remote sensing

MDPI

Article

A Hybrid Algorithm with Swin Transformer and Convolution for Cloud Detection

Chengjuan Gong [1,2], Tengfei Long [1,*], Ranyu Yin [1], Weili Jiao [1] and Guizhou Wang [1]

[1] Aerospace Information Research Institute (AIR), Chinese Academy of Sciences (CAS), Beijing 100094, China; gongcj@aircas.ac.cn (C.G.); yinry@aircas.ac.cn (R.Y.); jiaowl@aircas.ac.cn (W.J.); wanggz@aircas.ac.cn (G.W.)
[2] University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: longtf@aircas.ac.cn

Abstract: Cloud detection is critical in remote sensing image processing, and convolutional neural networks (CNNs) have significantly advanced this field. However, traditional CNNs primarily focus on extracting local features, which can be challenging for cloud detection due to the variability in the size, shape, and boundaries of clouds. To address this limitation, we propose a hybrid Swin transformer–CNN cloud detection (STCCD) network that combines the strengths of both architectures. The STCCD network employs a novel dual-stream encoder that integrates Swin transformer and CNN blocks. Swin transformers can capture global context features more effectively than traditional CNNs, while CNNs excel at extracting local features. The two streams are fused via a fusion coupling module (FCM) to produce a richer representation of the input image. To further enhance the network's ability in extracting cloud features, we incorporate a feature fusion module based on the attention mechanism (FFMAM) and an aggregation multiscale feature module (AMSFM). The FFMAM selectively merges global and local features based on their importance, while the AMSFM aggregates feature maps from different spatial scales to obtain a more comprehensive representation of the cloud mask. We evaluated the STCCD network on three challenging cloud detection datasets (GF1-WHU, SPARCS, and AIR-CD), as well as the L8-Biome dataset to assess its generalization capability. The results show that the STCCD network outperformed other state-of-the-art methods on all datasets. Notably, the STCCD model, trained on only four bands (visible and near-infrared) of the GF1-WHU dataset, outperformed the official Landsat-8 Fmask algorithm in the L8-Biome dataset, which uses additional bands (shortwave infrared, cirrus, and thermal).

Keywords: Swin transformer; cloud detection; image segmentation; attention; convolution

Citation: Gong, C.; Long, T.; Yin, R.; Jiao, W.; Wang, G. A Hybrid Algorithm with Swin Transformer and Convolution for Cloud Detection. *Remote Sens.* **2023**, *15*, 5264. https://doi.org/10.3390/rs15215264

Academic Editor: Filomena Romano

Received: 2 October 2023
Revised: 2 November 2023
Accepted: 2 November 2023
Published: 6 November 2023

1. Introduction

Remote sensing imagery has indeed been widely used for land cover detection, land surface change monitoring, and the estimation of biophysical parameters [1,2]. While the presence of clouds contributes to providing helpful information for weather forecasting and climate prediction [3,4], the cloud cover not only diminishes the quality of the optical remote sensing data, but also intensifies the complexity of subsequent data processing and downstream remote sensing analysis [5]. And the diverse types of clouds and intricate ground surfaces make it challenging to accurately distinguish between clouds and mixed ground objects. Consequently, the development of automatic cloud detection has become increasingly pivotal and crucial in the preprocessing of optical satellite remote sensing images, thereby significantly improving their utilization.

Recently, a wide range of techniques for identifying clouds have been put forward. Rule-based algorithms extract clouds based on significant disparities in both the physical characteristics and spatial characteristics between the clouds and most ground surfaces [6–9]. Using the ACCA algorithm [10], the authors applied several spectral filters and employed 32 fixed thresholds and three dynamic thresholds to estimate the overall percentage of

clouds in each Landsat 7 scene. The authors in [11] utilized 26 fixed thresholds according to different features to delineate cloud and cloud shadow regions. The spectral indices, namely, the method cloud index (CI) and cloud shadow index (CSI), were introduced in [12] to identify the clouds and cloud shadow regions using threshold segmentation. In summary, threshold-based algorithms are simple and effective, thereby making them well suited for small scenes or situations without complicated objects. However, it is essential to note that climate and ground surface conditions vary over time and space, and fixed thresholds may not be universally applicable to all areas and time frames, even for data from the same sensor. Additionally, adaptive thresholds are difficult to determine, especially in cloud-like features such as deserts, fog, haze, and ice/snow cover.

Furthermore, traditional machine learning algorithms are frequently employed in cloud detection. Those methods generate the cloud masks by carefully selecting relevant features and choosing an effective model. The authors in [13] proposed an algorithm that combines k-means clustering and random forest for cloud detection in Landsat images, which had better results than FMask [14]. Researchers applied the support vector machine (SVM) for cloud detection based on comparative analysis of the feature differences between clouds and backgrounds, and they verified the method using GF-1 and GF-2 satellite images [15]. An end-to-end PCANet cloud detection for Landsat 8 images obtained the cloud masks by employing an SVM based on the superpixels. Additionally, the boundaries of the clouds were refined through the fully connected conditional random field (CRF) [16]. Traditional machine learning algorithms have obtained promising results through statistical analysis. However, they heavily rely on artificial features or human-specified rules, thereby making it difficult to design a universal template to handle the diversity of cloud types.

With the rapid and extensive advancement of deep learning, it has been widely used in various industries. Deep learning-based methods can automatically extract data features from data and achieve remarkable results without the need for manual feature selection [17–22]. Numerous approaches based on deep learning have been proposed for cloud detection [23–30]. For example, DANet [31] utilizes space and channel attention to obtain the semantic interdependencies in different dimensions. CCNet [32] designs a crisscross attention module to harvest all pixels' contextual information using their respective crisscross paths. CSD-Net [33] composes the multiscale global attention feature fusion module and channel attention mechanism to refine the edges of cloud and cloud shadow masks. CSD-HFnet [34] combines the fundamental features, obtained through the local binary pattern, gray-level co-occurrence matrix, superpixel segmentation, and the deep semantic features, which are acquired form deep learning feature extraction network to distinguish the clouds from snow. BABFNet [35] introduces a boundary prediction branch to enhance the cloud detection results in confusing areas. CDUNet [36] uses a high-frequency feature extractor and multiscale convolutions to predict cloud masks. MAFANet [29] combines a multiscale strip pooling attention module, a multihead attention module, and a feature fusion module to acquire more accurate cloud and cloud shadow masks.

Nevertheless, CNN-based cloud detection models still have limitations due to the diversity of cloud forms and the difficulty of learning long-range dependencies using convolution operations. Recent advances in vision transformer (ViT) [37] technology have demonstrated its ability to learn long-term features and model global information effectively, thus resulting in satisfactory performance on image classification tasks [38]. To overcome the high memory demand of transformers, the Swin transformer [39] designs a hierarchical transformer to limit self-attention computation within nonoverlapping windows. Using the Swin transformer as the backbone, Swin-Unet [40] and TransDeepLab [38] have outperformed other methods in medical image segmentation. To leverage the complementary strengths of CNNs and ViTs, researchers have proposed hybrid models that combine convolution and transformer architectures to extract both local and global features for image classification tasks [41–43]. He et al. [44] integrated the global dependencies of the Swin transformer into the features from the UNet's encoder for remote sensing image

segmentation. BuildFormer [45] designs a dual-path structure with a CNN and transformer to extract the spatial details and global context for building segmentation. Yuan et al. [46] proposed the LiteST-Net model to extract building data from remote sensing images, which creatively simplifies the matrices Q, K, and V of the transformer to decrease the model computation. Alrfou et al. [47] concatenated the feature maps of CNN and Swin transformer encoders as the input into the corresponding decoder, and it has been demonstrated that combining transformers and CNN encoders consistently outperforms using CNN encoders along with image segmentation.

To enhance the capability of obtaining precise cloud boundaries and efficiently discerning clouds from bright ground objects, this paper proposes a novel network with an encoder–decoder structure for cloud detection, which has been named STCCD (a hybrid Swin transformer–CNN cloud detection network). The STCCD network has a parallel hybrid encoder that combines the Swin transformer layers and convolution blocks. Within this encoder, the feature coupling module (FCM) interacts with features from the Swin transformer layer and residual convolution blocks to allow the encoder to effectively learn the global representations while also capturing local features. Additionally, the feature fusion module based on the attention mechanism is designed to fuse the feature maps outputted from the Swin transformer and convolution branches to explore the relationships between channels. Next, we introduce an aggregation multiscale feature module (AMSFM) to extract multiscale features, which equips our network with the ability to recognize clouds at different scales. In the decoder part, the first four layers take three inputs: the output of the upper layer, the outputs of the corresponding residual layers, and the Swin transformer layer. Finally, a boundary refinement module (BRM) is utilized to capture edge details and optimize the result of cloud detection.

The main contributions of this paper are as follows. Firstly, we propose a novel cloud detection framework, the STCCD network, with an encoder–decoder architecture that leverages a combination of Swin transformer layers and residual convolution blocks to obtain both global representations and local features. Secondly, the STCCD network also includes two novel modules, FFMAM and AMSFM, which exploit the interplay between various network levels and the characteristics of cloud pixels. The STCCD netowork achieves state-of-the-art performance on cloud detection benchmarks, as evidenced by its superior results on the GF1-WHU, SPARCS, AIR-CCD, and L8-Biome datasets.

2. Methodology

The overall architecture of the STCCD network is presented in Figure 1, which is an encoder–decoder structure integrated by several feature extraction modules. In the encoder, the first two layers, composed of convolution blocks, are used to extract fine-grained information such as the spectral and texture features. Subsequently, the features extracted from each Swin transformer (ST) layer are processed by the FCM and then concatenated with the features achieved from the previous convolution layer. Likewise, the outputs of each residual convolution layer are also processed by the FCM and concatenated with the features from the previous ST layer. Overall, the convolution layers and the ST layers are paralleled and staggered. The FCM plays a pivotal role in eliminating the misalignment between the two branches, thereby effectively enabling global representations to interact with local features. Subsequently, the FFMAM uses a multihead spatial attention mechanism to fuse the outputs of these two branches, and a channel attention mechanism is utilized to enhance the communication between different channels. Following that, this paper introduces an AMSFM to capture multiscale contextual information. In the decoder, each decoder layer is composed of a basic convolution block and upsampling operation, and it fuses the low-level and high-level features. Finally, the BRM is utilized behind the decoder to capture intricate details and enhance the representations of cloud boundaries.

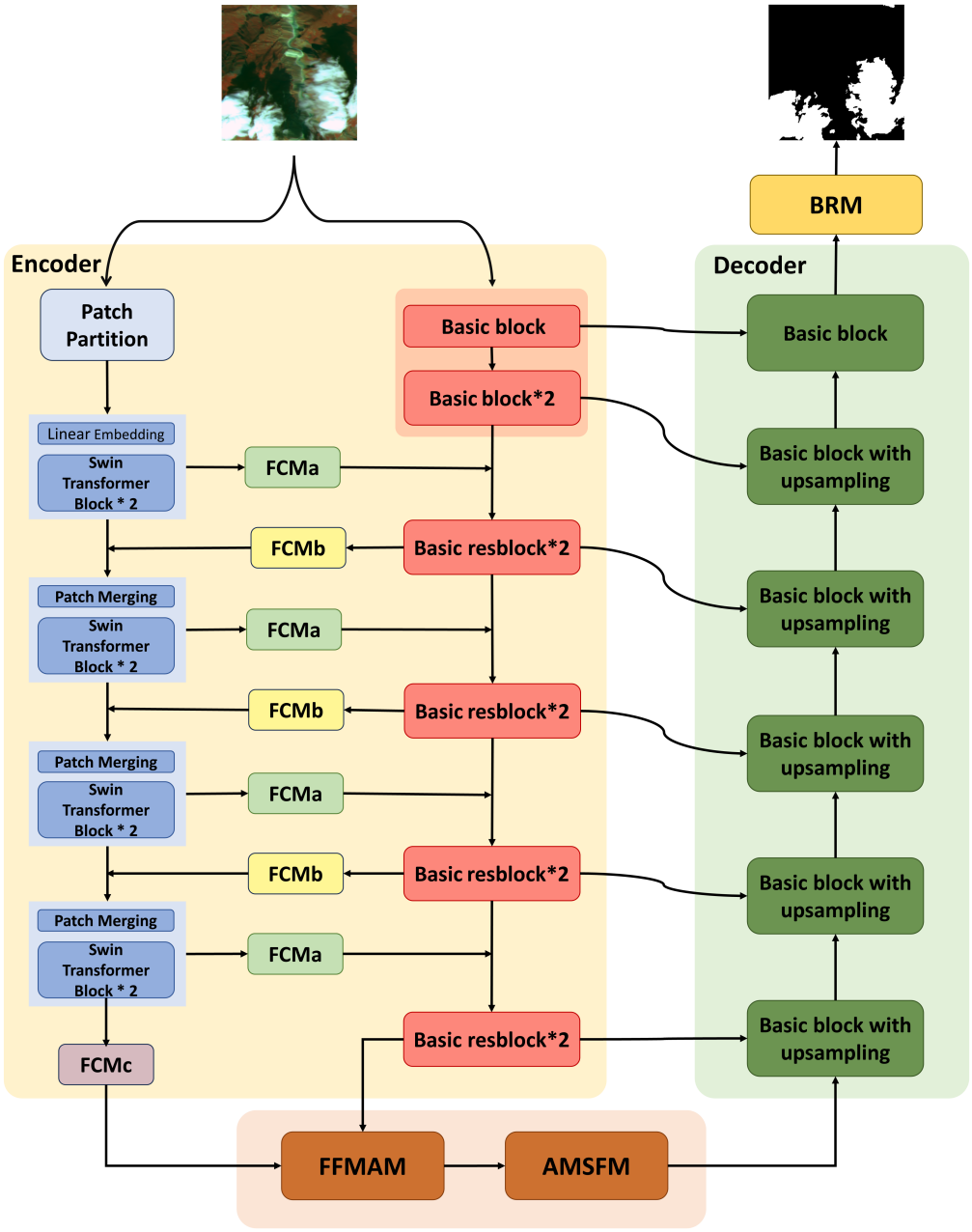

Figure 1. The structure of the STCCD network. The framework primarily consists of four components: encoder, decoder, bridge, and output. Two branches of Swin transformer layers and convolution layers, as well the FCM, are in the encoder stage, and the bridge stage includes FFMAM and AMSFM; there are six convolution blocks and five upsampling interaction layers in the decoder stage. Finally, the BRM is a simple boundary-refined module.

