symmetry

MDPI

Article

MCNet: Multi-Scale Feature Extraction and Content-Aware Reassembly Cloud Detection Model for Remote Sensing Images

Ziqiang Yao [1,2,3], Jinlu Jia [1,2,3] and Yurong Qian [1,2,3,*]

[1] College of Software, Xinjiang University, Urumqi 830000, China; yzq@stu.xju.edu.cn (Z.Y.); jjj@stu.xju.edu.cn (J.J.)
[2] Key Laboratory of Signal Detection and Processing, Xinjiang University, Urumqi 830046, China
[3] Key Laboratory of Software Engineering, Xinjiang University, Urumqi 830046, China
* Correspondence: qyr@xju.edu.cn

Abstract: Cloud detection plays a vital role in remote sensing data preprocessing. Traditional cloud detection algorithms have difficulties in feature extraction and thus produce a poor detection result when processing remote sensing images with uneven cloud distribution and complex surface background. To achieve better detection results, a cloud detection method with multi-scale feature extraction and content-aware reassembly network (MCNet) is proposed. Using pyramid convolution and channel attention mechanisms to enhance the model's feature extraction capability, MCNet can fully extract the spatial information and channel information of clouds in an image. The content-aware reassembly is used to ensure that sampling on the network can recover enough in-depth semantic information and improve the model cloud detection effect. The experimental results show that the proposed MCNet model has achieved good detection results in cloud detection tasks.

Keywords: cloud detection; convolutional neural network; multi-scale feature extraction; content-aware reassembly; deep learning

check for updates

Citation: Yao, Z.; Jia, J.; Qian, Y. MCNet: Multi-Scale Feature Extraction and Content-Aware Reassembly Cloud Detection Model for Remote Sensing Images. *Symmetry* **2021**, *13*, 28. https://dx.doi.org/10.3390/sym13010028

Received: 23 November 2020
Accepted: 22 December 2020
Published: 26 December 2020

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the progress of remote sensing technology, remote sensing images are widely used in navigation and positioning [1,2], ground object detection [3,4], environmental surveillance [5,6], and many other fields. However, many clouds will affect the imaging quality of remote sensing images, resulting in less available information in remote sensing images and a negative impact on tasks, such as subsequent target detection and tracking, segmentation, and classification. Therefore, it is of great significance to study high precision remote sensing cloud detection.

The traditional remote sensing cloud detection methods mainly include the physical threshold method, detection method based on cloud texture and spatial characteristics, and detection method based on machine learning. Limited by the spectrum of early remote sensing images, the ISCCP (The International Satellite Cloud Climatology Project) method [7], CLAVR (The NOAA Cloud Advanced Very High Resolution Radiometer) method [8], and other methods based on physical thresholds have low detection accuracy. As the spectrum of remote sensing satellites increases, multi-spectral cloud detection technology has been developed, using the physical characteristics of clouds such as high reflectivity and low-temperature characteristics, cloud detection is achieved through physical radiation threshold filtering [9–11]. The cloud detection method based on image space information [12,13], using the characteristic that the radiation change degree of cloud cover image is higher than that of a clean appearance, compares the field radiation value change of each pixel with a certain threshold to achieve pixel-level cloud segmentation. This method is suitable for remote sensing cloud detection with a single background such as in the sea area scene or an explicit scene. Cloud detection methods based on texture information [14,15] use the improved fractal dimension method to express the difference between

cloud texture features and surface object texture features to achieve a higher efficiency of cloud detection. The detection method based on cloud texture and spatial information relies on fixed features and has low robustness. With machine learning development, traditional machine learning methods such as decision trees and support vector machines (SVM) are used for cloud detection. The Fmask (Function of mask) method [16–18] uses a decision tree with optimized thresholds to classify each pixel. The Fmask method is widely used as a baseline for remote sensing image cloud detection and evaluation. The ACCA (automated cloud cover assessment) method [19] judges whether there is a cloud through a decision tree threshold. Latry et al. [20] use support vector machines to create a decision boundary for cloud detection by representing mapping data. The accuracy of traditional cloud detection and machine learning cloud detection algorithms depends on the correctness of training data and the effectiveness of features, and the detection accuracy of complex cloud layers is low.

In recent years, semantic segmentation combined with deep learning has achieved good results in cloud detection. Fully convolutional neural network (FCN) [21] is the pioneering work of image semantic segmentation algorithm. The image-level classification is extended to pixel-level classification through end-to-end training of the fully convolutional neural network, and semantic segmentation is realized. Mohajerani et al. [22] proposed a hybrid full convolution FCN and gradient identification algorithm and applied FCN to the cloud detection field. Mohajerani et al. [23] proposed that the Cloud-Net algorithm achieves a better detection effect by redesigning the convolution block based on FCN, and Cloud-Net is usually used as the baseline of deep learning cloud detection network. Chen et al. [24] proposed the Deeplab algorithm to expand the receptive field, extract more spatial features, and enrich network features' expression by introducing dilated convolution in the FCN. However, dilated convolution led to the loss of some detailed information of clouds [25], affecting local consistency. Zhao et al. [26] propose that the PSPNet algorithm realizes multi-scale feature fusion by embedding a pyramid pooling module (PPM) in the FCN to solve the problem of insufficient feature extraction ability FCN. Inspired by PSPNet, Yan et al. [27] proposed an MFFSNet algorithm that uses PPM to aggregate feature information of different scales to improve the utilization rate of local and global features of clouds in images. However, this algorithm easily loses partial boundary information. Ronneberger et al. [28] proposed that the Unet uses skip connection to connect the encoding and decoding network to solve the problem of rough mask generation by the FCN. Gonzales et al. [29] used the Unet based on ResNet34 [30] pretraining for cloud detection. It uses transfer learning to transfer the common feature attributes of different types of images to the feature extraction step of cloud detection to strengthen the network feature extraction capability. Guo et al. [31] proposed that the Cloud-AttU algorithm adds a spatial attention mechanism to the Unet and learns more effective feature information expression through jump layer connections. Compared with the cloud detection algorithm based on the traditional method, the above-mentioned remote sensing cloud detection methods based on deep learning can better mine the in-depth semantic information in the cloud detection image and has higher robustness and higher precision. However, the above remote sensing cloud detection algorithms pay more attention to the multi-scale information of the downsampling operation, insufficient utilization of channel features. The receptive field size is insufficient in the upsampling operation, and the upsampling process cannot be adaptively optimized according to the characteristic information. In the field of remote sensing cloud detection, good detection performance has not yet been achieved.

