BSANet: A Bilateral Segregation and Aggregation Network for Real-time Sky/Cloud Segmentation

Yijie Li*, Hewei Wang*, *Student Member, IEEE,* Shaofan Wang, *Member, IEEE,* Yee Hui Lee, *Senior Member, IEEE,* and Soumyabrata Dev, *Member, IEEE*

***Abstract*—Segmenting cloud from intensity images is an essential research topic at the intersection of atmospheric science and computer vision, which plays a vital role in weather forecasts, environmental monitoring, and climate evolution analysis. The ground-based sky/cloud image segmentation can help to extract the cloud from the original image and analyze the shape or additional features. The early approaches are mainly based on traditional methods and have limited segmentation performance on both day and night instances. After the advent of deep learning, many researches have been conducted to adopt convolutional neural networks (CNNs) to perform the end-to-end training of a segmentation model. However, these early CNN-based designs usually use a great number of parameters to guarantee accuracy, leading to a slow inference speed. In this paper, we introduced a novel sky/cloud segmentation network named Bilateral Segregation and Aggregation Network (BSANet) with 16.37 MBytes, which can reduce 70.68% of model size and achieve almost the same performance as the state-of-the-art method. After the deployment via TensorRT, BSANet-large configuration can achieve 392 fps in FP16, while BSANet-lite can achieve 1390 fps. Additionally, we proposed a novel and fast pre-training strategy for sky/cloud segmentation which can improve the accuracy of segmentation when ImageNet pre-training is not available. In the spirit of reproducible research, the model code, dataset, and results of the experiments in this paper are available at: https://github.com/Att100/BSANet-cloudseg.**

***Index Terms*—cloud segmentation, deep learning, pre-training, bilateral segregation and aggregation module (BSAM)**

I. Introduction

CLOUD/SKY relationship and distribution understanding have profound significance for the atmospheric science area. With the rise of computer vision and machine learning technology, it has been devoted to being applied in several interdisciplinary areas related to meteorology estimation and weather prediction [1]–[4]. This information can provide us with not only the status of the cloud but also ample low-level features. To retrieve this information, a common approach is to analyze the picture captured by a meteorological satellite in near-earth orbit, but this method is often expensive and requires large data storage. In recent years, the ground-based sky/cloud segmentation is more and more popular [5]–[7], and many related datasets have been introduced for researchers to study and reference, for example, SWIMSEG [8], SWINSEG [9], and SWINySEG [10]. After the release of these data, many approaches start to use traditional methods to split the cloud and sky parts of images. Due to the success of convolutional neural networks (CNNs) as well as the availability of annotated cloud images, the accuracy of sky/cloud segmentation is improved significantly, and the procedure is also simplified. In order to achieve better performance on the segmentation, many researchers begin to use the model with more parameters, but its corresponding segmentation speed is reduced. Therefore, the research focus today should be transferred to the trade-off of accuracy and inference speed.

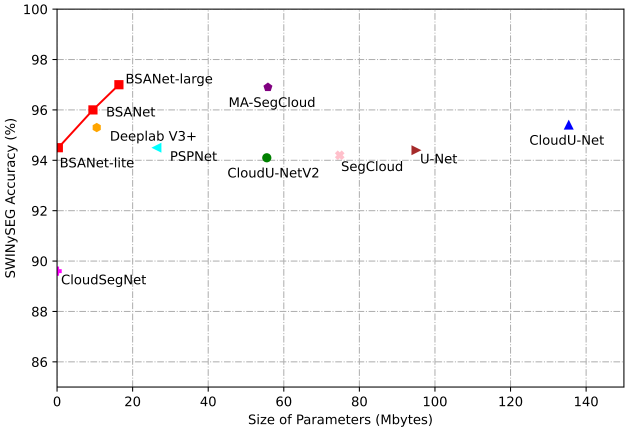

Fig. 1: Model Size vs. SWINySEG Accuracy. Our proposed BSANet successfully achieves a balance between the model size and accuracy. BSANet-large can achieve 97.0% of accuracy in SWINySEG with 16.34 million bytes of parameters, while BSANet-lite can achieve 94.5% of accuracy with only 0.34 million bytes of parameters.

In deep learning, the sky/cloud segmentation can be considered as binary semantic segmentation, after the fully convoluted neural network (FCN) [11] has been introduced for semantic segmentation, many advanced approaches have been released in the past decade. The research in real-time segmentation has played an essential role during the period whose most popular approaches can be split into two categories, the first category is lightweight backbone and decoder, such as DeepLab series [12]–[14], and the second category is redesigning the encoder-decoder structure and multiple resolution inputs, such as ICNet [15], BiseNetV1 [16], and BiseNetV2 [17].

Manuscript received Month Day, 2022; revised Month Day, 2022; accepted Month Day 2022.

* Authors contributed equally to this research.

Y. Li and H. Wang are with the Beijing-Dublin International College, Beijing University of Technology, Beijing 100124, China (e-mail: yijie.li@ucdconnect.ie, hewei.wang@ucdconnect.ie).