2.1. Convolution Branch

The convolution branch adopts a feature pyramid structure, in which the resolution of the feature maps decreases with network depth while the channel number increases. This branch has six layers. The initial two layers consist of different numbers of basic convolution blocks. The next four layers each contain two basic residual blocks, thus following the architecture of ResNet18 [48]. The structures of the basic convolution block and basic residual block are illustrated in Figure 2. Among these layers, max pooling is employed to decrease the spatial resolution.

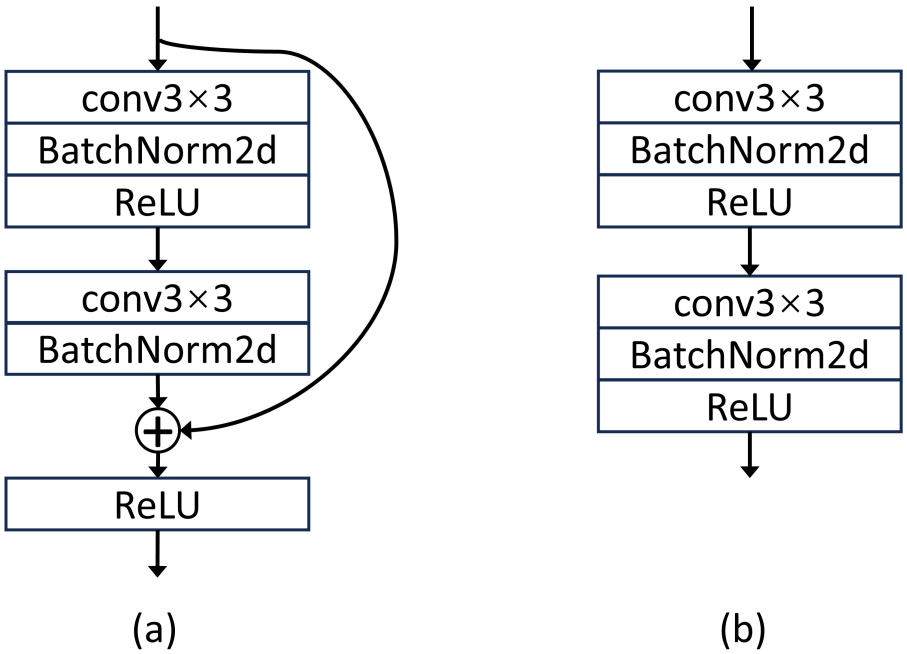

Figure 2. The structure of the basic residual block (**a**) and the basic convolution block (**b**).

2.2. Swin Transformer Branch

Liu et al. [39] proposed a hierarchical vision transformer based on a sliding window mechanism (Swin transformer). To decrease the computation complexity, Swin transformer calculates the multihead self-attention within local windows (W-MSA) and the shifted windows (SW-MSA), thereby evenly dividing the image without overlapping. Furthermore, in order to enhance the connections across windows while preserving the efficient computation of regular windows, Swin transformer alternately uses a regular window configuration and a shifted window configuration in consecutive Swin transformer blocks. As shown in Figure 3, the components of a Swin transformer block include a window based on an MSA module (W-MSA or SW-MSA), a two-layer MLP, and two LayerNorm layers. The process of the consecutive Swin transformer blocks is represented as follows:

$$\begin{aligned} \hat{z}^{l} &= W\text{-}MSA(LN(z^{l-1})) + z^{l-1} \\ z^{l} &= MLP(LN(\hat{z}^{l})) + \hat{z}^{l} \\ \hat{z}^{l+1} &= SW\text{-}MSA(LN(z^{l})) + z^{l} \\ z^{l+1} &= MLP(LN(\hat{z}^{l+1})) + \hat{z}^{l+1} \end{aligned} \tag{1}$$

where $\hat{z}^{l}$ and z^{l} represent the output features of the W-MSA and the MLP module, respectively, for block l, and $\hat{z}^{l+1}$ and z^{l+1} denote the output features of the SW-MSA and the MLP module, respectively, for block $l+1$.

In our Swin transformer branch, there are four stages, and each stage includes two consecutive ST blocks. As Figure 1 shows, starting from the second stage of the Swin transformer branch, the input feature maps are the concatenated features of the output features of the previous stage and the corresponding layers of the convolution branch. So, the linear layer in patch merging is applied to the 8C-dimensional concatenated features, with the output dimension remaining set at 2C, where C denotes the feature dimension after the process of the linear embedding layer.

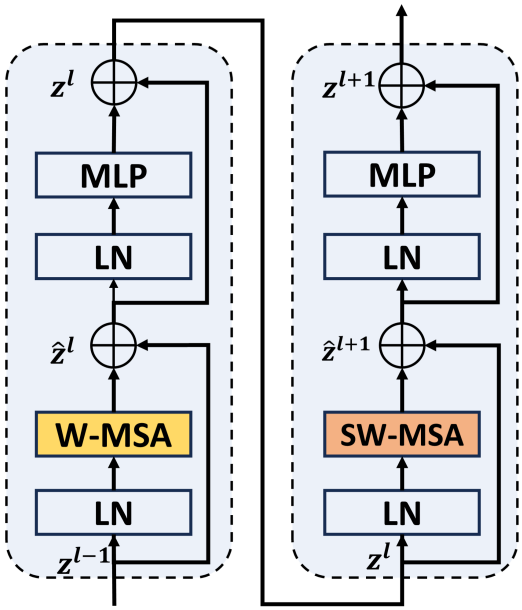

Figure 3. Two consecutive Swin transformer blocks.

2.3. Feature Coupling Module

The feature coupling module is designed to eliminate the misalignment between the convolution branch and the Swin transformer branch. The feature sizes of the convolution layer and Swin transformer layer are different. The feature maps of the convolution block have the size $B \times C \times H \times W$ (B, C, H, and W are the batch size, channels, height, and width, respectively), while the shape of the feature maps from the Swin transformer block is $B \times L \times C$ (B, L, and C are the batch size, number of tokens, and dimension, respectively). When fed to the convolution branch, the feature maps outputted from the Swin transformer layer must be upsampled to the same spatial scale. Next, the channel dimension is processed through a 1×1 convolution operation to align with that of the corresponding convolution layer; then, the resulting features are concatenated with those of the convolution layer. Those processes are shown in Figure 4a. When transitioning from the convolution branch back to the Swin transformer branch, the feature maps first undergo a 1×1 convolution to match the channel dimension. Afterward, the feature array is flattened to ensure that it has the same dimensionality as the feature maps from the Swin transformer layer, as depicted in Figure 4b. In the end, the spatial scale of the feature maps outputted from the Swin transformer branch is the same as that outputted from the convolution branch so that, as shown in Figure 4c, the process of the FCM(c) is identical to that of the FCM(a), with the sole difference being the absence of interpolating.

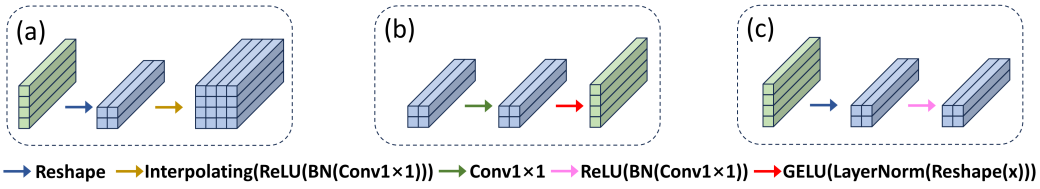

Figure 4. The operation process of different FCMs. BN denotes the BatchNorm operation.

2.4. Feature Fusion Module Based on Attention Mechanism

The attention mechanism has been extensively exploited in semantic segmentation because it enables the network to automatically learn and selectively focus on the critical information in the input, thereby improving the model's performance and generalization ability [18,49]. The self-attention mechanism works by calculating the relative importance and establishing an association between one pixel and all the other pixels, rather than just

relying on elements in adjacent positions, which aids in effectively capturing the long-term dependencies between pixels [50]. The multihead attention mechanism is developed on the basis of self-attention, which enhances the expressiveness and generalization ability of the model [51]. The channel attention mechanism operates by assessing the importance of each channel, and it generates more representative features. EcaNet [52] presents an efficient and lightweight correlation channel module, which is composed of a one-dimensional convolution determined by nonlinear adaptive control.

Considering the advantages of the attention mechanism, and inspired by the multihead feed-forward transfer attention module [29] and the boundary-guided context aggregation module [51]—which utilizes the multihead attention mechanism to fuse the feature maps of different convolution layers—we have designed the feature fusion module based on the attention mechanism (FFMAM) to promote the mutual guidance of two branches, integrate the features extracted, and explore the relationship between the channels. As the left of Figure 5 shows, the feature tensor $X \in R^{C\times H\times W}$, derived from the convolution branch, is used to generate the key vector (Key) and the value vector (Value) through different reshape modes; meanwhile, the feature tensor $Y \in R^{C\times H\times W}$, acquired from the Swin transformer branch, is employed for generating the query vector (Query). These vectors undergo processing via a 1×1 convolution layer and a batch normalization layer to consolidate the pixel-level crosschannel context information. The created Q, K, and V are shown as follows:

$$Q = BN(Conv_{1\times1}(Y)) \tag{2}$$

$$K = BN(Conv_{1\times1}(X)) \tag{3}$$

$$V = BN(Conv_{1\times1}(X)) \tag{4}$$

where $Conv_{1\times1}$ denotes a two-dimensional convolution with a kernel of 1×1, and BN denotes the batch normalization layer. Subsequently, the feature vectors Query and Key are reshaped to the size $R^{C\times L}$ and $R^{L\times C}$, respectively, where $L = H \times W$ represents the number of pixels. Matrix multiplication is performed between the Query and Key to produce a transposed attention graph, and then a Softmax function is applied. The above processes are shown in the following:

$$F' = V^{'} \cdot Softmax(K^{'} \cdot Q^{'}) \tag{5}$$

where F' is the output characteristic graph, and $V^{'}$, $Q^{'}$, and $K^{'}$ denote the reshaped vectors. After this, an efficient and lightweight channel attention module is applied to capture the local crosschannel interaction, as illustrated in the right side of Figure 5. Initially, the attention feature tensors are processed by a global average pooling (GAP). Subsequently, a 1×1 convolution operation with a kernel size of k is used to make all channels share the same learning parameters, which is followed by a sigmoid function. Finally, the output of the sigmoid function is multiplied by the attention feature tensors. The process can be expressed by the following formula:

$$GAP(x) = \frac{1}{H \times W} \sum_{i=1,j=1}^{H,W} x_{i,j} \tag{6}$$

$$E^{'} = x * (Sigmoid(Conv_{1\times1}(GAP(x)))).expand_as(x)) \tag{7}$$

where $E^{'}$ is the output characteristic graph, and the kernel size k of 1×1 convolution layer is determined by the following:

$$k = |\frac{log_2(C)}{\gamma} + \frac{b}{\gamma}|_{odd} \tag{8}$$

where C is the number of the channel dimension, $|n|_{odd}$ indices the neatest odd number of n. The parameters γ and b are set to two and one, respectively, by referring to [52]. In the experiment, the C of the input came out to be 512, so the k was set to five.

