Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

GCDB-UNet: A novel robust cloud detection approach for remote sensing images[☆]

Xian Li [a], Xiaofei Yang [a,1], Xutao Li [a,*], Shijian Lu [b], Yunming Ye [a], Yifang Ban [c]

[a] *Shenzhen Key Laboratory of Internet Information Collaboration in Harbin Institute of Technology (Shenzhen), Shenzhen, 518055, China*
[b] *School of Computer Science and Engineering in Nanyang Technological University, Singapore, 628798, Singapore*
[c] *Vice Chair of Department of Urban Planning and Environment at KTH Royal Institute of Technology in Stockholm, 10044, Sweden*

ARTICLE INFO

Article history:
Received 12 July 2021
Received in revised form 1 December 2021
Accepted 2 December 2021
Available online 14 December 2021

Keywords:
Convolution neural network(CNN)
Cloud detection
Attention

ABSTRACT

Cloud detection is a prerequisite in many remote sensing applications, and it has been tackled through different approaches from simple thresholding to complicated deep network training. On the other hand, existing approaches are susceptible to failures while handling thin clouds, largely because of their small sizes, sparse distributions, as well as high transparency and similarity to the non-cloud background regions. This paper presents global context dense block U-Net (GCDB-UNet), a robust cloud detection network that embeds global context dense block (GCDB) into the U-Net framework and is capable of detecting thin clouds effectively. GCDB consists of two feature extraction units for addressing the challenges in thin cloud detection, namely, a non-local self-attention unit that extracts sample correlation features by aggregating the sparsely distributed thin clouds and a squeeze excitation unit that extracts channel correlated features by differentiating their importance. In addition, a dense connection scheme is designed to exploit the multi-level fine-grained representations from the two types of extracted features and a recurrent refinement module is introduced for gradual enhancement of the predicted classification map. We also created a fully annotated cloud detection MODIS dataset that consists of 1192 training images, 80 validation images and 150 test images. Extensive experiments on Landsat8, SPARCS and MODIS datasets show that the proposed GCDB-UNet achieves superior cloud detection performance as compared with state-of-the-art methods. Our created MODIS cloud detection dataset is available at https://github.com/xiachangxue/MODIS-Dataset-for-Cloud-Detection.

1. Introduction

Remote sensing image analysis is an important and thriving research area, which can serve a lot of applications, such as environment monitoring, change detection, and satellite sensor calibration. However, the remote sensing images are often contaminated by ubiquitously distributed cloud. Zhang et al. [1] show that more than 66% of the earth's surface area is covered by cloud in general. The large area of cloud establishes a significant barrier to real applications. Hence, cloud detection becomes a prerequisite in many remote sensing analysis tasks, especially in numerical weather predictions [2,3], extreme weather [4–6] and terrestrial atmospheric dynamics [7]. However, the task is very challenging, especially for thin clouds, because of their high transparency and similarity to the no-cloud background, e.g., ice, snow or bright surface regions.

In the literature, a lot of approaches have been developed for cloud detection to address the task, including threshold-based approaches [8–11] and classification-based approaches [12–14]. The threshold-based approaches usually identify the cloud pixels and separate the cloud regions from the non-cloud regions by a proper threshold. For example, Zhu et al. [8] developed the function of mask (FMask), which can identify clouds and cloud shadows with a carefully-designed multichannel thresholding method. Irish et al. [9] investigated the characteristics of an automated cloud cover assessment (ACCA) algorithm on Landsat −7, which is a typical threshold based method for cloud detection. Xu et al. [15] proposed a cloud-ice discrimination algorithm in the Arctic based on a set of multi-band thresholds obtained with statistic analysis for NOAA-19 AVHRR. Thompson et al. [16] proposed a cloud screening method by using spectral unmixing to classify clouds. Due to the simplicity, the threshold based methods are widely adopted in real applications. However, such methods cannot tackle complex scenarios robustly, e.g., thin clouds or clouds with high-proximity backgrounds.

[☆] This work was partially supported by the Shenzhen Science and Technology Program, China under Grant JCYJ20210324120208022, and Grant JCYJ20200109113014456, Grant JCYJ20180507183823045.

[*] Corresponding author.
E-mail addresses: lixutao@hit.edu (X. Li), yifang@kth.se (Y. Ye).

[1] Xiaofei Yang is the co-first author with equal contributions.

https://doi.org/10.1016/j.knosys.2021.107890
0950-7051/© 2021 Elsevier B.V. All rights reserved.

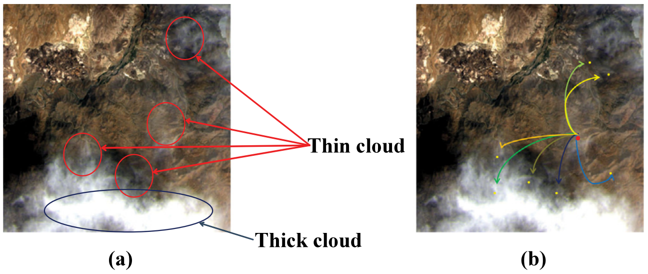

Fig. 1. An example to demonstrate the challenges for robust cloud detection. (a) thin cloud samples are scarce and dispersedly distributed; moreover, they are transparent and difficult to identify; (b) an intuition of our solution to characterize the challenges.

Another line of studies is based on machine learning techniques. For example, Tian et al. [17] improved the probabilistic neural network classifiers cloud detection by exploiting the temporal context information, and developed a prediction update scheme based on Markov chains. Ishida et al. [18] built the discriminant analysis method for classifying the typical cloudy and clear sky, and then applied support vector machine (SVM) classifier to adjust according to the incorrect predictions. As a result, more promising boundary can be produced for cloud and non-cloud regions. However, these conventional machine learning approaches require good manual features to perform well. Recently, researchers start turning to deep learning techniques for cloud detection and propose many deep learning-based methods for cloud detection [19–24]. For instance, Mateo-García et al. [25] investigated the performance of convolutional neural networks (CNNs) with different configurations for cloud detection, and the results demonstrate that CNNs are more promising than conventional machine learning approaches. Xie et al. [12] put forward a two-branch CNN model, which detects cloud with super-pixels based classification. Shao et al. [14] developed a multi-scale CNN, which extracts multi-level features for cloud identification.

Despite of the great number of existing methods, they are not robust enough because they do not carefully take the thin cloud identification into account. Thin cloud detection is very challenging, because of two reasons. **Challenge 1:** thin cloud samples (pixels) are relatively scarce and dispersedly distributed compared to thick clouds. Taking Fig. 1(a) as an example, we can see that the number of thin cloud samples (pixels) are usually smaller than that of thick cloud samples. Moreover, it can be seen that the thick cloud samples group into clusters while the thin cloud samples dispersedly distribute in an image. **Challenge 2:** as shown in Fig. 1(a), the thin cloud is often transparent. As a result, the visual features to distinguish it from background are very weak and we thus need to carefully leverage the multi-channel information of satellite images.

In this paper, we propose a novel robust cloud detection approach, called global context dense block U-Net (GCDB-UNet), which can tackle the two challenges. Specifically, we develop a novel neural network unit, called the global context dense block (GCDB) and embed it in the U-Net framework. Our intuition to develop the GCDB unit is to characterize the two challenges by introducing some special components and mechanisms. To address the challenge 1, we leverage the non-local self-attention mechanism to exploit the global contexts. In the mechanism, each pixel will consider their feature proximity to other non-local pixels when performing feature transformation. As a result, the dispersedly distributed thin cloud samples (pixels) can help each other (as shown in Fig. 1(b)), which can remedy the sample scarce and disperse distribution issue. Conventional convolution unit cannot tackle this issue due to its local operation property. To address challenge 2, we adopt the squeeze excitation technique to extract channel features, which can automatically exploit and leverage the correlations between multi-channels of satellite images. Finally, the features extracted from the two parts are fed into a specially designed local feature fusion (LFF) based dense connection block, which aims to further exploit multi-level fine-grained features for cloud detection. Moreover, a recurrent refinement module is appended in the U-Net to gradually enhance the output classification map. Extensive experiments on Landsat8, SPARCS and MODIS cloud detection data sets have been conducted, and the results show that the proposed GCDB-UNet outperforms state-of-the-art methods. The main contributions of the paper are summarized as follows.

- First, we propose a global context dense block as basic unit to construct the U-Net, which nicely addresses the two challenges for robust cloud detection.
- Second, to enhance the output classification map of U-Net, we append a recurrent gated block and gradually improve cloud detection results.
- Third, we have manually labeled and published a new cloud detection MODIS dataset, which contains of 1192 images in training set, 80 images in verification set and 150 images in test set. Our dataset includes the complex cloud detection scenes, such as Polar region, snow/ice region and bright surface region. For each image, it includes ten channels ranging from short wave to long wave, selected from 36 original channels.
- Fourth, extensive experimental results on Landsat8, SPARCS and MODIS have shown that the proposed method outperforms state-of-the-art techniques.