S. Wang is with Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China (email: wangshaofan@bjut.edu.cn).

Y. H. Lee is with Nanyang Technological University Singapore, Singapore (e-mail: EYHLee@ntu.edu.sg).

S. Dev is with the School of Computer Science, University College Dublin (e-mail: soumyabrata.dev@ucd.ie).

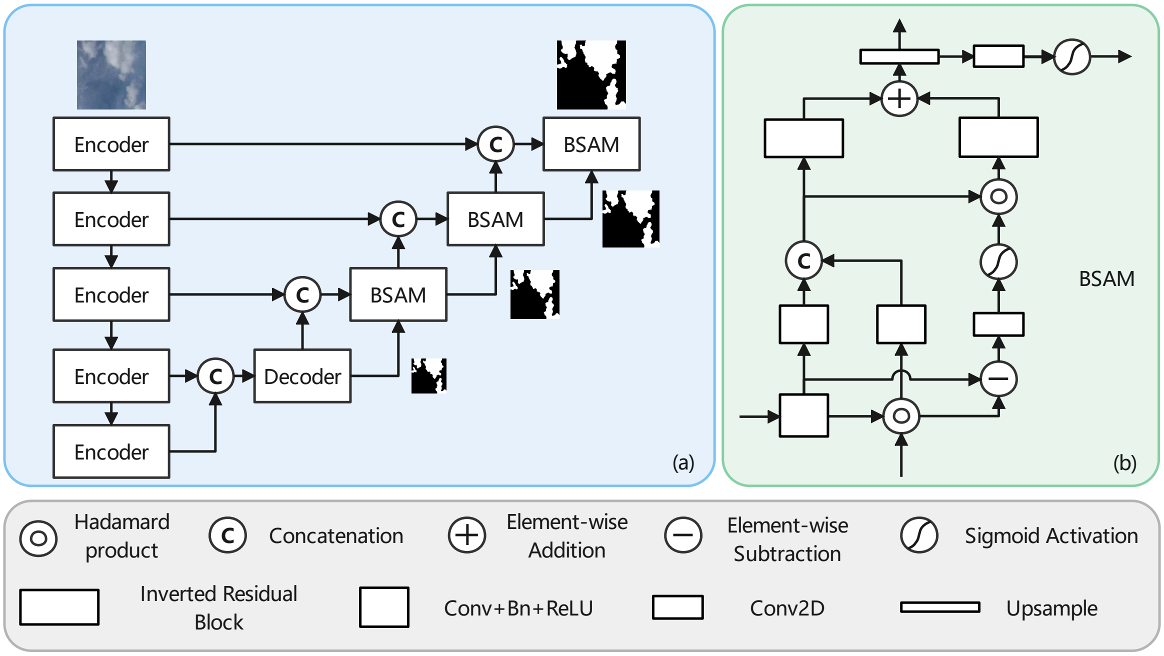

Fig. 2: The overall architecture of BSANet and BSAM. (a) illustrate the overall pipeline of BSANet and (b) indicate the detailed design of bilateral segregation and aggregation module (BSAM). The procedure between the output of the model and the segmentation mask has been omitted in this figure.

In our work, we take the first designing approach and adopt a light-weight backbone, including MobileNetv2 [18], and EfficientNet-B0 [19] with our proposed bilateral segregation and aggregation module (BSAM). Our BSANet adopts a U-Net [20] based encoder-decoder structure, which takes the feature maps of the multiple-level output of the backbone network and generates the pixel-wise prediction through a series of BSAM decoders. The BSAM encoder process the feature map element of cloud and sky separately by using the prediction map of the previous stage, which can improve the ability of information aggregation. Not like other previous work, we apply supervision in the last three stages, which can reduce the time consumption of model convergence. The ablation study was conducted to analyze the efficiency of our proposed model and pre-training strategy, which will be discussed in the following section. In the smallest configuration of BSANet (BSANet-lite), we modified the parameter configuration of MobileNetV2, and utilized our proposed new pre-training strategy with SWINySEG to perform pre-training instead of using ImageNet [21].

The main contributions of BSANet is threefold:

- We introduced a real-time CNN-based approach, the BSANet, which can achieve nearly the same performance and reduce 70.68% of parameters by comparing with state-of-the-art approach.
- We proposed a novel BSAM decoder for sky/cloud segmentation which guarantee the performance under the light-weight and real-time conditions.
- We designed an novel pre-training strategy for sky/cloud segmentation when ImageNet pre-training is not available.

II. Related Works

Several techniques were developed to solve the sky/cloud image segmentation task, which can be broadly classified into traditional computer vision methods [22]–[24] and deep learning methods [10], [25], [26]. As for some of these traditional methods, they utilize color features, pre-defined fixed convolution filters, and a gradient of pixels to handle the image segmentation problem. For instance, in 2014, Dev *et al.* [23] adopted principal component analysis (PCA) and fuzzy cluster to evaluate the color model with the goal of capturing the most color variation between the cloud and the sky. While the overall distribution of the sky can be captured using those methods, numerous details are lost during the extraction process, leading to a sub-optimal segmentation accuracy. Additionally, the generated binary masks of the sky and cloud images are ill-defined and do not adhere to the image boundaries. The generation of binary cloud masks has improved with the introduction of deep learning techniques for sky/cloud image segmentation tasks. In 2019, Dev *et al.* [10] introduced a novel model, CloudSegNet, which makes use of a standard FCN structure to down-sample the original images to compress information into high-dimensional feature maps, then perform a series of up-sample operations to recover the segmentation results. These operations noticeably improve the overall quality, and the details of the boundary are more accurate than previous methods. Dev *et al.* [26] introduce another method for multi-label sky/cloud segmentation in the same year. This method divides each sky/cloud image's label into three classes: thin cloud, thick cloud, and sky. They trained a multi-class U-Net to produce predictions, enabling researchers to analyze the three-class segmentation map with greater accuracy.