The FFMAM facilitates mutual guidance between the two branches for feature extraction, thereby enabling the integration of global and local feature information while investigating interchannel relationships.

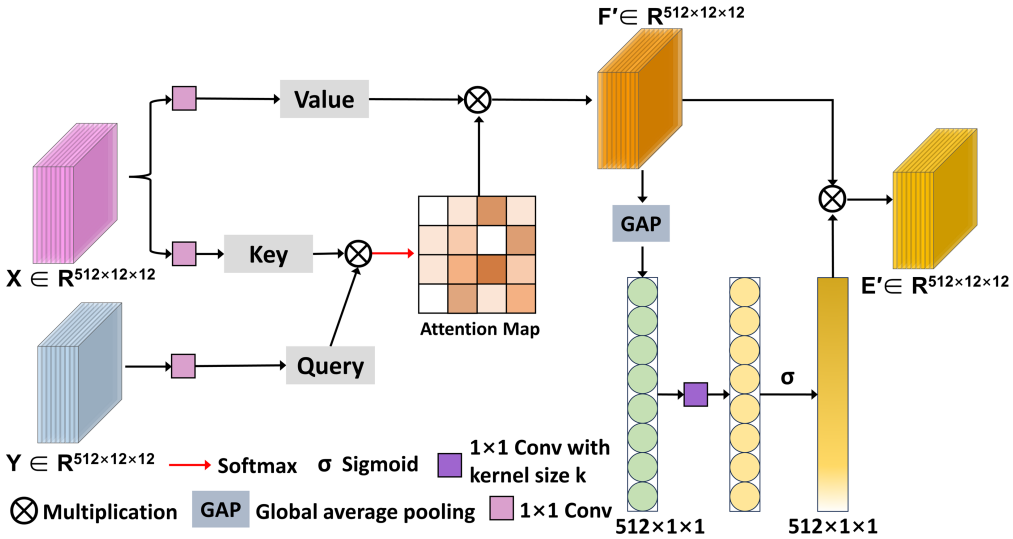

Figure 5. The structure of the FFMAM module. The feature tensor X is derived from the convolution branch, and the feature tensor Y is acquired from the Swin transformer branch. The feature tensor F' is the output characteristic graph of the multihead attention mechanism part, and the feature tensor E' is the output characteristic graph of the channel attention mechanism part.

2.5. Aggregation Multiscale Feature Module

Capturing representative features at multiple scales plays a crucial role in numerous visual tasks [53]. Dilation convolution [54] and the pyramid structure [55] are widely adopted to augment semantic information. Dilation convolution can effectively capture long-distance correlations, which are beneficial to the segmentation of large objects. Pooling is frequently utilized to acquire feature maps at different scales, yet it may have poor, limited effectiveness when dealing with scattered small-scale objects. The use of large square kernels in dilation convolution and pooling operations often results in the extraction of excessive information from irrelevant regions, thereby leading to the loss of fine-grained details. In response to these problems, the Atrous Spatial Pyramid Pooling (ASPP) module proposed in DeepLabv3 [56] is a spatial pyramid module composed of dilation convolution with different rates and an average pooling. Zhu et al. [57] added the attention mechanism in DeepLabv3+ [58] to detect foreign object debris on an airport runway. Additionally, in the task of cloud detection, where clouds can display a wide range of shapes and frequently have indistinct boundaries, the acquisition of more effective features becomes a critical consideration.

We designed the AMSFM to capture characteristics across various spatial scales in cloud detection. Figure 6 illustrates the structure of the AMSFM, which comprises four dilation convolution layers with different dilation rates, two spatial pooling operations, and a spatial attention mechanism. Four dilation convolution layers, as shown in Figure 7, facilitate the extraction of feature maps with varying receptive fields. Meanwhile, the pooling layers, global average pooling, and global max pooling, are used to obtain the average value and max value of each channel, respectively, which help to reserve essential semantic features. Then, the feature maps outputted from the dilation convolution layers and pooling layers are concatenated and operated by 1×1 convolution to produce the multiscale feature maps $X \in R^{512\times12\times12}$. The spatial attention mechanism is used to

generate the feature maps with enhanced spatial information. Initially, the max pool (MP) and the average pool (AP) are employed to obtain the maximum and average values, respectively, for each spatial position along the channel dimension. Subsequently, we concatenate the feature maps acquired from both pooling operations along the channel dimension. Following this, a convolution operation with a kernel size of 7×7 is performed to decrease the number of channels from two to one. Finally, the spatial weight feature maps of each spatial location generated by a nonlinear activation function (sigmoid) is used to multiply with the input feature maps to get the final feature map $Y \in R^{512\times12\times12}$. The calculation process can be summarized as follows:

$$SA(x) = x*(Sigmoid(Conv_{7\times7}(Concat(AP(x), MP(x))))) \quad (9)$$

where $SA(x)$ is the result of the space attention module, $Conv_{7\times7}$ denotes a two-dimensional convolution with the kernel size of 7×7, and $Concat$ represents the connect operation. At the end of the AMSFM, the results of the aforementioned feature maps X and Y are concatenated along the channel dimension and operated through a 1×1 convolution to form the input features for the decoder part.

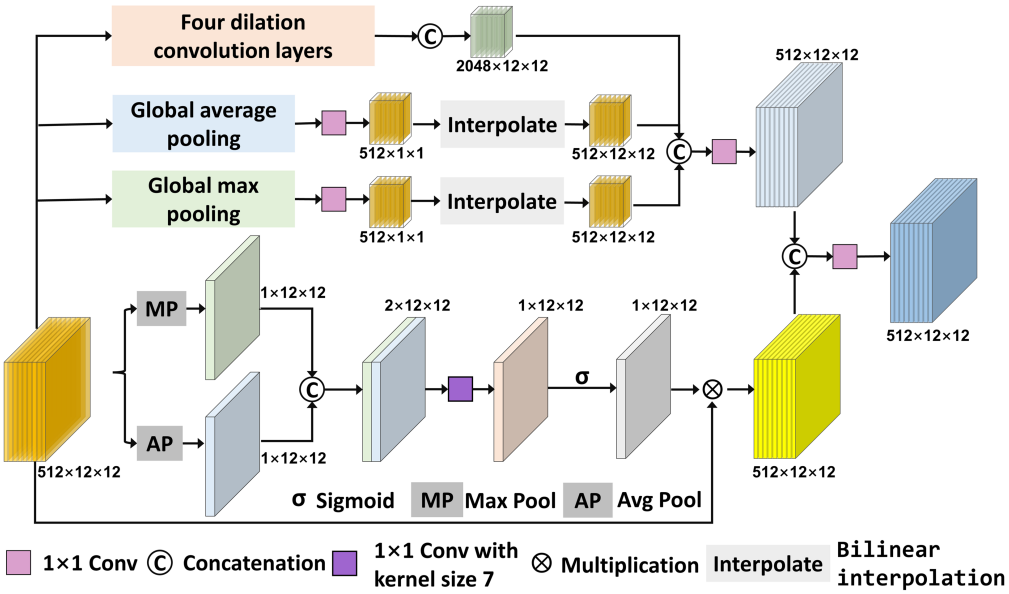

Figure 6. The structure of AMSFM.

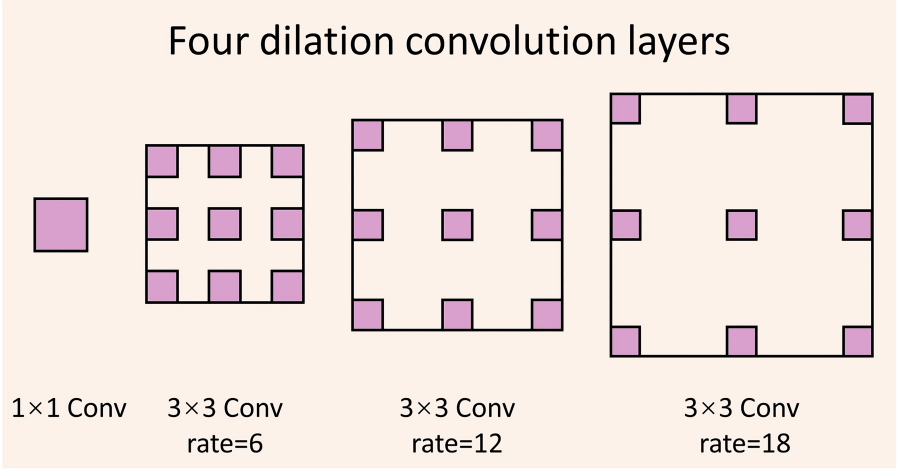

Figure 7. Four dilation convolution layers in parallel.

2.6. Boundary Refinement Module

Boundary features are significant in segmentation tasks, as the delineation of regions and boundaries are mutually determined. Presently, many researchers utilize the boundary features to assist segmentation. BANet [59] is a boundary-aware segmentation network that concatenates three streams, including a boundary localization stream, an interior perception stream, and a transition compensation stream to form a boundary-aware feature mosaic map. A multiscale boundaries extractor was proposed in the BCANet [51], which is an independent protocol used to predict the binary boundary of an image. Given the variability in the cloud shapes and the potential confusion between thin cloud boundaries and ground objects, the extraction of the boundary information assumes significant importance in cloud detection.

In this paper, we still utilized a simple encoder–decoder module to process the output of the decoder part to refine the cloud boundary. Subsequently, we combined the refined boundary information to the decoder's output to produce the final cloud mask by numerical addition. Figure 8 shows the structure of our boundary refinement module. There are three parts in the BRM: an encoder, a decoder, and a transition layer. The BRM operation serves to enhance boundary features and capture fine details, thus effectively mitigating the potential blurring of the boundaries of cloud masks. In summary, the boundary-refined architecture provides a simple yet powerful result in cloud detection.

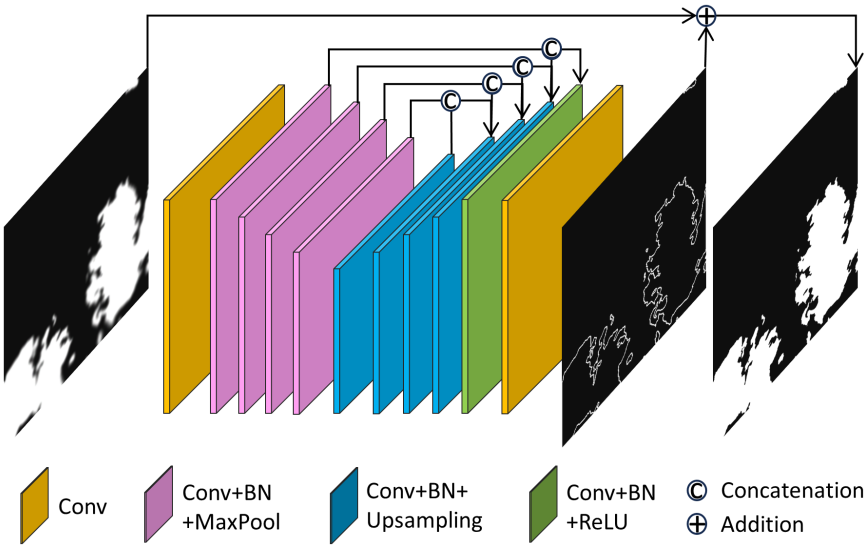

Figure 8. The structure of boundary refinement module.

3. Experiment

3.1. Datasets

3.1.1. GF1-WHU Dataset

The GF1-WHU dataset is constructed based on GF-1 WFV (wide field view) imagery to verify the performer of the MFC algorithm [11]. It contains 108 Level-2A scenes collected from 2013 to 2016 with different geomorphic environments and varying cloud conditions. In this dataset, the approximate size of an image is $17,000 \times 16,000$ pixels, and the spatial resolution is 16 m. Each image has four multispectral bands, including visible and near-infrared bands. In our experiment, we randomly selected 66 training images and 10 validating images. The 32 leftover images from the year 2013 were excluded due to abnormal top of atmosphere (TOA) reflectance values, which had been calibrated using the official parameters and fell outside the range of [0, 1].

3.1.2. SPRACS Dataset

The SPARCS dataset [60,61] was created by M. Joseph Hughes of Oregon State University and was delineated manually based on Landsat 8 images to verify the performance

of the Spatial Program for Automatic Cloud and Shadow Removal (SPARCS) method. There are 80 images with 1000 × 1000 pixels in this dataset, and it contains six categories, including cloud, cloud shadow, snow/ice, water, land, and flooded. The spatial resolution is 30 m, and each image has ten bands. In this paper, 70 images were selected randomly as the training data, and the remaining ten images were used as the verification data. In the training process, only visible and near-infrared bands were used.