The remainder of this paper is organized as follows. In Section 2, existing cloud detection methods are reviewed. In Section 3, the proposed method is introduced. The MODIS dataset contribution is presented in Section 4. Section 5 presents the experiments. Finally, we conclude the paper in Section 6.

2. Related work

2.1. Threshold-based methods

Threshold-based approaches detect clouds by determining some proper thresholds for given channels. Due to the simplicity, many algorithms and systems have been developed. For example, an automated cloud cover assessment (ACCA) algorithm is a typical threshold based method. Irish et al. [9] studied and compared its performance on Landsat 7 Enhanced Thematic Mapper Plus (ETM+). Gao et al. [26] found that the near-IR channel (1.375um) on Terra Spacecraft, which was equipped to analyzing high-altitude clouds in the low/mid-latitude regions, delivered very promising performance on the identification of high clouds in polar regions during the daytime. Ackerman et al. [27] implemented the MODIS cloud masking algorithm, which is able to run on a limited processing and storage computer. Also, the result can be delivered in near real time. The main idea of the algorithm is to find proper thresholds for different types of surfaces, e.g., land, ocean, snow/ices and deserts. Huang et al. [28] presented an automated flagging method to separate clouds from clear view surfaces in a spectral-temperature space and predict cloud shadows according to its location, cloud height and solar illumination geometry. Oreopoulos et al. [29] transferred the cloud detection algorithm developed for MODIS to be used on Landsat −7 images and assessed the performance. Zhu et al. [8] utilized the Fmask algorithm to identify the cloud regions by using the potential cloud pixels and the cloud probability mask together, and predict

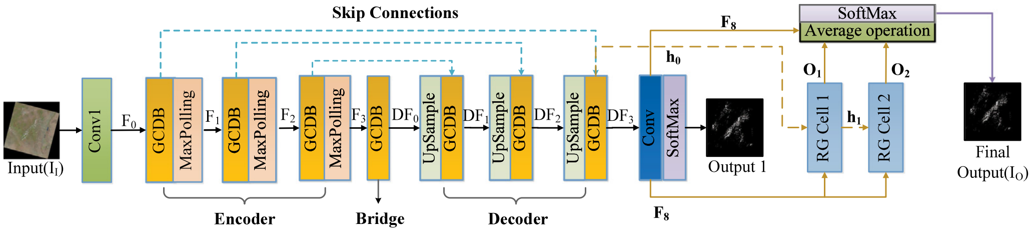

Fig. 2. The proposed GCDB-UNet architecture, with six GCDB layers embedded in U-Net for feature maps extraction and classification map generation, and a recurrent gate block with two Recurrent Gate cells (RGCs) that acts a result refinement module.

the cloud shadows by applying the flood-fill transformation. Zhang et al. [30] extended the haze optimized transform (HOT) for the identification of clouds and cloud shadows. In this method, the HOT was utilized to identify the clouds and generate dense haze features to detect and characterize the shadows. However, the threshold based methods are often too simple to robustly tackle complex situations, e.g., the thin cloud or cloud with high proximity background, e.g., snow or ices [30–35].

2.2. Classification-based methods

In essence, cloud detection is a pixel classification problem. As state-of-the-art techniques, deep learning methods especially the convolutional neural network (CNN), have been developed to address the problem [36–39]. For example, Johnston et al. [37] developed a two-phase strategy to optimize the CNN configurations for cloud detection. Zhan et al. [40] proposed a fully convolutional neural network with a multi-level feature fusion strategy to distinguish clouds from snow. Jiang et al. [41] designed a multi-scale residual 3D convolutional neural network (MRCNN) to extract spatial–spectral correlation information for haze removal of remote sensing. Yan et al. [42] proposed an end-to-end network, namely multilevel feature fused segmentation network (MFFSNet), to fuse different level of semantic features for cloud and cloud shadow detection. Recently, some semantic segmentation methods, such as fully convolutional neural network (FCN) [43], Unet [44], Deeplab [45], etc., have also been developed and utilized for clouds and cloud shadows detection. For instance, Mohajerani et al. [13] utilized a FCN to determine the cloud regions in Landsat8, where a complicated thresholding strategy was introduced as a pre-processing step to exclude snow and ice first. Jacob et al. [46] proposed a deep learning model, namely remote sensing network (RS-Net) to identify cloud regions based on U-Net architecture [44]. Mohajerani et al. [47] proposed a new cloud detection network (Cloud-Net). Each block of Cloud-Net is specifically designed to capture the local and global information with a hybrid of concatenation, copy and addition layers. Sadjadi et al. [48] introduced the moving singular value decomposition (MSVD) method to exploit the temporal information for cloud detection. Xie et al. [12] designed a multi-level deep learning framework for cloud detection. The framework firstly used an improved simple linear iterative clustering (SLIC) method to segment the images into super-pixels and obtain accurate cloud boundaries, and then built a two-branch CNN to extract the cloud features from the super-pixels and detect the clouds. Shao et al. [14] proposed a multiscale features-convolutional neural network (MF-CNN), by stacking visible, near-infrared, short-wave, and thermal infrared channels, upon which multi-scale features are extracted to identify cloud regions. Guo et al. [49] utilized “U"-architecture and designed a novel end-to-end CNN-based method called CD-netV2. In CDnetV2, an adaptive feature fusing model is devised to extract much more useful information, which comprises three components: a channel attention fusion model, a spatial attention fusion model, and a channel attention refinement model. Zhang et al. [50] applied the wavelet transform in cloud detection and proposed a deep encoder–decoder network to learn the multi-scale global features. Li et al. [51] proposed a novel deep convolution neural network, namely spatial folding–unfolding remote sensing network (SFRS-Net) adopting an encoder–decoder structure. The authors inserted the folding and unfolding operations instead of the unsample and max-pooling operations. Liu et al. [52] employed the deformable convolution blocks to capture saliency spatial context information, and presented an encoder–decoder network called DCNet. The above deep learning techniques, albeit developed for cloud detection, do not carefully tackle the two aforementioned challenges for thin cloud identification. Hence, they are not robust enough.

In this paper, we aim to propose a novel deep learning cloud detection approach, where the network structure can carefully characterize the two challenges for thin cloud identification.

3. Global context dense block U-net for cloud detection

In this section, we introduce the proposed method. First, we present the overall network architecture of our approach (as shown in Fig. 2), which essentially a U-Net [44] based segmentation model. Then, the developed Global Context Dense Block is introduced, which is utilized to build our detection model.

3.1. Overall network architecture

The overall network architecture of our GCDB-UNet is depicted in Fig. 2. We can see that our GCDB-UNet consists of two key components, namely a U-Net component with encoder, bridge and decoder parts and a recurrent gate refinement module. Next, we elaborate the two key components respectively. In the U-Net component, given a multi-channel remote sensing image I_I (with the size $m \times m$), a conventional convolution layer is applied to extract shallow feature maps F_0, which can be denoted as:

$$F_0 = H_{conv}(I_I), \tag{1}$$

where $H_{conv}(\cdot)$ denotes the first convolution operation, whose size is 3×3. Then, F_0 is input to the encoder part of U-Net. It is noted that the computational complexity of this layer is $O(3m^2n)$, and n is the size of this convolution filters. The encoder path contains three $E_l(l \in 1, 2, 3)$ blocks, each of which includes a GCDB operation and a 2×2 Max Pooling. Thus, the feature map F_l of encoder path can be obtained by

$$F_l = Pool(H_{GCDB}(F_{l-1})), \tag{2}$$

where $Pool(\cdot)$ is the Max Pooling, and $H_{GCDB}(\cdot)$ denotes the GCDB operation. Here we first focus on the overall architecture, and the detailed structure inside the GCDB will be discussed in Section 3.2.

After the encoder operation, we extract new features with the bridge path. Then, the output DF_0 can be written as

$$DF_0 = H_{GCDB}(F_3). \tag{3}$$

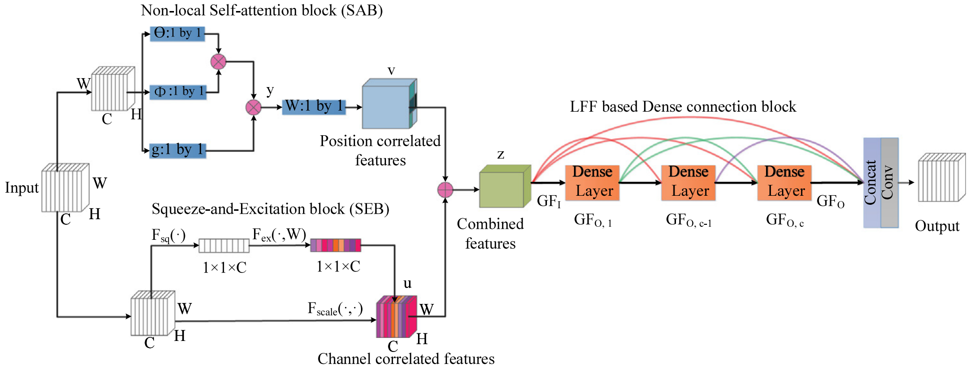

Fig. 3. The proposed global context dense block, which consists of non-local self-attention block(SAB), SE block(SEB) and a LFF based Dense connection block.