In 2021, Shi *et al.* [27] proposed the UCloud-Net, a U-Net based encoder-decoder network and a series of fully connected conditional random field (CRF) layers are adopted at the final stage to refine the segmentation results from U-Net. They also use dilated convolution in U-Net to increase the receptive field. In the same year, Shi *et al.* perfected their previous work and proposed the UCloud-NetV2 [28], which replaced the 'upsampling' in CloudU-Net with 'bilinear upsampling' and equipped their model with non-local position and channel attention. They also use Radam optimizer in their training to resolve the problem of falling into a local optimum solution. In 2022, Zhang *et al.* [29] introduced a novel approach for sky/cloud segmentation named MS-SegCloud, which is a U-Net based structure like previous approaches. However, it is integrated with convolutional block attention module (CBAM) [30], squeeze-and-excitation module (SEM) [31], and asymmetric convolution module, which is proved to have better performance than previous approaches.

III. BSANet

Our BSANet is designed based on the U-Net [20] structure. We use different backbone networks to extract high-dimensional features in our experiments. In decoders, we use basic 2D convolution blocks in the first stage, and our proposed Bilateral Segregation and Aggregation Module (BSAM) is used in the remaining four stages. The architecture of BSANet is shown in Fig. 2 (a), and the BSAM is illustrated in Fig. 2 (b). As for the training strategy, we use deep supervision [32] in which objective function is applied to outputs of all stages. Additionally, we proposed a new pre-training strategy for sky/cloud segmentation which is used in the training of our BSANet-lite. The details of our model will be discussed in the following sections.

A. Backbone Networks

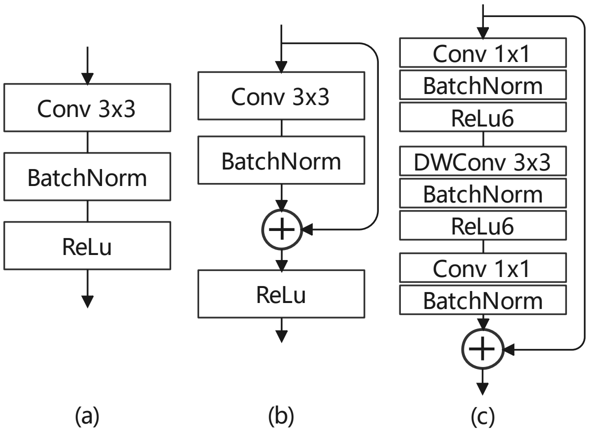

Fig. 3: Comparison of popular and basic modules used in backbone networks. (a) indicate straight structure (b) is a residual block (c) illustrates the inverted-residual [18]

We equipped our BSANet with MobileNetV2 [18] and EfficientNet-B0 [19]. The most widely used backbone network for segmentation usually contains derivatives of three fundamental modules: straight structure, residual block, and inverted residual [18], shown in Fig. 3. In MobileNetV2, the basic block is inverted-residual which first goes through a group of Conv2D-BatchNorm-ReLU to expand the channels, and then passes features to a similar group of layers but replaces the original Conv2D with depth-wise Conv2D. Finally, a 1×1 convolution layer and batch-normalization layer are used to reduce the number of channels to match the input and then apply an element-wise addition. The MobileNetV2 is a lite-weight image classification model that can be used for inference on mobile-device, the depth-wise convolution used in MobileNetV2 set group number to input channels which significantly reduce the number of parameters without losing too much accuracy. The EfficientNet is a model whose hyper-parameters are determined by large-scale neural network architecture searching, and the best configuration is selected. In our design, we use MobileNetV2 for BSANet and EfficientNet for BSANet-large. As for BSANet-lite, we modified the predefined model configuration with a smaller one and performed our proposed pre-training strategy before training on SWINySEG.

B. Bilateral Segregation and Aggregation Module (BSAM)

The Bilateral Segregation and Aggregation Module (BSAM) is a lightweight decoder that we introduced for sky/cloud segmentation, as shown in Fig. 2 (b). This module is used in the 2nd, 3rd, and 4th stages. The sky/cloud segmentation can be considered as a binary semantic segmentation that only contains sky and cloud. This problem setting motivates us to design a module to process the features of different categories separately after segregating the features belonging to different categories based on rough segmentation results output by the previous stage and then aggregating the two features. The BSAM needs two inputs, a concatenation of the U-Net shortcut and feature maps from the previous layer, c_{i-1}, and the segmentation prediction of the previous stage, s_{i-1}. Then the main branch can be formulated as follow:

$$f_i = Cat(Conv(c_{i-1}), Conv(c_{i-1} \times s_{i-1})) \tag{1}$$

f_i represent features of foreground (sky), and Cat indicates the channel level concatenation and the background mask can be given as:

$$m_i = Sigmoid(Conv(c_{i-1} \times (1 - s_{i-1}))) \tag{2}$$

we then apply the background mask on the core branch to segregate weighted features to represent the features of the background, the calculation formula is shown as follows:

$$b_i = f_i \times m_i \tag{3}$$

finally, we use element-wise addition to aggregate f_i and background (sky) b_i to get the output variable o_i, the formula is shown as below:

