TransCloudSeg: Ground-Based Cloud Image Segmentation With Transformer

Shuang Liu, *Senior Member, IEEE*, Jiafeng Zhang, Zhong Zhang, *Senior Member, IEEE*, Xiaozhong Cao, and Tariq S. Durrani, *Life Fellow, IEEE*

***Abstract*—Cloud image segmentation plays an important role in ground-based cloud observation. Recently, most existing methods for ground-based cloud image segmentation learn feature representations using the convolutional neural network (CNN), which results in the loss of global information because of the limited receptive field size of the filters in the CNN. In this article, we propose a novel deep model named TransCloudSeg, which makes full use of the advantages of the CNN and transformer to extract detailed information and global contextual information for ground-based cloud image segmentation. Specifically, TransCloudSeg hybridizes the CNN and transformer as the encoders to obtain different features. To recover and fuse the feature maps from the encoders, we design the CNN decoder and the transformer decoder for TransCloudSeg. After obtaining two sets of feature maps from two different decoders, we propose the heterogeneous fusion module to effectively fuse the heterogeneous feature maps by applying the self-attention mechanism. We conduct a series of experiments on Tianjin Normal University large-scale cloud detection database and Tianjin Normal University cloud detection database, and the results show that our method achieves a better performance than other state-of-the-art methods, thus proving the effectiveness of the proposed TransCloudSeg.**

***Index Terms*—Convolutional neural network (CNN), cloud image segmentation, heterogeneous feature maps, transformer.**

I. Introduction

CLOUDS are the visible aggregations of a large number of small water droplets or ice crystals in the atmosphere. They play an important role in the Earth's atmospheric movement, surface temperature regulation, and hydrological cycle [1]–[3]. Hence, accurate cloud observation is crucial in environmental monitoring, weather forecasting, etc. In general, there are two types of cloud observation, i.e., satellite-based [4] and ground-based [5], [6]. Ground-based cloud observation is highly flexible and accessible, and it is good at monitoring the bottom characteristics of clouds in local areas for capturing information [7].

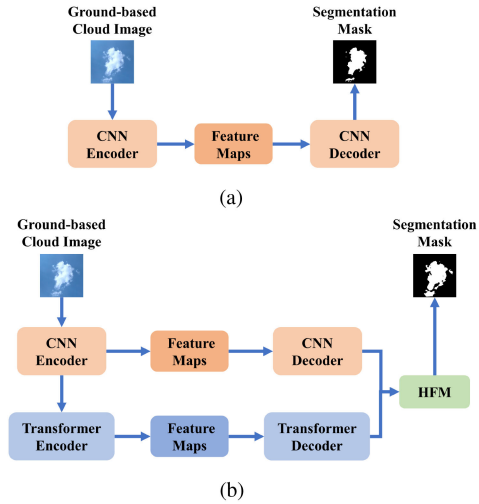

Fig. 1. Structures of (a) common-used encoder–decoder network and (b) proposed TransCloudSeg for ground-based cloud image segmentation.

Due to blurred edges and varied shapes of clouds, ground-based cloud observation is quite challenging. Moreover, it primarily relies on professional technicians, which not only leads to a waste of human resources but also may obtain inconsistent results toward the same observation. In order to make the results of ground-based cloud observation more objective, automatic algorithms are in great need. Many automatic algorithms are proposed for ground-based cloud observation including cloud base height measurement [8]–[10], cloud type classification [11], [12], and cloud cover estimation [13]–[15]. In this article, we focus on cloud cover estimation for ground-based cloud observation.

The goal of ground-based cloud image segmentation is to generate a segmentation mask in which each pixel is classified as cloud or sky, so the cloud cover estimation can be implemented by ground-based cloud image segmentation [16]. Since the light scattering causes sky and clouds to show different colors, traditional methods [1], [14], [17] for ground-based cloud image segmentation usually apply color values as features. Some performance improvements are achieved, but it is still not good enough for actual applications.

Manuscript received 3 June 2022; revised 11 July 2022; accepted 24 July 2022. Date of publication 27 July 2022; date of current version 5 August 2022. This work was supported in part by National Natural Science Foundation of China under Grant 62171321, in part by the Natural Science Foundation of Tianjin under Grant 20JCZDJC00180 and Grant 19JCZDJC31500, and in part by the Open Projects Program of National Laboratory of Pattern Recognition under Grant 202000002. (*Corresponding author: Zhong Zhang.*)

Shuang Liu, Jiafeng Zhang, and Zhong Zhang are with the Tianjin Key Laboratory of Wireless Mobile Communications and Power Transmission, Tianjin Normal University, Tianjin 300387, China (e-mail: shuangliu.tjnu@gmail.com; m648167095@gmail.com; zhong.zhang8848@gmail.com).

Xiaozhong Cao is with the Meteorological Observation Centre, China Meteorological Administration, Beijing 100081, China (e-mail: xzhongcao@163.com).

Tariq S. Durrani is with the Department of Electronic and Electrical Engineering, University of Strathclyde, G1 1XQ Glasgow, U.K. (e-mail: t.durrani@strath.ac.uk).

Digital Object Identifier 10.1109/JSTARS.2022.3194316

Because of the powerful representation ability of deep learning, convolutional neural network (CNN) has become the mainstream for many computer vision tasks [18]–[20]. For example, Hong et al. [21] proposed a unified multimodality learning framework for remote sensing image classification and designed the cross fusion strategy for effective transfer information across modalities. Hence, some methods are proposed to employ the CNN for ground-based cloud image segmentation, such as CloudU-Net [5], CloudSegNet [22], and SegCloud [23]. As shown in Fig. 1(a), these methods are usually designed as the encoder–decoder architecture where the encoder is composed of the CNN. The encoder is used to learn high-level and low-resolution features, and the decoder outputs the segmentation mask. However, the CNN encoder results in the loss of global information, because the receptive field size of the filter in the CNN is limited.

To overcome the aforementioned limitation, a transformer [24] is proposed, which is first successfully applied in natural language processing (NLP). Since transformer relies on the self-attention (SA) mechanism to learn discriminative features, it is naturally introduced into the computer vision field [25]–[27]. Many transformer-based methods for computer vision [28]–[30] treat the input image as a sequence of image patches, and then, they add the position embedding to build location information. Although transformer-based methods have achieved new state-of-the-art performance compared with CNN-based methods in many topics of computer vision, they are easy to lose detailed information of images.

In this article, we propose a novel deep learning method named TransCloudSeg, which makes full use of the advantages of the CNN and transformer for ground-based cloud image segmentation. To the best of our knowledge, it is the first time to introduce transformer into ground-based cloud image segmentation. Specifically, TransCloudSeg mainly consists of two encoders, two decoders, and the heterogeneous fusion module (HFM) as shown in Fig. 1(b). The two encoders are the CNN encoder and the transformer encoder, which focus on extracting detailed information and global contextual information, simultaneously. Meanwhile, we design two decoders, i.e., CNN decoder and transformer decoder corresponding to the two encoders. The CNN decoder utilizes the skip connections to integrate the feature maps from different scales of the CNN encoder for recovering detailed information. The transformer decoder aggregates the feature maps from different levels of the transformer encoder by assigning different weights. The outputs of the two decoders are two sets of feature maps, and these feature maps are heterogeneous because they contain different semantic information.

After obtaining the heterogeneous feature maps from the CNN and Transformer decoders, the direct methods to fuse them are concatenation or addition, but it is difficult to mine useful information from them . Hence, we propose HFM to effectively fuse the heterogeneous feature maps. The proposed HFM transforms the feature maps into the sequences, and then, employs the SA mechanism to learn discriminative features.

The main contributions of this article are threefold.

1) We hybridize the CNN and transformer as the encoder of TransCloudSeg, and to the best of our knowledge, we are the first to apply the transformer for ground-based cloud image segmentation.

2) We design two corresponding decoders to fuse feature maps from the two encoders. Furthermore, as a newly proposed component, the HFM is proposed to perform effective heterogeneous feature maps fusion.