3.1.3. AIR-CD Dataset

The AIR-CD dataset [62] is a publicly available cloud detection dataset with a high spatial resolution, which contains 34 GF-2 images from different regions across China. This dataset poses considerable challenges due to its complex and diverse backgrounds, thereby encompassing urban areas, snow-covered regions, forests, and bare lands. Those 34 images were collected from the PMS1 and PMS2 sensors of the GF-2 satellite imaging system, and each image has visible and near-infrared bands. Furthermore, these images have a spatial resolution of 4 m and a size of 7300 × 6908 pixels. In the experiment, we randomly selected 27 images as the training data, and the remaining 7 images were the test data.

3.1.4. L8-Biome Dataset

USGS EROS createed the Landsat 8 cloud cover validation dataset named L8-Biome [63,64]. The L8-Biome dataset comprises 96 Landsat 8 OLI/TIRS terrain-corrected scenes from various global locations. Among these, 64 images are labeled as containing only clouds, while 32 images are labeled as containing both clouds and cloud shadows. There are four classes in the L8-Biome dataset, i.e., clear, cloud shadow, thin cloud, and cloud. We used this dataset to verify the generalization ability of our method in the extended experiment.

The images we used in the GF1-WHU, SPARCS, and L8-Biome datasets are distributed in global regions, which are shown in Figure 9. The details of those datasets are listed in Table 1.

Table 1. The information about the datasets we used.

Dataset Name	Number	Resolution	Size	References
GF1-WHU	76	16 m	17,000 × 16,000	Li et al., 2017 [11]
SPARCS	80	30 m	1000 × 1000	Hughes and Hayes, 2014; USGS., 2016c [60,61]
AIR-CD	34	4 m	7300 × 6908	He et al., 2021 [62]
L8-Biome	96	30 m	7500 × 7300	Foga et al., 2017; USGS., 2016b [63,64]

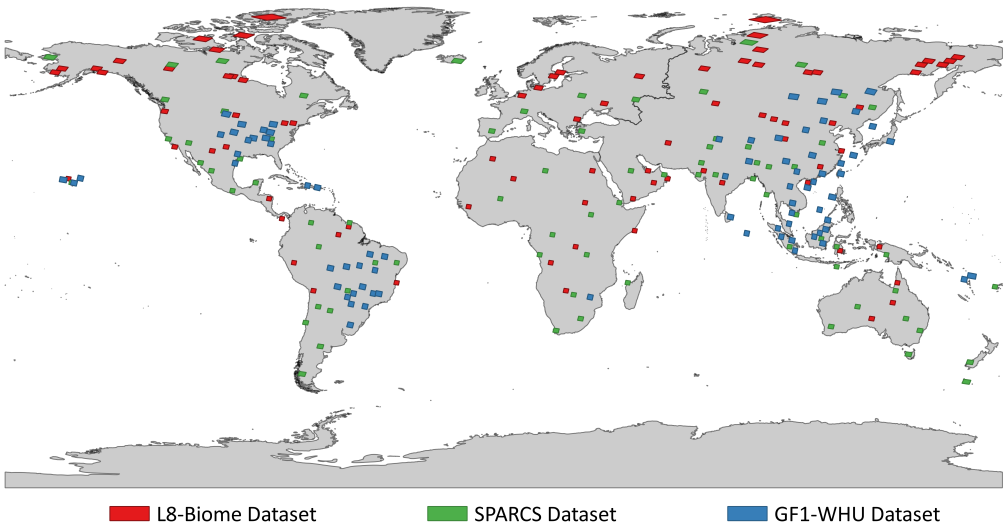

Figure 9. Global distribution of the datasets, including GF1-WHU, SPARCS, and L8-Biome. The AIR-CD dataset is not shown due to lack of geolocation information.

3.2. Training Details

This experiment was implemented using the PyTorch deep learning framework [65] on an NVIDIA A100 GPU with 40G of memory. In this work, the adaptive motion estimation (Adam) optimizer [66] was used, with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The initial learning rate was 10^{-4}, and a total of 200 epochs were trained.

3.2.1. Data Processing

We used a randomly cropped tool from the Albumentations library to partition each image of these datasets into 384×384 image patches. Due to the variation in image sizes across datasets, we selected 160 patches per image in the GF-WHU dataset, 40 patches per image in the SPARCS dataset, and 100 patches per image in the AIR-CD dataset.

In order to improve the generalization of our network, we used the data enhancement tool from the Albumentations library to perform brightness contrast changes, blurring, and flips with a probability of 0.5.

3.2.2. Loss Function

To achieve high-quality cloud detection results, we used a hybrid loss defined as the sum of the binary crossentropy loss and the intersection over union loss. This loss function learns the difference between the true and predicted values of the cloud pixels. The calculation formula is as follows:

$$\ell_h = \lambda_1 \ell_{bca} + \lambda_2 \ell_{IoU} \tag{10}$$

where the λ_1 and λ_2 represent the weights of the corresponding loss function, and ℓ_{bca} and ℓ_{IoU} denote the BCE loss [67] and IoU loss [68], respectively.

The BCE loss is most widely used in binary segmentation, which is calculated in a pixelwise manner. The formula of the BCE loss is shown as follows:

$$\ell_{bce} = -\sum_{(i,j)} W[G(i,j)log(S(i,j)) + (1 - G(i,j))log(1 - S(i,j))] \tag{11}$$

where $G(i,j)$ is zero or one and is the ground truth label of the pixel (i,j)—zero is the noncloud pixel and one denotes the cloud pixel—and $S(i,j) \in (0,1)$ is the predicted probability of the cloud mask. W is the weight given to the loss of each element.

The IoU loss is a map-level measure used in image segmentation and object detection, which quantifies the ratio of the intersection of the true and predicted areas to their union. The IoU loss is given as follows:

$$\ell_{IoU} = 1 - \frac{\sum_{i=1}^{H}\sum_{j=1}^{W} S(i,j)G(i,j)}{\sum_{i=1}^{H}\sum_{j=1}^{W}[S(i,j) + G(i,j) - S(i,j)G(i,j)]} \tag{12}$$

where $G(i,j)$ is zero or one and is the ground truth label of the pixel (i,j)—zero is the noncloud pixel 1 denotes the cloud pixel—and $S(i,j) \in (0,1)$ is the predicted probability of the cloud mask.

In the decoder part, the output of each block undergoes processing through a 1×1 convolution operation followed by a bilinear interpolation layer to generate an interim cloud mask. As shown in Figure 1, there are six blocks in the decoder, and counting the output of BRM, our segmentation model is deeply supervised, thus resulting in seven outputs. Therefore, the overall training loss function is computed by summing the losses from all the blocks as follows:

$$\ell_{overall} = \sum_{n=1}^{N} \ell_h^n \tag{13}$$

where $N = 7$ denotes that there are seven outputs of this network, and ℓ_h^n denotes the loss of the nth block output.

3.2.3. Evaluation Metrics

In order to evaluate the performance of this network in the cloud segmentation task, the overall accuracy (OA), mean intersection over union (MIoU), and F1 score of the cloud were chosen as the evaluating indices. The F1 score is a harmonic mean of the precision and recall, and a higher F1 score indicates greater robustness of the model. Those indicators are calculated based on the confusion matrix, which includes the true positive (TP), true negative (TN), false positive (FP), and false negative (FN), and their calculation formulas are listed as follows:

$$OA = \frac{TP + TN}{TP + FN + FP + TN} \tag{14}$$

$$F1_{cloud} = 2 \times \frac{precision \times recall}{precision + recall} \tag{15}$$

$$MIoU = [\frac{TP}{TP + FP + FN} + \frac{TN}{TN + FP + FN}]/2 \tag{16}$$

where $precision = \frac{TP}{TP+FP}$ and $recall = \frac{TP}{TP+FN}$, $F1_{cloud}$ represent the F1 score of cloud. The MIoU represents the average of the intersection over union from two categories (cloud and noncloud).

3.3. Ablation Study

To ascertain the validity and necessity of each module, we conducted an ablation study using the GF1-WHU dataset. The base model is established as a U-shaped architecture based on ResNet18. To verify the role of the FCM, we conducted experiments using a dual-branch encoder composed of a convolution and Swin running in parallel but without interaction. Table 2 illustrates the incremental accuracy improvements achieved through the gradual addition of individual modules.

Table 2. Evaluation results with the proposed modules on GF1-WHU dataset.

Method	MIoU (%)	F1_cloud (%)
Base	91.14	91.32
Base+SwinTransformer	91.52	91.59
Base+SwinTransformer+FCM	91.67	91.73
Base+SwinTransformer+FFMAM	91.70	91.79
Base+SwinTransformer+FCM+FFMAM	91.79	92.08
Base+SwinTransformer+FCM+FFMAM+AMSFM	91.90	92.42
Base+SwinTransformer+FCM+FFMAM+AMSFM+BRM	<u>91.96</u>	<u>92.45</u>

3.3.1. Ablation for Swin Transformer Branch

On the basis of the base network, We added the Swin transformer branch, which is composed of four stages, and each stage includes two consecutive ST blocks to acquire a parallel but noninteractive encoder. The output features from each convolution layer and the corresponding Swin transformer stages were reshaped to the same dimensionality and concatenated along the channel and then as the input feature maps to the corresponding decoder layers. The Swin transformer branch helped to obtain the global context information, which is beneficial to distinguish clouds from ground objects and to extract thin clouds. The result shows that the addition of the Swin transformer branch led to an improvement in the F1_cloud to 91.59% and in the MIoU to 91.52%.

3.3.2. Ablation for FCM

The addition of the FCM serves to enhance the interplay between the global information representation and local features. These two elements complement each other, thereby facilitating the model's ability to comprehend both the broader context and specific, fine-grained details. This synergy between the global and local information contributes

to the model's overall effectiveness in understanding and processing complex data. As indicated in Table 2, the cooperative operation of the dual branches, Swin transformer, and convolution led to a 0.14% improvement in the F1_cloud and a 0.15% enhancement in the MIoU. In addition, on the basis of adding the FFMAM to fuse the global and local features, the addition of the FCM making those two branches interact still led to an improvement in the F1_cloud to 92.08% and in the MIoU to 91.79%.

3.3.3. Ablation for FFMAM

We employed the FFMAM to enhance the fusion of features obtained from the dual branches in both the spatial and dimensional aspects instead of directly through concatenation. By coordinating the multihead self-attention and the channel attention mechanisms, it further enables the global features to blend with the local features, thereby improving the global feature connectivity and acquiring more discriminative characteristics. Consequently, this enables the more precise differentiation of clouds from other objects. As demonstrated in Table 2, there were two ablation experiments to verify the effect of the FFMAM. The addition of the FFMAM on the noninteractive encoder with a convolution and Swin transformer led to a 0.20% increase in the F1_cloud and a 0.18% boost in the MIoU. In comparison, the inclusion of the FFMAM on the synergy encoder led to a 0.35% increase in the F1_cloud and a 0.12% boost in the MIoU.

3.3.4. Ablation for AMSFM

The AMSFM extracts large-scale information from various receptive fields and employs an attention mechanism to emphasize the spatial importance of each pixel. In remote sensing images with different resolutions, clouds exhibit distinct characteristics, shapes, and scales. Therefore, by integrating multiscale features, the AMSFM can not only improve the accuracy of cloud detection, but also enhance the generalization ability of the model. Experimental results show that the inclusion of the AMSFM led to an improvement in the F1_cloud to 92.42% and in the MIoU to 91.90%.

3.3.5. Ablation for BRM

At the end of the STCCD model, we added the enhanced edge features obtained from the BRM into the feature maps produced by the decoder section. This integration results in cloud masks exhibiting sharper boundaries and finer details. In our experimental results, the inclusion of the BRM led to an improvement in the F1_cloud from 92.42% to 92.45% and in the MIoU from 91.90% to 91.96%.

3.4. Comparison Test of the GF1-WHU Dataset

In this section, the proposed method is compared with other semantic segmentation models, such as UNet [69], DeepLabv3+ [58], U^{2}Net [70], Swin-Unet [40], BoundaryNet [25], LiteST-Net [46], BuildFormer [45], and ST-UNet [44]. The quantitative results are listed in Table 3, and the best result is underlined. Analysis of the data presented in Table 3 reveals that the models with U-shaped structures outperformed other models, likely because their decoder components combined the deep context information with shallow features, which is beneficial for capturing finer details. In addition, in direct comparison, our network obtained a higher F1 score, thus substantiating that the STCCD model exhibits the best overall performance in cloud detection.