The symmetric decoder path also contains three $D_l(l \in 1, 2, 3)$ blocks, each of which includes a upsampling operation via the learnable bilinear interpolation, a skip connection with the corresponding feature map from the encoder path, and a GCDB transformation. The feature map DF_l for the lth block can be calculated by

$$DF_l = H_{GCDB}[(DeConv(DF_{l-1}))\bigoplus H_{GCDB}(F_{3-l})], \tag{4}$$

where F_{3-l} is the feature map transferred from the encoder path, whose size is the same with $DeConv(DF_{l-1})$. $\bigoplus$ denotes the concatenation operation. $DeConv(\cdot)$ indicates the learnable bilinear interpolation operation. The final output of decoder will be fed into a conventional convolution layer, followed by a softmax to produce the output classification map. Here cross entropy loss is adopted.

Next, we introduce our second key component, namely the recurrent refinement module. In the module, we aim to fine tune the detection results by a gradually approaching strategy. The strategy is enforced by a specially designed recurrent gate cell (RGC). The RGC is a Gate Recurrent Unit (GRU) with two modifications: (i) to maintain the spatial properties in feature maps, we replace the dot product in GRU with a convolution operator; (ii) the reset gate is always set as 1 and it forgets the status of previous layer merely by the update gate. Then, we append two RGCs to produce refined results from the feature map DF_3:

$$\begin{aligned} I_{O_1} &= H_{RGC}(DF_3), \\ I_{O_2} &= H_{RGC}(H_{RGC}(DF_3)). \end{aligned} \tag{5}$$

where I_{O_1} and I_{O_2} are the gradually approaching results, guided by their own cross entropy losses, and $H_{RGC}(\cdot)$ denotes the RGC operation. The final classification map will be an average of I_{O_1}, I_{O_2} and F_8.

3.2. Global context dense block

As shown in Fig. 3, the proposed GCDB is composed of three key ingredients, which are a non-local self attention block, a squeeze-extraction block and a local feature fusion (LFF) based dense connection block. Next, we elaborate the three ingredients and discuss how they are designed to characterize the two challenges of thin cloud detection.

Non-local Self-Attention Block. To detect the thin clouds, one challenge we face is the sample scarcity and disperse distribution issue. In remote sensing images, the number of thin clouds is relatively small compared to that of thick clouds and background. Moreover, the thin cloud samples often distribute dispersedly. To tackle the challenge, we leverage the non-local self-attention mechanism [53] to exploit correlated sample features. The core idea is aggregating the samples at other positions in terms of their proximity to form a better feature for each pixel. Specifically, let $X = \{x_i\}_{i=1}^{N_p}$ denote all pixel samples in an remote sensing image with height H, width W and d channels, $N_p = H \cdot W$, and each x_i is a d-dimensional vector. Then, the output of ith position(sample) by the non-local self-attention block is computed as:

$$y = W_y \sum_{j=1}^{N_p} \frac{f(x_i, x_j)}{\zeta(x)}(W_v, x_j), \tag{6}$$

where $f(x_i, x_j)$ is the proximity relationship between position i and j, and $\zeta(x)$ is a normalization factor, W_v and W_y denote the linear transform matrices to be learned. In the implementation, we use a 1 × 1 convolution for both W_v and W_y. Following [53], we leverage a Gaussian kernel to calculate the proximity between the positions i and j, and thus formally $f(x_i, x_j)$ and $\zeta(x)$ are defined as follows:

$$f(x_i, x_j) = e^{x_i^T x_j}, \quad \zeta(x) = \Sigma_{\forall j} f(x_i, x_j) \tag{7}$$

Here $x_i^T x_j$ denotes the dot-product similarity. In fact, the non-local block captures long-range dependence and aggregates query-specific global context features by weighting global context features of all locations, such as spatial, temporal and Spatio-temporal. Since attention maps are calculated for each query position, the computational complexity of the non-local block are quadratic with the number of positions N_p. We note that the computational complexity of this block is $O(m^2n + m^2n + m^2n)$, and n is the size of convolution filters.

Squeeze Excitation Block. Another challenge of thin clouds detection is the weak feature and channel information fusion issue. Here we employ the squeeze-excitation [54] block to characterize channel correlation and fuse multiple channel features, in which the size of convolution layers is 1 × 1. By doing so, the channel correlated features are extracted. As the features for thin cloud are too weak, we overcome the shortcoming by a combination of the two types of features, namely sample correlated features and channel correlated features. Specifically, we adopt the following scheme for feature strengthening:

$$z = u + v, \tag{8}$$

where z is the combined features, v denotes the position correlated features by non-local self-attention block, and u denotes channel correlated features by squeeze excitation block. We note that the computational complexity of this block is $O(m^2n)$, and n is the size of convolution filters.

LFF based Dense Connection Block. Combining sample correlated and channel correlated features with a simple addition operator is not strong enough for thin cloud detection. Previous studies [12,14] have proven that extracting multi-level features is very effective for cloud detection. Hence, we design a novel LFF based dense connection block to extract multi-level fine-grained features from the combined ones. In the block, dense connection scheme can naturally model the multi-level features thanks to its dense short-cuts between layers. Each dense layer consists of two convolution operations followed by BatchNorm [55] and ReLU [56]. Moreover, as cloud detection is pixel-wise classification, the multi-level features should be fine-grained. Inspired by RDN [57], we introduce a 1 × 1 convolution layer into dense connection block. We name the operation as local feature fusion (LFF) and its output is computed as:

$$GF_O = H_{LFF}([z, GF_{O,1}, \dots, GF_{O,c}]), \tag{9}$$

where the $H_{LFF}(\cdot)$ denotes the 1 × 1 convolution layer.

In the LFF based dense connection block, we also employ the contiguous memory mechanism, which is similar to [57]. Specifically, the state of preceding layer is transferred to the current layer. Let GF_I and GF_O be the input and output of the GCDB respectively, and both of them have G_0 feature-maps. The output of c_{th} convolution layer of current GCDB can be written as

$$GF_{O,c} = \sigma(W_{d,c}[z, GF_{O,1}, \dots, GF_{O,c-1}]), \tag{10}$$

where σ is the ReLU [56] activation function. $W_{d,c}$ denotes the weights of the c_{th} convolution layer, and its size is 1 × 1. Here, $GF_{O,c}$ is assumed to contain G features. $[z, GF_{O,1}, \dots, GF_{O,c-1}]$ refers to the concatenation of the features produced in $1, \dots, (c-1)$ convolutional layers. Then, we obtain a result of $G_0+(c-1)\times G$ features. The outputs z and each layer in LFF have direct connections to all subsequent layers. Benefits from the structure of LFF-based dense connection block, the proposed GCDB-UNet can extract the local dense features while retaining the feed-forward characteristics.

Computational Complexity Analysis. Since GCDB U-Net consists of seven GCDB blocks and up-down pooling layers, the computational complexity of the proposed GCDB U-Net is mainly dominated by GCDB. GCDB consists of two feature extraction modules and one feature fusion module, namely a Non-Local Self-Attention Block for extraction the spatial attention features, a Squeeze Excitation Block for channel attention features extraction and an LFF-based Dense Connection Block for features fusion. In addition, the Non-Local Self-Attention module and the Squeeze Excitation module can be conducted in parallel. As a result, for a given input image $m \times m$, the per GCDB computational complexity is dominated by Non-local Self-Attention Block and LFF-based Dense Connection Block that require an overall $O(m^2n + m^2n + m^2n + km^2n)$, where n is the size of convolution filters and k denotes the kernel width. Table 1 lists the detailed computation complexity of all methods with input size $(1, 4, 192, 192)$. We can see that the proposed GCDB U-Net has a higher Flops of 14.87 GB, and a small number of parameters of 1.23MB. The main reason is that the number of kernels at each GCDB layer is 64, while the number of kernels in U-Net is $[64, 128, 256, 512, 1024]$. In addition, the kernel size of the Non-Local Self-Attention module is 1 and the Squeeze Excitation module adopts linear operations.