3) We evaluate the segmentation performance of the proposed TransCloudSeg on Tianjin Normal University large-scale cloud detection database (TLCDD) [31] and Tianjin Normal University cloud detection database (TCDD) [32]. The experimental results outperform other state-of-the-art methods, demonstrating the superiority of the proposed method.

II. Related Work

A. Ground-Based Cloud Image Segmentation

With the development of acquisition devices of images, many powerful algorithms have been proposed for ground-based cloud image segmentation. Long et al. [17] treated the ratio of red and blue (R/B) as the fixed threshold for cloud image segmentation. Afterwards, Heinle et al. [1] utilized R-B as the threshold. Shi et al. [14] combined texture and color features on the basis of superpixels for cloud image segmentation, which allows the aggregation of pixels to consider location information.

Recently, many approaches employ the encoder–decoder architecture combined with the CNN for ground-based cloud image segmentation. For example, Dev et al. [22] proposed CloudSegNet, which is a lightweight deep learning model for effectively segmenting daytime and nighttime cloud images. Xie et al. [23] presented the SegCloud model, which has a CNN encoder–decoder architecture trained on 400 all-sky images with annotation. In order to obtain better segmentation results, Shi et al. [5] proposed to replace traditional convolution operations with dilated convolution operations, which could obtain more information. Since the sample number of cloud database is limited, Zhou et al. [33] employed transfer learning to train the deep network on other databases, and then, fine-tune on the ground-based cloud image database.

The main contribution of CNN-based methods for ground-based cloud image segmentation is to introduce the encoder–decoder architecture into this field. The encoder is designed to learn the representation features, which enables to mine semantic information. The decoder recovers the representation features into the segmentation mask so as to implement the pixel-level classification.

B. Transformer

Transformer is first proposed in [24] and it has achieved promising performance in NLP due to well dealing with long-range spatial dependencies of sequences. Recently, researchers modify transformer for computer vision tasks [34],

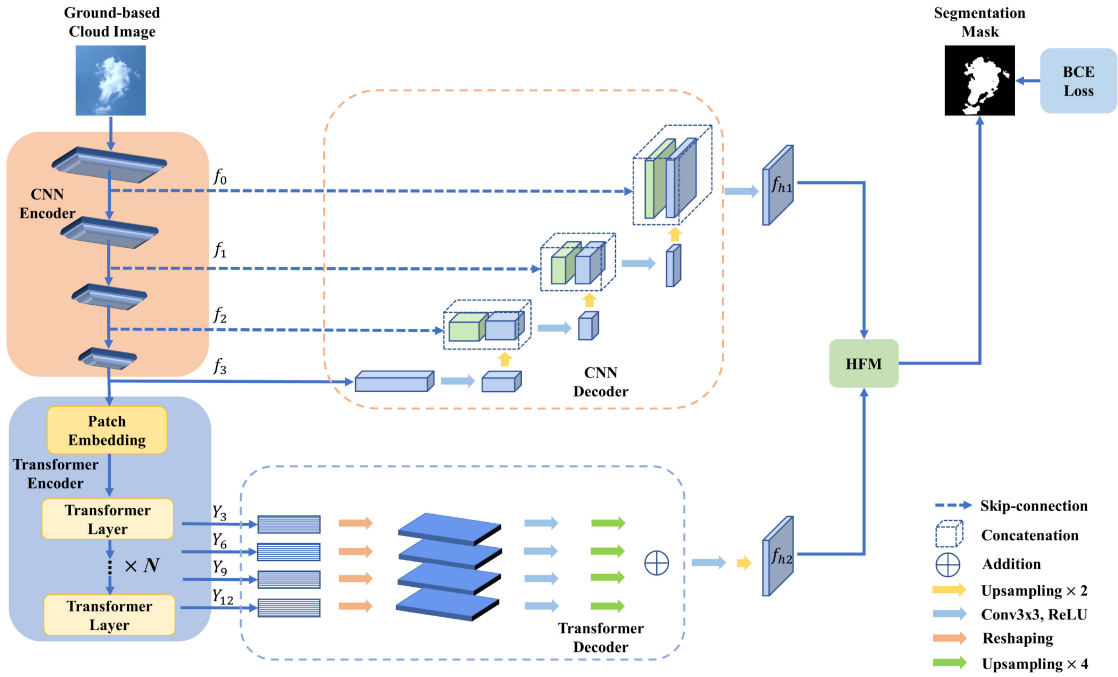

Fig. 2. Framework of the proposed TransCloudSeg. We first utilize the CNN encoder to learn the local information, and then, apply the transformer encoder to learn the global information. Afterwards, we design the CNN decoder and the transformer decoder to combine the multiscale feature maps from the CNN encoder and the multilevel feature maps from the transformer encoder, respectively. Finally, the two sets of feature maps from the CNN decoder and the transformer decoder are fed into the HFM to generate the segmentation mask.

and these transformer-based methods show new state-of-the-art performance in image classification [35], [36], segmentation [37]–[39], object detection [26], and so on.

Most transformer-based methods [28], [29], [40] transform the input image to a sequence of patches so as to capture long-range spatial dependencies. In addition, the position embedding is utilized to build position information among patches. Vision transformer (ViT) [25] demonstrated that a pure transformer could achieve the state-of-the-art through directly sequencing images for ImageNet classification. Zheng et al. [28] applied transformer to semantic segmentation and yielded new state of the art on publicly available segmentation databases, which provides an alternative solution to image segmentation. The contributions of [28] lie in twofolds.

1) They utilized the transformer with a global receptive field to learn global information.
2) To extensively examine the SA representation features, it designs three different kinds of decoders.

Hong et al. [41] applied the transformer to the HS image classification task and designed the groupwise spectral embedding and cross-layer adaptive fusion modules to improve the detail capture of subtle spectral differences and enhance the interaction between layers.

Different from the aforementioned transformer-based methods, we hybridize the CNN and transformer as the encoders, and design the two corresponding decoders. Furthermore, we propose the HFM to fuse the heterogenous feature maps from the two decoders in order to generate the segmentation mask.

III. Approach

In this section, we first clarify the motivation of the proposed TransCloudSeg. Then, we present an overall framework of the proposed TransCloudSeg, and then, describe the CNN encoder and the transformer encoder in detail. Afterwards, we present two different decoders corresponding to the CNN encoder and the transformer encoder, respectively. Finally, we introduce how to fuse the heterogeneous feature maps extracted from the two decoders in order to generate the segmentation mask.

A. Motivation

Clouds are aggregated with water droplets or ice crystals in the atmosphere, so they possess blurred boundaries and irregular shapes. Furthermore, imaging ground-based cloud images is easily affected by illumination. These factors are the challenges of ground-based cloud image segmentation.

Most existing deep learning methods for ground-based cloud image segmentation are under the framework of the CNN. Although the CNN has excellent representational capabilities but neglects to learn global contextual information due to the limited receptive field size of the filter. As shown in Fig. 3(a), the CNN-based methods lack the whole consistency for cloud image segmentation. Meanwhile, the transformer adopts SA mechanism to learn global contextual information, but lacks local detail information processing. From Fig. 3(b), we can see that the transformer-based methods are unfavorable to segment the boundaries of the cloud. Hence, we hybridize the CNN and transformer as the encoder for cloud representation so as to make

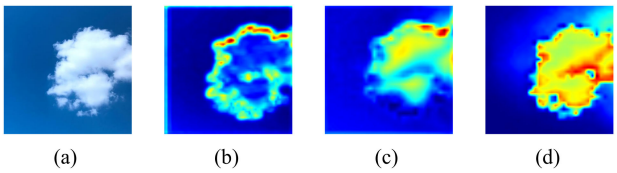

Fig. 3. Visualization of encoder feature maps for different methods. (a) Ground-based cloud map sample. (b) CNN-based methods. (c) Transformer-based methods. (d) Our method.

full use of the properties of the CNN and transformer, which could learn local information and global contextual information, simultaneously. As shown in Fig. 3(c), our method could balance the whole cloud distribution and the detailed boundaries for accurate ground-based cloud image segmentation.