Figure 10 displays the cloud detection results obtained by applying the different models to various scenarios within the GF1-WHU dataset. As seen in the first row, the image is predominantly covered by a large expanse of thin clouds. U^{2}Net, DeepLabV3+, ST-UNet, and BuildFormer only extracted the thick clouds and missed all the thin clouds. BoundaryNet and LiteST-Net identified a few thin clouds. and the STCCD network combined the local features and the global information representation while also incorporating a spatial attention mechanism, which is essential for accurately detecting thin clouds and achieving optimal results. As seen in the second row, the background of the image is the ocean, and the fractus clouds in this scene

are dispersive and fragmentized. As the result shows, almost all of the models missed most of the broken clouds. However, compared with the other models, the STCCD network accurately identified the spatial locations of the most clouds in this challenging scene. As seen in the third row, we encounter scenarios with snow, which has high brightness and irregular shapes like clouds. Consequently, the initial four models exhibited significant challenges in terms of false positives due to the complexity of distinguishing between clouds and snow when they coexist. In the remaining three models, the STCCD network, benefiting from a two-branch feature extraction structure, had a superior capability to distinguish between clouds and snow in such scenarios and obtained the fewest errors and missed marks. As seen in the fourth row, almost all of the models had splendid results in this scenario. However, it can be seen that the STCCD model had better performance in the detection of details and small-scale clouds. As seen in the fifth row, distinguishing between white buildings beneath clouds and the clouds themselves based solely on physical features is challenging. The FFMAM and AMSFM modules of our network effectively integrated the global information representation and multiscale semantic information, which is advantageous for learning the distinctive features that can be used to accurately determine the specific location of clouds. Overall, the STCCD network demonstrates the ability to extract clouds of different scales and thicknesses while also discriminating between clouds and other highlighted objects. This leads to more accurate cloud detection results.

Table 3. Comparison of evaluation metrics of different models on GF1-WHU dataset.

Method	OA (%)	MIoU (%)	F1_cloud (%)
DeepLabv3+	97.88	90.83	91.06
UNet	97.92	90.90	91.04
U^2Net	97.91	91.19	91.54
BoundaryNet	97.92	91.13	91.76
Swin-Unet	97.72	90.34	90.54
ST-UNet	97.66	90.70	91.30
BuildFormer	97.13	90.15	91.15
LiteST-Net	97.61	91.07	91.52
Our network	<u>98.06</u>	<u>91.96</u>	<u>92.45</u>

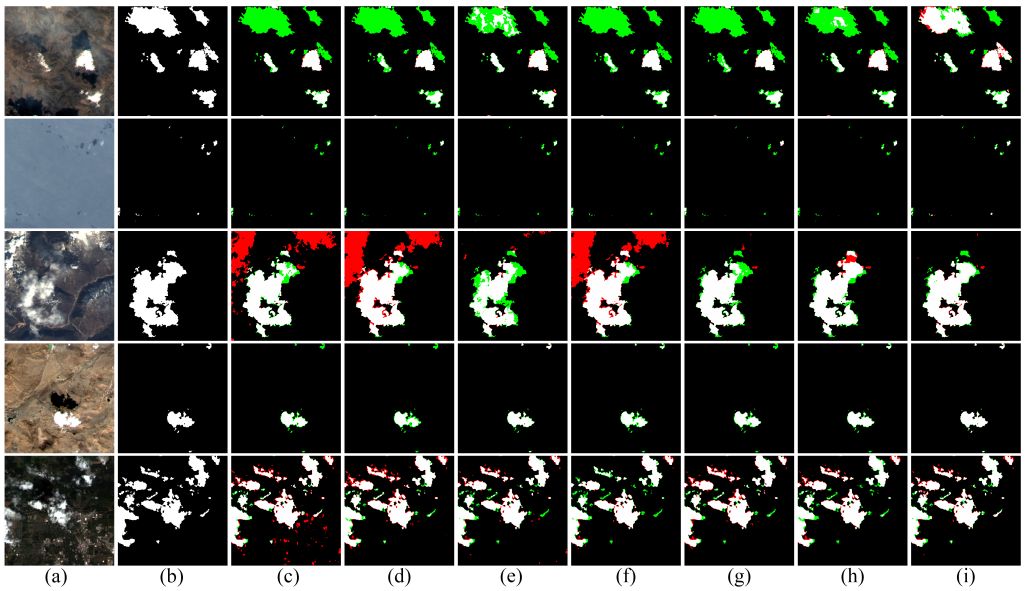

Figure 10. Prediction results of different models on GF1-WFV dataset. Black and white represent the noncloud and cloud pixels, respectively. (**a**) RGB image. (**b**) Label. (**c**) U^2Net. (**d**) DeepLabV3+. (**e**) BoundaryNet. (**f**) ST-UNet. (**g**) BuildFormer. (**h**) LiteST-Net. (**i**) Our network.

3.5. Comparison Test of the SPARCS Dataset

In this section, we used the SPARCS dataset to further verify the effectiveness of the STCCD network. The quantitative results of the different models are shown in Table 4, and the best results are underlined. Compared to the GF1-WHU dataset, the SPARCS dataset has only 80 images with 1000 × 1000 pixel resolution, which may result in less training data and cause the model to overfit. According to Tables 3 and 4, the UNet, PSPNet, U^2Net DeepLabV3+, ST-UNet, BuildFormer, and SwinUNet achieved lower accuracies on the SPARCS dataset than those on the GF1-WHU dataset, while the BoundaryNet, LiteST-Net, and STCCD network achieved similar F1 scores on both datasets.

Table 4. Comparison of evaluation metrics of different models on SPARCS dataset.

Method	OA (%)	MIoU (%)	F1_cloud (%)
DeepLabv3+	96.89	87.40	87.09
UNet	97.23	88.31	87.95
U^2Net	97.14	89.08	89.39
BoundaryNet	<u>97.85</u>	91.22	91.65
Swin-Unet	96.09	86.21	86.75
ST-UNet	96.14	88.26	90.08
BuildFormer	96.91	89.44	90.67
LiteST-Net	96.79	89.87	91.16
STCCD	97.78	<u>91.48</u>	<u>92.20</u>

Figure 11 shows the different performance outcomes of those models. The five samples include bare soil, thin clouds, thick clouds, white buildings, and small clouds. As seen in the top row of Figure 11, most of the models exhibited misclassifications with respect to the highlighted bare soil areas adjacent to a cluster of clouds. Notably, both the ST-UNet and STCCD models displayed comparatively less noise in their results. As seen in the second row, some small-scale clouds were easily confused with bright buildings. The STCCD model exceled in accurately identifying the clouds in this scenario, thus owing to its robust feature extraction and integration modules. As seen in the third row, it is noteworthy that the fractus clouds may be overlooked, and there is a potential for misidentification of the boundaries of thick clouds. We can see from the results that our model successfully detected all of the clouds' locations and had the fewest false positive pixels. As seen in the fourth row, the image background consists of a river and bare soil. The results show that the U^2Net, DeepLabV3+, ST-UNet, and BuildFormer mistakenly detected bare soil as clouds. The BoundaryNet and LiteST-Net missed many small-scale clouds. The STCCD network's unique blend of Swin transformer and convolutional techniques can aggregate the global context information and establish detailed context connections, which significantly contributes to the enhancement of prediction results.

These examples illustrate that the STCCD network has the ability to effectively detect clouds in a variety of complex backgrounds.

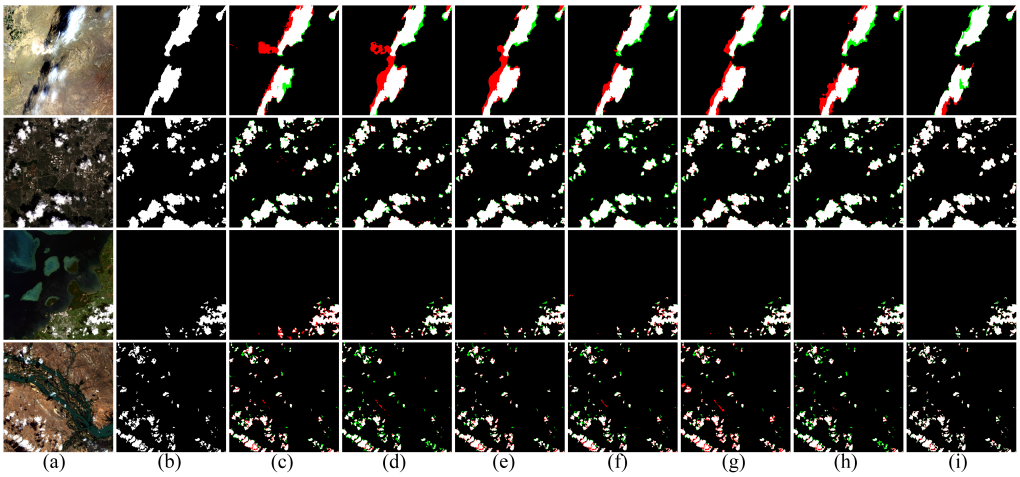

Figure 11. Prediction results of different models on the SPARCS dataset. Black and white represent the noncloud and cloud pixels, respectively, green represents cloud pixels that are missed, and red represents that the noncloud pixels are detected as clouds. (**a**) RGB image. (**b**) Label. (**c**) U^2Net. (**d**) DeepLabV3+. (**e**) BoundaryNet. (**f**) ST-UNet. (**g**) BuildFormer. (**h**) LiteST-Net. (**i**) Our network.

3.6. Comparison Test of the AIR-CD Dataset

The accuracy assessment results on the AIR-CD dataset are given in Table 5. As the results show, compared with the other models, the STCCD model still obtained the best performance regarding the OA, MIoU, and F1 score. However, compared to the other datasets, the backgrounds in the AIR-CD dataset are more complex and easily confused with the cloud. Consequently, some small-scale clouds were overlooked, thereby leading to lower performance scores for these models on this particular dataset when contrasted with their performance on other datasets.

Table 5. Comparison of evaluation metrics of different models on AIR-CD dataset.

Method	OA (%)	MIoU (%)	F1_cloud (%)
DeepLabv3+	97.50	87.82	84.62
UNet	96.76	86.96	85.38
U^2Net	97.16	87.15	84.31
BoundaryNet	97.12	89.18	88.81
Swin-Unet	96.89	87.43	85.67
ST-UNet	96.32	87.29	87.08
BuildFormer	96.54	86.72	86.36
LiteST-Net	97.21	88.25	85.81
Our network	<u>97.59</u>	<u>90.47</u>	<u>90.12</u>

Several prediction results for the GF-2 images are displayed in Figure 12. As seen in the first row, there is a big area of bare soil in the image background. The U^2Net, BoundaryNet, and ST-UNet mistakenly detected the bare soil as clouds. The DeepLabv3+ and BuildFormer missed most of the low-brightness clouds visible in the lower right corner of the image. Compared to the LiteST-Net, our model had fewer false negative pixels. As seen in the second row, the thin clouds in the lower right corner cover the cloud shadow, thus leading to decreases in the brightness of those thin clouds. As a result, most of the models failed to detect those thin clouds, and only the STCCD network achieved the fewest negative pixels. As seen in the third row, the thick clouds are surrounded by flocculent clouds, thus rendering it challenging to recognize the boundary between the clouds and the ground. The BRM played a crucial role in refining these boundaries, thus making the edges of our results closer to the real label. In the image of the fourth row, there is a large expanse of thin clouds, thus presenting challenges for boundary prediction. However, the STCCD model exceled in predicting the thin clouds. The success of STCCD model can

be attributed to the fusion of global representation information and local features in the encoder part, as well as the combination of deep context information and shallow features in the decoder part.