From what has been discussed above, cloud detection in remote sensing images still faces enormous challenges because there are many types of clouds in remote sensing images and their distribution is uneven, which brings specific difficulties to detection, and the background contains many similar ground objects, which makes of cloud detection even harder. Currently, remote sensing cloud detection still has the following problems.

1. The remote sensing cloud image obtained in the real scene contains many surface objects (such as snow, ice, trees, and white human-made objects) with similar re-

flection characteristics to the clouds, and the background interference is serious. As a result, it is challenging to capture clouds under a large amount of background interference accurately.
2. The uneven distribution and thickness of clouds in remote sensing images make the detection accuracy low.
3. Clouds are affected by shooting angles and wind speeds, resulting in different scales and various shapes, and the accuracy of cloud mask generation in complex scenes is low.

This paper proposes a deep neural network cloud detection method that combines multi-scale feature extraction based on channel attention mechanism and content perception reorganization to solve the above problems. Experiments show that this method has excellent detection performance. The main contributions of this article are as follows.

- To solve the detection problems of uneven cloud layer thickness, uneven cloud layer distribution, and background interference, a pyramidal convolution residual network with an efficient channel attention (ECA) module (EPResNet-50) is proposed, which uses pyramidal convolution [32] to capture multi-scale feature information and increase the network's attention to effective channels through the ECA [33] module, comprehensively consider the channel characteristics and spatial characteristics, and enhance the ability of network feature extraction.
- To solve the problem of low accuracy of cloud generation masks in complex scenes, the CARAFE (Content-Aware ReAssembly of FEatures) [34] upsampling module is introduced. The semantic information of feature maps is fully utilized for feature restoration through adaptive kernels, and the accuracy of network generation masks is improved.
- Conduct a comparative experiment between the proposed algorithm and the current mainstream algorithm on the 38-cloud [22,23] dataset. Experimental results show that this method has better detection performance.

The other parts of the paper are organized as follows. In Section 2, we introduce an overview of the method in this paper. In Section 3, we compare and analyze the experimental results between the methods proposed in this paper and other methods. In Section 4, we provide a comprehensive conclusion.

2. Methods

This research introduces pyramidal convolution based on the efficient channel attention mechanism in the Unet to make full use of the red, green, blue, and near-infrared (RGBN) multi-spectral channel features and improves the network feature extraction ability. Content-aware reorganization is introduced in the upsampling. The network can adaptively optimize the upsampling operation according to the extracted feature information and improve mask generation accuracy.

2.1. MCNet

The proposed MCNet is based on the encoder and decoder structure, which connects the encoder and the decoder through a skip connection. The backbone of the MCNet encoder is a pyramidal convolutional residual network base on efficient channel attention (EPResNet-50). Each block of EPResNet-50 can process the input feature map at multiple scales and extract the interdependence of local channels to improve the encoder's feature expression ability. Each step of the decoder consists of CARAFE upsampling operation and two convolution operations. CARAFE makes full use of the in-depth semantic information of features through an adaptive kernel, strengthens the decoder's feature recovery capability, and improves the accuracy of mask generation. In the skip connection process, we use 1×1 convolution to adjust the number of output channels of each module of EPResNet-50. It has the same number of channels as the decoder as well as a convenient encoder–decoder combination. The MCNet conceptual structure is shown in Figure 1.

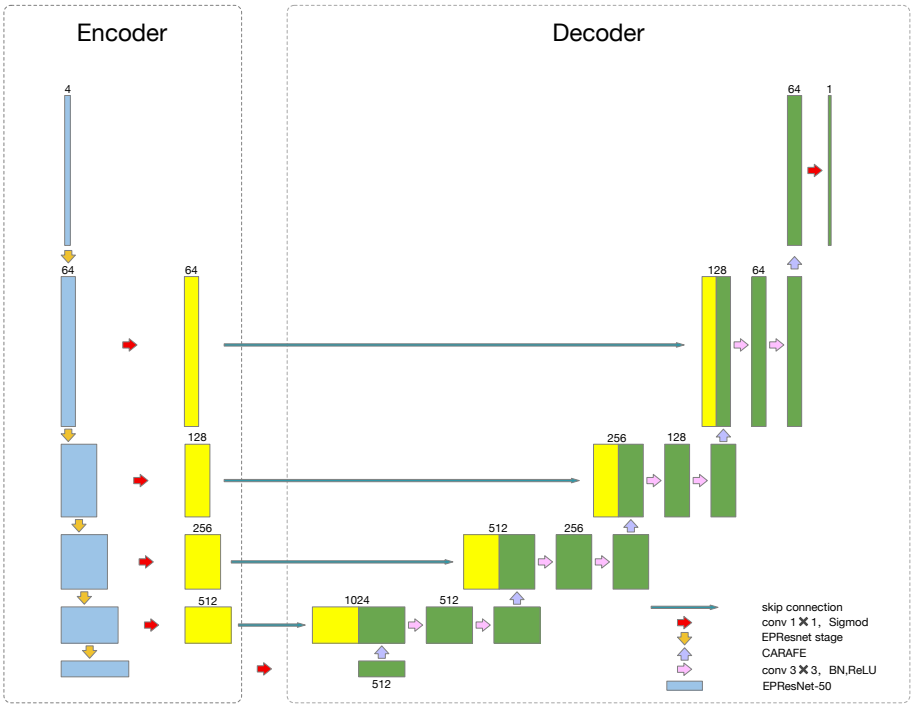


Figure 1. The MCNet conceptual structure.

2.2. EPResNet-50

In the encoder, EPResNet-50 was used as the backbone network for feature extraction, which enhanced feature extraction capability by adding pyramidal convolution and efficient channel attention to take into account the channel characteristics and spatial characteristics.

2.2.1. Pyramidal Convolution

Due to the uneven distribution of clouds in remote sensing images and large-scale differences, the proportion of cloud coverage in different images varies greatly. Traditional convolution uses a single-scale convolution kernel, which cannot extract features from multiple scales and is not suitable for remote sensing cloud detection. In this paper, by introducing pyramidal convolution (PyConv) in the encoder, it is used to capture cloud information in different spaces and depths on multiple scales and improve the network's ability to detect clouds uneven distribution and large scale differences. Pyramidal convolution operation is shown in Figure 2. For the input feature map of FM_i channels, the convolution kernel with different sizes of K_x and different depths of FM_{ox} (x is the grouping convolution number) is used to conduct grouping convolution processing for the input features. Finally, the feature splice generates the final output feature.

