

3) Lightweight Feature Pyramid Module: To evaluate the effectiveness and efficiency of the proposed LWFPM and ASPP, we conducted the ablation study by replacing the LWFPM with ASPP. Table VIII shows the results. When considering the performance, the model with LWFPM achieves higher scores in all metrics except recall in LandSat8 and precision in MODIS. In addition, the model equipped with LWFPM has fewer parameters and less computation complexity.

4) Inclusion of Context Information in Backbone: In backbone, we adopt the proposed LWDBB, which is of a dual-branch structure. The left branch in LWDBB is designed to extract spatial information with normal convolution, and the right branch in LWDBB is proposed to extract context information with dilated convolution to make full use of semantic information. To evaluate the effectiveness by introducing the branch for context information, we replace the dilated convolution with normal convolution in the right branch in LWCDnet. Table IX presents the results on LandSat8 and

TABLE VII

ABLATION STUDY OF BOTTLENECK EMPLOYED IN MOBILENET [48] [SEE FIG. 2(a)] AND NONBOTTLENECK EMPLOYED IN LEDNET [11] [SEE FIG. 2(b)] AND LWDBB [SEE FIG. 2(c)]

		Performance						Efficiency	
Dataset	**Methods**	**Jaccard**	**Precision**	**Recall**	**MIoU**	**Overall**	**F1-Score**	**GFLOPS**	**Params** (10^6)
LandSat8	bottleneck - Backbone	**87.4**	94.2	**96.5**	**91.0**	**96.0**	**96.4**	5.99	1.47
	non-bottleneck - Backbone	86.5	94.8	94.6	90.0	95.2	94.7	2.96	0.73
	LWDBB - Backbone	86.2	**95.1**	94.3	89.9	95.3	94.7	**1.41**	**0.32**
MODIS	bottleneck - Backbone	**89.6**	**93.1**	**91.9**	**86.0**	**96.0**	**92.5**	6.03	1.47
	non-bottleneck - Backbone	89.1	93.0	91.4	85.5	92.7	92.2	3.0	0.73
	LWDBB - Backbone	89.0	92.6	91.7	85.5	92.7	92.1	**1.59**	**0.32**

TABLE VIII

ABLATION STUDY OF THE PROPOSED LWFPM AND ASPP

		Performance						Efficiency	
Dataset	**Methods**	**Jaccard**	**Precision**	**Recall**	**MIoU**	**Overall**	**F1-Score**	**GFLOPS**	**Params** (10^6)
LandSat8	ASPP	86.0	94.2	**95.0**	89.8	**95.4**	94.6	1.51	0.36
	LWFPM	**86.2**	**95.1**	94.3	**89.9**	95.3	**94.7**	**1.41**	**0.32**
MODIS	ASPP	88.9	**93.0**	91.0	85.0	92.5	92.0	1.65	0.36
	LWFPM	**89.0**	92.6	**91.7**	**85.5**	**92.7**	**92.1**	**1.59**	**0.32**

TABLE IX

ABLATION STUDY OF THE INCLUSION OF CONTEXT INFORMATION IN BACKBONE

		Performance						Efficiency	
Dataset	**Methods**	**Jaccard**	**Precision**	**Recall**	**MIoU**	**Overall**	**F1-Score**	**GFLOPS**	**Params** (10^6)
LandSat8	LWCDnet(without context information)	85.4	94.6	94.0	89.2	95.1	94.3	1.41	0.32
	LWCDnet	**86.2**	**95.1**	**94.3**	**89.9**	**95.3**	**94.7**	1.41	0.32
MODIS	LWCDnet(without context information)	88.8	**92.9**	91.1	85.0	92.5	92.0	1.59	0.32
	LWCDnet	**89.0**	92.6	**91.7**	**85.5**	**92.7**	**92.1**	1.59	0.32

MODIS. In comparison, the efficiency of both models is the same; however, the model included context information shows better cloud detection performance on both datasets.

E. Discussion

On dataset LandSat8, the lightweight semantic segmentation methods, in general, outperform the cloud detection methods, and our proposed LWCDnet is inferior to the lightweight semantic segmentation methods HANet and DDP-Net. This is because these models extract enough multispectral context information by increasing the receptive field (in LEDNet and DDP-Net) in the feature extraction process. As well, the results of our LWCDnet also indicate that the context information is critical for cloud detection accuracy. The HANet in comparison is based on the ResNet-50. Also, in each layer, an HANet model is injected. Distinguishable features are captured by stacking convolution layers and the additional attention module. Also, the model size is large with a deep stack of layers. In addition, the attention affinity matrix is memory intensive. In similarity, DDP-Net has over 40 layers stacked. Even though the number of parameters in DDP-Net is reduced by feature reuse in dense connection, the model needs much more memory to store the intermediate feature maps.