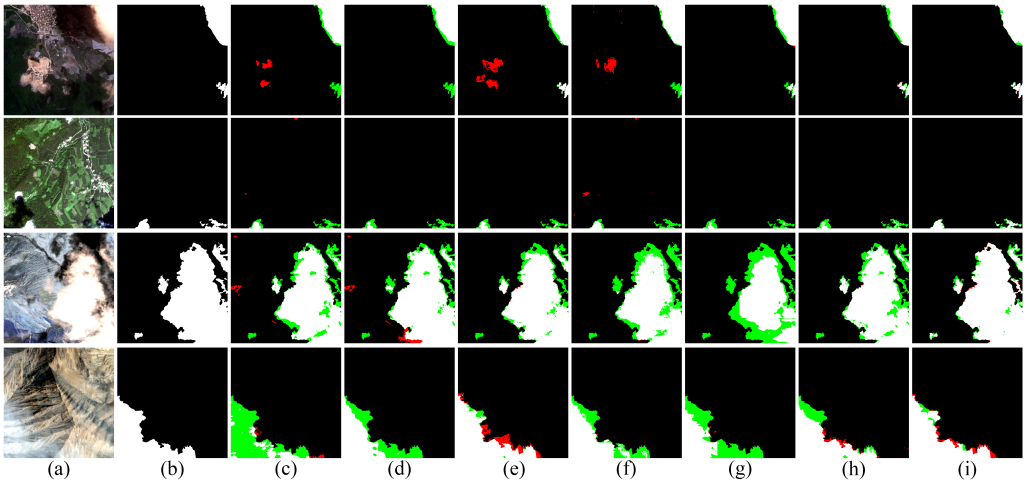

Figure 12. Prediction results of different models on AIR-CD dataset. Black and white represent the non cloud and cloud pixels, respectively, green represents cloud pixels that are missed, and red represents that the noncloud pixels are detected as clouds. (**a**) RGB image. (**b**) Label. (**c**) U^{2}Net. (**d**) DeepLabV3+. (**e**) BoundaryNet. (**f**) ST-UNet. (**g**) BuildFormer. (**h**) LiteST-Net. (**i**) Our network.

3.7. Extended Experiment

3.7.1. Crossvalidation

Crossvalidation was conducted to assess the robustness of the STCCD model using the SPARCS dataset. In the comparison test of the SPARCS dataset, the STCCD model, LiteST-Net, and BoundaryNet exhibited similar accuracy metrics. Therefore, we proceeded to further compare the stability of these three models through crossvalidation. During the crossvalidation process, we randomly partitioned the 80 images into five sets, with each containing 16 images. Subsequently, we selected one set sequentially as the testing dataset while using the remaining four sets as the training data. The mean and standard deviation (std) of each indicator are listed in Table 6, which are expressed as mean ± std.

Table 6. Crossvalidation of STCCD model, LiteST-Net, and BoundaryNet on the SPARCS dataset.

Method	OA (%)	MIoU (%)	F1_cloud (%)
BoundaryNet	96.70 ± 0.61	88.55 ± 0.91	89.10 ± 1.16
LiteST-Net	96.66 ± 0.46	87.79 ± 0.85	88.07 ± 1.30
Our network	97.41 ± 0.27	90.50 ± 0.87	91.04 ± 1.12

Table 6 reveals that, among these three models, the STCCD model had the highest mean values of the OA, MIoU, and *F1_cloud* score. The mean *F1_cloud* score of the STCCD model was 1.94% higher than that of the BoundaryNet, while the std of the *F1_cloud* score for the STCCD network was the lowest. Moreover, the std of the OA of the STCCD model was also the lowest, and the std of the MIoU was only 0.2% higher than that of the LiteST-Net. The results demonstrate that the STCCD network not only achieved the highest accuracy in cloud detection, but also excels in reliability and stability.

3.7.2. Extend Validation

In this section, we conducted an extended experiment to evaluate the generalization performance of the STCCD model. We used the model parameters trained on the GF1-WHU dataset to predict the cloud mask for images from the L8-Biome dataset. As described in Section 3.1.4, the L8-Biome dataset consists of 96 images distributed globally with diverse

backgrounds, thereby making it challenging to correctly detect clouds in all of the images. The L8 cloud cover assessment system uses the C Function of Mask (CFMask) method derived from the FMask algorithm to identify clouds, cloud confidence, cloud shadows, and snow/ice in Landsat 8 scenes, and it has been validated for cloud detection [71]. Therefore, we compared the predicted results of the STCCD model on the L8-Biome dataset with those of the FMask. The quantitative results are presented in Table 7. The Mask and STCCD models achieved similar overall accuracy values. However, it is noteworthy that the STCCD model achieved a higher *F*1_*cloud* score. Our model was trained on the GF1-WHU dataset with only four spectral bands, while the FMask used additional spectral information from the shortwave infrared bands, the cirrus band, and the thermal bands. This indicates a higher extensibility of the STCCD model for general satellite remote sensing imagery.

Table 7. Comparison of evaluation metrics of FMask and STCCD on L8-Biome dataset.

Method	OA(%)	MIoU (%)	F1_cloud (%)
FMask	91.19	75.65	77.54
Our network	92.07	76.37	80.62

We present selected samples in Figures 13 and 14 to illustrate the model's performance. As seen in the first two rows of Figure 13, the exposed riverbed appears white and bright, which can be easily confused with clouds. The FMask not only misclassified the riverbed as clouds, but also missed almost all the actual clouds. In contrast, the STCCD model correctly distinguished between the riverbed and clouds, thus achieving high accuracy. However, due to the riverbed's distance, the STCCD model did not detect all the true clouds. As seen in the third and fourth rows, the result detected by the FMask has numerous holes in the thick clouds, and it incorrectly identified bare soil as clouds. Comparatively, the STCCD model obtained superior results. As seen in the fifth and sixth rows, our STCCD model successfully discriminated between the snow and the clouds and accurately located the clouds. Moving to Figure 14, as seen in the first two rows, a comparison with the true label reveals that the STCCD model attained a higher recall, whereas the FMask had a higher precision. However, as seen in the enlarged view of the second row, the FMask mistakenly detected parts of the town's impervious surface as clouds. As seen in the subsequent two rows, the STCCD model accurately extracted clouds from the bright images, bare soil, and ice.

In summary, we used the model parameters trained on the GH1-WHU dataset to detect the cloud masks of the images from the L8-Biome dataset and obtained superior results compared to the FMask method. The complete inconsistency of the time, location, and resolution of images in these two datasets demonstrates the robustness and generalizability of our STCCD model. Despite outperforming the FMask, the STCCD model achieved an F1_cloud score of only 80.62%. This may be due to the differences in radiation and resolution of the different satellites.

In addition, the STCCD network exhibited lower accuracy in complex scenarios. As Figure 15 shows, the situations where the entire image is covered with snow posed a significant challenge. In such cases, clouds may be even darker than snow due to the influence of cloud shadows, making them difficult to distinguish from snow using only visible and near-infrared bands. This challenging problem remains an open research area in cloud detection.

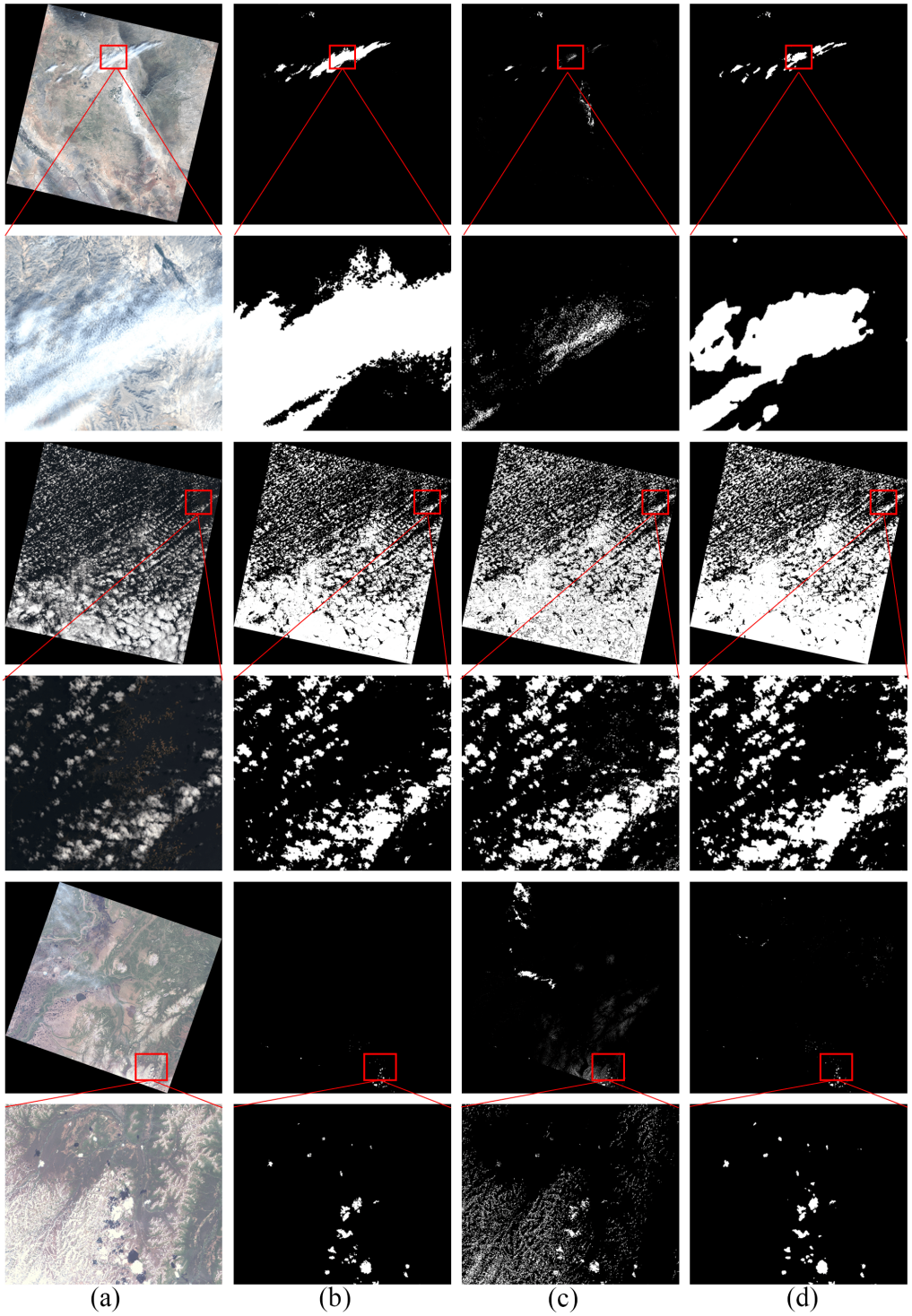

Figure 13. Prediction results of the STCCD model and FMask method on L8-Biome dataset. Black and white represent the noncloud and cloud pixels, respectively, green represents cloud pixels that are missed, and red represents that the noncloud pixels are detected as clouds. (**a**) RGB image. (**b**) Label. (**c**) FMask. (**d**) Our network.

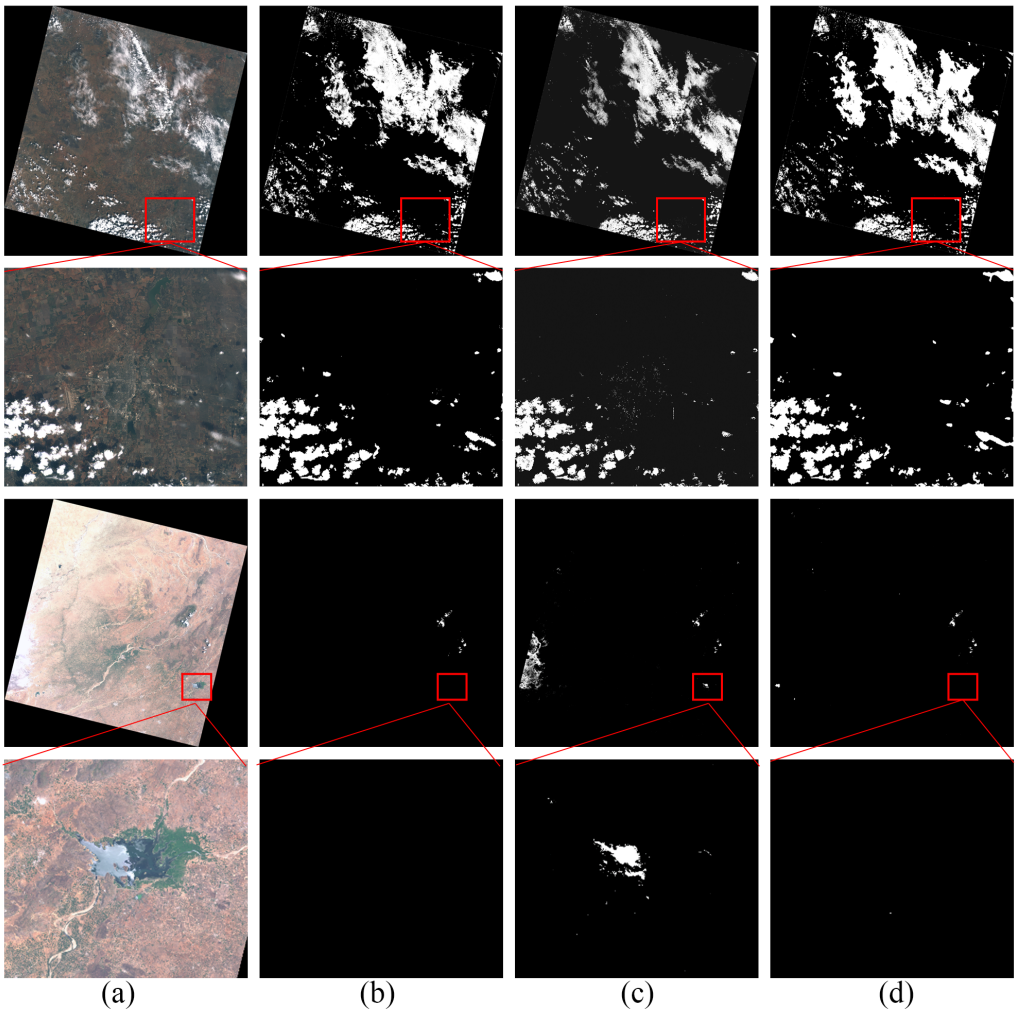

Figure 14. Prediction results of STCCD model and FMask method on L8-Biome dataset. Black and white represent the noncloud and cloud pixels, respectively. (**a**) RGB image. (**b**) Label. (**c**) FMask. (**d**) Our network.