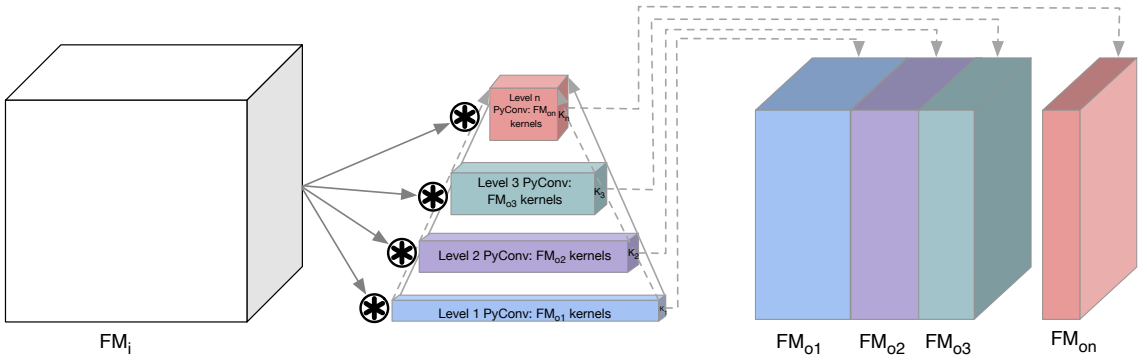


Figure 2. Pyramidal convolution model structure.

2.2.2. ECA

Many ground objects in remote sensing images are covered by ice and snow, and the temperature at the covered area is low. This results in the near-infrared radiation of ground objects being similar to that of clouds, and the color of ice and snow is identical to that of clouds. Therefore, it is difficult to accurately distinguish clouds under a large amount of similar background interference. In order to reduce similar background interference and improve detection accuracy, we add channel attention module ECA to the encoder to realize local cross-channel information interaction without dimension reduction, allowing network to focus on the aggregation of image channel features, which improve the feature expression ability of the encoder, and enhance the network discrimination ability.

The ECA module is mainly composed of global-average-pooling (GAP), Conv1d, and Scale. GAP compresses $W \times H \times C$ input features along the spatial dimension into $1 \times 1 \times C$ global pooling information, using the channel size adaptive function $\psi(C)$:

$$k = \psi(C) = \left| \frac{\log_2(C)}{\gamma} + \frac{b}{\gamma} \right|_{\text{odd}} \tag{1}$$

Reach the number of adjacent channels k that participates in the channel weight calculation. The parameters b and γ are set to 1 and 2, respectively, in this article, and C is the number of available channels; $|\bullet|_{\text{odd}}$ means to take adjacent odd numbers. Conv1d obtains the local cross-channel weight value ω by taking the adjoining k-channel according to the formula 2, scale multiplies the weight value to weight the previous feature, and finally obtains a feature map with channel weights:

$$\omega_i = \sigma\left(\sum_{j=1}^{k} \alpha^j y_i^j\right), y_i^j \in \Omega_i^k \tag{2}$$

where Ω_i^k represents the set of k adjacent channels of the channel y_i, α is the channel sharing parameter, and σ is the Sigmoid function. The ECA model structure is shown in Figure 3.

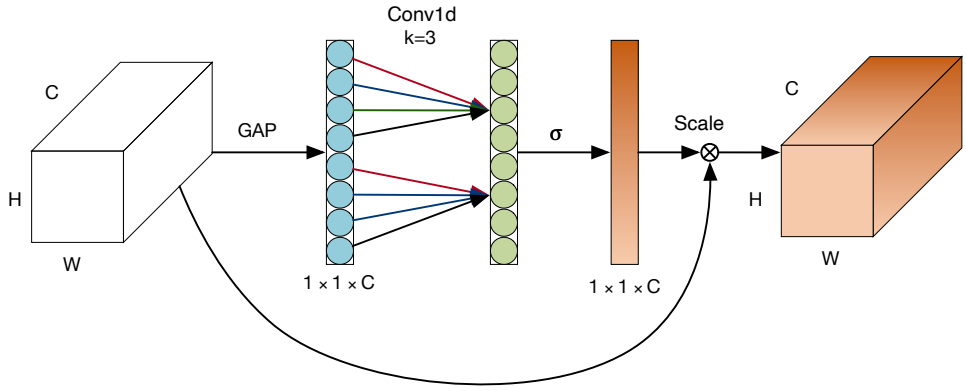


Figure 3. ECA model structure.

2.2.3. EPResNet-50 Architectures

The basic model of the EPResNet-50 model in this article is ResNet-50. EPResNet-50 introduces feature pyramid convolution to capture more comprehensive spatial information by multi-scale extraction of cloud feature, and introduces the ECA module to aggregate feature channel information, using ECA to improve the effectiveness of feature expression by paying attention to the dependency between channels. EPResNet-50 consists of a stack of similar structural blocks, the overall structure of EPResNet-50 is shown in Table 1.

To extract cloud layer information at multiple scales, the depth and size of each block's pyramid convolution kernel in EPResNet-50 are different. The stage1 block of EPResNet-

50 is shown in Figure 4. The input features are first subjected to 1×1 convolution for feature mapping. After the mapping, the features are input into pyramidal convolution for multi-scale feature extraction. The number of feature channels is adjusted to 256 through 1×1 convolution. The 256 channels feature information is input to the ECA module to achieve local cross-channel interaction without dimension reduction, and to aggregate channel information, and finally, to combine the aggregated features with the original input to produce the output features. Compared with ResNet-50, EPResNet-50 can get more effective features in space and channel dimensions.

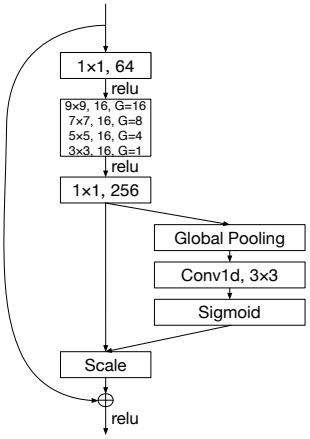


Figure 4. The Stage1 block of EPResNet-50.

Table 1. ResNet-50, EPResNet-50 structure comparison.

