Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105 (Online)
Volume **6**, Number **2**, May **2021**

High-resolution Satellite Multi-class Cloud Detection Based on Improved AlexNet

Wenli Lv

College of Electrics Engineering
Heilongjiang University
Harbin, 150080, PR China
Guangzhou Branch
China Telecom Co.,LTd..
Guangzhou, 510000, PR China
512636201@qq.com

Wenjing Lv

College of Electrics Engineering
Heilongjiang University
Harbin, 150080, PR China
jlee1020@163.com

Jianxiong Li

College of Electrics Engineering
Heilongjiang University
Harbin, 150080, PR China
2628466879@qq.com

Xiaofei Wang*

College of Electrics Engineering
Heilongjiang University
Harbin, 150080, PR China
*Corresponding author:nk_wxf@hlju.edu.cn

Received January 2021; revised March 2021

Abstract. *Today, relevant statistics show that the surface of the earth is always covered by clouds 50% of the time. Under different sensors and different application scenarios, the solution of cloud occlusion and interference to remote sensing images is extremely complicated. Need to choose different solutions according to the actual situation. Therefore, performing cloud inspections is critical to us. However, due to the small number of bands and narrow spectral range of high-resolution remote sensing images in China, the accuracy of traditional cloud detection is low. In this study, we will use the PlanetScope and Sentinel-2 images collected in the tropical region of Australia as datasets to perform four classifications on our efficient and streamlined AlexNet network proposed in this paper. In model training, these two images were divided into 9936* 128×128 *images, and four labels were made manually. The training works well, with an overall accuracy of 99. The accuracy of label 0 is 91.22, the accuracy of label 1 is 98.79, the accuracy of label 2 is 74.79, and the accuracy of label 3 is 76.28. Since the total number of samples used for label 2 and 3 is small, accuracy is less than 80. We compare it with some previous studies, and the accuracy and overall accuracy of each class of the algorithm have been improved. At the same time, compared with other single models of CNN and multiple combined models in experiments, the accuracy evaluation index F score and OA have improved. In addition, the algorithm is compared with the most popular clouds and cloud shadows. The shielding algorithms (Sen2Cor and MACCS algorithms) are compared. The improved AlexNet model in this paper is beneficial to the research of high-resolution multi-class cloud detection.*

Keywords: Deep learning, Remote sensing, Sentinel-2, PlanetScope

1. **Introduction.** At present, with the significant improvement of the spatial resolution of various sensors in the aerospace field, the application value of remote sensing images in resources and environment, disaster monitoring, urban planning, and other areas has become increasingly more significant. For example, through cloud detection technology and Internet of Things technology to achieve typhoon monitoring in order to achieve the purpose of early warning. [1]. The satellite transmission frequency and sensor update speed are rapidly increasing. The brightness and resolution of the remote sensing images obtained are different. The applicability of traditional feature design and fixed empirical parameters are getting worse and worse. Cloud detection and compensation issues have not been a perfect solution. To assure that earth observation-based geo-information products have the highest available spatial and temporal resolutions and information content, it is critical to simultaneously harvest data from a large variety of satellite sensors. While Synthetic Aperture Radar (SAR) sensors have the capability of penetrating through clouds and acquiring images during day and night. A host of tasks require the use of a high-resolution multispectral image to overcome limitations of SAR data and to improve the temporal resolution and timespan [2]. Also, urban [3]. and flood [4]. mapping and monitoring systems widely use high-resolution multispectral images due to their extended thematic information content and more straightforward reflection characteristics in complex environments. For example, spatiotemporal fusion technology is used to fuse different sensors to obtain high-resolution multispectral images. [5] The inherent potentials of multi-spectral satellite images are, however, being hampered by the presence of clouds and cloud shadows that may obstruct objects of interest and can bias image statistics, which can negatively impact on the performance of image analysis methods. Therefore, the usefulness of this imagery depends strongly on the ability to reliably mask clouds and cloud shadows from clear-sky pixels. Being able to quickly and accurately identify cloud and cloud shadow pixels is fundamental for unbiased down-stream analysis [6].