3.3. Implementation details

As shown in Fig. 2, the proposed GCDB-UNet consists of one 3 × 3 convolutional operation, six GCDBs and two RGCs. In this paper, the size of all convolutional layers in GCDBs is 3 × 3. And, the size of convolutional layers in non-local self-attention and last layer in local feature fusion of GCDBs is 1 × 1. To keep the size fixed, zero padding strategy is adopted in the proposed GCDB-UNet. Similar to [57], there are 64 filters in convolutional layers of local feature fusion.

Table 1
The computational complexity of all methods.

Methods	Flops (GB)	Parameters (MB)
FCN	14.37	18.64
U-Net	6.53	31.04
Deeplab	12.49	59.24
RS-Net	7.89	7.85
MF-CNN	17.64	17.41
CloudNet	0.07	34.78
SFRS-Net	19.91	88.10
DCNet	5.61	0.81
GCDB-UNet (ours)	14.87	1.23

4. MODIS cloud dataset construction

Moderate-Resolution Imaging Spectroradiometer (MODIS) is an important earth observation system in NASA. Our data set is composed of a subset of MODIS level 1B cloud product images in 2005. Specifically, we include the images of the first five days of each month in the year in our data set. Each image sample has 36 band channels. We remove the obviously abnormal samples by checking their synthesized false-color images, e.g., discarding the images taken at day and night boundary as shown in Fig. 4.

Next, we introduce how the bands are selected. To this end, we first perform some distinguishable analysis of different wavelength channels regarding the cloud and non-cloud pixels. According to the analysis, we find there are three groups. The first group is the short and medium wavelength channels. The first row in Fig. 5 shows the distinguish ability of three such wavelength channels. We can see that the cloud pixels are mainly distributed in the interval of high pixel values. On the contrary, non-cloud pixels are in the lower interval ones. This is because cloud usually has higher reflectance than other objects on the ground. The overlapping distribution between cloud and non-cloud is probably due to the bright background, such as snow and ice. The second group is long wavelength channels, which are shown in the second row in Fig. 5. In this case, we can see that the non-cloud pixels are mainly distributed in the high pixel value parts and the cloud pixels are in the lower pixel value interval. This is because long wavelength emissive channels generally reflect the temperature of objects, and cloud usually has a lower temperature. Anyway, the two distributions are still overlapping. The third groups refer to the invalid wavelength channels, where the cloud and non-cloud are totally inseparable. According to the analysis above, we select the ten most distinguishable channels from the 36 bands, which are bands 1, 3, 4, 18, 20, 23, 28, 29, 31 and 32.

The data set mainly contains images in four scenarios: ocean, land, land and ocean, and Polar glaciers, shown in Fig. 6. Different from the 38-Cloud Landsat8 dataset, the MODIS dataset has some special scenes, such as oceans, and Polar ice regions which are very difficult to identify clouds from the background. The reasons are two-fold. First, ice and clouds both have the similar low-temperature characteristics. Second, the reflectivity characteristics of the ice resembles those of clouds. Moreover, thin clouds often appear in such scenes. Hence, the dataset poses a great challenge to existing methods.

5. Experimental results

In this section, we conduct experiments on three remote sensing datasets, namely Landsat8, SPARCS and MODIS to evaluate the performance of the proposed methods. As for a comparison, the state-of-the-art cloud detection approaches are utilized as baselines.

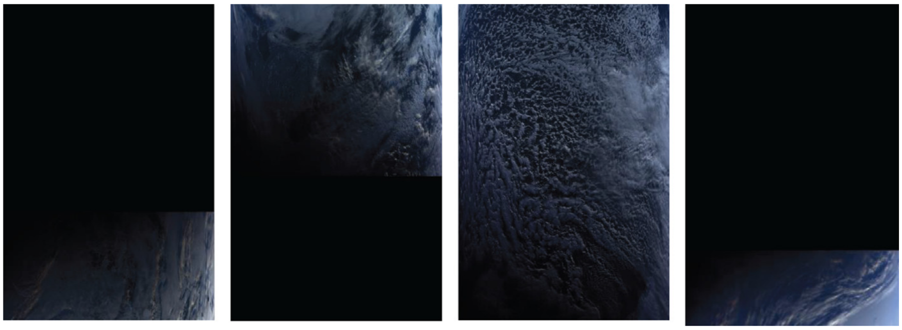

Fig. 4. The samples of the removed false color images, which are basically the images captured at day and night boundary.

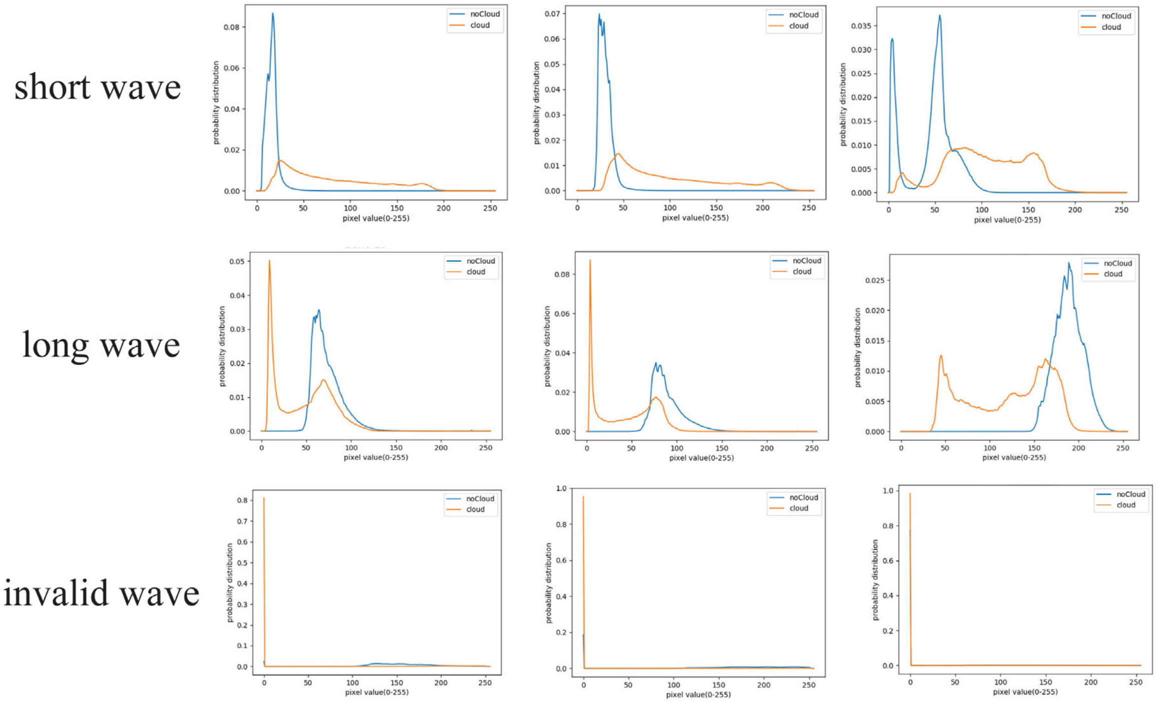

Fig. 5. The values of cloud pixels and non-cloud pixels in different channels. The first row represents the short wave and medium wave channels, the second row is the long wave channels, and the third row represents the invalid wave channels.

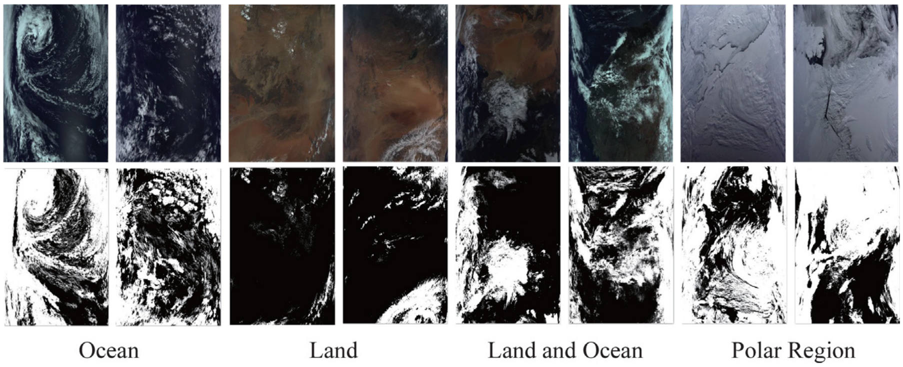

Fig. 6. Cloud images with different underlying surfaces in MODIS dataset. The images of the first row are the false-color images, and the corresponding images in the second row are the masks.

5.1. Experimental setup

5.1.1. Landsat8 dataset

Recently, Mohajerani et al. [47] have released a Landsat8 remote sensing dataset, for cloud detection, which is named as 38-Cloud dataset. The dataset has 18 Landsat8 images for training and 20 images for test. Each Landsat8 image is cropped into 384 × 384 patches. As a result, 8400 training samples and 9201 test samples are obtained.