On the thin-cloud subdataset of LandSat8, our proposed LWCDnet is inferior to CDNet in performance comparison; however, the GFLOPs of CDNet are 400 times that of LWCDnet and the inference time of CDNet is over 15 times that of LWCDnet. From Figs. 11 and 12, LWCDnet has some small true-negative detection on the clear areas surrounded by thin cloud. The introduction of long-range context information achieves feature alignment to get rid of the impact of noise; meanwhile, such area is probably to be treated as cloud.

On dataset MODIS, the cloud detection methods outperform in contrast the lightweight semantic segmentation methods. Also, LWCDnet is inferior to the cloud detection methods. The experiment results indicate that in multispectral imagery with complex scenarios, the normal convolution is better than separable depthwise convolution. It is because in the separable depthwise convolution, each filter channel works first at one input channel in depthwise convolution, and then, the outputs are fused in pointwise convolution. However, in some of the single channels, features of cloud and other objects have some overlap. The separation of feature extraction and fusion suppresses the feature expression of cloud, while the separable depthwise convolution is normally adopted by lightweight models to reduce the computation complexity. In our proposed LWCDnet, the designed module LWFFM uses normal convolution in SKBlock. From ablation study Table VI, the inclusion of LWFFM increases the MIoU from 83.3% to 85.3% in MODIS. However, the performance increase caused by LWFFM is only 0.2% in LandSat8.

From the efficiency comparison results, our proposed LWCDnet has the smallest GFLOPS and the fewest parameters because of the use of lightweight modules LWDBB and LWFPM in the encoder. However, the inference time of LWCDnet is longer than that of CS-CNN and MobileNet. In [54], it is proven that the degree of parallelism is reduced by network fragmentation. The "multipath" structure is unfriendly for devices with strong parallel computing powers such as GPU. In CS-CNN, each block consists of one convolution operation, while in MobileNet, this number is 2 or 3. In our LWCDnet, LWDBB contains two parallel convolution operations and LWFPM contains five ones. Therefore, LWCDnet is more fragmental and needs more time in inference. In addition, the smallest feature map in LWCDnet is scaled to 1/8 size of the input image for accuracy. Also, in CS-CNN and MobileNet, this number is 1/16. With the larger intermediate feature maps and more complex model structure, LWCDnet needs more memory. However, when taking efficiency and accuracy into consideration, LWCDnet has the best tradeoff.

V. Conclusion and Future Work

In this article, we are the first to investigate lightweight methods for cloud detection. We develop a novel model LWCDnet to tackle the lightweight cloud detection problem with a deep CNN. In LWCDnet, we effectively utilize context information to achieve the feature alignment in RSIs. The accuracy is maintained with less storage and computation burden. In the encoder, spatial and context features are captured simultaneously via the proposed LWDBB in the three-stage ResNet backbone. In the designed LWFPM, the global and multiscale context information is extracted to further improve the accuracy. Then, the feature maps are compensated with lost low-level features from the designed LWFFM and restored to the size of the input image in the decoder. Extensive experiments have been conducted on LandSat8 and MODIS datasets, and the results demonstrate that LWCDnet can achieve excellent performance and reduce the calculation at the same time.

Although the LWCDnet has satisfactory results in the lightweight cloud detection task, it has difficulty in considering the following scenarios because of the limitation of used datasets: first, the discrimination between cloud and noncloud bright surfaces, and second, cloud shadows detection. In future work, it is meaningful to use datasets with more difficult cloud detection scenarios, and datasets with the mask of multiclass label such as cloud, thin cloud, and cloud shadow to extend the LWCDnet to multiclass classification.

References

[1] Y. Zhang, W. B. Rossow, A. A. Lacis, V. Oinas, and M. I. Mishchenko, "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, p. D19105, 2004.

[2] Y. Zhang, B. Guindon, and J. Cihlar, "An image transform to characterize and compensate for spatial variations in thin cloud contamination of Landsat images," *Remote Sens. Environ.*, vol. 82, nos. 2–3, pp. 173–187, 2002.

[3] Z. Li *et al.*, "Remote sensing of atmospheric particulate mass of dry $PM_{2.5}$ near the ground: Method validation using ground-based measurements," *Remote Sens. Environ.*, vol. 173, pp. 59–68, Feb. 2016.