The emerging machine learning algorithms provide an effective technical way to solve cloud detection. By mixing the input of training data, the model can be applied to

different sensors, and it can achieve better detection under different imaging lighting conditions and imaging ratio results [7]. The research on cloud detection has been going on for decades, and the evolution of algorithms has also changed from early physical spectral characteristics to artificially designed texture feature structures to current machine learning algorithms. Convolutional neural networks, as one of the most extensive algorithms for machine learning, are also continually developing. The first successful applications were developed by Yann LeCun in the 1990s. The most famous of these is the LeNet architecture for identifying postal codes, numbers, and more [8]. It was followed by AlexNet, which was popular in computer vision, and was developed by Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton. AlexNet was applied to the ImageNet ILSVRC challenge in 2012, and it clearly surpassed the runner-up (top 5 error rate was 16%, and the Biya army was 26%) [9]. This convolutional neural network has a very similar architecture to LeNet, but it is more profound, more significant, and has convolutional layers stacked on top of each other (usually a convolutional layer always immediately follows the pooling layer). Later there was GoogLeNet, a convolutional network developed by ILSVRC 2014 winner Szegedy and others. Its main contribution is the development of an Inception module, which significantly reduces the number of parameters in the network (4M compared to AlexNet with 60M). In addition, this paper uses Average Pooling instead of Fully Connected layers on top of the convolutional neural network, thereby eliminating a large number of seemingly unimportant parameters. The 2014 ILSVRC runner-up is a convolutional neural network from Karen Simonyan and Andrew Zisserman, known as VGGNet [10]. Its main contribution is to demonstrate that the depth of the network is a critical component of excellent performance. Their final best network contains 16 CONV / FC layers, and it is desirable that this convolutional neural network has a very uniform architecture, which only performs 3×3 convolution and 2×2 pooling from the beginning to the end. Their pre-trained model can be used directly in Caffe. The disadvantage labele of VGGNet is that it costs more to evaluate and uses more memory and parameters (140M). Most of these parameters are located in the first fully connected layer, because it was found that these FC layers can be removed without degrading performance, thereby significantly reducing the number of indispensable parameters.

Finally, the residual network (ResNet) was developed by Kaiming He et al. It is the winner of ILSVRC 2015. It has special skip connections and a large number of uses for batch normalization [11]. The architecture also does not have a fully connected layer at the end of the network. ResNet is by far the most advanced convolutional neural network model and the default choice for using convolutional neural networks in practice (as of May 10, 2016). In particular, you can see more recent developments, adjusting the original architecture of Kaiming He et al. Identity Mappings in Deep Residual Networks (posted March 2016). Significant work has also been undertaken to detect and segment clouds and cloud shadows in multi-spectral satellite images [12]. The existing methods can broadly be categorized into rule-based and machine learning approaches. The majority of rule-based methods exploit variations of reflectance in visible, shortwave-infrared and thermal bands and develop rule-sets that combine thresholds [13] or functions over several spectral bands [14] to distinguish clouds from clear-sky pixels. Cloud shadows are more difficult to detect than clouds because their spectral signatures overlap with other dark surface materials [15]. In particular, water surfaces are known to introduce false positives into shadow segmentation.

In recent years, machine learning methods have been proposed to extract more robust high-level information from images [16]. Li [17] used a Support Vector Machine to segment clouds from reflectance and texture information. Hollstein [18] presented an overview of several ready-to-use machine learning algorithms to detect "cloud", "cloud shadow",

"snow/ice", "water", "flooded" and "clear sky" pixels in Sentinel-2 images. The algorithms they presented include Classical Bayes, Decision Trees, Support Vector Machine and Hayes [19]introduced the Spatial Procedures for Automated Removal of Cloud and Shadow (SPARCS) algorithm that used neural networks and rule-based post-processing to determine "cloud," "cloud shadow," "water," "snow/ice" and "clear sky pixels" in Landsat images. Convolutional Neural Networks (CNNs), which extract features directly from raw images by combining convolutional and pooling layers, gradually appear in recent studies on cloud segmentation [20]. First results manifest superior accuracy, generalization ability, and inference speed compared to rule-based and classical machine learning approaches. Zhaoxiang [21] used a lightweight U-Net [22] architecture and combined it with wavelet image compression for computationally efficient cloud detection on-board small satellites.

Ozkan [23] adapted a deep pyramid network to produce cloud masks from noisy labeled RGB colour images. Zhan [24]specifically focused on distinguishing cloud and snow from Gaofen1 imagery using a modified VGG network [25]. Despite the promising results for cloud detection achieved in above mentioned deep learning studies, none of them specifically considers cloud shadows. Isikdogan [26] uses a Fully Convolutional Network for semantic segmentation of Landsat ETM+ images into five classes that include, amongst others, "cloud" and "shadow." Their main objective is large-scale surface water mapping, but generalization ability across sensors is not considered. Sholar [27]confirms the observed lack of research and available deep learning models that specifically consider a shadow class and concludes that future work should focus on improving in this regard. Furthermore, additional research is needed to train models that generalize well across different sensors and images with varying atmospheric conditions and surface reflectance characteristics.