B. Overall Framework

The framework of TransCloudSeg is shown in Fig. 2, where it contains three main components: two encoders, two decoders, and HFM.

1) Encoders: The proposed TransCloudSeg consists of two encoders, i.e., CNN encoder and transformer encoder, which aim to learn high-level features from input cloud images. The CNN encoder is first utilized to extract multiscale feature maps. Then, the transformer encoder changes the feature maps outputted from the CNN encoder into 1-D sequences. We add the position embedding to retain positional information, and stack multiple transformer layers to obtain multilevel feature maps.

2) Decoders: We design two different decoders corresponding to the CNN encoder and the transformer encoder for TransSegCloud. The CNN decoder targets to gradually recover the resolution of feature maps and integrate the feature maps via skip connections. The purpose of the transformer decoder is to aggregate the feature maps with different levels from the transformer encoder. The outputs of the two decoders are two sets of heterogeneous feature maps.

3) Heterogeneous Fusion Module (HFM): We propose HFM to fuse the heterogeneous feature maps. Specifically, we first transform the two sets of feature maps into two sequences. Afterwards, we concatenate the two sequences and feed them into the SA operation. Finally, the output of SA is reshaped to generate the segmentation mask.

C. Encoders

Since the receptive field size of the filter in the CNN is limited, the CNN encoder results in losing global information, and meanwhile, the transformer encoder focuses on learning global contextual information. Hence, we design the encoder of TransCloudSeg as a CNN–Transformer hybrid form so as to learn local and global information, which makes full use of the advantages of the CNN and transformer.

1) CNN Encoder: We apply ResNet-50 (BiT) [25] as the backbone of the CNN encoder to extract multiscale feature maps, and the structure of ResNet-50 (BiT) is shown in Table I. The size of input cloud image is $H \times W \times 3$, where

TABLE I
STRUCTURE OF RESNET-50 (BIT)

Name	Output Size	Filters	Padding
Conv1	$\frac{H}{2} \times \frac{W}{2}$	$[7 \times 7, 64]$, stride = 2	(3, 3)
Max pooling	$\frac{H}{4} \times \frac{W}{4}$	3×3, stride = 2	(1, 1)
Stage1	$\frac{H}{4} \times \frac{W}{4}$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} (0,0) \\ (1,1) \\ (0,0) \end{bmatrix} \times 3$
Stage2	$\frac{H}{8} \times \frac{W}{8}$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} (0,0) \\ (1,1) \\ (0,0) \end{bmatrix} \times 4$
Stage3	$\frac{H}{16} \times \frac{W}{16}$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 9$	$\begin{bmatrix} (0,0) \\ (1,1) \\ (0,0) \end{bmatrix} \times 9$

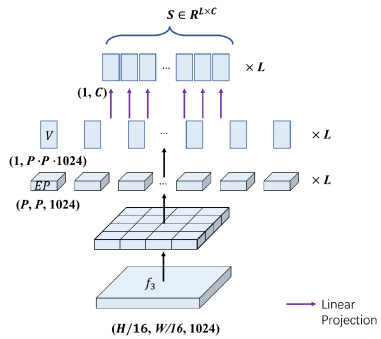

Fig. 4. Flowchart of the way to transform from the 2-D feature map f_3 into the 1-D sequence S.

H and W are the height and the width of cloud image, respectively, and 3 is the channel number of cloud image. $f_0 \in \mathbb{R}^{H/2\times W/2\times 64}$, $f_1 \in \mathbb{R}^{H/4\times W/4\times 256}$, $f_2 \in \mathbb{R}^{H/8\times W/8\times 512}$, and $f_3 \in \mathbb{R}^{H/16\times W/16\times 1024}$ are the multiscale feature maps of the CNN encoder.

2) Transformer Encoder: We treat the output feature maps of the CNN encoder f_3 as the input of the transformer encoder. The transformer encoder is composed of patch embedding and N transformer layers.

As for the patch embedding, we tokenize f_3 into the 1-D sequence $S \in \mathbb{R}^{L\times C}$ where L is the length of sequences and C is the hidden channel size. Since the size of $f_3 \in \mathbb{R}^{H/16\times W/16\times 1024}$, L and C are equal to $H/16 \times W/16$ and 1024, respectively. Specifically, similar to SETR [28] and ViT [25], we utilize fixed-size patches to transform 2-D feature maps into 1-D sequence as shown in Fig. 4. We first uniformly partition the feature maps $f_3 \in \mathbb{R}^{H/16\times W/16\times 1024}$ into L patches with the size of $P \times P$, where $L = \frac{H/16\times W/16}{P\times P}$. For each patch, EP is with the size of $P \times P \times 1024$. Afterwards, we flatten EP to a 1-D vector $V \in \mathbb{R}^{1\times P\cdot P\cdot 1024}$, and then, project the vector dimension to C through the linear layer.

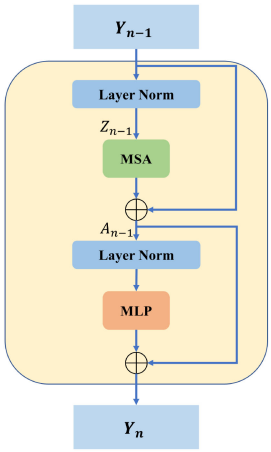

Fig. 5. Flowchart of the transformer layer.

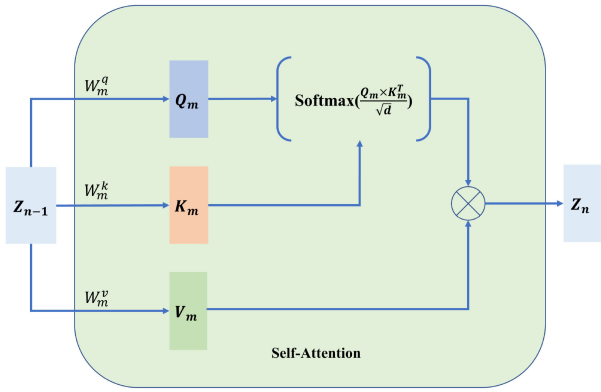

Fig. 6. Flowchart of SA.

Hence, the 2-D feature maps f_3 are mapped into L number of C-dimensional vectors, i.e., 1-D sequence S. Furthermore, we apply the learnable position embedding $E_p \in \mathbb{R}^{L\times D}$ to learn the spatial dependencies between different sequences. Hence, the patch embedding is defined as

$$X = SE + E_p \tag{1}$$

where $E \in \mathbb{R}^{C\times D}$ is a trainable linear projection. Here, to reduce the computational cost, we set $D < C$ to change the size of the hidden channel from C to D. The patch embedding X contains the information of 1-D sequence S and the position information, and it is employed to compensate for the spatial information between sequences when transforming 2-D feature maps into 1-D sequences. Furthermore, the patch embedding X is treated as the input of transformer. The transformer utilizes the SA mechanism [24] to enhance the interactions within the patch embedding for global information.

The SA mechanism is vital to learning global contextual information for the transformer encoder. We take the output of patch embedding X as the input of transformer layer and apply N transformer layers to learn complex feature representations. Fig. 5 shows the flowchart of the transformer layer, which consists of multihead SA (MSA), layer normalization (LN), residual connections, and multilayer perceptron (MLP). The nth transformer layer is defined as

$$Y_n = \text{MLP}(\text{LN}(A_{n-1})) + A_{n-1} \tag{2}$$

$$A_{n-1} = \text{MSA}(Z_{n-1}) + Y_{n-1} \tag{3}$$

$$Z_{n-1} = \text{LN}(Y_{n-1}) \tag{4}$$

where LN$(\cdot)$ denotes the layer normalization operation.

The MSA utilizes several independent SA operations to learn global contextual information from different aspects. The MSA in the nth transformer layer is formulated as

$$MSA(Z_{n-1}) = [SA_1(Z_{n-1}); \ldots; SA_M(Z_{n-1})]\, W_n \tag{5}$$

where M is the number of independent SA operations and W_n is the trainable linear projection in the nth transformer layer.