[4] J. Xue and B. Su, "Significant remote sensing vegetation indices: A review of developments and applications," *J. Sensors*, vol. 2017, May 2017, Art. no. 1353691.

[5] V. Kothari, E. Liberis, and N. D. Lane, "The final frontier: Deep learning in space," in *Proc. 21st Int. Workshop Mobile Comput. Syst. Appl.*, 2020, pp. 45–49.

[6] *Phisat-1—Satellite-Missions—Eoportal Directory*. Accessed: Jun. 20, 2021. [Online]. Available: https://directory.eoportal.org/web/eoportal/satellite-missions/p/phisat-1

[7] G. Giuffrida *et al.*, "CloudScout: A deep neural network for on-board cloud detection on hyperspectral images," *Remote Sens.*, vol. 12, no. 14, p. 2205, Jul. 2020.

[8] Z. Zhu and C. E. Woodcock, "Object-based cloud and cloud shadow detection in Landsat imagery," *Remote Sens. Environ.*, vol. 118, pp. 83–94, Mar. 2012.

[9] N. Ghasemian and M. Akhoondzadeh, "Introducing two random forest based methods for cloud detection in remote sensing images," *Adv. Space Res.*, vol. 62, no. 2, pp. 288–303, Jul. 2018.

[10] R. Rossi, R. Basili, F. Del Frate, M. Luciani, and F. Mesiano, "Techniques based on support vector machines for cloud detection on QuickBird satellite imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2011, pp. 515–518.

[11] Y. Wang *et al.*, "Lednet: A lightweight encoder-decoder network for real-time semantic segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 1860–1864.

[12] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[13] 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.

[14] *MODIS Web*. Accessed: Jun. 20, 2021. [Online]. Available: https://modis.gsfc.nasa.gov/data/dataprod/mod35.php

[15] 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.

[16] M. Shi, F. Xie, Y. Zi, and J. Yin, "Cloud detection of remote sensing images by deep learning," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2016, pp. 701–704.

[17] J. Yu, Y. Li, X. Zheng, Y. Zhong, and P. He, "An effective cloud detection method for Gaofen-5 images via deep learning," *Remote Sens.*, vol. 12, no. 13, p. 2106, Jul. 2020.

[18] S. Platnick *et al.*, "The MODIS cloud products: Algorithms and examples from Terra," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 2, pp. 459–473, Feb. 2003.

[19] S. Qiu, B. He, Z. Zhu, Z. Liao, and X. Quan, "Improving Fmask cloud and cloud shadow detection in mountainous area for Landsats 4–8 images," *Remote Sens. Environ.*, vol. 199, pp. 107–119, Sep. 2017.

[20] S. Choi, J. T. Kim, and J. Choo, "Cars can't fly up in the sky: Improving urban-scene segmentation via height-driven attention networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 9373–9383.

[21] J. Wei *et al.*, "Dynamic threshold cloud detection algorithms for MODIS and Landsat 8 data," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2016, pp. 566–569.

[22] R. R. Irish, J. L. Barker, S. N. Goward, and T. Arvidson, "Characterization of the Landsat-7 ETM+ automated cloud-cover assessment (ACCA) algorithm," *Photogramm. Eng. Remote Sens.*, vol. 72, no. 10, pp. 1179–1188, 2006.

[23] C. Li, J. Ma, P. Yang, and Z. Li, "Detection of cloud cover using dynamic thresholds and radiative transfer models from the polarization satellite image," *J. Quant. Spectrosc. Radiat. Transf.*, vols. 222–223, pp. 196–214, Jan. 2019.

[24] S. Ray, "A quick review of machine learning algorithms," in *Proc. Int. Conf. Mach. Learn., Big Data, Cloud Parallel Comput. (COMITCon)*, Feb. 2019, pp. 35–39.

[25] 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.

[26] Y. Oishi, H. Ishida, T. Y. Nakajima, R. Nakamura, and T. Matsunaga, "Preliminary verification for application of a support vector machine-based cloud detection method to GOSAT-2 CAI-2," *Atmos. Meas. Techn.*, vol. 11, no. 5, pp. 2863–2878, May 2018.

[27] C. Bulgin, J. Mittaz, O. Embury, S. Eastwood, and C. Merchant, "Bayesian cloud detection for 37 years of advanced very high resolution radiometer (AVHRR) global area coverage (GAC) data," *Remote Sens.*, vol. 10, no. 2, p. 97, Jan. 2018.