In this research, we propose a multi-classification method based on clouds and cloud shadows, which is based on the improved AlexNet. Our goal is to provide a fast and easy-to-use method that can cover up from two Clouds and cloud-shadow pixels in a single date image of a high-resolution satellite sensor without retraining or human intervention. We demonstrate the generalization ability of our method on multiple satellite sensors (Palntscope and Sentinel-2).

As a comparison, we investigated the performance of the ensemble CNN model trained and validated on the combination of T-S2 and T-PS datasets, comparing it with the most popular cloud and cloud shadow mask algorithms (Sen2Cor and MACCS algorithms) on the Sentinel-2 image dataset [28], it turns out that our method performs better than its classification performance. In this case, we also manifest the impact of different trained CNN models on the performance of cloud multi-classification. Compared with previous work in this direction, our research has more computational efficiency of the model for simple label classification and considers the data set combination of the two sensors. In addition, we consider shadow categories and distinguish between cloudy and partially shadowed pixels.

2. **Methods.**

2.1. **Datasets.**

Our satellite data consists of two datasets consisting of PlanetScope and Sentinel-2 satellite images. All PlanetScope images contain three data bands: red, green, blue (RGB), and ground sampling distance (GSD) of 3.125m [29]. We also trimmed the Sentinel-2 image to include only RGB bands and resampled it to a 3.125m resolution to match the PlanetScope image. In this study, we refer to the data of the PlanetScope and Sentinel-2 images as the data sets T-PS and T-S2, respectively.

The image sizes of the T-PS and T-S2 datasets are 128×128 pixels (ie 400

In this study, we used Yuri's dataset, but we changed multiple labels to a single label, that is, each image corresponds to a label, and all images were set to 4 categories. In addition to manually modifying the labels, we emphasize keeping the two categories of cloud and cloud shadow, and then subdividing into three types of labels. The remaining land cover categories were classified into one category.

Each data set was further divided into training and test sets. In the experiment, the entire data set was scrambled, and 10% is randomly selected as the test set used and 90% is used as the training set.

2.2. **Cloud and shadow CNN architectures.** Convolutional neural network is a form of machine learning. It describes linear and non-linear functions by computing nodes. The input layer, convolution layer and pooling layer can be used as a complete FC layer. Each additional layer results in increased computation. The image is generally input to the network as a vector, and then the parameters are determined for training.

However, traditional CNNs will expand more connected neurons due to hidden layers during training, and the network will have difficulty converging. As shown in Figure 1, it is an improved general CNN pixel-level classification architecture [30] that extracts pixel information from three-channel images.

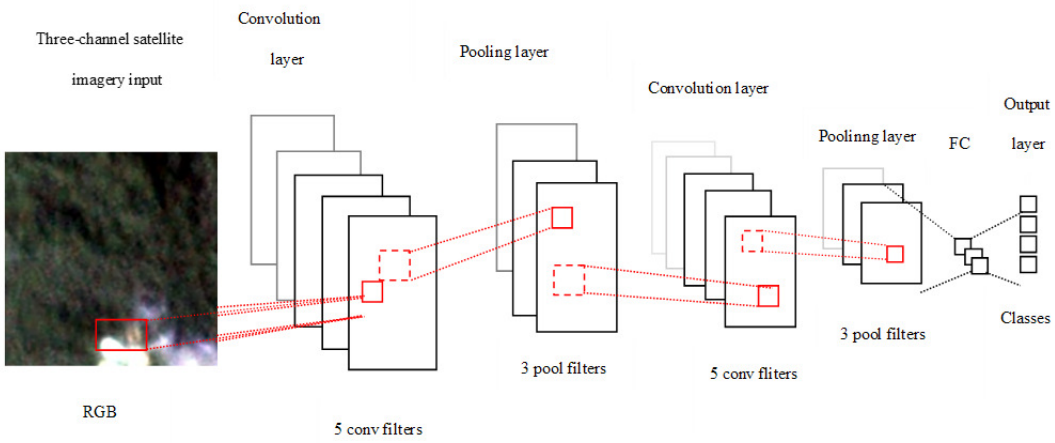

FIGURE 1. Improved on a generic CNN pixel-wide classification architecture

The architecture is a 5×5 convolutional layer, a 3×3 pooling layer, and the last three fully connected layers. Finally, we get our four classifications, which are based on pixel-level classification. This architecture is suitable for multi-channel images. The dimensions of each convolution layer include width, height, and depth.During convolution, the next layer connects a small area on the previous layer, so that the weight and redundancy are reduced.