$$o_i = Conv(Conv(b_i) + Conv(f_i)) \tag{4}$$

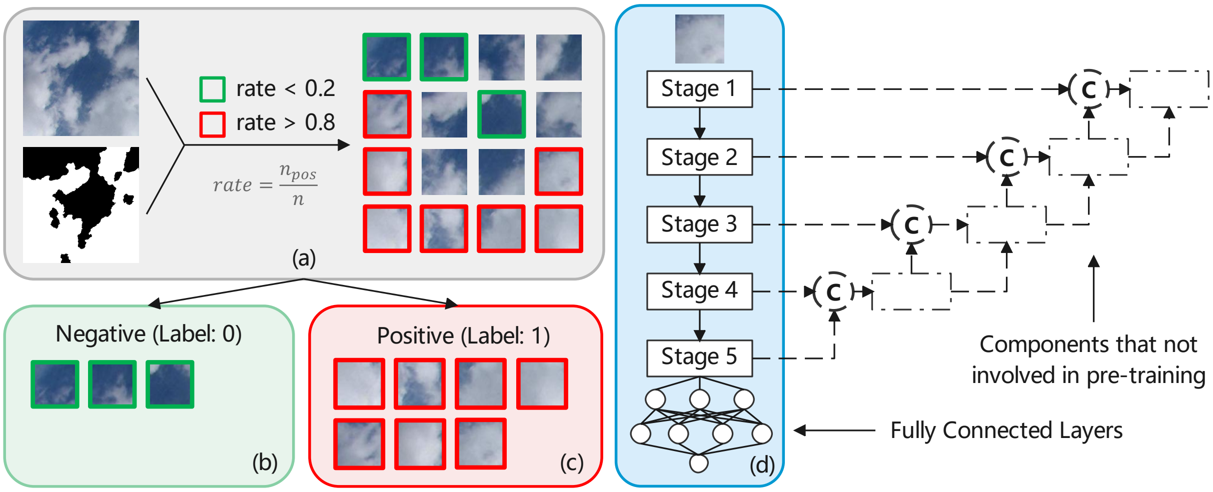

Fig. 4: Schematic diagram of our proposed SWINySEG-based pre-training. (a) illustrates the positive and negative sample generation process. (b) indicates the negative samples. (c) is positive samples. (d) represents the modules involved in pre-training.

The stage prediction s_i can also be generated after a convolution layer and sigmoid activation shown below:

$$s_i = Sigmoid(Conv(o_i)) \tag{5}$$

In order to reduce the amount of parameters and computation complexity, we also adopt inverted residual blocks in BSAM, which didn't show any visible drop in performance.

C. SWINySEG-based Pre-training

Pre-training is a commonly used strategy for many downstream tasks in the computer vision area. Because it is difficult for people to train a semantic segmentation or object detection from scratch, and pre-training is introduced to first train the backbone model on ImageNet [21] to improve the performance and reduce the time needed to converge. In BSANet and BSANet-large, we can reuse the pre-trained weights directly because we didn't apply any modification. In BSANet-lite, we modify the configuration and the backbone needs to be pre-trained from scratch. However, pre-training a model on ImageNet is time-consuming and expensive, which motivates us to design a new strategy, shown in Fig. 4, to complete such a process by reusing the SWINySEG dataset.

We first iterate the image set of SWINySEG; for each image, we split them into 16 patches, and for each patch, we use their corresponding ground truth to generate labels as follows:

$$rate = \frac{n_{pos}}{n} \tag{6}$$

where n_{pos} indicates the number of pixel with the positive label, and n represents the total number of pixels which should be a constant value. If $rate > 0.8$, we label this patch as positive, and if $rate < 0.2$, we label it with the negative sample. And we ignore the image with rate in the range between 0.2 and 0.8.

Note that when pre-training, we remove all decoders and the connections between them and the backbone, then a fully connected layer is added after the backbone.

D. Loss Functions

In our research, We use binary cross entropy (BCE) as the loss function in training of BSANet-lite and BSANet. When training BSANet-large, we use the combination of BCE loss and IOU loss. These loss functions can be defined as follow:

$$\mathcal{L}_{bce}(p,y) = -\frac{1}{N} * \sum_{j=0}^{N}(y_j * \log p_j + (1-y_j) * \log(1-p_j)) \tag{7}$$

$$\mathcal{L}_{iou}(p,y) = 1 - \frac{1}{n}\sum_{j=0}^{N}(\frac{y_j \times p_j}{y_j + p_j - y_j \times p_j}) \tag{8}$$

then our total training loss under deep supervision can be formulated as:

$$\mathcal{L}(p,y) = \sum_{i=2}^{4} \alpha_i * (\mathcal{L}_{bce}(p_i,y_i) + \mathcal{L}_{iou}(p_i,y_i)) \tag{9}$$

in which α_i represents the coefficient of i th decoder.

IV. Datasets & Configurations

A. Dataset

We use Singapore Whole sky Nychthemeron Image SEGmentation Database (SWINySEG) as our training set, which contains 6078 day-time cloud images and 690 night-time cloud images. These images are captured in Singapore using a calibrated camera. We split the training set and test set with a ratio of 9:1 following the setting of Zhang *et al.* [29]. In evaluation, we test our model with three different portions: day-time images (augmented SWIMSEG), night-time images (augmented SWINSEG), and the full SWINySEG dataset. Note that we only trained our model once on the full SWINySEG dataset.