[28] Y. LeCun, Y. Bengio, and G. E. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, Dec. 2015.

[29] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson, "Deep learning in remote sensing applications: A meta-analysis and review," *ISPRS J. Photogramm. Remote Sens.*, vol. 152, pp. 166–177, Jun. 2019.

[30] 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.

[31] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

[32] 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.

[33] 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.

[34] J. Drönner *et al.*, "Fast cloud segmentation using convolutional neural networks," *Remote Sens.*, vol. 10, no. 11, p. 1782, Nov. 2018.

[35] 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.

[36] 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, Aug. 2019.

[37] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," 2016, *arXiv:1606.02147*.

[38] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[39] H. Li, P. Xiong, H. Fan, and J. Sun, "DFANet: Deep feature aggregation for real-time semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9522–9531.

[40] X. Yang, Y. Wu, J. Zhao, and F. Liu, "Dense dual-path network for real-time semantic segmentation," in *Proc. Asian Conf. Comput. Vis.*, 2020, pp. 553–570.

[41] C. Li, R. Cong, J. Hou, S. Zhang, Y. Qian, and S. Kwong, "Nested network with two-stream pyramid for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9156–9166, Nov. 2019.

[42] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2814–2821.

[43] R. Cong, J. Lei, H. Fu, M.-M. Cheng, W. Lin, and Q. Huang, "Review of visual saliency detection with comprehensive information," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 10, pp. 2941–2959, Oct. 2019.

[44] Q. Zhang *et al.*, "Dense attention fluid network for salient object detection in optical remote sensing images," *IEEE Trans. Image Process.*, vol. 30, pp. 1305–1317, 2021.

[45] P. Wang *et al.*, "Understanding convolution for semantic segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1451–1460.

[46] M. Lin, Q. Chen, and S. Yan, "Network in network," 2013, *arXiv:1312.4400*.

[47] 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.

[48] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[49] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.

[50] S. Zhenfeng *et al.*, "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.

[51] R. P. K. Poudel, S. Liwicki, and R. Cipolla, "Fast-SCNN: Fast semantic segmentation network," 2019, *arXiv:1902.04502*.

[52] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 405–420.

[53] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.

[54] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 116–131.

[55] Z. Li, H. Shen, H. Li, G. Xia, 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.

[56] 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.

Chen Luo received the B.Sc. degree from Xidian University, Xi'an, China, in 2011, and the M.Sc. degree from Hannover University, Hannover, Germany, in 2014. She is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China.

Her research interests include deep learning, remote sensing, and computer vision.

Shanshan Feng received the Ph.D. degree in computer science from Nanyang Technological University, Singapore, in 2017.

He is currently an Associate Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. His research interests include sequential data mining and social network analysis.

Xiaofei Yang received the B.S. degree from Suihua University, Suihua, China, in 2011, and the M.S. and Ph.D. degrees from the Harbin Institute of Technology, Harbin, China, in 2014 and 2019, respectively.

He is currently a Post-Doctoral Researcher with the Department of Computer and Information Science, University of Macau, Macau, China. His research interests include semi-supervised learning, deep learning, remote sensing, transfer learning and graph mining.

Yunming Ye received the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China, in 2004.

He is currently a Professor with the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. His research interests include data mining, text mining, and ensemble learning algorithms.

Xutao Li received the bachelor's degree from the Lanzhou University of Technology, Lanzhou, China, in 2007, and the master's and Ph.D. degrees in computer science from the Harbin Institute of Technology, Shenzhen, China, in 2009 and 2013, respectively.

He is currently an Associate Professor with the Department of Computer Science and Technology, Harbin Institute of Technology. His research interests include data mining, machine learning, graph mining, and social network analysis, especially tensor-based learning and mining algorithms.

Baoquan Zhang received the B.S. degree from the Harbin Institute of Technology, Weihai, China, in 2015, and the M.S. degree from the Harbin Institute of Technology, Shenzhen, China, in 2017, where he is currently pursuing the Ph.D. degree with the School of Computer Science and Technology.

His research interests include meta learning, few-shot learning, and machine learning.

Zhihao Chen received the B.S. degree from the Harbin Institute of Technology, Shenzhen, China, in 2020, where he is currently pursuing the M.Sc. degree with the Department of Computer Science and Technology.

His research interests include machine learning, remote sensing, and computer vision.

Yingling Quan received the B.Sc. degree from the University of Science and Technology of China, Hefei, China, in 2021. She is currently pursuing the M.Sc. degree with the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China.

Her research interests include machine learning, remote sensing, and computer vision.