The flowchart of SA operation is shown in Fig. 6. The mth SA operation in the nth transformer layer is defined as

$$\text{SA}_m(Z_{n-1}) = \text{softmax}\left(\frac{Q_m K_m^T}{\sqrt{d}}\right) V_m \tag{6}$$

$$Q_m = Z_{n-1}W_m^q, K_m = Z_{n-1}W_m^k, V_m = Z_{n-1}W_m^v \tag{7}$$

where $W_m^q, W_m^k, W_m^v \in \mathbb{R}^{D\times d}$ are three independent trainable linear projection for the mth SA operation.

The outputs of the transformer layer $Y_n \in \mathbb{R}^{L\times D}(n = 1, .., N)$ consist of multilevel feature maps of the transformer encoder. Since the transformer layer keeps the size of input and output, the size of multilevel feature maps is the same.

D. Decoders

Since the CNN encoder and the transformer encoder possess different mechanisms when extracting features, we design two decoders, i.e., CNN decoder and transformer decoder for TransCloudSeg so as to effectively recover and fuse the feature maps extracted from the encoders.

1) CNN Decoder: We design the CNN decoder, which leverages the skip connections to integrate the multiscale feature maps from the CNN encoder for retaining detailed information. In the CNN decoder, $[f_0, f_1, f_2, f_3]$ is regarded as the input. The flowchart of the CNN decoder can be concisely described as

$f_3 \rightarrow$Conv+R $\rightarrow$ Upsampling$\times 2\rightarrow$Cat(f_2)
$\rightarrow$Conv+R $\rightarrow$ Upsampling$\times 2\rightarrow$Cat(f_1) $\rightarrow$Conv+R
$\rightarrow$Upsampling $\times 2\rightarrow$ Cat(f_0) $\rightarrow$Conv+R$\rightarrow f_{h1}$.

Here, Conv+R is the convolution layer followed by ReLU, Upsampling$\times I$ indicates that the feature maps are enlarged by a factor of I using bilinear interpolation, and Cat(f) represents the concatenation of the input feature maps and f. Specifically, the convolution layer is with the filter size of 3×3, the stride of 1 and the padding of $(1, 1)$. $f_{h1} \in \mathbb{R}^{H/2\times W/2\times 16}$ is the output of the CNN decoder.

2) Transformer Decoder: We propose the transformer decoder, which utilizes the weight aggregation to integrate the multilevel feature maps from the transformer encoder for retaining global contextual information.

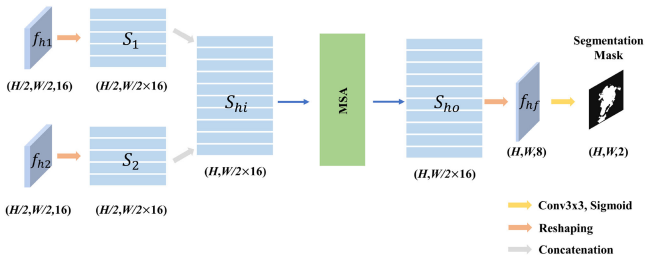

Fig. 7. Framework of HFM.

The output of the transformer decoder is defined as

$$f_{h2} = g\left(\sum_{n=1}^{N} \frac{n}{N} h(Y_n)\right) \tag{8}$$

where $h(\cdot)$ indicates that we first reshape Y_n from the sequence with the size of $L \times D$ to the feature maps with the size of $H/16 \times W/16 \times D$, then apply the convolution layers and ReLU, and finally, employ the bilinear interpolation to enlarge the size of the feature maps by a factor of 4. Here, $g(\cdot)$ denotes applying the convolution layers, ReLU and the bilinear interpolation to obtain the feature maps where the size is the same as f_{h1}. From (8), we can see that it aggregate the multiscale feature maps using a weighted strategy to take full advantage of different contextual information.

E. Heterogeneous Fusion Module (HFM)

The outputs of two decoders are two sets of feature maps f_{h1} and f_{h2}. They are heterogeneous because they are generated by different encoder–decoder architectures and contain different semantic information. For the heterogeneous feature maps f_{h1} and f_{h2}, it is difficult to mine useful information from them by directly concatenating or adding them together. In order to fully exploit the information from heterogeneous feature maps, we propose the HFM to increase the interactions between the features by using MSA.

Fig. 7 shows the framework of the HFM. We first transform f_{h1} and f_{h2} into the sequences $S_1 \in \mathbb{R}^{H/2\times W/2\cdot 16}$ and $S_2 \in \mathbb{R}^{H/2\times W/2\cdot 16}$, respectively, where $H/2$ is the length of sequences and $W/2 \cdot 16$ is the hidden channel size. The process of sequence construction is shown in Fig. 4. Specifically, we first uniformly partition the feature maps $f_{h1} \in \mathbb{R}^{H/2\times W/2\times 16}$ into L patches with the size of $P \times P$, where $L = (H/2)/(P \times P)$. For each patch, the patch EP is with the size of $P \times P \times W/2 \cdot 16$. Afterwards, we flatten EP into a 1-D vector $V \in \mathbb{R}^{1\times P\cdot P\cdot W/2\cdot 16}$, and then, project the vector dimension to $C = W/2 \cdot 16$ through the linear layer. The feature f_{h2} constructs the sequence S_2 in the same way. Afterwards, we concatenate S_1 and S_2 as the MSA input $S_{hi} \in \mathbb{R}^{H\times W/2\cdot 16}$. Furthermore, we utilize MSA to enhance the interactions between sequences. The MSA operation is described in (9). Finally, we transform the output of MSA $S_{ho} \in \mathbb{R}^{H\times W/2\cdot 16}$ into $f_{hf} \in \mathbb{R}^{H\times W\times 8}$ by the reshaping operation.

$$f_{hf} = r\left(\text{MSA}\left(\text{Cat}(S_1, S_2)\right)\right) \tag{9}$$

where $r(\cdot)$ is the reshaping operation. From (9), we can see that the HFM could sufficiently fuse the heterogenous feature maps and obtain discriminative features.

After obtaining f_{hf}, we employ the convolution layer to reduce its channel number to 2, and apply the Sigmoid function to generate the segmentation mask. Furthermore, we treat the binary cross-entropy (BCE) loss as the objective function, which optimizes the difference between the ground-truth distribution and the predicted distribution. It is formulated as

$$L = -\frac{1}{T}\sum_{i=1}^{T}[u_i \log(p_i) + (1-u_i)\log(1-p_i)] \tag{10}$$

where T is the total number of pixels, u_i is the ground-truth label for the ith pixel, and p_i is the predicted probability that the ith pixel belongs to cloud.

IV. Experiments

In this section, we evaluate the performance of TransCloudSeg for ground-based cloud image segmentation on the TLCDD and TCDD. We first introduce TLCDD and TCDD, and then, the implementation details of our experiments. Afterwards, we conduct a series of experiments to verify the superiority of TransCloudSeg, and study the influence of several important parameters.

A. Tianjin Normal University Large-Scale Cloud Detection Database (TLCDD)

The TLCDD [31] is a large-scale ground-based cloud detection database, and it is collected over two years from nine provinces of China including Tianjin, Anhui, Sichuan, Gansu, Shandong, Hebei, Liaoning, Jiangsu, and Hainan. Hence, the TLCDD possesses high diversity of cloud images, which makes the experimental results more convincing. The TLCDD consists of 5000 ground-based cloud images with corresponding ground-truth cloud masks manually annotated by meteorologists and cloud-related researchers. To the best of our knowledge, it is the largest public ground-based cloud segmentation database. This database is divided into 4208 training images and 792 test images. The cloud images are acquired by the visual sensor and stored with the resolution of 512×512. Fig. 8 shows some cloud images and their corresponding cloud segmentation masks.

The TCDD [32] is composed of 2300 ground-based cloud images. This dataset has high diversity of cloud shapes, and variability of scenes, which makes the experiment more convincing.

B. Implementation Details and Evaluation Criteria

In the training stage, we perform the preprocessing operations on the ground-based cloud images. Specifically, we apply random resizing with the ratio between 0.5 and 1.5, random cropping, random horizontal flipping, and normalization by the mean and standard deviation values.