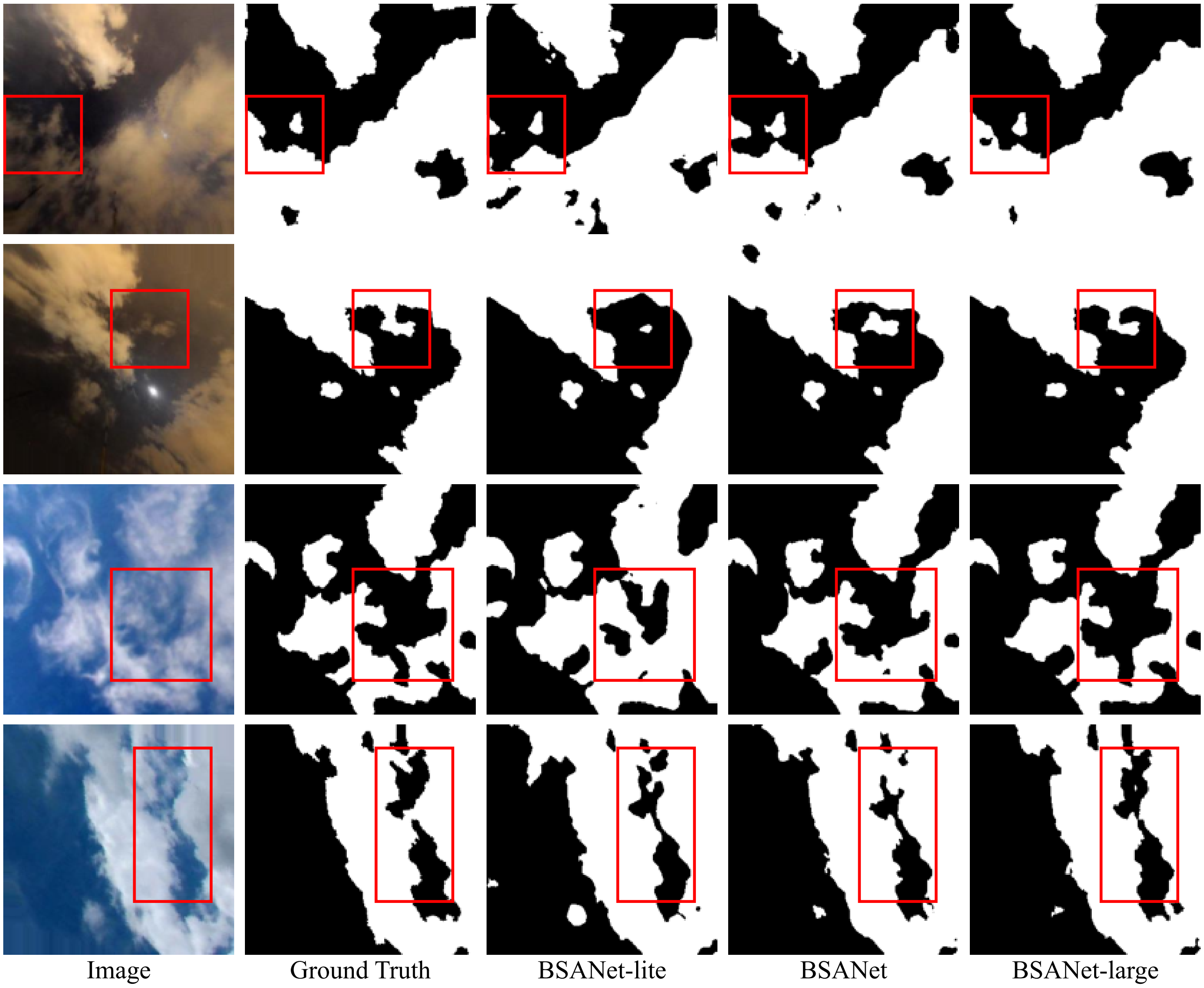

Fig. 5: Qualitative visualization of BSANet-lite, BSANet, and BSANet-large on day-time and night-time images of SWINySEG dataset

B. Training Configurations

We use PaddlePaddle to implement our model and perform the training on a single NVIDIA Tesla V100-SXM2 16GB GPU. We set the training batch-size to 16 and trained for 100 epochs in all experiments. As for the optimizer, we use Adam with an initialized learning rate of 1e-3, beta1 to 0.9, beta2 to 0.999, and epsilon with 1e-8. We also use exponential learning-rate decay with gamma equal to 0.95 after each training epoch. We evaluate our model on the test set after every 5 epochs. As for data augmentation, we only apply random horizontal flip and vertical flip in training after resizing images to 320 × 320.

V. Experiments & Results

We conduct experiments under the training setup introduced in the previous section. In this section, we will introduce the metrics used to evaluate BSANet and then illustrate experiment results qualitatively and quantitatively.

A. Metrics

In our experiments, we evaluate our model with four widely-used metrics: accuracy, precision, recall, F-Score, error rate, and MIOU. F-Score is usually used to describe the overall performance of a model, which is equal to the harmonic mean of precision and recall, $\frac{2 \times Precision \times Recall}{Precision + Recall}$. Precision, can be expressed as $\frac{TP}{TP+FP}$, recall, is equal to $\frac{TP}{TP+FN}$, and error rate can be expressed as $\frac{FP+FN}{P+N}$. MIoU is commonly used in semantic segmentation and can be defined as:

$$\text{MIoU} = 0.5 * (miou_{+} + miou_{-}) \tag{10}$$

where $miou_{+}$ and $miou_{-}$ are defined as below:

$$\begin{aligned} miou_{+} &= TP/(FN + FP + TP) \\ miou_{-} &= TN/(TN + FN + FP) \end{aligned} \tag{11}$$