Stage	Output	ResNet-50	EPResNet-50
0	192×192	$7 \times 7,\ 64,\ s = 2$ $3 \times 3\ max\ pool,\ s = 2$	$7 \times 7,\ 64,\ s = 2$
1	96×96	$\begin{bmatrix} 1 \times 1,\ 64 \\ 3 \times 3,\ 64 \\ 1 \times 1,\ 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1,\ 64 \\ \overline{Pyconv4,\ 64:} \\ \begin{bmatrix} 9 \times 9,\ 16,\ G = 16 \\ 7 \times 7,\ 16,\ G = 8 \\ 5 \times 5,\ 16,\ G = 4 \\ 3 \times 3,\ 16,\ G = 1 \end{bmatrix} \\ 1 \times 1,\ 256 \\ ECA_block,\ k = 3 \end{bmatrix} \times 3$
2	48×48	$\begin{bmatrix} 1 \times 1,\ 128 \\ 3 \times 3,\ 128 \\ 1 \times 1,\ 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1,\ 128 \\ \overline{Pyconv3,\ 128:} \\ \begin{bmatrix} 7 \times 7,\ 64,\ G = 8 \\ 5 \times 5,\ 32,\ G = 4 \\ 3 \times 3,\ 32,\ G = 1 \end{bmatrix} \\ 1 \times 1,\ 512 \\ ECA_block,\ k = 3 \end{bmatrix} \times 4$
3	24×24	$\begin{bmatrix} 1 \times 1,\ 256 \\ 3 \times 3,\ 256 \\ 1 \times 1,\ 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1,\ 256 \\ \overline{Pyconv2,\ 256:} \\ \begin{bmatrix} 5 \times 5,\ 128,\ G = 4 \\ 3 \times 3,\ 128,\ G = 1 \end{bmatrix} \\ 1 \times 1,\ 1024 \\ ECA_block,\ k = 3 \end{bmatrix} \times 6$
4	12×12	$\begin{bmatrix} 1 \times 1,\ 512 \\ 3 \times 3,\ 512 \\ 1 \times 1,\ 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1,\ 512 \\ \overline{Pyconv1,\ 512:} \\ [3 \times 3,\ 512,\ G = 1] \\ 1 \times 1,\ 2048 \\ ECA_block,\ k = 3 \end{bmatrix} \times 3$

2.3. CARAFE

The decoder structure recovers cloud effective features through stack upsampling, and the feature upsampling module is the key operation module in the convolution network structure. Unet uses bilinear interpolation method for upsampling and uses a preset sampling core. The semantic information of feature map does not play a role, and the perceptual domain is only 2×2, so the network cannot make full use of the information of nearby regions, resulting in the decoder cannot completely recover the effective features of the cloud layer. The CARAFE module adaptively generates the upsampling kernel based on the input features, which can not only make full use of the semantic information of the feature map, but also obtains a larger receptive field, which enhances the feature recovery capability of the decoder and improve the mask generation accuracy.

We use the CARAFE module for upsampling. CARAFE combines the kernel prediction module with the content perception recombination module, uses the underlying content information to predict the adaptively optimized recombination kernel, and performs feature recombination in the vicinity of each pixel to achieve upsampling of the content perception recombination. The CARAFE module is shown in Figure 5.

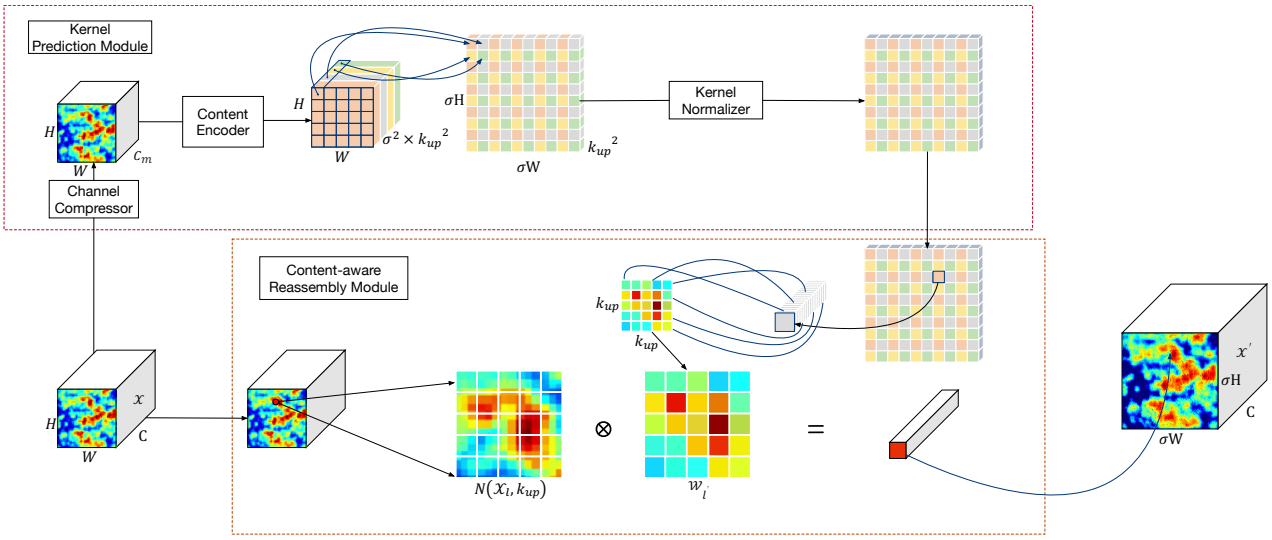


Figure 5. The CARAFE module structure.

In CARAFE, the feature map $\mathcal{X}$ of size $C \times H \times W$ and the upsampling ratio σ (assuming σ is an integer) are input into the kernel prediction module and the content reorganization module at the same time. After module calculation, the output feature map $\mathcal{X}'$ with the size of $C \times \sigma H \times \sigma W$ is obtained, and the upsampling operation of the feature map is realized. Any position $l' = (i', j')$ on the output feature map $\mathcal{X}'$. There is a corresponding $l = (i, j)$ at the input feature map $\mathcal{X}$, where $i = |i'/\sigma|$,$j = |j'/\sigma|$.

The kernel prediction module ψ adaptively generates a kernel $\mathcal{W}_{l'}$ for each target position l' in the output feature map $\mathcal{X}'$, and the calculation is depicted as follows,

$$\mathcal{W}_{l'} = \psi(N(\mathcal{X}_l, k_{\text{encoder}})) \tag{3}$$

where $\mathcal{X}_l$ is a certain pixel position on the input feature map $\mathcal{X}$, k_{encoder} is the kernel, and $N(\mathcal{X}_l, k_{\text{encoder}})$ is the target position l' on the input feature map $\mathcal{X}$, which corresponds to a square area with l as the center and size k_{encoder}.

The kernel prediction module is composed of three sub-modules: channel compressor, content encoder, and kernel normalizer. First, the channel compressor compresses the feature map of $C \times H \times W$ into $C_m \times H \times W$. Then, through the $k_{\text{encoder}} \times k_{\text{encoder}} \times C_{up}$ size convolution kernel convolution calculation in the content encoder, the feature map of size $\sigma^2 \times {k_{up}}^2 \times H \times W$ is obtained, where $C_{up} = \sigma^2 \times {k_{up}}^2$. Next, convert the feature map

to the size of ${k_{up}}^2 \times \sigma H \times \sigma W$, finally, use the softmax function of the kernel normalizer to obtain the standardized output.