We set the total number of epoch to 150 and the batch size to 2 in the experiments. We employ the stochastic gradient descent (SGD) [42] as the optimizer with the initial learning rate of

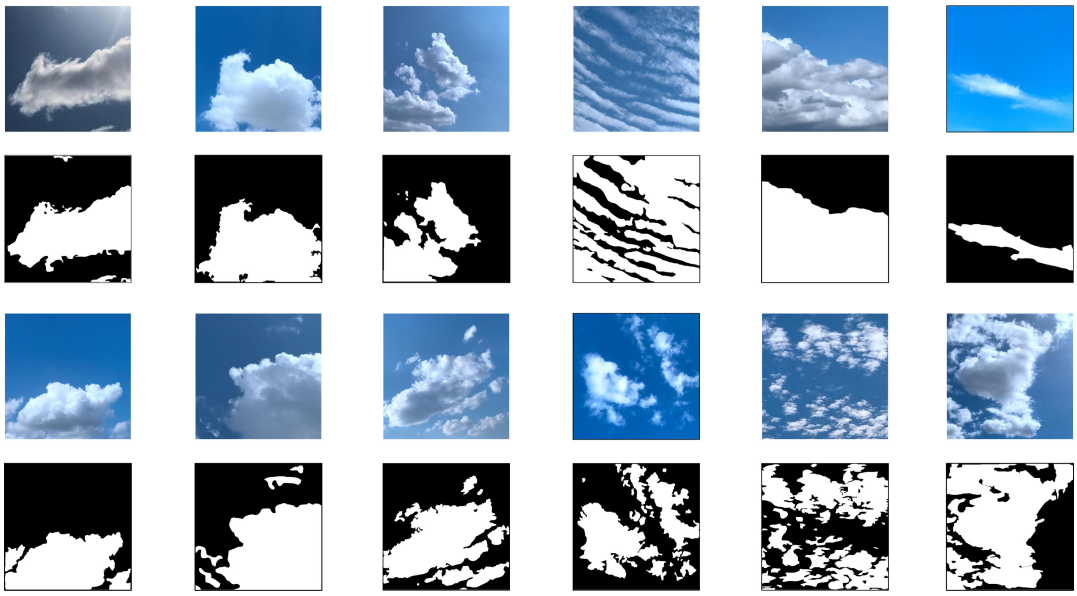

Fig. 8. Some cloud images and the corresponding segmentation masks.

0.01, the weight decay of 0.0001, and the momentum of 0.9. We apply the "poly" learning rate strategy [43], [44], where the learning rate of the current iteration is equal to the initial learning rate multiplied by a factor of $(1 - \frac{\text{iter}}{\text{iter\_num}})^{\text{power}}$. Here, iter and iter_num are the number of current iteration and the total number of iterations respectively, and power $= 0.9$.

We initialize the CNN encoder and the transformer encoder for TransCloudSeg with pretrained model [30]. We set the number of transformer layers in the transformer encoder to 12 and the number of heads in MSA to 16. We select $[Y_3, Y_6, Y_9, Y_{12}]$ as the input of the transformer decoder.

In order to quantitatively evaluate the proposed method, we utilize five evaluation criteria, i.e., Precision (Pre), Recall (Rec), F-score (F_s), Accuracy (Acc), and intersection over union (IoU), which are commonly used in the segmentation task. These evaluation criteria are defined as

$$\text{Pre} = \text{TP}/(\text{TP} + \text{FP}) \tag{11}$$

$$\text{Rec} = \text{TP}/(\text{TP} + \text{FN}) \tag{12}$$

$$F\_s = 2\text{Pre} \times \text{Rec}/(\text{Pre} + \text{Rec}) \tag{13}$$

$$\text{Acc} = (\text{TP} + \text{TN})/(\text{TP} + \text{FP} + \text{TN} + \text{FN}) \tag{14}$$

$$\text{IoU} = \text{TP}/(\text{TP} + \text{FP} + \text{FN}) \tag{15}$$

where TP, TN, FP, and FN indicate true positive, true negative, false positive, and false negative, respectively.

C. Experimental Results

1) Ablation Studies: The advantages of TransCloudSeg are to make full use of the CNN and transformer so as to learn detail information and global information for ground-based cloud image segmentation, simultaneously. These advantages depend on the three main components: two encoders, two decoders, and HFM. To demonstrate their contributions on the performance improvement of TransCloudSeg, we perform ablation experiments on TransCloudSeg, and Table II lists the segmentation results of different ablation methods.

TABLE II
COMPARISON WITH DIFFERENT ABLATION METHODS

Methods	Pre	Rec	F_s	Acc	IoU
C-C	69.19	78.45	69.09	78.25	59.26
T-T	70.54	79.85	70.37	84.72	60.43
C+T-C	72.39	81.03	74.42	86.93	64.01
C+T-T	71.60	80.89	72.25	85.51	63.11
C+T-C+T(A)	73.26	81.35	75.24	88.44	66.76
C+T-C+T(C)	74.28	81.05	77.10	88.96	67.82
TransCloudSeg	**75.39**	**82.53**	**77.27**	**90.55**	**69.48**

a) C-C version: We design C-C as the CNN encoder–decoder architecture. Specifically, the CNN encoder is used to learn the multiscale feature maps from the input cloud image. Then, the CNN decoder utilizes these feature maps to generate segmentation mask.

b) T-T version: We implement T-T using the transformer encoder–decoder architecture. Concretely, the transformer encoder is used to learn the multilevel feature maps from the input cloud image. Then, the transformer decoder applies these feature maps to generate segmentation mask.

c) C+T-C version: We design C+T-C as the CNN–transformer encoder and the CNN decoder, which is similar to TransUNet [30]. The CNN encoder and the transformer encoder are hybridized to learn different feature maps from the input cloud image. These feature maps are reshaped, and then, fed into the CNN decoder to generate the segmentation mask.

d) C+T-T version: We implement C+T-T using the CNN–transformer encoder and the transformer decoder. Specifically, the hybrid form of the CNN–transformer encoder is used to

learn different feature maps from the input cloud image. Then, the transformer decoder utilizes these feature maps to generate segmentation mask.

e) C+T-C+T(A) version: To prove the validity of the HFM, we design C+T-C+T(A) as the hybrid encoder–decoder structure including the CNN and transformer encoders, and the CNN and transformer decoders. After obtaining the feature maps from the CNN and transformer decoders, we directly add them, and then, learn the segmentation mask.

f) C+T-C+T(C) version: C+T-C+T(C) has similar structure with C+T-C+T(A) expect that the two sets of output feature maps from the CNN and transformer decoders are directly concatenated.

From Table II, we can draw several conclusions. First, the proposed TransCloudSeg achieves the best results in all five evaluation criteria, which validates the effectiveness of our method. Second, the methods with the hybrid CNN–Transformer encoder, i.e., "C+T-C" and "C+T-T" exceed "C-C" and "T-T." It demonstrates that the hybrid encoders could extract detailed information and global contextual information simultaneously, which is beneficial to the ground-based cloud image segmentation. Third, TransCloudSeg, "C+T-C+T(A)," and "C+T-C+T(C)" surpass "C+T-C" and "C+T-T" because the hybrid decoders, i.e., CNN decoder and transformer decoder could learn information from different aspects. Finally, TransCloudSeg obtains better results than "C+T-C+T(A)" and "C+T-C+T(C)," which validates the effectiveness of the HFM. It is because the HFM considers the interactions between heterogeneous feature maps.

2) Comparisons With State-of-the-Art Methods: In this section, we compare TransCloudSeg with other state-of-the-art ground-based cloud image segmentation methods. The compared methods include both traditional methods and CNN-based methods. Traditional methods usually apply color values as the thresholds including R/B (0.6) [17], B/R (Otsu) [45], B-R (Otsu) [45], and (B-R)/(B+R) (Otsu) [45]. Here, R and B represent the values of the red and blue channels in the ground-based cloud image, respectively. For example, R/B (0.6) indicates that when the ratio of R and B is smaller than 0.6, the pixel is identified as sky, otherwise as cloud. Furthermore, (Otsu) means that the algorithm proposed by Otsu [46] is used to automatically select the segmentation threshold.