Table 2
Results comparison on Landsat8 dataset. (in %).

Methods	Jaccard	Precision	Recall	MIoU	Overall	F1-score
FCN [43]	83.2	95.3	86.7	87.7	94.4	90.8
U-Net [44]	85.0	93.2	90.6	88.9	94.9	91.9
DeepLap [45]	84.7	92.5	90.8	88.6	94.7	91.6
RS-Net [46]	85.8	93.4	91.4	89.5	95.2	92.4
MF-CNN [14]	87.1	94.1	92.1	90.5	95.6	93.1
CloudNet [47]	87.3	**96.7**	89.9	90.7	95.8	93.2
SFRS-Net [51]	85.6	94.3	90.3	89.4	95.1	92.26
DCNet [52]	86.1	96.1	89.1	89.7	95.3	92.47
GCDB-UNet (ours)	**89.0**	95.3	**93.0**	**91.9**	**96.3**	**94.2**

5.1.2. SPARCS dataset

The SPARCS dataset includes 80 Landsat8 images with ten channels and a resolution of 1000 × 1000. We crop each image into 384 × 384 patches. As a result, 540 training samples and 180 test samples are obtained.

5.1.3. MODIS dataset

Moderate-Resolution Imaging Spectroradiometer(MODIS) is an important earth observation system in NASA. In our data set, we have 1422 MODIS images, which are separated into a training set with 1192 samples, a validation set with 80 samples and a test set with 150 samples. Similar to Landsat8 dataset, we crop them into 512 × 512 overlapping patches and obtain 21330 images, where each image has 10 channels. After cropping, the training, validation and test sets include 17880, 1200 and 2250 images, respectively.

5.1.4. Evaluation metrics

The predicted mask has two classes, i.e., cloud or non-cloud. To measure the performance, we adopt the widely utilized metrics, including Jaccard Index, Precision, Recall, MIoU, F1-score and Overall Accuracy. The six metrics are defined as follows:

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

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

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

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

$$OverallAccuracy = \frac{TP + TN}{TP + TN + FN + FP}, \tag{15}$$

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

where TP, TN, FP, and FN are the total number of true positive, true negative, false positive, and false negative pixels, respectively. We note that the higher the six metrics are, the better the performance is. The MIoU is the Mean Intersection over Union, which is a widely used metric in semantic segmentation. Different from the Jaccard, the MIoU considers both the accuracies of cloud pixels and non-cloud pixels, namely there are two classes. Here $k = 2$, and p_{ij} denotes the number of pixels that are from class i but predicted as class j. The p_{ji} is the number of pixels that are from class j but predicted as class i, and p_{ii} denotes the number of pixels that are from class i and predicted as class i.

5.1.5. Parameter settings

Following setting of [14], the size of input Landsat8 patches is 384 × 384 extracted from the raw remote sensing images. In order to fed into the GCDB-UNet, each input patch will be resized to 192 × 192 (but the input patch of MODIS is 256 × 256). We implement the proposed GCDB-UNet on the Pytorch framework and set 10^{-3} as the learning rate up for all layers. It is updated with SGD optimizer. According to [58], we employ an adaptive learning rate policy where the initial learning rate is multiplied by $1-(iter/maxiter)^{0.9}$ after each iteration. Training a GCDB-UNet roughly takes 1 day with two 1080Ti GPU for 100 epoches.

5.1.6. Baseline approaches

In this paper, we compare our approach with six state-of-the-art methods, which are RS-Net [46], MF-CNN [14], CloudNet [47], FCN [43], U-Net [44] and DeepLab [45], respectively. We note that for a fare comparison all the baseline methods are re-trained with the same training set as our approach, where the hyper-parameters are tuned based on the same validation set.

5.2. Results on Landsat8 dataset

Table 2 summarizes the results of different methods on Landsat8 dataset. We can see that the proposed GCDB-UNet performs the best in terms of all the metrics, except on Precision metric. This is attributed to our specially designed GCDB unit, which can nicely characterize the two key challenges for cloud detection. Though CloudNet delivers better precision than our GCDB-UNet, we can see that its recall is inferior. The fact indicates that CloudNet tends to produce conservative cloud regions.

To further understand the results, we make visual comparisons on three thin cloud detection examples in Fig. 7. We can see from the first row example that the proposed GCDB-UNet can nicely find all thin cloud regions, whereas the baseline approaches FCN, U-Net, DeepLab, RS-Net, MF-CNN, SFRS-Net, DC-Net, and CloudNet miss many regions with thin cloud. In the second row, we observe that FCN, RS-Net, MF-CNN and CloudNet overestimate thin cloud regions, marked as red parts, and U-Net, DeepLab and GCDB-UNet deliver promising results. Again, in the third row, it can be seen that the six baseline approaches fail to detect some thin cloud regions, while our method delivers the best performance. All the observations validate the robustness of the proposed GCDB-UNet approaches.

5.3. Results on SPARCS dataset

Table 3 shows the results of all the methods on SPARCS. We can see that the proposed GCDB-UNet delivers the best performance for all the metrics, except the precision and recall. Again, we find that CloudNet yields better precision but much worse recall than our GCDB-UNet, which is because CloudNet is more conservative for cloud detection. By contrast, Fmask produces better recall but much worse precision than our approach, which is more aggressive. In terms of F1-score, which is a harmonic

Table 3
Results comparison on SPARCS dataset. (in %).

Methods	Jaccard	Precision	Recall	MIoU	Overall	F1-score
Fmask [8]	44.5	44.6	**99.4**	52.6	70.1	61.6
FCN [43]	79.8	88.1	89.5	86.4	94.5	88.8
U-Net [44]	72.1	81.4	86.4	81.0	91.9	83.8
DeepLap [45]	74.1	77.3	94.7	81.9	92.0	85.1
RS-Net [46]	80.2	92.3	85.9	86.8	94.8	88.9
MF-CNN [14]	80.3	88.2	89.9	86.8	94.7	89.0
CloudNet [47]	82.4	**94.0**	87.0	88.4	95.5	90.4
SFRS-Net [51]	77.1	88.5	85.7	85.2	94.5	87.08
DCNet [52]	73.1	86.0	83.0	82.6	93.4	84.47
GCDB-UNet (ours)	**83.7**	91.0	91.2	**89.1**	**95.7**	**91.1**

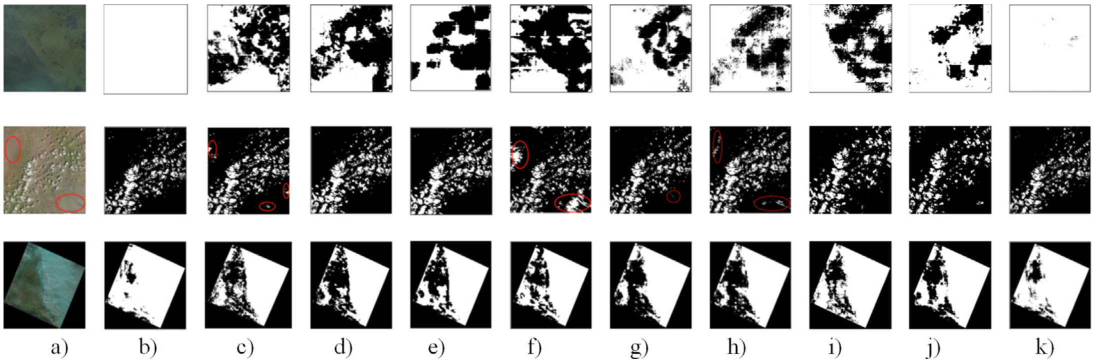

Fig. 7. Visual comparisons of different cloud detection methods in the partial scene of three examples from Landsat dataset. (a) denotes the input image; (b) the ground-truth; (c) Result of FCN; (d) Result of U-Net; (e) Result of DeepLab; (f) Result of CloudNet; (g) Result of RS-Net; (h) Result of MF-CNN; (i) Result of SFRS-Net; (j) Result of DCNet; k) Result of GCDB-UNet (ours). The red part denotes the mistakes of cloud pixel predicted by the methods.