The recombination module uses a weighted operation to obtain the upsampling feature of the pixel l'. The formula for weighting is as follows,

$$\mathcal{X}'_{l'} = \phi\left(N\left(\mathcal{X}_l, k_{up}\right), \mathcal{W}_{l\prime}\right) = \sum_{n=-r}^{r} \sum_{m=-r}^{r} \mathcal{W}_{l'(n,m)} \cdot \mathcal{X}_{(i+n,j+m)} \quad (4)$$

where k_{up} is the size of the reorganized kernel, $N\left(\mathcal{X}_l, k_{up}\right)$ is the target position l' in the input feature $\mathcal{X}$ corresponding to a square area with l as the center and size of $k_{up} \times k_{up}$, $\mathcal{W}_{l\prime}$ is the reorganized kernel generated by the kernel prediction module at the target location l', $r = \lfloor kup/2 \rfloor$.

3. Experiments

3.1. Datasets

In this paper, we use 38-cloud as the benchmark cloud detection data set for evaluating model performance. The 38-cloud dataset has 38 scenarios, including 18 training scenarios and 20 test scenarios. Each scene data is provided by the Landsat 8 satellite [35]. The Landsat 8 satellite carries two sensors, namely, the Operational Land Imager (OLI) and the Thermal Infrared Sensor (TIRS). OLI has 9 spectral bands, the spatial resolution for bands 1 to 7 and 9 is 30 m, the spatial resolution for band 8 is 15 m. TIRS has 2 spectral bands, the spatial resolution for bands 10 to 11 is 30 m. The information for all bands of the Landsat 8 is shown in Table 2. The 38-cloud dataset selects the red, green, blue, and near-infrared spectral bands of the landsat 8 satellite, and provides corresponding hand-labeled segmentation masks. The scene image is cropped into 384×384 patches. There are 8400 patches in the training set and 9200 patches in the test set. We use holdout cross-validation to partition dataset. We randomly use 85% of the patches in training set for training and 15% of patches for verification. To prevent the model from overfitting and enhance the robustness of the model, we use random translation, symmetry, rotation, and scaling to enhance the training set.

Table 2. Landsat 8 Spectral Bands.

Spectral Bands	Wavelength (μm)	Resolution (m)
Band 1—Ultra Blue	0.435–0.451	30
Band 2—Blue	0.452–0.512	30
Band 3—Green	0.533–0.590	30
Band 4—Red	0.636–0.673	30
Band 5—Near Infrared	0.851–0.879	30
Band 6—Shortwave Infrared 1	1.566–1.651	30
Band 7 -Shortwave Infrared 2	2.107–2.294	30
Band 8—Panchromatic	0.503–0.676	15
Band 9—Cirrus	1.363–1.384	30
Band 10—Thermal Infrared 1	10.60–11.19	30
Band 11—Thermal Infrared 2	11.50–12.51	30

3.2. Evaluation Metrics

We use Overall Accuracy, Recall, Precision, Specificity, Jaccard Index, and F1-Score as evaluation indicators. The Precision indicator is used to judge the precision of the algorithm, and the Recall indicator is used to judge the precision of the algorithm. The Specificity index is used to measure the completeness of the error prediction, and the Overall Accuracy index is used to indicate the accuracy of the two classifications. The Jaccard Index indicator is used to describe the similarity between the predicted mask and the real mask. The F1-Score considers the relationship between Precision and Recall. Moreover, the

Jaccard Index indicator is an important index to judge the performance of cloud detection algorithms [22,23,31].

$$\text{Jaccard Index} = \frac{TP}{TP + FN + FP} \tag{5}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{6}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{7}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{8}$$

$$\text{Overall Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

Among them, TP represents the number of positive samples whose judgment result is positive, TN represents the number of negative samples whose judgment result is negative, FP represents the number of negative samples whose judgment result is positive, and FN represents the number of positive samples whose judgment result is negative.

3.3. Experimental Details

In this study, all experiments are programmed and implemented using the PyTorch framework running on Ubuntu16.04 with NVIDIA RTX 2080Ti GPU. The experiment uses python3.6 as the software environment. In the experiment, we use the 384 × 384 pixel RGBN 4-channel patch image in 38-cloud as the input of the neural network. The training batch is 8, using the BCE (Binary Cross Entropy) loss function, a total of 400 epochs are trained, and the Adam (Adaptive Moment Optimization) optimizer is used for training optimization. The initial learning rate is set to 0.001, and the learning rate is attenuated by 0.1 at 200, 300, 350, and 390, respectively.

3.4. Experimental Result Analysis

3.4.1. Ablation Experiment

To verify the rationality of the module combination in the network, we use the same computing environment to perform eight sets of ablation experiments on the same data set. Table 3 shows the experimental results. The first line is Unet, which uses ResNet-50 as the backbone network. The ECA, PyConv, or CARAFE conditions were activated separately or conjunctively. The network performance was comparable to the reference network. The Jaccard Index increase indicator 0.58%, 2.62%, and 1.46%, indicating that the ECA, PyConv, and CARAFE modules' validity added separately. Adding any two modules of ECA, PyConv, and CARAFE to Unet simultaneously, the test results are further improved than adding a single module, which verifies the mutual assistance effect between the three modules. Under the condition of adding three modules at the same time, all the index results are optimal. Jaccard Index reaches 83.05%, precision reaches 94.83%, Recall reaches 86.69%, Specificity reaches 98.67%, and Overall Accuracy reaches 96.44%, which proves the rationality of the network structure of this article. ECA considers the interaction of local channels to enhance the importance of effective channels and extract unique information from the cloud. PyConv extracts cloud spatial information from a multi-scale perspective. The PyConv integrated with the ECA module comprehensively considers the spatial relationship and channel relationship of the features extracts comprehensive and representative cloud information, and improves the feature expression ability of the encoder. The upsampling process of the CARAFE module can adaptively generate a kernel according to the input features, which can make full use of the semantic information of the feature map to restore more cloud layer details and improve the mask generation accuracy of the decoder.

Table 3. Ablation test of ECA, PyConv, and CARAFE on the 38-cloud dataset.