As for the CNN-based methods, we compare TransCloudSeg with five state-of-the-art CNN-based methods for ground-based cloud image segmentation, including FCN [47], CloudSegNet [22], U-Net [48], SegCloud [23], and PSPNet [49].

The comparison results between TransCloudSeg and different methods on TLCDD are shown in Table III. From the table, we can see that the proposed TransCloudSeg outperforms other methods on all five evaluation criteria. Specifically, it surpasses the second best results by 6.59%, 1.03%, 9.95%, 11.91%, and 11.32% in Precision, Recall, F-score, Accuracy, and IoU, respectively. Furthermore, the CNN-based methods generally perform better than the traditional methods. It is because the multilayer structure of CNN could learn complex feature transformations, and therefore, the discriminative features are obtained.

TABLE III
COMPARISON WITH STATE-OF-THE-ART METHODS

Methods	Pre	Rec	F_s	Acc	IoU
R/B (0.6) [17]	69.47	51.59	46.12	71.76	36.48
B/R (Otsu [45]	55.98	77.48	57.26	67.72	45.39
B-R (Otsu) [45]	57.91	61.47	50.80	66.92	38.34
(B-R)/(B+R) (Otsu) [45]	63.00	69.60	59.11	73.61	47.23
FCN [47]	63.20	73.77	57.00	66.49	46.75
CloudSegNet [22]	64.46	77.61	57.79	64.59	47.78
U-Net [48]	68.80	80.43	67.32	74.13	58.16
SegCloud [23]	68.35	81.50	66.95	73.06	57.76
PSPNet [49]	68.74	77.75	67.00	78.64	57.43
TransCloudSeg	**75.39**	**82.53**	**77.27**	**90.55**	**69.48**

TABLE IV
RUN TIME COMPARISON OF DIFFERENT METHODS

Methods	Running Times (ms)
R/B (0.6)	11.31
B/R (Otsu)	13.94
B-R (Otsu)	12.98
(B-R)/(B+R) (Otsu)	14.89
FCN	51.56
CloudSegNet	40.56
U-Net	59.39
SegCloud	60.75
PSPNet	55.94
TransCloudSeg	87.21

TABLE V
COMPARISON OF THE RECEPTIVE FIELD SIZES OF THE ENCODER FROM DIFFERENT CNN-BASED METHODS

Methods	Receptive Field Sizes
FCN	100×100
CloudSegNet	22×22
U-Net	140×140
SegCloud	76×76
PSPNet	427×427
TransCloudSeg	Global

We present the runtime comparison of different methods as shown in Table IV. From the table, we can see that the traditional methods have less runtime than the CNN-based methods. It is because the CNN-based methods require a large number of parameters to represent complex features. Moreover, the network with large number of parameters can learn more complex features, but requires more running time.

The proposed method makes tradeoff between performance and runtime, and the runtime to process one ground-based cloud image is 87.21 ms. The acquisition device takes approximately 2 min to acquire one ground-based cloud image. Therefore, our method could satisfy the practical requirements. Note that all the runtime tests are performed with the workstation equipped with E5-1620V4 CPU, 32-GB RAM and NVIDIA RTX 2080Ti 12-GB GPUs.

We list the comparison of the receptive field sizes of the encoder from different CNN-based methods in Table V. The

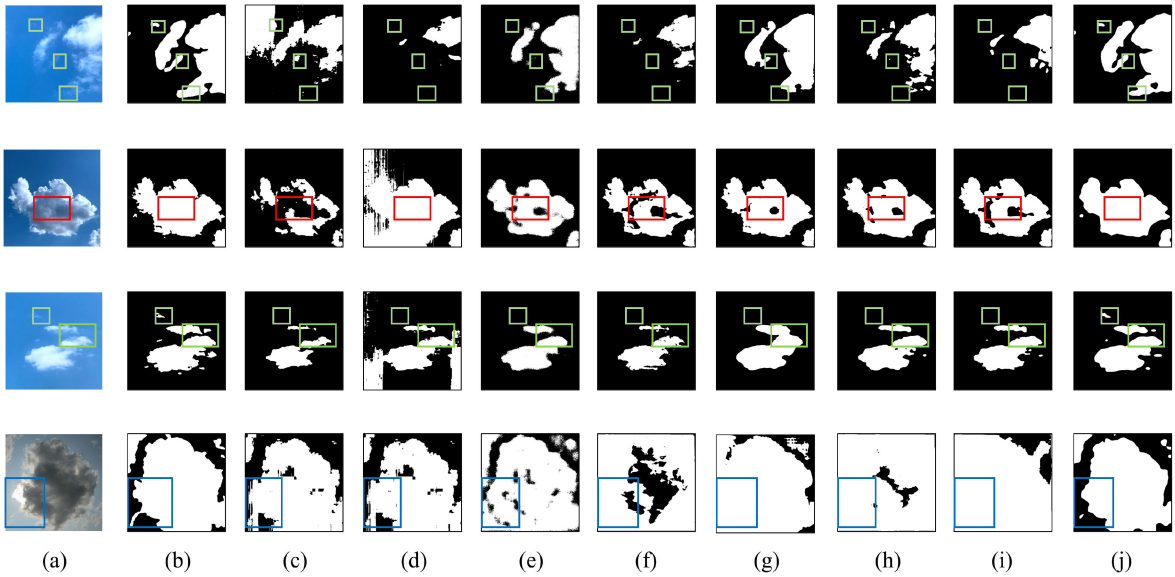

Fig. 9. Segmentation masks of different methods. (a) Cloud images. (b) Ground-truth segmentation masks. (c) R/B (0.6). (d) (B-R)/(B+R)(Otsu). (e) FCN. (f) U-Net. (g) PSPNet. (h) CloudSegNet. (i) SegCloud. (j) TransCloudSeg.

TABLE VI
COMPARISON WITH STATE-OF-THE-ART METHODS ON THE TCDD

Methods	Pre	Rec	F_s	Acc	IoU
R/B (0.6)	70.81	42.53	42.33	74.85	36.63
B/R (Otsu)	51.35	79.74	53.97	62.32	42.36
B-R (Otsu)	51.52	57.29	45.65	61.54	32.92
(B-R)/(B+R) (Otsu)	50.41	68.32	55.88	70.51	43.82
FCN	70.41	72.54	61.02	79.16	51.06
CloudSegNet	69.60	71.57	60.91	80.05	51.02
U-Net	75.98	77.97	70.47	80.63	61.12
SegCloud	77.70	78.91	71.89	82.49	63.26
PSPNet	64.52	77.56	70.34	79.31	61.08
TransCloudSeg	**81.35**	**84.81**	**81.72**	**92.28**	**74.21**

proposed TransCloudSeg has a global receptive field regardless of the size of the input images. From Tables III and V, we can see that the proposed method obtains a better performance than these CNN-based methods.

To demonstrate the robustness of the proposed method, we also perform a series of comparison experiments on the TCDD. The comparison results of different methods on TCDD are shown in Table VI. As we can see from the table, the proposed TransCloudSeg achieves the best results on all five evaluation criteria once again. The experimental results demonstrate the robustness of the proposed method.

3) Visualization: To qualitatively prove the effectiveness of TransCloudSeg, we visualize some segmentation masks of the comparison methods in Fig. 9. We can see from the figure that our method achieves superior performance than other methods, especially for thin clouds, thick clouds, and the areas affected by illumination. Furthermore, the CNN-based methods generally perform better than the traditional methods.

In Fig. 9, the green, red, and blue rectangles indicate thin clouds, thick clouds, and areas affected by illumination. As for the thin clouds, they are easily ignored, and as for the thick clouds, they are usually classified into the sky labels. While the proposed TransCloudSeg could classify them correctly. In the fourth row of Fig. 9, the sample is affected by illumination, which is a challenge for cloud image segmentation. From the blue rectangles in the fourth row of Fig. 9, we can see that most methods misclassify the sky as clouds because the illumination causes the sky to appear white. While our method ensures the promising performance with large illumination variations.