B. Ablation Study

In order to evaluate the effectiveness of our proposed modules and training strategies, we perform an ablation study which is shown in Table III. Our ablation study is performed under five different components, including backbone networks, BSAM, SWINySEG-based pre-training, ImageNet-based pre-training, and IOU loss. The results show that compared with

TABLE I: Comparison with other state-of-the-art methods on day time and night time images

Methods	Size(MBytes)	Day-time					Night-time				
		Accuray	Precision	Recall	F1-Score	MIoU	Accuray	Precision	Recall	F1-Score	MIoU
General Semantic Segmentation Models											
U-Net [20]	95.00	0.943	0.945	0.945	0.945	0.891	0.953	0.949	0.943	0.946	0.909
PSPNet [33]	26.40	0.945	0.953	0.942	0.948	0.896	0.938	0.927	0.931	0.929	0.882
DeeplabV3+ [14]	10.50	0.953	0.962	0.948	0.955	0.911	0.947	0.931	0.948	0.939	0.898
Special Designed Sky/Cloud Segmentation Models											
CloudSegNet [10]	0.02	0.893	0.888	0.909	0.898	0.806	0.880	0.870	0.922	0.895	0.813
SegCloud [34]	74.80	0.941	0.953	0.934	0.943	0.889	0.955	0.936	0.960	0.948	0.912
CloudU-Net [27]	135.40	0.954	0.956	0.957	0.956	0.912	0.954	0.925	**0.972**	0.949	0.912
CloudU-NetV2 [28]	55.50	0.940	0.967	0.917	0.941	0.887	0.954	0.931	0.965	0.948	0.911
MA-SegCloud [29]	55.80	0.969	0.971	**0.970**	0.970	**0.940**	0.969	0.960	0.970	**0.965**	**0.940**
BSANet-lite	0.34	0.945	0.940	0.939	0.940	0.863	0.948	0.935	0.934	0.934	0.889
BSANet	9.50	0.960	0.957	0.954	**0.995**	0.897	0.966	0.956	0.957	0.957	0.925
BSANet-large	16.37	**0.970**	**0.972**	0.960	0.966	0.921	**0.972**	**0.971**	0.955	0.963	0.937

TABLE II: Comparison with other state-of-the-art methods on day+night time images

Methods	Size(MBytes)	Day+Night time				
		Accuray	Precision	Recall	F1-Score	MIoU
General Semantic Segmentation Models						
U-Net [20]	95.00	0.944	0.945	0.945	0.945	0.893
PSPNet [33]	26.40	0.945	0.951	0.941	0.946	0.895
DeeplabV3+ [14]	10.50	0.953	0.960	0.948	0.954	0.910
Special Designed Sky/Cloud Segmentation Models						
CloudSegNet [10]	0.02	0.896	0.899	0.899	0.899	0.811
SegCloud [34]	74.80	0.942	0.952	0.936	0.944	0.891
CloudU-Net [27]	135.40	0.954	0.954	0.958	0.956	0.913
CloudU-NetV2 [28]	55.50	0.941	0.964	0.920	0.941	0.889
MA-SegCloud [29]	55.80	0.969	0.970	**0.970**	**0.970**	**0.940**
BSANet-lite	0.34	0.945	0.940	0.938	0.939	0.866
BSANet	9.50	0.960	0.957	0.954	0.955	0.900
BSANet-large	16.37	**0.970**	**0.971**	0.960	0.966	0.923

the baseline model (U-Net), our proposed model has better performance on SWINySEG, which shows the effectiveness of BSAM. By observing the metric of BSANet with MobileNet2-l backbone with and without SWINySEG-based pre-training, it is evident that after using our proposed training strategy, error rate is smaller. Experiments No. 5 and No. 6 also show that when adding IOU loss to the original BCE loss, the performance can be improved but not significantly.

TABLE III: Ablation study on different module compositions, objective function, backbone networks, and training strategy. CSPT indicates SWINySEG-based pre-training and INPT is ImageNet-based pre-training. We selected U-Net with 95 MBytes as baseline model.

No.	Backbone	Model configs				SWINySEG
		BSAM	CSPT	INPT	IOU	error rate
1	baseline					0.056
2	MobileNet2-l	✓				0.058
3	MobileNet2-l	✓	✓			0.055
4	MobileNet2	✓		✓		0.040
5	EfficientNet-B0	✓		✓		0.031
6	EfficientNet-B0	✓		✓	✓	**0.030**

C. Qualitative Evaluation

The qualitative results of our proposed models are illustrated in Fig. 5. The two leftmost columns show the source images and their corresponding ground truth. The red rectangle indicates the part that can reflect the performance of the model the most. It shows that our BSANet-lite, the lightest version, can accurately capture the overall shape and content of clouds in both night-time and day-time, but the details of the images are not clear and precise enough. In the results of BSANet, some details, such as the boundary and the middle-size patch of the source image can be captured more accurately. As for our biggest configuration, BSANet-large, the overall shape and details can both be predicted precisely. For example, in two night-time images, the small patch of cloud in the red square is completely captured compared with our two smaller ones.