ECA	PyConv	CARAFE	Jaccard Index [%]	Precision [%]	Recall [%]	Specificity [%]	Overall Accuracy [%]	F1-Score [%]
			78.00	86.61	87.92	98.65	96.30	87.26
✓			78.58	90.89	84.40	98.59	96.37	87.52
	✓		80.62	94.43	84.50	98.87	96.47	89.19
		✓	79.46	90.71	85.95	98.67	96.15	88.27
✓	✓		80.94	92.76	86.01	98.48	96.45	89.26
✓		✓	80.63	92.74	86.06	98.43	96.50	89.28
	✓	✓	81.20	92.69	86.11	98.59	96.53	89.28
✓	✓	✓	83.05	94.83	86.69	98.67	96.49	90.58

3.4.2. Comparison with Some Popular Methods

To prove the effectiveness of the MCNet algorithm, we conducted a quantitative analysis with some popular cloud detection algorithms and semantic segmentation algorithms. Including Gradient Boosting Classifier (BGC), Random Forest (RF), FCN, Fmask, Cloud-net, Att-unet, PSPNet, and Deeplabv3, Table 4 shows the cloud detection results on the 38-cloud dataset. According to the experimental results in Table 4, MCNet's Jaccard Index reached 83.05%, Precision reached 94.83%, Recall reached 86.69%, Specificity reached 98.67%, and Overall Accuracy reached 96.49%. MCNet algorithm is 6.8%, 10.4%, 1.5%, and 0.7% higher than Unet algorithm in Jaccard Index, Precision, Specification, and Overall Accuracy, respectively. The MCNet algorithm is compared with the Cloud-net algorithm of cloud detection deep learning, Jaccard Index is 4.5% higher, and the precision is 3.6% higher; compared with the sub-optimal PSPNet algorithm, Jaccard Index is 3.7% higher and Precision is 7.7% higher. The experimental results indicated the algorithm proposed in this paper has achieved higher performance accuracy under each index, which demonstrates the effectiveness of this algorithm.

Table 4. Comparative test of different methods on the 38-cloud dataset.

Method	Jaccard Index [%]	Precision [%]	Recall [%]	Specificity [%]	Overall Accuracy [%]	F1-Score [%]
GBC	51.77	65.34	66.78	88.74	83.49	66.05
RF	56.52	71.65	68.12	91.79	87.11	69.84
FCN	72.17	84.59	81.37	98.45	95.23	82.95
Fmask	75.16	77.71	97.22	93.96	94.89	86.38
Cloud-net	78.50	91.23	84.85	98.67	96.48	87.92
Att-unet	74.54	87.27	84.59	96.74	94.63	85.91
PSPNet	79.36	87.12	86.09	96.97	96.08	86.60
Deeplabv3	74.20	92.33	78.30	98.55	95.45	84.74
Unet	76.20	84.47	87.91	97.13	95.72	86.16
MCNet	83.05	94.83	86.69	98.67	96.49	90.58

Figures 6 and 7 show the qualitative comparison between MCNet and other cloud detection algorithms. In Figures 6 and 7, the first column is the test pseudo-color image; the second column is the ground truth; and the third, fourth, fifth, and sixth columns are masks generated by Cloud-net, PSPNet, Unet, and MCNet, respectively, in which white represents clouds and black represents background.

The blue box in Figure 6 is the detection result under complex clouds, the yellow box is the detection result under similar background interference such as ice and snow, and the red box is the detection result under the simultaneous existence of complex clouds and similar background interference such as ice and snow. Figure 8 shows the detailed information in the blue box, yellow box and red box from Figure 6. The experimental results show that the results from MCNet are less disturbed by similar backgrounds such

as ice and snow, the generated mask is more real, closer to ground truth, and the detection accuracy is higher than other algorithms.

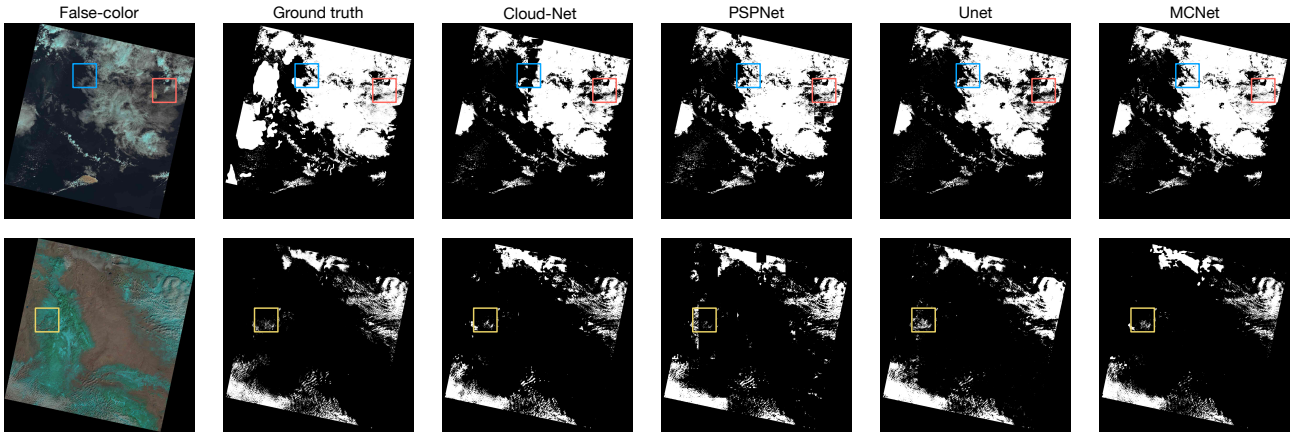


Figure 6. Visual results of two overall scene image obtained by different models on the 38-cloud dataset.

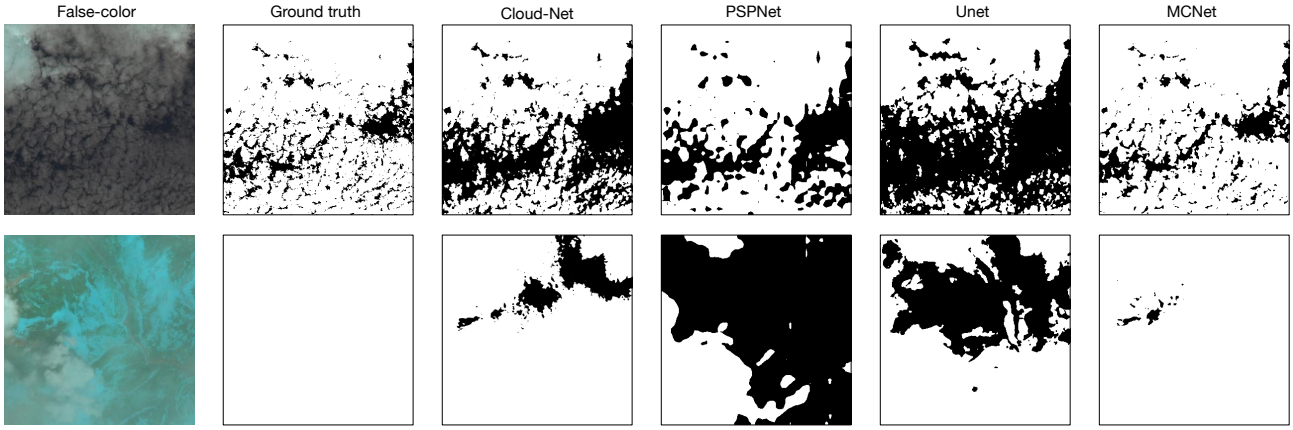


Figure 7. Some patch image visual examples obtained by different models on the 38-cloud dataset.