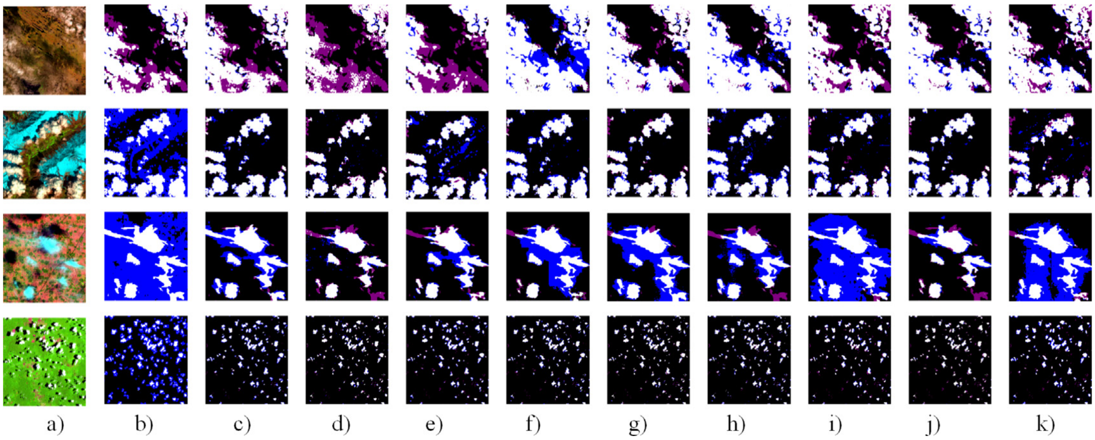

Fig. 8. Visual comparisons of different cloud detection methods in the partial scene of three examples from SPARCS dataset. (a) denotes the input image; (b) Result of Fmask; (c) Result of FCN; (d) Result of U-Net; (e) Result of DeepLab; (f) Result of CloudNet; (g) Result of RS-Net; (h) Result of MF-CNN; (i) Result of SFRS-Net; (j) Result of DCNet; (k) Result of GCDB-UNet (ours).

mean of precision and recall, our method is better than the two methods.

To make a visual comparison, we depict in Fig. 8 the cloud detection results of all the methods regarding three example images from SPARCS dataset. In the figures, the blue represents the error recognition pixels, and purple represents the missed recognition pixels. From the first row, we find that Fmask, FCN, U-Net, SFRS-Net, DCNet, and DeepLab tend to miss the thin cloud regions. From the second and third rows, we observe that Fmask, FCN, U-Net, DeepLab, CloudNet, RS-Net and MF-CNN are likely to overestimate the thick cloud regions. Compared to the baseline methods, the proposed GCDB-UNet is more robust and produces the most accurate predictions. All the observations demonstrate the superiority and robustness of our GCDB-UNet.

5.4. Results on MODIS dataset

Table 4 reports the results comparison on MODIS dataset. Again, the proposed GCDN-UNet outperforms the state-of-the-art methods. Similarly, we depict the detection results on three examples from MODIS for a visual comparison in Fig. 9. We find that the proposed GCDB-UNet can accurately identify the cloud from ice or snow background; while the six baseline approaches either miss some cloud regions or overestimate the regions. The results further validate the superiority and robustness of the proposed method.

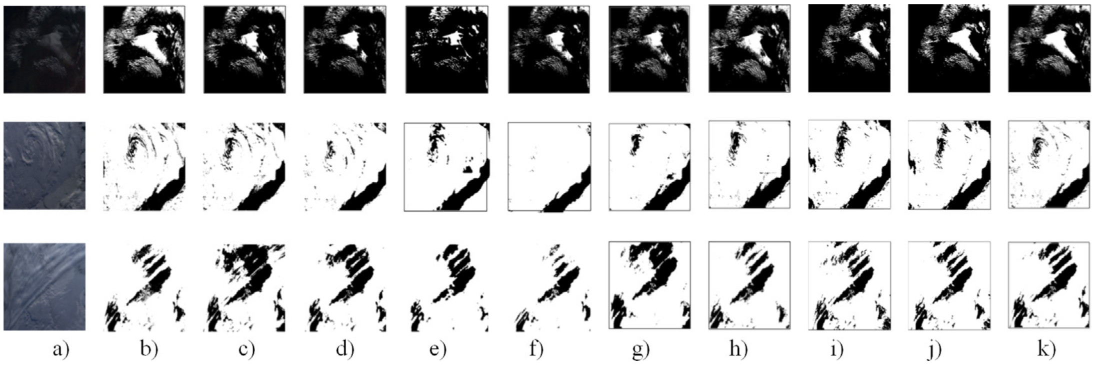

Fig. 9. Visual comparisons of different cloud detection methods in the partial scene of three examples from MODIS. (a) denotes the input image; (b) the ground-truth; (c) Result of FCN; (d) Result of U-Net; (e) Result of DeepLab; (f) Result of CloudNet; (g) Result of RS-Net; (h) Result of MF-CNN; (i) Result of SFRS-Net; (j) Result of DCNet; (k) Result of GCDB-UNet (ours).

Table 4
Results comparison on MODIS dataset. (in %).

Methods	Jaccard	Precision	Recall	MIoU	Overall	F1-score
FCN [43]	89.4	97.3	91.6	86.3	93.1	94.4
U-Net [44]	89.7	97.7	91.5	86.8	93.2	94.6
DeepLap [45]	86.9	94.1	91.9	82.8	91.1	93.0
RS-Net [46]	89.7	96.7	92.5	86.6	93.3	94.6
MF-CNN [14]	90.0	96.4	93.1	86.9	93.4	94.7
CloudNet [47]	90.8	**97.8**	92.7	88.1	94.0	95.2
SFRS-Net [51]	89.3	96.3	92.4	86.0	92.9	94.31
DCNet [52]	90.2	96.0	93.6	87.0	93.5	94.79
GCDB-UNet (ours)	**91.7**	97.6	**93.9**	**89.5**	**94.6**	**95.7**

Table 5
Ablation study of non-local self-attention (NL), squeeze-excitation (SE), and recurrent gated block (RGB) on Landsat8 data set.

	Different combinations of NL, SE, and RGB							
NL	×	√	×	×	√	√	×	√
SE	×	×	√	×	√	×	√	√
RGB	×	×	×	√	×	√	√	√
Jaccard	85.0	86.3	88.1	85.6	88.4	87.3	88.3	**88.9**
Precision	93.2	92.6	95.2	93.8	93.9	95.5	94.4	**95.6**
Recall	90.6	92.6	92.2	90.8	**93.8**	91.1	93.1	93.7
MIoU	88.9	89.8	91.3	89.2	91.4	90.8	91.3	**91.8**
Overall	94.9	95.3	96.0	95.0	96.1	96.0	96.0	**96.3**
F1-score	91.9	92.7	93.7	92.2	93.8	93.2	93.7	**94.1**

5.5. Ablation investigation

The proposed GCDB-UNet contains three key blocks, which are non-local self-attention block, squeeze-excitation block, and recurrent gated block. Here, we would to study the effectiveness of the blocks by conducting different ablation investigations. Table 5 shows the result.

We can see that the best performance is obtained generally when all the three blocks are utilized; the worst performance is delivered when all the three blocks are removed; and ablating two blocks yields worse performance than ablating merely one. The observations implies that the developed three blocks are effective, which all contribute to the final performance. According to the improvement comparison by adding respectively the three blocks, we find that their impacts on the final performance follows the order: $SE > NL > RGB$. In other words, channel correlated features are more important than the sample correlated ones, and the gradual refinement module plays the least important role.

In addition to the three key components, LFF based dense connection block is also an important part in our GCDB-UNet. Hence, it is interesting to investigate its effectiveness by ablating this part. We refer to the ablation model as GCDB-UNet without LFF, denoted as GCDB-UNet w.o. LFF. Table 6 shows the result. We can see that a removal of the LFF leads a performance degeneration, which implies the effectiveness of the LFF based dense connection part. The reason is that the module helps to exploit the multi-level fine grained feature for cloud identification.

Table 6
Ablation study of LFF based dense connection block on Landsat8 data set.

Metrics	GCDB-UNet w.o. LFF	GCDB-UNet
Jaccard	88.1	**89.0** (↑ 0.9)
Precision	93.5	**95.3** (↑ 1.8)
Recall	**93.9**	93.0
MIoU	91.2	**91.9** (↑ 0.7)
Overall	95.9	**96.3** (↑ 0.4)
F1-score	93.7	**94.2** (↑ 0.5)

5.6. Transfer test across datasets

In this part, we investigate the generalization ability of each method across datasets. To this end, we train all the methods on

Table 7
Comparison results of transfer (T) and non-transfer (N/T) test on SPARCS dataset.