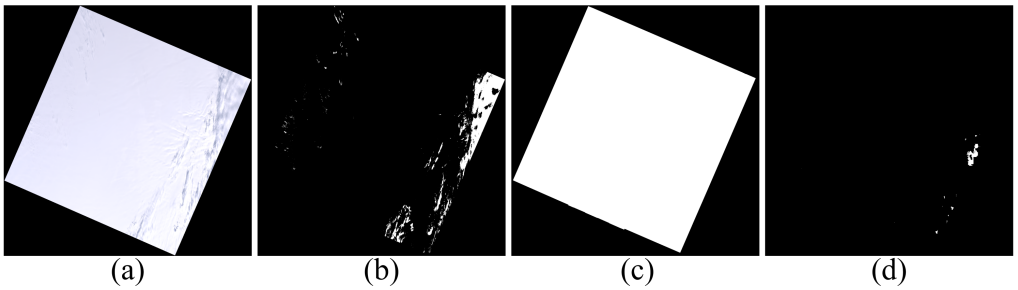

Figure 15. Failure sample of the STCCD model. Black and white represent the noncloud and cloud pixels, respectively. (**a**) RGB image. (**b**) Label. (**c**) FMask. (**d**) Our network.

4. Discussion

4.1. Advantage Analysis

The STCCD network, LiteST-Net, and BoundaryNet showed similar accuracy metrics, and the performance differences for different underlying surface types were further investigated. The F1 scores of the snow and bare scenarios, which are usually confused with clouds, of the SPARCS dataset are listed in Table 8. An example of the comparison of these two scenarios is shown in Figure 16. The result shows a better accuracy of the STCCD model. As shown in the first row of Figure 16, all three methods successfully distinguished clouds from snow. However, the STCCD model had the least false positive pixels and false

negative pixels. As seen in the second row, there are thin clouds covering the bare soil, and the results show that the STCCD model had the highest F1_cloud and the least false negative pixels.

Table 8. Comparison of evaluation metrics of STCCD, LiteST-Net, and BoundaryNet on the images with snow and bare soil.

Method	F1_cloud(%) (Snow)	F1_cloud(%) (Bare Soil)
BoundaryNet	81.89	88.90
LiteST-Net	83.40	87.60
Our network	86.41	90.45

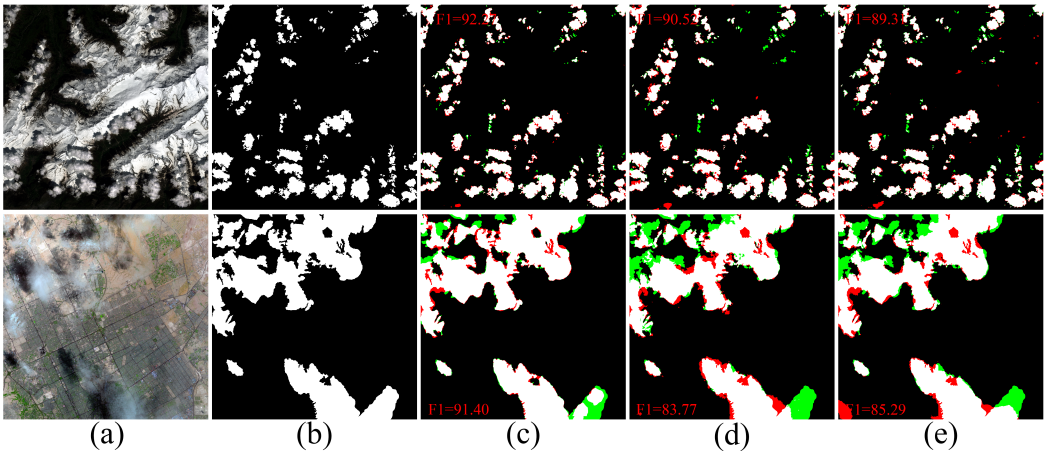

Figure 16. Prediction of STCCD, LiteST-Net, and BoundaryNet for clouds over snow and clouds over bare soil on the SPARCS dataset. Black and white represent the noncloud and cloud pixels, respectively, green represents cloud pixels that are missed, and red represents that the noncloud pixels are detected as clouds. (**a**) RGB image. (**b**) Label. (**c**) Our network. (**d**) LiteST-Net. (**e**) BoundaryNet.

Overall, the STCCD network has achieved superior results in these comparative experiments, even when considering the variations in spatial resolutions and the number of images across the three datasets. The extended experiment indicates the higher extensibility of the STCCD model regarding general satellite remote sensing imagery. The reason that the STCCD network can effectively distinguish clouds from other objects, when confronted with images featuring complex backgrounds, is because of its robust feature extraction and integration capabilities derived from the dual branches and features fusion modules. Moreover, the skip connection operation of the U-shaped structure, combining deep context information and shallow fine-grained features, proves to be a crucial feature splicing strategy, and it exhibits excellent performance in many end-to-end semantic segmentation algorithms. In addition, the FFMAM and AMSFM play significant roles in enhancing the extraction of crucial information for cloud detection. The BRM optimizes the edge of the cloud masks and makes them visually fit more closely with real cloud labels.

4.2. Limitations and Future Perspectives

However, despite achieving the best segmentation accuracy, there were still some misclassifications and omissions in the edges of clouds and the small scale of the thin clouds. In addition, Figure 15 shows the failure of the STCCD model in bad conditions. We hope to solve this problem by adding auxiliary data in future work. Additionally, our model incorporates the Swin transformer, the attention mechanisms, and the multiscale dilation convolution layers, which are helpful for improving the accuracy of cloud detection, but they also increase the model parameters and reduce the prediction speed. As shown in Table 9, the parameters and frames per second (FPS) for the various models on the SPARCS dataset are listed. The FPS represents the number of images processed by the model per

second. Table 9 indicates that both the STCCD and the ST-UNet models possess a greater number of parameters and exhibit slower processing speeds compared to other models. However, it is noteworthy that the STCCD model outperformed the ST-UNet in terms of overall performance. Therefore, reducing the model's parameter count while maintaining the performance of our model is another meaningful work for us. Furthermore, it is essential to acknowledge that, like most fully supervised semantic segmentation methods, model performance still inherently relies on the quality of the datasets. Hence, it's necessary to explore how to reduce the dependence on labels and improve the utilization of datasets through methods such as semisupervised learning or domain adaptation.

Table 9. Comparison of parameters and speed of different models on SPARCS dataset.

Method	Parameters (MB)	FPS
DeepLabv3+	17.59	405
UNet	31.04	224
U^2Net	44.01	160
BoundaryNet	53.32	86
Swin-Unet	27.17	378
ST-UNet	168.8	23
BuildFormer	40.52	140
LiteST-Net	18.03	55
Our network	164.71	63

5. Conclusions

This paper introduces the STCCD network, an encoder–decoder network tailored for cloud detection, which has demonstrated remarkable performance. The STCCD network embodies a holistic methodology, with multiple pivotal modules synergistically collaborating to attain its overarching success. The dual branches seamlessly combine Swin transformer and convolutional components. These components harness convolutional operators to extract local features and utilize the self-attention mechanisms within shift windows to capture global representations. The feature coupling module, in various forms, facilitates the transformation of the global representations and local features. On the basis of the harmonious branches, the feature fusion based on the attention mechanism leverages multihead attention and channel attention to effectively fuse the local and global features, thereby enhancing the overall feature representation. The aggregation multiscale feature module plays a pivotal role by extensively employing dilated convolution, pooling layers, and spatial attention mechanisms to extract discriminative information from the fused features. The boundary refinement module finetunes the cloud mask boundaries, thus further improving the accuracy of the cloud detections.

The experimental results prove the effectiveness of the STCCD network in cloud detection across diverse datasets, including the SPARCS, GF1-WHU, and AIR-CD. Quantitatively, the STCCD network achieved an overall accuracy (OA) that was greater than 97.59%, a mean intersection over union (MIoU) that was greater than 90.5%, and a cloud F1 score that was greater than 90.1% on these three datasets, thereby evidencing its versatility and superior performance. Moreover, we set up an extended experiment to verify the generalization capability of the STCCD model, and the results show that our model has high extensibility.

Future work will investigate the relationship between clouds and cloud shadows based on our accurate cloud mask generation to enhance cloud shadow extraction capabilities. Additionally, we will explore cloud detection methods based on domain adaptation to alleviate the reliance on limited sample data.

Author Contributions: Conceptualization, C.G. and R.Y.; methodology, C.G.; software, R.Y.; validation, T.L., W.J. and G.W.; formal analysis, R.Y.; investigation, C.G.; resources, C.G.; data curation, C.G.; writing—original draft preparation, C.G.; writing—review and editing, C.G. and T.L.; visualization, R.Y.; supervision, T.L. and W.J.; project administration, G.W.; funding acquisition, G.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Fund for Pioneering Research in Science and Disruptive Technologies through the Aerospace Information Research Institute at the Chinese Academy of Sciences (Grant No. E3Z218010F), as well as the National Natural Science Foundation of China under Grants 61860206004 and 61731022.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data and the code of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Long, T.; Zhang, Z.; He, G.; Jiao, W.; Tang, C.; Wu, B.; Zhang, X.; Wang, G.; Yin, R. 30 m Resolution Global Annual Burned Area Mapping Based on Landsat Images and Google Earth Engine. *Remote Sens.* **2019**, *11*, 489. [CrossRef]
2. Yin, R.; He, G.; Jiang, W.; Peng, Y.; Zhang, Z.; Li, M.; Gong, C. Night-Time Light Imagery Reveals China's City Activity During the COVID-19 Pandemic Period in Early 2020. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 5111–5122. [CrossRef]
3. Kuma, P.; Bender, F.A.M.; Schuddeboom, A.; McDonald, A.J.; Seland, Ø. Machine learning of cloud types shows higher climate sensitivity is associated with lower cloud biases. *Atmos. Chem. Phys. Discuss.* **2022**, *32*, 523–549. [CrossRef]
4. Zheng, X.; Ye, J.; Chen, Y.; Wistar, S.; Li, J.; Piedra-Fernández, J.A.; Steinberg, M.A.; Wang, J.Z. Detecting Comma-shaped Clouds for Severe Weather Forecasting using Shape and Motion. *IEEE Trans. Geosci. Remote. Sens.* **2019**, *57*, 3788–3801. [CrossRef]
5. Ju, J.; Roy, D.P. The availability of cloud-free Landsat ETM+ data over the conterminous United States and globally. *Remote Sens. Environ.* **2008**, *112*, 1196–1211. [CrossRef]
6. Zhu, X.; Helmer, E.H. An automatic method for screening clouds and cloud shadows in optical satellite image time series in cloudy regions. *Remote Sens. Environ.* **2018**, *214*, 135–153. [CrossRef]
7. 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]
8. Ge, K.; Liu, J.; Wang, F.; Chen, B.; Hu, Y. A Cloud Detection Method Based on Spectral and Gradient Features for SDGSAT-1 Multispectral Images. *Remote Sens.* **2022**, *15*, 24. [CrossRef]
9. Main-Knorn, M.; Pflug, B.; Louis, J.; Debaecker, V.; Müller-Wilm, U.; Gascon, F. Sen2Cor for Sentinel-2. In *Proceedings of the Image and Signal Processing for Remote Sensing XXIII*; Bruzzone, L., Bovolo, F., Benediktsson, J.A., Eds.; SPIE: Tokyo, Japen, 2018; p. 3. [CrossRef]
10. 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]
11. Li, Z.; Shen, H.; Li, H.; Xia, G.; Gamba, P.; Zhang, L. Multi-feature combined cloud and cloud shadow detection in GaoFen-1 wide field of view imagery. *Remote Sens. Environ.* **2017**, *191*, 342–358. [CrossRef]
12. Zhai, H.; Zhang, H.; Zhang, L.; Li, P. Cloud/shadow detection based on spectral indices for multi/hyperspectral optical remote sensing imagery. *ISPRS J. Photogramm. Remote Sens.* **2018**, *144*, 235–253. [CrossRef]
13. Deng, J.; Wang, H.; Ma, J. An automatic cloud detection algorithm for Landsat remote sensing image. In Proceedings of the 2016 4th International Workshop on Earth Observation and Remote Sensing Applications (EORSA), Guangzhou, China, 4–6 July 2016; pp. 395–399. [CrossRef]
14. Zhu, Z.; Woodcock, C.E. Object-based cloud and cloud shadow detection in Landsat imagery. *Remote Sens. Environ.* **2012**, *118*, 83–94. [CrossRef]
15. Bai, T.; Li, D.; Sun, K.; Chen, Y.; Li, W. Cloud Detection for High-Resolution Satellite Imagery Using Machine Learning and Multi-Feature Fusion. *Remote Sens.* **2016**, *8*, 715. [CrossRef]
16. Zi, Y.; Xie, F.; Jiang, Z. A Cloud Detection Method for Landsat 8 Images Based on PCANet. *Remote Sens.* **2018**, *10*, 877. [CrossRef]
17. Yang, L.; Zhuo, W.; Qi, L.; Shi, Y.; Gao, Y. ST++: Make Self-trainingWork Better for Semi-supervised Semantic Segmentation. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 19–24 June 2022; pp. 4258–4267. [CrossRef]
18. Cao, R.; Fang, L.; Lu, T.; He, N. Self-Attention-Based Deep Feature Fusion for Remote Sensing Scene Classification. *IEEE Geosci. Remote Sens. Lett.* **2021**, *18*, 43–47. [CrossRef]
19. Zhu, X.X.; Tuia, D.; Mou, L.; Xia, G.S.; Zhang, L.; Xu, F.; Fraundorfer, F. Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 8–36. [CrossRef]