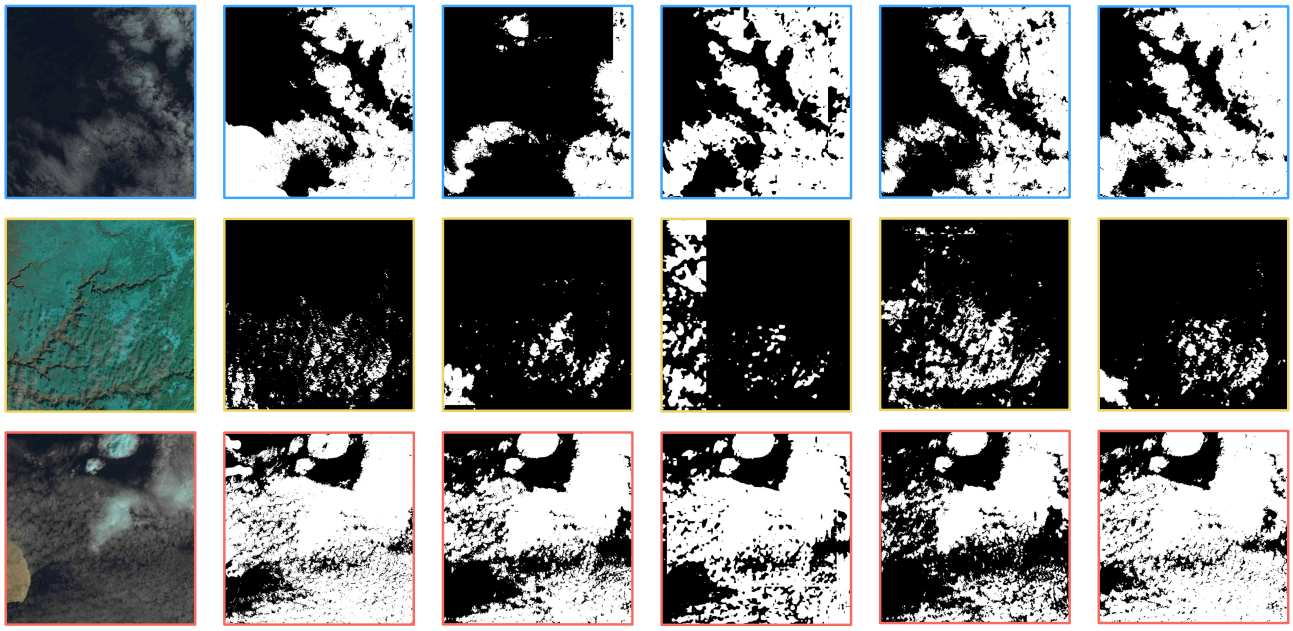


Figure 8. Details of blue box, yellow box, and red box in Figure 6.

Seen from the experimental results of Figure 7, the masks generated by other detection algorithms have the problems of missing information and inaccurate positioning. The mask generated by MCNet retains more detailed cloud information, proving that MCNet can accurately and comprehensively capture cloud information, thus effectively avoiding this problem.

4. Conclusions

This paper proposes a deep neural network cloud detection method that combines multi-scale feature extraction and content perception reorganization based on the attention mechanism. To solve the interferences in cloud detection caused by cloud thickness, uneven cloud distribution, and ground objects in remote sensing images, the encoder uses EPResNet-50 to extract multi-scale spatial features and channel dependencies. It improves the model's utilization of multi-spectral image features. The CARAFE module is introduced in the decoder stage, and the size of the upsampled receptive field is automatically adjusted through the adaptive kernel, and the deep semantic information is fully utilized for feature recovery to improve the mask generation accuracy. We use the 38-cloud data set captured by the Landsat 8 satellite for a validation. The experimental results prove the MCNet algorithm's effectiveness, Jaccard Index reaches 83.05%, precision reaches 94.83%, Recall reaches 86.69%, and Specificity reaches 98.67%, even in complex background interference. It also has a better detection effect in scenes with large differences in cloud distribution. In future research, we will further optimize the model. By introducing an adaptive dynamic filter, the adaptability of the model will be enhanced, and therefore the model can adaptively optimize convolution operation in different scenes.

Author Contributions: All authors have worked on this manuscript together and All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Science Foundation of China under Grant 61966035, by the Autonomous Region graduate Student Innovation project "K-means Algorithm Initial Center Point Optimization Research under Spark Environment" (XJ2019G072), by the International Cooperation Project of the Science and Technology Department of the Autonomous Region "Data-Driven Construction of Sino-Russian Cloud Computing Sharing Platform" (2020E01023), and by the National Science Foundation of China under Grant U1803261.

Conflicts of Interest: The authors declare no conflict of interest. The funders helped reviewing the manuscript and provided the decision to publish the result.