Methods	Jaccard		Precision		Recall		MIoU		Overall		F1score	
	T	N/T	T	N/T	T	N/T	T	N/T	T	N/T	T	N/T
FCN	62.3	75.4	72.0	84.9	82.2	87.1	74.6	83.9	89.3	93.9	76.8	86.0
U-Net	50.9	74.5	58.1	89.8	80.3	81.4	65.4	83.6	83.3	94.0	67.4	85.4
Deeplab	61.6	74.9	69.8	86.3	84.1	85	73.9	83.7	88.8	93.9	76.3	85.6
RS-Net	57.6	76.3	66.7	84.3	80.8	88.9	71.1	84.5	87.2	94.1	73.1	86.5
MF-CNN	59.3	78.4	65.5	83.8	86.1	92.4	71.8	85.8	87.3	94.5	74.4	87.9
CloudNet	57.2	77.9	68	84.2	78.2	91.3	71	85.5	87.4	94.4	72.7	87.6
SFRS-Net	58.2	76.3	68.8	84.1	79.1	89.2	71.7	84.5	87.8	94.0	73.59	86.58
DCNet	61.7	71.2	75.2	76.4	77.5	91.4	74.6	80.7	89.6	92.0	76.33	83.23
GCDB-UNet	62	80.7	69.3	89.4	85.5	89.2	74.1	87.5	88.8	95.4	76.6	89.3

Table 8
Comparison results of transfer (T) and non-transfer (N/T) test on MODIS dataset.

Methods	Jaccard		Precision		Recall		MIoU		Overall		F1score	
	T	N/T	T	N/T	T	N/T	T	N/T	T	N/T	T	N/T
FCN	70.2	85.2	92.2	88.7	74.6	95.6	65.9	79.1	79.8	89.4	82.5	92.0
U-Net	43.8	86.6	95.9	94.7	44.6	91.0	46.4	82.5	63.5	91.0	60.9	92.8
Deeplab	43.1	85.7	91.7	92.3	44.9	92.4	45.1	80.9	62.3	90.2	60.3	92.3
RS-Net	53.8	87.0	95.4	95.4	55.2	90.8	53.5	83.3	69.7	91.4	69.9	93.0
MF-CNN	64.0	87.1	93.7	94.1	66.9	92.2	61.1	83.1	76.0	91.3	78.1	93.1
CloudNet	62.7	88.9	95.7	93.2	64.5	95.1	60.6	84.9	75.5	91.5	77.1	94.1
SFRS-Net	50.2	85.2	94.9	91.4	51.6	92.6	50.8	80.1	67.4	89.7	66.85	91.99
DCNet	48.5	85.2	95.2	94.1	49.7	90.0	49.6	81.0	66.3	90.0	65.31	92.01
GCDB-UNet	60.9	89.3	95.2	95.1	62.9	93.5	59.0	85.7	74.3	92.8	75.8	94.3

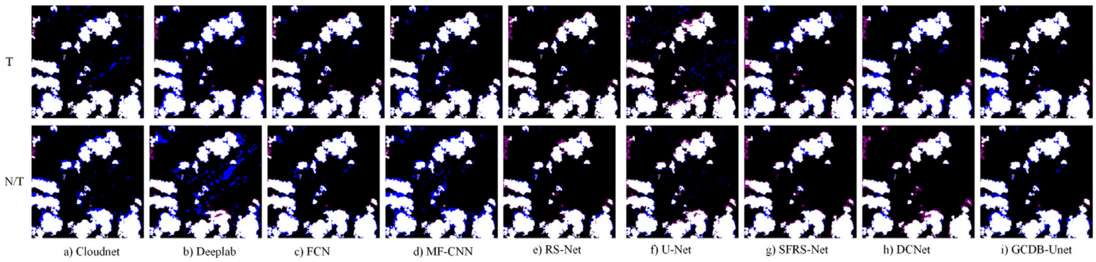

Fig. 10. Visual comparisons of transfer (T) and non-transfer (N/T) test in the partial scene of one example from SPARCS. The first row N/P denotes the results of No-Pretrained model, and the second row W/P is the results of With-Pretrained model.

Landsat8 datasets and then apply them on SPARCS and MODIS datasets for test. To make the input bands consistent, we correspond the four bands of Landsat8 with those in SPARCS and MODIS, which are bands 2, 3, 4, 5 and bands 3, 4, 1, 2, respectively. As for a comparison, we report the test results (mentioned as *T* for transfer test) as well as the results that models are directly trained and applied on SPARCS and MODIS (mentioned as N/T for non-transfer test).

Table 7 shows the results for SPARCS. We can see though the transfer test shows acceptable performance for all the methods, but there is still a large gap compared to directly training a model on SPARCS with sufficient samples. Fig. 10 depicts an example to compare the results in the two settings. In the figures, the blue represents the error recognition pixels, and purple represents the missed recognition pixels. We can see that in the example the results of two settings are very competitive to each other. This means for some samples the transfer setting is successful. However, the overall performance in Table 6 degenerates for transfer setting. The reason may be because the sample distribution in Landsat8 is much different from that in SPARCS. Table 8 shows the results on MODIS dataset. We find similar observations but big gaps between transfer and non-transfer test. This is because our MODIS is constructed in totally different scenes. Moreover, the four bands are roughly matched, because the band wavelengths are different between Landsat8 and MODIS. Hence, how to adapt a cloud detection model to a totally different satellite with none or few labeled samples is an interesting problem, which is worth studying in the future.

6. Conclusions

In this paper, we propose a novel robust cloud detection approach based on the U-Net architecture. Previous state-of-the-art methods are not robust because they fail to nicely tackle thin cloud detection. By analyzing the two challenges of thin cloud detection, we develop a novel global context dense block and build a U-Net upon the block. A recurrent gated block is appended to the U-Net for the detection map refinement. Extensive experiments have been conducted on three real-world remote sensing data sets, and the results show that the proposed GCDB-UNet outperforms the state-of-the-art cloud detection methods. In the future, we would like to study how to quickly adapt the GCDB-UNet to a new satellite by transfer learning techniques.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Y.-C. Zhang, W.B. Rossow, A.A. Lacis, V. Oinas, 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. 109 (2004) D19105.

[2] T. Wang, G. Yan, X. Mu, Z. Jiao, L. Chen, Q. Chu, Toward operational shortwave radiation modeling and retrieval over rugged terrain, Remote Sens. Environ. 205 (2018) 419–433.

[3] T. Wang, J. Shi, Y. Yu, L. Husi, B. Gao, W. Zhou, D. Ji, T. Zhao, C. Xiong, L. Chen, Cloudy-sky land surface longwave downward radiation (LWDR) estimation by integrating MODIS and AIRS/AMSU measurements, Remote Sens. Environ. 205 (2018) 100–111.

[4] N. Shastri, K.K. Pathak, New cloud detection index (CDI) for forecasting the extreme rain events, Adv. Remote Sens. 8 (1) (2019) 30–39.

[5] R. Biondi, P.-Y. Tournigand, M. Hammouti, Machine learning cloud top height detection based on GNSS radio occultations: a step ahead towards an operational use, in: EGU General Assembly Conference Abstracts, Remote Sensing, pp. EGU21–8789.

[6] W. Fang, Q. Xue, L. Shen, V.S. Sheng, Survey on the application of deep learning in extreme weather prediction, Atmosphere 12 (6) (2021) 661.

[7] K.M. Cooper, S.J. Goldstein, K.W. Sims, M.T. Murrell, Uranium-series chronology of gorda ridge volcanism: new evidence from the 1996 eruption, Earth Planet. Sci. Lett. 206 (3–4) (2003) 459–475.

[8] Z. Zhu, C.E. Woodcock, Object-based cloud and cloud shadow detection in landsat imagery, Remote Sens. Environ. 118 (none) (2012) 0–94.

[9] R. Irish, J. Barker, S. Goward, T. Arvidson, Characterization of the landsat-7 ETM+ automated cloud-cover assessment (ACCA) algorithm, Photogramm. Eng. Remote Sens. 72 (2006) 1179–1188.

[10] G. Vivone, P. Addesso, R. Conte, M. Longo, R. Restaino, A class of cloud detection algorithms based on a MAP-MRF approach in space and time, IEEE Trans. Geosci. Remote Sens. 52 (8) (2014) 5100–5115.

[11] X. Zhu, E.H. Helmer, An automatic method for screening clouds and cloud shadows in optical satellite image time series in cloudy regions, Remote Sens. Environ. 214 (2018) 135–153.

[12] F. Xie, M. Shi, Z. Shi, J. Yin, D. Zhao, Multilevel cloud detection in remote sensing images based on deep learning, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 10 (2017) 3631–3640.

[13] S. Mohajerani, T.A. Krammer, P. Saeedi, A cloud detection algorithm for remote sensing images using fully convolutional neural networks, in: 2018 IEEE 20th International Workshop on Multimedia Signal Processing, MMSP, 2018, pp. 1–5.

[14] Z. Shao, Y. Pan, C. Diao, J. Cai, Cloud detection in remote sensing images based on multiscale features-convolutional neural network, IEEE Trans. Geosci. Remote Sens. 57 (2019) 4062–4076.

[15] D. Xu, J. Su, Z. Liu, W. Wang, X. Tang, Application of cloud detection algorithm in the Arctic region based on AVHRR satellite data, in: 2011 International Conference on Remote Sensing, Environment and Transportation Engineering, 2011, pp. 1026–1030.