20. Mountrakis, G.; Li, J.; Lu, X.; Hellwich, O. Deep learning for remotely sensed data. *ISPRS J. Photogramm. Remote Sens.* **2018**, *145*, 1–2. [CrossRef]
21. Yin, R.; He, G.; Wang, G.; Long, T.; Li, H.; Zhou, D.; Gong, C. Automatic Framework of Mapping Impervious Surface Growth With Long-Term Landsat Imagery Based on Temporal Deep Learning Model. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 1–5. [CrossRef]
22. Li, J.; Wu, Z.; Sheng, Q.; Wang, B.; Hu, Z.; Zheng, S.; Camps-Valls, G.; Molinier, M. A hybrid generative adversarial network for weakly-supervised cloud detection in multispectral images. *Remote Sens. Environ.* **2022**, *280*, 113197. [CrossRef]
23. Liu, C.C.; Zhang, Y.C.; Chen, P.Y.; Lai, C.C.; Chen, Y.H.; Cheng, J.H.; Ko, M.H. Clouds Classification from Sentinel-2 Imagery with Deep Residual Learning and Semantic Image Segmentation. *Remote Sens.* **2019**, *11*, 119. [CrossRef]
24. Yin, M.; Wang, P.; Ni, C.; Hao, W. Cloud and snow detection of remote sensing images based on improved Unet3+. *Sci. Rep.* **2022**, *12*, 14415. [CrossRef] [PubMed]
25. Wu, K.; Xu, Z.; Lyu, X.; Ren, P. Cloud detection with boundary nets. *ISPRS J. Photogramm. Remote Sens.* **2022**, *186*, 218–231. [CrossRef]
26. Mazza, A.; Sepe, P.; Poggi, G.; Scarpa, G. Cloud Segmentation of Sentinel-2 Images Using Convolutional Neural Network with Domain Adaptation. In Proceedings of the 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS, Brussels, Belgium, 11–16 July 2021; pp. 7236–7239. [CrossRef]
27. Pang, S.; Sun, L.; Tian, Y.; Ma, Y.; Wei, J. Convolutional Neural Network-Driven Improvements in Global Cloud Detection for Landsat 8 and Transfer Learning on Sentinel-2 Imagery. *Remote Sens.* **2023**, *15*, 1706. [CrossRef]
28. Zhang, C.; Weng, L.; Ding, L.; Xia, M.; Lin, H. CRSNet: Cloud and Cloud Shadow Refinement Segmentation Networks for Remote Sensing Imagery. *Remote Sens.* **2023**, *15*, 1664. [CrossRef]
29. Chen, K.; Xia, M.; Lin, H.; Qian, M. Multi-scale Attention Feature Aggregation Network for Cloud and Cloud Shadow Segmentation. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5612216. [CrossRef]
30. Guo, J.; Yang, J.; Yue, H.; Liu, X.; Li, K. Unsupervised Domain-Invariant Feature Learning for Cloud Detection of Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5405715. [CrossRef]
31. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual Attention Network for Scene Segmentation. *arXiv* **2019**, arXiv:1809.02983.
32. Huang, Z.; Wang, X.; Wei, Y.; Huang, L.; Shi, H.; Liu, W.; Huang, T.S. CCNet: Criss-Cross Attention for Semantic Segmentation. *arXiv* **2020**, arXiv:1811.11721.
33. Zhang, G.; Gao, X.; Yang, Y.; Wang, M.; Ran, S. Controllably Deep Supervision and Multi-Scale Feature Fusion Network for Cloud and Snow Detection Based on Medium- and High-Resolution Imagery Dataset. *Remote Sens.* **2021**, *13*, 4805. [CrossRef]
34. Wang, Y.; Gu, L.; Li, X.; Gao, F.; Jiang, T. Coexisting Cloud and Snow Detection based on a Hybrid Features Network applied to Remote Sensing Images. *IEEE Trans. Geosci. Remote. Sens.* **2023**, *61*, 5405515. [CrossRef]
35. Zhao, C.; Zhang, X.; Kuang, N.; Luo, H.; Zhong, S.; Fan, J. Boundary-Aware Bilateral Fusion Network for Cloud Detection. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–14. [CrossRef]
36. Hu, K.; Zhang, D.; Xia, M. CDUNet: Cloud Detection UNet for Remote Sensing Imagery. *Remote Sens.* **2021**, *13*, 4533. [CrossRef]
37. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2021**, arXiv:2010.11929.
38. Azad, R.; Heidari, M.; Shariatnia, M.; Aghdam, E.K.; Karimijafarbigloo, S.; Adeli, E.; Merhof, D. TransDeepLab: Convolution-Free Transformer-based DeepLab v3+ for Medical Image Segmentation. *arXiv* **2022**, arXiv:2208.00713.
39. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. *arXiv* **2021**, arXiv:2103.14030.
40. Cao, H.; Wang, Y.; Chen, J.; Jiang, D.; Zhang, X.; Tian, Q.; Wang, M. Swin-Unet: Unet-like Pure Transformer for Medical Image Segmentation. *arXiv* **2021**, arXiv:2105.05537.
41. Gulati, A.; Qin, J.; Chiu, C.C.; Parmar, N.; Zhang, Y.; Yu, J.; Han, W.; Wang, S.; Zhang, Z.; Wu, Y.; et al. Conformer: Convolution-augmented Transformer for Speech Recognition. *arXiv* **2020**, arXiv:2005.08100.
42. Feng, D.; Zhang, Z.; Yan, K. A Semantic Segmentation Method for Remote Sensing Images Based on the Swin Transformer Fusion Gabor Filter. *IEEE Access* **2022**, *10*, 77432–77451. [CrossRef]
43. Chen, H.; Qi, Z.; Shi, Z. Remote Sensing Image Change Detection With Transformers. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–14. [CrossRef]
44. He, X.; Zhou, Y.; Zhao, J.; Zhang, D.; Yao, R.; Xue, Y. Swin Transformer Embedding UNet for Remote Sensing Image Semantic Segmentation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 4408715. [CrossRef]
45. Wang, L.; Fang, S.; Meng, X.; Li, R. Building Extraction With Vision Transformer. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5625711. [CrossRef]
46. Yuan, W.; Zhang, X.; Shi, J.; Wang, J. LiteST-Net: A Hybrid Model of Lite Swin Transformer and Convolution for Building Extraction from Remote Sensing Image. *Remote Sens.* **2023**, *15*, 1996. [CrossRef]
47. Alrfou, K.; Zhao, T.; Kordijazi, A. Transfer Learning for Microstructure Segmentation with CS-UNet: A Hybrid Algorithm with Transformer and CNN Encoders. *arXiv* **2023**, arXiv:2308.13917.
48. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2015**, arXiv:1512.03385.
49. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2023**, arXiv:1706.03762.

50. Shaw, P.; Uszkoreit, J.; Vaswani, A. Self-Attention with Relative Position Representations. *arXiv* **2018**, arXiv:1803.02155.
51. Ma, H.; Yang, H.; Huang, D. Boundary Guided Context Aggregation for Semantic Segmentation. *arXiv* **2021**, arXiv:2110.14587.
52. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. *arXiv* **2020**, arXiv:1910.03151.
53. Luo, W.; Li, Y.; Urtasun, R.; Zemel, R. Understanding the Effective Receptive Field in Deep Convolutional Neural Networks. *arXiv* **2017**, arXiv:1701.04128.
54. Yu, F.; Koltun, V. Multi-Scale Context Aggregation by Dilated Convolutions. *arXiv* **2016**, arXiv:1511.07122.
55. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid Scene Parsing Network. *arXiv* **2017**, arXiv:1612.01105.
56. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *arXiv* **2018**, arXiv:1802.02611. [CrossRef]
57. Zhu, Z.; Liu, G.; Hui, G.; Guo, X.; Cao, Y.; Wu, H.; Liu, T.; Tian, G. Semantic Segmentation of FOD Using an Improved Deeplab V3+ Model. In Proceedings of the 2022 12th International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Baishan, China, 27–31 July 2022; pp. 791–796. [CrossRef]
58. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In *Computer Vision—ECCV 2018*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2018; Volume 11211, pp. 833–851. [CrossRef]
59. Su, J.; Li, J.; Zhang, Y.; Xia, C.; Tian, Y. Selectivity or Invariance: Boundary-Aware Salient Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3798–3807. [CrossRef]
60. Hughes, M.; Hayes, D. Automated Detection of Cloud and Cloud Shadow in Single-Date Landsat Imagery Using Neural Networks and Spatial Post-Processing. *Remote Sens.* **2014**, *6*, 4907–4926. [CrossRef]
61. Hughes, M. *L8 SPARCS Cloud Validation Masks*; US Geological Survey: Sioux Falls, SD, USA, 2016.
62. He, Q.; Sun, X.; Yan, Z.; Fu, K. DABNet: Deformable Contextual and Boundary-Weighted Network for Cloud Detection in Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–16. [CrossRef]
63. Foga, S.; Scaramuzza, P.L.; Guo, S.; Zhu, Z.; Dilley, R.D.; Beckmann, T.; Schmidt, G.L.; Dwyer, J.L.; Joseph Hughes, M.; Laue, B. Cloud detection algorithm comparison and validation for operational Landsat data products. *Remote Sens. Environ.* **2017**, *194*, 379–390. [CrossRef]
64. USGS. *Landsat 8 Cloud Cover Assessment Validation Data*; USGS: Reston, VA, USA, 2016.
65. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in PyTorch. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4 December 2017.
66. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2017**, arXiv:1412.6980.
67. De Boer, P.T.; Kroese, D.P.; Mannor, S.; Rubinstein, R.Y. A Tutorial on the Cross-Entropy Method. *Ann. Oper. Res.* **2005**, *134*, 19–67. [CrossRef]
68. Mattyus, G.; Luo, W.; Urtasun, R. DeepRoadMapper: Extracting Road Topology from Aerial Images. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017; pp. 3458–3466. [CrossRef]
69. Navab, N.; Hornegger, J.; Wells, W.M.; Frangi, A.F. (Eds.) *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015*; Proceedings, Part III; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2015; Volume 9351. [CrossRef]
70. Qin, X.; Zhang, Z.; Huang, C.; Dehghan, M.; Zaiane, O.R.; Jagersand, M. U^2-Net: Going Deeper with Nested U-Structure for Salient Object Detection. *Pattern Recognit.* **2020**, *106*, 107404.
71. Landsat 8 (L8) Data Users Handbook. Available online: https://www.usgs.gov/landsat-missions/landsat-8-data-users-handbook (accessed on 15 September 2023).