References

1. Yang, Y. Resilient PNT Concept Frame. *J. Geod. Geoinf. Sin.* **2019**, *2*, 1–7.
2. Yang, Y. Concepts of comprehensive PNT and related key technologies. *Acta Geod. Cartogr. Sin.* **2016**, *45*, 505–510.
3. Cleve, C.; Kelly, M.; Kearns, F.R.; Moritz, M. Classification of the wildland–urban interface: A comparison of pixel-and object-based classifications using high-resolution aerial photography. *Comput. Environ. Urban Syst.* **2008**, *32*, 317–326. [CrossRef]
4. Mena, J.B. State of the art on automatic road extraction for GIS update: A novel classification. *Pattern Recognit. Lett.* **2003**, *24*, 3037–3058. [CrossRef]
5. Friedrich, T.; Oschlies, A. Neural network-based estimates of North Atlantic surface pCO2 from satellite data: A methodological study. *J. Geophys. Res. Ocean.* **2009**, *114*. [CrossRef]
6. Hall, R.; Skakun, R.; Arsenault, E.; Case, B. Modeling forest stand structure attributes using Landsat ETM+ data: Application to mapping of aboveground biomass and stand volume. *For. Ecol. Manag.* **2006**, *225*, 378–390. [CrossRef]
7. Schiffer, R.A.; Rossow, W.B. The International Satellite Cloud Climatology Project (ISCCP): The first project of the world climate research programme. *Bull. Am. Meteorol. Soc.* **1983**, *64*, 779–784. [CrossRef]
8. Price, J.C. Land surface temperature measurements from the split window channels of the NOAA 7 Advanced Very High Resolution Radiometer. *J. Geophys. Res. Atmos.* **1984**, *89*, 7231–7237. [CrossRef]
9. Li, W.; Li, D. The universal cloud detection algorithm of MODIS data. In Proceedings of the Geoinformatics 2006: Remotely Sensed Data and Information, Wuhan, China, 28–29 October 2006; Volume 6419, p. 64190F. [CrossRef]
10. Wu, X.; Cheng, Q. Study on methods of cloud identification and data recovery for MODIS data. In Proceedings of the Remote Sensing of Clouds and the Atmosphere XII, Florence, Italy, 17–20 September 2007; Volume 6745, p. 67450P. [CrossRef]

11. Ren, R.; Guo, S.; Gu, L.; Wang, L.; Wang, X. An effective method for the detection and removal of thin clouds from MODIS image. In Proceedings of the Satellite Data Compression, Communication, and Processing V, San Diego, CA, USA, 2–6 August 2009; Volume 7455, p. 74550Z. [CrossRef]
12. Solvsteen, C. Correlation-based cloud detection and an examination of the split-window method. In Proceedings of the Global Process Monitoring and Remote Sensing of the Ocean and Sea Ice, Paris, France, 25–28 September 1995; Volume 2586, pp. 86–97, [CrossRef]
13. Ping, Q.W.; Qing, L.W.; Guo, L.J.; Huai, L.Y.; Jun, Z.; Min, Q.; Cheng, L. Application of Single-Band Brightness Variance Ratio to the Interference Dissociation of Cloud for Satellite Data. *Spectrosc. Spectr. Anal.* **2006**, *26*, 2011–2015.
14. Shan, N.; Zheng, T.; Wang, Z. High-speed and high-accuracy algorithm for cloud detection and its application. *J. Remote Sens.* **2009**, *13*, 1138–1146.
15. Chen, P.; Zhang, R.; Liu, Z. Feature detection for cloud classification in remote sensing images. *J. Univ. Ence Technol. China* **2009**, *5*, 484–488.
16. Zhu, Z.; Woodcock, C.E. Object-based cloud and cloud shadow detection in Landsat imagery. *Remote Sens. Environ.* **2012**, *118*, 83–94. [CrossRef]
17. Zhu, Z.; Wang, S.; Woodcock, C.E. Improvement and expansion of the Fmask algorithm: Cloud, cloud shadow, and snow detection for Landsats 4–7, 8, and Sentinel 2 images. *Remote Sens. Environ.* **2015**, *159*, 269–277. [CrossRef]
18. Qiu, S.; Zhu, Z.; He, B. Fmask 4.0: Improved cloud and cloud shadow detection in Landsats 4–8 and Sentinel-2 imagery. *Remote Sens. Environ.* **2019**, *231*, 111205. [CrossRef]
19. Irish, R.R.; Barker, J.L.; Goward, S.N.; Arvidson, T. Characterization of the Landsat-7 ETM+ automated cloud-cover assessment (ACCA) algorithm. *Photogramm. Eng. Remote Sens.* **2006**, *72*, 1179–1188. [CrossRef]
20. Latry, C.; Panem, C.; Dejean, P. Cloud detection with SVM technique. In Proceedings of the 2007 IEEE International Geoscience and Remote Sensing Symposium, Barcelona, Spain, 23–28 July 2007; pp. 448–451.
21. Shelhamer, E.; Long, J.; Darrell, T. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 640–651. [CrossRef]
22. Mohajerani, S.; Krammer, T.A.; Saeedi, P. Cloud detection algorithm for remote sensing images using fully convolutional neural networks. *arXiv* **2018**, arXiv:1810.05782.
23. Mohajerani, S.; Saeedi, P. Cloud-Net: An end-to-end cloud detection algorithm for Landsat 8 imagery. In Proceedings of the IGARSS 2019—2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 1029–1032.
24. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef]
25. Segal-Rozenhaimer, M.; Li, A.; Das, K.; Chirayath, V. Cloud detection algorithm for multi-modal satellite imagery using convolutional neural-networks (CNN). *Remote Sens. Environ.* **2020**, *237*, 111446. [CrossRef]
26. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
27. Yan, Z.; Yan, M.; Sun, H.; Fu, K.; Hong, J.; Sun, J.; Zhang, Y.; Sun, X. Cloud and cloud shadow detection using multilevel feature fused segmentation network. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 1600–1604. [CrossRef]
28. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Cham, Switzerland, 2015; pp. 234–241.
29. Gonzales, C.; Sakla, W. *Semantic Segmentation of Clouds in Satellite Imagery Using Deep Pre-trained U-Nets*; Technical Report; Lawrence Livermore National Lab. (LLNL): Livermore, CA, USA, 2019.
30. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
31. Guo, Y.; Cao, X.; Liu, B.; Gao, M. Cloud Detection for Satellite Imagery Using Attention-Based U-Net Convolutional Neural Network. *Symmetry* **2020**, *12*, 1056. [CrossRef]
32. Duta, I.C.; Liu, L.; Zhu, F.; Shao, L. Pyramidal Convolution: Rethinking Convolutional Neural Networks for Visual Recognition. *arXiv* **2020**, arXiv:2006.11538.
33. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11534–11542.
34. Wang, J.; Chen, K.; Xu, R.; Liu, Z.; Loy, C.C.; Lin, D. Carafe: Content-aware reassembly of features. In Proceedings of the IEEE International Conference on Computer Vision, Long Beach, CA, USA, 16–20 June 2019; pp. 3007–3016.
35. Roy, D.P.; Wulder, M.A.; Loveland, T.R.; Woodcock, C.; Allen, R.G.; Anderson, M.C.; Helder, D.; Irons, J.R.; Johnson, D.M.; Kennedy, R.; et al. Landsat-8: Science and product vision for terrestrial global change research. *Remote Sens. Environ.* **2014**, *145*, 154–172. [CrossRef]