4) Analysis of Different Training Sample Proportion: The proportion of training and test samples in Table III is about 4:1. To demonstrate the generalizability of the model, we add two sets of experiments with different proportions of training and test samples on TLCDD. We list the comparison performance of different methods with different proportions of training and test samples in Table VII. From the table, our method still outperforms other methods when the number of training samples is different, which proves the generalization of the model. Furthermore, the traditional methods utilize the prior information to segment the cloud images, which is irrelevant to the number of training samples.

5) Parameters Analysis: We investigate the impact of several important parameters for our method. They are the number of skip connections in the CNN decoder, and the level number of the aggregated feature maps in the transformer decoder.

a) Number of skip connections: In the CNN decoder of TransCloudSeg, we utilize the skip connections to integrate multiscale feature maps from the CNN encoder, which recovers the detail information. We conduct the experiments with different numbers of skip connections as shown in Fig. 10. Here, "$Skip = 0, 1, 2, 3$" indicates that we treat $[f_3]$, $[f_2, f_3]$, $[f_1, f_2, f_3]$, and $[f_0, f_1, f_2, f_3]$ as the inputs of the CNN decoder, respectively. From the figure, we can see that the performance of

TABLE VII
COMPARISON WITH STATE-OF-THE-ART METHODS WITH DIFFERENT PROPORTIONS OF TRAINING AND TEST SAMPLES

Train:Test	Methods	Pre	Rec	F_s	Acc	IoU
2:1	R/B (0.6)	66.26	63.02	64.21	73.45	49.35
	B/R (Otsu)	64.34	77.48	59.15	73.53	53.54
	B-R (Otsu)	64.36	62.17	63.24	68.26	43.82
	(B-R)/(B+R) (Otsu)	66.93	72.55	65.64	75.15	53.34
	FCN	60.35	75.28	55.91	67.12	45.68
	CloudSegNet	55.14	68.89	56.64	59.45	45.56
	U-Net	67.12	76.93	64.42	66.26	55.72
	SegCloud	67.28	78.06	65.78	69.68	54.69
	PSPNet	67.64	73.56	63.48	75.99	55.16
	TransCloudSeg	**71.23**	**81.68**	**70.74**	**84.56**	**62.48**
1:1	R/B (0.6)	67.13	63.58	64.62	72.23	50.41
	B/R (Otsu)	66.19	76.46	60.51	73.72	53.15
	B-R (Otsu)	65.17	62.97	62.83	68.85	44.56
	(B-R)/(B+R) (Otsu)	66.55	72.77	65.32	75.83	53.89
	FCN	52.85	73.02	49.43	57.20	44.09
	CloudSegNet	51.83	68.47	55.52	58.39	44.79
	U-Net	65.62	68.51	61.46	58.7	53.38
	SegCloud	61.42	73.48	60.31	68.28	52.76
	PSPNet	65.83	70.54	63.07	72.56	53.78
	TransCloudSeg	**70.93**	**80.16**	**68.61**	**78.72**	**58.10**

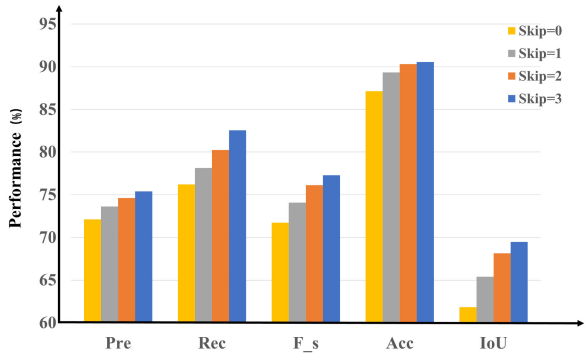

Fig. 10. Performance of TransCloudSeg with different number of skip connections in the CNN decoder.

five evaluation criteria improves as the number of skip connections increases. Hence, we set the number of the skip connections to 3 in our experiments.

b) Level Number of the Aggregated Feature Maps: In the transformer decoder of TransCloudSeg, we apply the weight aggregation to integrate the multilevel feature maps. We conduct the experiments with different level number of the aggregated feature maps as shown in Fig. 11. Here, "$Layer = 2, 4, 6$" indicates that we treat $[Y_6, Y_{12}]$, $[Y_3, Y_6, Y_9, Y_{12}]$, and $[Y_2, Y_4, Y_6, Y_8, Y_{10}, Y_{12}]$ as the inputs of the transformer decoder, respectively. As can be seen from the figure, the best performance of the proposed TransCloudSeg is achieved when the level number of aggregated feature maps is set to 4.

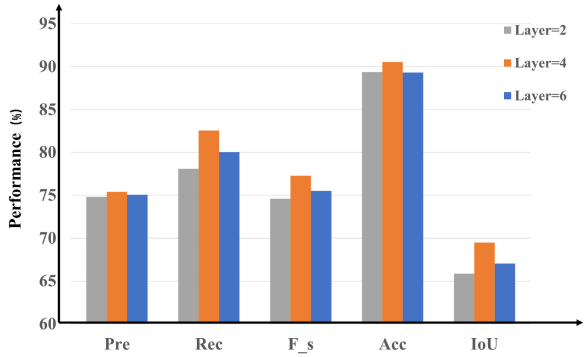

Fig. 11. Performance of TransCloudSeg with different level number of the aggregated feature maps in the transformer decoder.

6) Computational Complexity Analysis: The computational complexity of the proposed TransCloudSeg is dominated by CNN and MSA. According to [24], [41], and [50], the jth layer computational complexity is $O(C_{j-1} \cdot C_j \cdot k^2 \cdot r^2)$, where C_{j-1} is the number of input channels of the jth layer, C_j is the number of output channels of the jth layer, i.e., the number of filters of the jth layer convolution operation, k is the spatial size of each convolution filter, and r is the spatial size of output feature maps. The computational complexity of each layer of MSA is $O(L \cdot D^2 + L^2 \cdot D)$, where L is the length of sequences and D is the hidden channel size.

V. CONCLUSION

In this article, we have proposed TransCloudSeg for the ground-based cloud image segmentation. To the best of our knowledge, we are the first to apply transformer to the ground-based cloud image segmentation task. Specifically, we present the CNN encoder and the transformer encoder to extract detailed information and global information from ground-based cloud images. Meanwhile, we design two decoders, i.e., CNN decoder and transformer decoder to integrate the multiscale and multilevel feature maps extracted from the two encoders. Furthermore, we propose HFM to effectively exploit the information contained in the heterogeneous feature maps from the CNN and transformer decoders in order to generate accurate segmentation mask. Extensive experimental results have demonstrated the effectiveness of the proposed TransCloudSeg on TLCDD and TCDD. In the future, we will try to build light-weight transformer-based architecture to reduce the number of parameters so as to decrease the complexity and speed up the training process. Furthermore, we will study some specific scenes for ground-based cloud image segmentation, such as nighttime, etc.

REFERENCES

[1] A. Heinle, A. Macke, and A. Srivastav, "Automatic cloud classification of whole sky images," *Atmospheric Meas. Techn.*, vol. 3, no. 3, pp. 557–567, 2010.

[2] Y. Wang, C. Wang, C. Shi, and B. Xiao, "A selection criterion for the optimal resolution of ground-based remote sensing cloud images for cloud classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1358–1367, Mar. 2019.

[3] S. Liu, L. Duan, Z. Zhang, X. Cao, and T. S. Durrani, "Multimodal ground-based remote sensing cloud classification via learning heterogeneous deep features," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 11, pp. 7790–7800, Nov. 2020.

[4] D. Hong, N. Yokoya, J. Chanussot,, and X. X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1923–1938, Apr. 2019.

[5] C. Shi, Y. Zhou, B. Qiu, D. Guo, and M. Li, "CloudU-Net: A deep convolutional neural network architecture for daytime and nighttime cloud images' segmentation," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 10, pp. 1688–1692, 2021.

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

[7] A. Taravat, F. D. Frate, C. Cornaro, and S. Vergari, "Neural networks and support vector machine algorithms for automatic cloud classification of whole-sky ground-based images," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 3, pp. 666–670, Mar. 2015.