[16] D.R. Thompson, R.O. Green, D. Keymeulen, S.K. Lundeen, Y. Mouradi, D.C. Nunes, R. Castaño, S.A. Chien, Rapid spectral cloud screening onboard aircraft and spacecraft, IEEE Trans. Geosci. Remote Sens. 52 (11) (2014) 6779–6792.

[17] B. Tian, M.R. Azimi-Sadjadi, T.H. Vonder Haar, D. Reinke, Temporal updating scheme for probabilistic neural network with application to satellite cloud classification, IEEE Trans. Neural Netw. 11 (4) (2000) 903–920.

[18] H. Ishida, Y. Oishi, K. Morita, K. Moriwaki, T.Y. Nakajima, Development of a support vector machine based cloud detection method for MODIS with the adjustability to various conditions, Remote Sens. Environ. 205 (2018) 390–407.

[19] J. Wang, D. Yang, S. Chen, X. Zhu, S. Wu, M. Bogonovich, Z. Guo, Z. Zhu, J. Wu, Automatic cloud and cloud shadow detection in tropical areas for PlanetScope satellite images, Remote Sens. Environ. 264 (2021) 112604.

[20] K. Tarrio, X. Tang, J.G. Masek, M. Claverie, J. Ju, S. Qiu, Z. Zhu, C.E. Woodcock, Comparison of cloud detection algorithms for sentinel-2 imagery, Sci. Remote Sens. 2 (2020) 100010.

[21] Y. Li, W. Chen, Y. Zhang, C. Tao, R. Xiao, Y. Tan, Accurate cloud detection in high-resolution remote sensing imagery by weakly supervised deep learning, Remote Sens. Environ. 250 (2020) 112045.

[22] M. Segal-Rozenhaimer, A. Li, K. Das, V. Chirayath, Cloud detection algorithm for multi-modal satellite imagery using convolutional neural-networks (CNN), Remote Sens. Environ. 237 (2020) 111446.

[23] D. López-Puigdollers, G. Mateo-García, L. Gómez-Chova, Benchmarking deep learning models for cloud detection in landsat-8 and sentinel-2 images, Remote Sens. 13 (5) (2021) 992.

[24] M. Shao, Y. Zou, Multi-spectral cloud detection based on a multi-dimensional and multi-grained dense cascade forest, J. Appl. Remote Sens. 15 (2) (2021) 028507.

[25] G. Mateo-García, L. Gómez-Chova, G. Camps-Valls, Convolutional neural networks for multispectral image cloud masking, in: 2017 IEEE International Geoscience and Remote Sensing Symposium, IGARSS, 2017, pp. 2255–2258.

[26] B.-C. Gao, P. Yang, R.-R. Li, Detection of high clouds in polar regions during the daytime using the MODIS 1.375-/spl mu/m channel, IEEE Trans. Geosci. Remote Sens. 41 (2) (2003) 474–481.

[27] S.A. Ackerman, K.I. Strabala, W.P. Menzel, R.A. Frey, C.C. Moeller, L.E. Gumley, Discriminating clear-sky from clouds with MODIS, J. Geophys. Res. 103 (1998) 141–157.

[28] C. Huang, N. Thomas, S.N. Goward, J.G. Masek, Z. Zhu, J.R. Townshend, J.E. Vogelmann, Automated masking of cloud and cloud shadow for forest change analysis using landsat images, 2010.

[29] L. Oreopoulos, M.J. Wilson, T. Várnai, Implementation on landsat data of a simple cloud-mask algorithm developed for MODIS land bands, IEEE Geosci. Remote Sens. Lett. 8 (4) (2011) 597–601.

[30] Y. Zhang, B. Guindon, X. Li, A robust approach for object-based detection and radiometric characterization of cloud shadow using haze optimized transformation, IEEE Trans. Geosci. Remote Sens. 52 (9) (2014) 5540–5547.

[31] Y. Oishi, H. Ishida, R. Nakamura, A new landsat 8 cloud discrimination algorithm using thresholding tests, Int. J. Remote Sens. 39 (23) (2018) 9113–9133.

[32] Q. Shi, B. He, Z. Zhe, Z. Liao, X. Quan, Improving fmask cloud and cloud shadow detection in mountainous area for landsats 4–8 images, Remote Sens. Environ. 199 (2017) 107–119.

[33] R. Zhang, D. Sun, S. Li, Y. Yu, A stepwise cloud shadow detection approach combining geometry determination and SVM classification for MODIS data, Int. J. Remote Sens. 34 (1) (2013) 211–226.

[34] P. Li, L. Dong, H. Xiao, M. Xu, A cloud image detection method based on SVM vector machine, Neurocomputing 169 (2015) 34–42.

[35] X. Hu, Y. Wang, J. Shan, Automatic recognition of cloud images by using visual saliency features, IEEE Geosci. Remote Sens. Lett. 12 (8) (2015) 1760–1764.

[36] M. Le Goff, J.Y. Tourneret, H. Wendt, M. Ortner, M. Spigai, Deep learning for cloud detection, in: 8th International Conference of Pattern Recognition Systems ICPRS 2017, 2017, pp. 1–6.

[37] T. Johnston, S.R. Young, D. Hughes, R.M. Patton, D. White, Optimizing convolutional neural networks for cloud detection, in: Proceedings of the Machine Learning on HPC Environments, 2017.

[38] M. Shi, F. Xie, Y. Zi, J. Yin, Cloud detection of remote sensing images by deep learning, in: 2016 IEEE International Geoscience and Remote Sensing Symposium, IGARSS, 2016, pp. 701–704.

[39] G. Terrén-Serrano, M. Martínez-Ramón, Explicit basis function kernel methods for cloud segmentation in infrared sky images, 2021, arXiv preprint arXiv:2102.06646.

[40] Y. Zhan, J. Wang, J. Shi, G. Cheng, L. Yao, W. Sun, Distinguishing cloud and snow in satellite images via deep convolutional network, IEEE Geosci. Remote Sens. Lett. 14 (10) (2017) 1785–1789.

[41] H. Jiang, N. Lu, Multi-scale residual convolutional neural network for haze removal of remote sensing images, Remote Sens. 10 (2018) 945.

[42] Z. Yan, M. Yan, H. Sun, K. Fu, J. Hong, J. Sun, Y. Zhang, X. Sun, Cloud and cloud shadow detection using multilevel feature fused segmentation network, IEEE Geosci. Remote Sens. Lett. 15 (10) (2018) 1600–1604.

[43] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2015, pp. 3431–3440.

[44] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, 2015, pp. 234–241.

[45] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: ECCV, 2018.

[46] J.H. Jeppesen, R.H. Jacobsen, F. Inceoglu, T.S. Toftegaard, A cloud detection algorithm for satellite imagery based on deep learning, Remote Sens. Environ. 229 (2019) 247–259.

[47] S. Mohajerani, P. Saeedi, Cloud-Net: An end-to-end cloud detection algorithm for landsat 8 imagery, in: IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, 2019.

[48] M.R. Azimi-Sadjadi, W. Gao, T.H. Vonder Haar, D. Reinke, Temporal updating scheme for probabilistic neural network with application to satellite cloud classification - further results, IEEE Trans. Neural Netw. 12 (5) (2001) 1196–1203.

[49] J. Guo, J. Yang, H. Yue, H. Tan, C. Hou, K. Li, CDNetv2: CNN-based cloud detection for remote sensing imagery with cloud-snow coexistence, IEEE Trans. Geosci. Remote Sens. 59 (1) (2020) 700–713.

[50] J. Zhang, H. Wang, Q. Zhou, Y. Wang, Y. Li, Deep encoder-decoder network based on the up and down blocks using wavelet transform for cloud detection, in: IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium, IEEE, 2020, pp. 2583–2586.

[51] X. Li, H. Zheng, C. Han, W. Zheng, H. Chen, Y. Jing, K. Dong, SFRS-net: A cloud-detection method based on deep convolutional neural networks for GF-1 remote-sensing images, Remote Sens. 13 (15) (2021) 2910.

[52] Y. Liu, W. Wang, Q. Li, M. Min, Z. Yao, DCNet: A deformable convolutional cloud detection network for remote sensing imagery, IEEE Geosci. Remote Sens. Lett. (2021).

[53] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7794–7803.

[54] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141, http://dx.doi.org/10.1109/CVPR.2018.00745.

[55] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: International Conference on Machine Learning, PMLR, 2015, pp. 448–456.

[56] X. Glorot, A. Bordes, Y. Bengio, Deep sparse rectifier neural networks, in: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, 2011, pp. 315–323.

[57] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, Y. Fu, Residual dense network for image super-resolution, in: CVPR, 2018.

[58] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, A. Agrawal, Context encoding for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, 7151–7160.