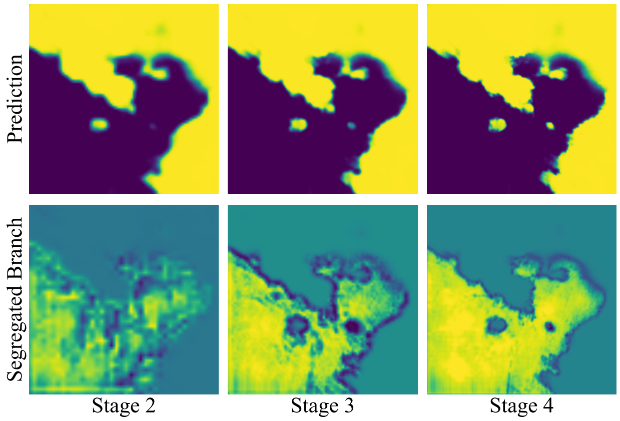

Fig. 6: Visualization of BSAM output s_i (first row) and background mask m_i (second row)

Additionally, we evaluate the effectiveness by visualizing the middle output of the bilateral segregation and aggregation module, which is shown in Fig. 6. The resolutions of segregated branch output (background mask) from stage 2 to stage 4 are 40 × 40, 80 × 80, and 160 × 160. The resolutions of prediction of stages are 80 × 80, 160 × 160, and 320 × 320. In stage 2, the segregated branch output is rough, but it has already captured the overall shape of the segmentation. In stage 3, the background branch is refined based on the mask in the previous stage and its boundary is more apparent. In the final stage, the background mask and stage prediction are clear enough and their combination is close to pure yellow which validates the effectiveness of BSAM.

D. Quantitative Evaluation

Table I and Table II show the quantitative results of our BSANet on day-time, night-time, and day+night time SWINy-SEG datasets by comparing them with other state-of-the-art methods. Noted that we reference the research conducted by Zhang et al. [29] to retrieve results of previous approaches, and in order to compare these data with our results equitable, we test our model under the same setting. Table I and Table II show that our BSANet-large has the best accuracy (0.970) and precision (0.971) under three different datasets compared with 8 previous methods including the current state-of-the-art approach, MA-SegCloud whose accuracy is 0.969 and precision is 0.970. Considering the model parameters size, our largest version BSANet-large only has 16.37 Million Bytes (Mbytes) which has 70.68% less size of MA-SegCloud whose size of parameters is 55.80 MBytes. The results of our standard configuration, BSANet, are also competitive whose accuracy is 0.960 and MIoU is 0.900. As for our smallest version with only 0.34 MBytes of parameters, BSANet-lite, the accuracy is 0.945, which outperform many approaches with large model size such as SegCloud with 74.80 MBytes of parameters and accuracy is 0.942. To sum up, our proposed BSANet has reached the balance of model size and performance, while our BSANet-large has better accuracy and precision than the state-of-the-art approach with a smaller size of parameters.

TABLE IV: FPS and Infer Speed of BSANets

Methods	FP32		FP16	
	FPS	Infer Time	FPS	Infer Time
BSANet-lite	750	1.3 ms	1390	0.7 ms
BSANet	465	2.1 ms	1124	0.8 ms
BSANet-large	299	3.3 ms	392	2.6 ms

To better understand the effectiveness of our lightweight design in inference, we deploy our model via TensorRT, which gives us the results of inference speed in a production environment, shown in Table IV. In our experiments, we first transfer our PaddlePaddle model into Open Neural Network Exchange (ONNX) format and then use TensorRT to compile it into a TensorRT model with FP32 and FP16 data types. We test our model with NVIDIA Tesla V100-SXM2 16GB GPU, and infer time is calculated by running single image inference 1000 times. The table shows that all three configurations of BSANet can run in real time. The BSANet-large can provide inference at about 299 fps under FP32 and 392 fps under FP16. Our BSANet-lite with 0.34 MBytes of parameters can run in FP32 with 750 fps (infer time: 1.3ms) and up to 1390 fps under FP16.

VI. Conclusion and Future works

In this paper, we introduce the Bilateral Segregation and Aggregation Network (BSANet), a real-time sky/cloud segmentation network that can eliminate 70.68% of parameters while achieving almost the same performance as the state-of-the-art method. BSANet-large configuration can obtain 392 fps under FP16 after deployment via TensorRT, whereas BSANet-lite with only 0.34 MBytes of parameters can achieve 1390 fps. In addition, we proposed an innovative and efficient pre-training technique for sky/cloud segmentation that can increase segmentation accuracy when ImageNet pre-training is unavailable. Besides, we apply ablation studies and layer output visualization on our models and training strategies which show the effectiveness of our BSANet. Furthermore, we qualitatively and quantitatively compared our approach with state-of-the-art methods, which shows the advanced nature of our design and contributes to further research in accurate and efficient sky/cloud segmentation areas.

For future work, we intend to concentrate on the design of models for down-stream tasks with higher accuracy with fewer parameters and higher inference speed based on our current approach. Moreover, we will focus on the interdisciplinary research between remote sensing and computer vision, such as multi-classes pixel-wise classification, cloud depth estimation, and sky/cloud-based weather prediction. We will conduct research and experiments on transfer learning, semi-supervised learning, and few-shot learning to solve these problems under resource-limited situations.

Acknowledgment

This research was conducted with the financial support of Science Foundation Ireland under Grant Agreement No. 13/RC/2106_P2 at the ADAPT SFI Research Centre at University College Dublin. ADAPT, the SFI Research Centre for AI-Driven Digital Content Technology, is funded by Science Foundation Ireland through the SFI Research Centres Programme.

References

[1] S. Manandhar, S. Dev, Y. H. Lee, Y. S. Meng, and S. Winkler, "A Data-Driven Approach for Accurate Rainfall Prediction," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 11, pp. 9323–9331, 2019.