[8] M. C. Allmen and W. P. Kegelmeyer Jr., "The computation of cloud base height from paired whole-sky imaging cameras," *Mach. Vis. Appl.*, vol. 9, no. 4, pp. 160–165, 1997.

[9] E. Kassianov, C. N. Long, and J. Christy, "Cloud-base-height estimation from paired ground-based hemispherical observations," *J. Appl. Meteorol. Climatol.*, vol. 44, no. 8, pp. 1221–1233, 2005.

[10] N. B. Blum et al., "Cloud height measurement by a network of all-sky imagers," *Atmospheric Meas. Techn.*, vol. 14, no. 7, pp. 5199–5224, 2021.

[11] S. Liu, C. Wang, B. Xiao, Z. Zhang, and X. Cao, "Tensor ensemble of ground-based cloud sequences: Its modeling, classification, and synthesis," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 5, pp. 1190–1194, Sep. 2013.

[12] S. Liu, C. Wang, B. Xiao, Z. Zhang, and Y. Shao, "Salient local binary pattern for ground-based cloud classification," *Acta Meteorologica Sinica*, vol. 27, no. 2, pp. 211–220, 2013.

[13] S. Liu, Z. Zhang, B. Xiao, and X. Cao, "Ground-based cloud detection using automatic graph cut," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 6, pp. 1342–1346, Jun. 2015.

[14] C. Shi, Y. Wang, C. Wang, and B. Xiao, "Ground-based cloud detection using graph model built upon superpixels," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 5, pp. 719–723, May 2017.

[15] S. Dev, Y. H. Lee, and S. Winkler, "Color-based segmentation of sky/cloud images from ground-based cameras," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 1, pp. 231–242, Jan. 2017.

[16] W. Li, Z. Zou, and Z. Shi, "Deep matting for cloud detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8490–8502, Dec. 2020.

[17] C. N. Long, J. M. Sabburg, J. Calb, and D. Pages, "Retrieving cloud characteristics from ground-based daytime color all-sky images," *J. Atmospheric Ocean. Technol.*, vol. 23, no. 5, pp. 633–653, 2006.

[18] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

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

[20] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza,, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5966–5978, 2020.

[21] D. Hong et al., "More diverse means better: Multimodal deep learning meets remote-sensing imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4340–4354, May 2021.

[22] S. Dev, A. Nautiyal, Y. H. Lee, and S. Winkler, "CloudSegNet: A deep network for nychthemeron cloud image segmentation," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 12, pp. 1814–1818, 2019.

[23] W. Xie et al., "SegCloud: A novel cloud image segmentation model using a deep convolutional neural network for ground-based all-sky-view camera observation," *Atmospheric Meas. Techn.*, vol. 13, no. 4, pp. 1953–1961, 2020.

[24] A. Vaswani, N. Shazeer, N. Parmar, and J. Uszkoreit, "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.

[25] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*. [Online]. Available: https://arxiv.org/abs/2010.11929

[26] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," in *Proc. Int. Conf. Learn. Representations*, 2021, pp. 1–26.

[27] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2019, *arXiv:1810.04805*. [Online]. Available: https://arxiv.org/abs/1810.04805

[28] S. Zheng et al., "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 6881–6890.

[29] O. Petit, N. Thome, C. Rambour, L. Themyr, T. Collins, and L. Soler, "U-net transformer: Self and cross attention for medical image segmentation," in *Proc. Int. Workshop Mach. Learn. Med. Imag.*, 2021, pp. 267–276.

[30] J. Chen et al., "Transunet: Transformers make strong encoders for medical image segmentation," 2021, *arXiv:2102.04306*. [Online]. Available: https://arxiv.org/abs/2102.04306

[31] Z. Zhang, S. Wang, S. Liu, X. Cao, and T. S. Durrani., "Ground-based remote sensing cloud detection using dual pyramid network and encoder decoder constraint," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5620912, doi: 10.1109/TGRS.2022.3163917.

[32] Z. Zhang, S. Wang, S. Liu, B. Xiao, and X. Cao, "Ground based cloud detection using multiscale attention convolutional neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 8019605, doi: 10.1109/LGRS.2021.3106337.

[33] Z. Zhou et al., "A novel ground-based cloud image segmentation method by using deep transfer learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 8010805, doi: 10.1109/LGRS.2021.3072618.

[34] K. Han et al., "A survey on visual transformer," 2020, *arXiv:2012.12556*. [Online]. Available: https://arxiv.org/abs/2012.12556

[35] B. Wu et al., "Visual transformers: Token-based image representation and processing for computer vision," 2020, *arXiv:2006.03677*. [Online]. Available: https://arxiv.org/abs/2006.03677

[36] L. Yuan et al., "Tokens-to-token ViT: Training vision transformers from scratch on imagenet," 2021, *arXiv:2101.11986*. [Online]. Available: https://arxiv.org/abs/2101.11986

[37] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," 2021, *arXiv:2105.15203*. [Online]. Available: https://arxiv.org/abs/2105.15203

[38] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," 2021, *arXiv:2103.14030*. [Online]. Available: https://arxiv.org/abs/2103.14030

[39] F. Zhu, Y. Zhu, L. Zhang, C. Wu, Y. Fu, and M. Li, "A unified efficient pyramid transformer for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2667–2677.

[40] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5607514, doi: 10.1109/TGRS.2021.3095166.

[41] D. Hong et al., "SpectralFormer: Rethinking hyperspectral image classification with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5518615.

[42] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1139–1147.

[43] L. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*. [Online]. Available: https://arxiv.org/abs/1706.05587

[44] F. Zhang et al., "ACFNet: Attentional class feature network for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6798–6807.

[45] J. Yang, W. Lu, Y. Ma, W. Yao, and Q. Li, "An automatic ground-based cloud detection method based on adaptive threshold," *J. Appl. Meteorological Sci.*, vol. 20, no. 6, pp. 713–721, 2009.

[46] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

[47] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[48] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervention*, 2015, pp. 234–241.

[49] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2881–2890.

[50] K. He and J. Sun, "Convolutional neural networks at constrained time cost.," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5353–5360.

Shuang Liu (Senior Member, IEEE) received the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2014 .

She is currently a Professor with Tianjin Normal University, Tianjin, China. She has authored and coauthored more than 50 articles in major international journals and conferences. Her research interests include remote sensing, computer vision, and deep learning.

Jiafeng Zhang is currently working toward the master degree in information and communication engineering with Tianjin Normal University, Tianjin, China.

His research interests include ground-based cloud analysis and deep learning.

Zhong Zhang (Senior Member, IEEE) received the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China.

He is a Professor with Tianjin Normal University, Tianjin, China. He has authored and co-authored about 110 articles in international journals and conferences such as IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE TRANSACTIONS ON FUZZY SYSTEMS, *Pattern Recognition*, IEEE TRANSACTIONS ON CIRCUITS SYSTEMS VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, *Signal Processing* (Elsevier), IEEE Conference on Computer Vision and Pattern Recognition), International Conference on Pattern Recognition, and International Conference on Image Processing. His research interests include remote sensing, computer vision, and deep learning.

Xiaozhong Cao received the Ph.D. degree in automatic control theory and application from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 1996.

He is currently a Professor with Meteorological Observation Centre, China Meteorological Administration. His current research interests include the theory of meteorological observation and climate change, and the automatic meteorological observation.

Tariq S. Durrani (Life Fellow, IEEE) received the Ph.D. degree from the University of Southampton, U.K., in 1970.

He is a Research Professor with the University of Strathclyde, Glasgow, U.K. His research interests include artificial intelligence, signal processing, and technology management. He has authored 350 publications and supervised 45 Ph.Ds.

Prof. Durrani is a Fellow of the U.K. Royal Academy of Engineering, Royal Society of Edinburgh, IET, and the Third World Academy of Sciences. He was elected Foreign Member of the Chinese Academy of Sciences and the U.S. National Academy of Engineering in 2021 and 2018, respectively.