[2] H. Wang, M. S. Pathan, Y. H. Lee, and S. Dev, "Day-ahead Forecasts of Air Temperature," in *2021 IEEE USNC-URSI Radio Science Meeting (Joint with AP-S Symposium)*. IEEE, 2021, pp. 94–95.

[3] S. Dev, F. M. Savoy, Y. H. Lee, and S. Winkler, "Estimating Solar Irradiance Using Sky Imagers," *Atmospheric Measurement Techniques*, vol. 12, no. 10, pp. 5417–5429, 2019.

[4] D. Tulpan, C. Bouchard, K. Ellis, and C. Minwalla, "Detection of clouds in sky/cloud and aerial images using moment based texture segmentation," in *Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS)*, 2017, pp. 1124–1133.

[5] M. Jain, I. Gollini, M. Bertolotto, G. McArdle, and S. Dev, "An Extremely-Low Cost Ground-Based Whole Sky Imager," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2021, pp. 8209–8212.

[6] S. Dev, F. M. Savoy, Y. H. Lee, and S. Winkler, "Design of low-cost, compact and weather-proof whole sky imagers for high-dynamic-range captures," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2015, pp. 5359–5362.

[7] ——, "High-Dynamic-Range Imaging for Cloud Segmentation," *Atmospheric Measurement Techniques*, vol. 11, no. 4, pp. 2041–2049, 2018.

[8] S. Dev, Y. H. Lee, and S. Winkler, "Color-Based Segmentation of Sky/Cloud Images From Ground-Based Cameras," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 1, pp. 231–242, 2016.

[9] S. Dev, F. M. Savoy, Y. H. Lee, and S. Winkler, "Nighttime sky/cloud image segmentation," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 345–349.

[10] S. Dev, A. Nautiyal, Y. H. Lee, and S. Winkler, "CloudSegNet: A Deep Network for Nychthemeron Cloud Image Segmentation," *IEEE Geoscience and Remote Sensing Letters (GRSL)*, vol. 16, no. 12, pp. 1814–1818, 2019.

[11] J. Long, E. Shelhamer, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3431–3440.

[12] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs," *arXiv preprint arXiv:1412.7062*, 2014.

[13] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE T-PAMI)*, vol. 40, no. 4, pp. 834–848, 2018.

[14] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[15] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for Real-Time Semantic Segmentation on High-Resolution Images," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 405–420.

[16] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 325–341.

[17] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, "BiSeNet V2: Bilateral Network with Guided Aggregation for Real-Time Semantic Segmentation," *International Journal of Computer Vision (IJCV)*, vol. 129, no. 11, pp. 3051–3068, 2021.

[18] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks ," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4510–4520.

[19] M. Tan and Q. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *Proceedings of the International Conference on Machine Learning (PMLR)*. PMLR, 2019, pp. 6105–6114.

[20] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[21] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009, pp. 248–255.

[22] C. N. Long, J. M. Sabburg, J. Calbó, and D. Pagès, "Retrieving Cloud Characteristics from Ground-Based Daytime Color All-Sky Images," *Journal of Atmospheric and Oceanic Technology*, vol. 23, no. 5, pp. 633–652, 2006.

[23] S. Dev, Y. H. Lee, and S. Winkler, "Systematic study of color spaces and components for the segmentation of sky/cloud images," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 5102–5106.

[24] J. Yang, W. Lv, Y. Ma, W. Yao, and Q. Li, "An Automatic Groundbased Cloud Detection Method based on Local Threshold Interpolation," *Acta Meteorologica Sinica*, vol. 68, no. 6, pp. 1007–1017, 2010.

[25] M. Jain, C. Meegan, and S. Dev, "Using GANs to Augment Data for Cloud Image Segmentation Task," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2021, pp. 3452–3455.

[26] S. Dev, S. Manandhar, Y. H. Lee, and S. Winkler, "Multi-label Cloud Segmentation Using a Deep Network," in *Proceedings of the USNC-URSI Radio Science Meeting (Joint with AP-S Symposium)*. IEEE, 2019, pp. 113–114.

[27] 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 Geoscience and Remote Sensing Letters (GRSL)*, vol. 18, no. 10, pp. 1688–1692, 2020.

[28] C. Shi, Y. Zhou, and B. Qiu, "CloudU-Netv2: A Cloud Segmentation Method for Ground-Based Cloud Images Based on Deep Learning," *Neural Processing Letters*, vol. 53, no. 4, pp. 2715–2728, 2021.

[29] L. Zhang, W. Wei, B. Qiu, A. Luo, M. Zhang, and X. Li, "A Novel Ground-Based Cloud Image Segmentation Method Based on a Multi-branch Asymmetric Convolution Module and Attention Mechanism," *Remote Sensing*, vol. 14, no. 16, p. 3970, 2022.

[30] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 3–19.

[31] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7132–7141.

[32] L. Wang, C.-Y. Lee, Z. Tu, and S. Lazebnik, "Training Deeper Convolutional Networks with Deep Supervision," *arXiv preprint arXiv:1505.02496*, 2015.

[33] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid Scene Parsing Network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[34] W. Xie, D. Liu, M. Yang, S. Chen, B. Wang, Z. Wang, Y. Xia, Y. Liu, Y. Wang, and C. Zhang, "SegCloud: A novel cloud image segmentation model using a deep convolutional neural network for ground-based all-sky-view camera observation," *Atmospheric Measurement Techniques*, vol. 13, no. 4, pp. 1953–1961, 2020.