The AlexNet network structure model proposed by Alex in 2012, sparked a wave of application of neural networks and won the 2012 image recognition competition, making CNN the core algorithm model in image classification. For traditional machine learning classification algorithms, you can see that the official data model accuracy rate has achieved an excellent result. The benefits of using AlexNet are:

1)Using Relu, Relu is an activation function in neural networks, which is superior to tan and sigmoid functions. when sigmoid and other features are used to calculate the activation function (exponential operation), the amount of calculation is large; when backpropagation is used to calculate the error gradient, the derivation involves division,

and the amount of computation is relatively large; when Relu activation function is used, the amount of calculation in the whole process is much saved; for deep networks, when the sigmoid function is back-propagated, it is easy for the gradient to disappear. ReLu will make a slice of neurons output 0, which will cause the network to be sparse and reduce the interdependence of the parameters, which will alleviate the problem of overfitting.

2)The overlapping pooling improves accuracy and is not easy to overfit. In previous CNNs, the average pooling layer was hardly used. AlexNet all used the largest pooling layer to avoid the blurring effect of the average pooling layer. The length is longer than the size of the pooled kernel so that there is overlap between the outputs of the pooling layer, which improves the feature richness.

3)Local response normalization improves accuracy. It creates a competition mechanism for local neurons, making the response small in response to larger values and suppressing smaller feedback.

4)Data gain Dropout, reduce overfitting, and use data enhancement methods to alleviate overfitting.

Based on the above advantages, I began to use the AlexNet model as my own classification framework, but because the data in this study is too small, each picture is only 128×128, which is too large for network training, and the extracted features that are easy to appear at the end are almost 0. Therefore, inspired by MNIST, the number of training layers is reduced. Through continuous debugging during training, it is found that the accuracy is quite high when it is reduced to two layers. At the same time, it also retains the LRN processing in AlexNet, creating a competition mechanism for the activity of local neurons, making the larger response values relatively more significant, and suppressing other neurons with smaller feedback, which enhances the generalization ability of the model. The optimization section selects AdamOptimizer as the optimizer.

The data set used in this paper has 9936 pictures, which are randomly shuffled into 8936 training sets and 1000 test sets. Its data set size is 128*128 pixels. The system used in this study is Win10, the hardware is anaconda 3, the platform is pycharm, and the library used is tensorflow1.4. The network is trained using a CPU. In this study, the training data was passed in before training. The new module tf.data framework module of tensorflow1.4 was used to perform pre-processing data operations. The tf.data module has corresponding data enhancement operations, such as rotation and noise, and contrast enhancement. The middle also uses tf.image.resize to resize the data to $32 \times 32 \times 3$. Convolution kernel size used during the training is 5*5, the number of iterations to 50 times, using Adam optimizer, bitchsize set to 128. The Relu activation function is used after each layer of convolution. The kernel span of each convolution layer is fixed at 1. Pooling is generally divided into maximum pooling and average pooling. This multi-class cloud detection convolutional neural network AlexNet framework uses the biggest pooling. Compared to classic AlexNet, we have made the following improvements:

1) Resampling is used. Change the pixel size of the original image 128×128 to 32×32.

2) Reduce the number of convolutional layers. The 5-layer convolutional layer was changed to 2 layers, achieving a total accuracy of 99.

3) Change the number of fully connected nodes. Change the number of fully connected connection points from 4096 and 1000 to 384 and 192.

4) Change the output category. Change the 1000 categories into 4 categories.

The hyperparameters of each layer of AlexNet are shown in Figure 2 below. The central idea of this network is to resize the input size of $128 \times 128 \times 3$ to $32 \times 32 \times 3$, using two convolution layers, and the picture size is not suitable for classic. The first layer of AlexNet uses a larger kernel size of 11×11, but uses a kernel size of 5×5, with a step size of

1, which scans all pixels in the full image and has 64 convolution kernels; immediately followed by a layer of LRN; Then the maximum pooling layer with a step size of 2.

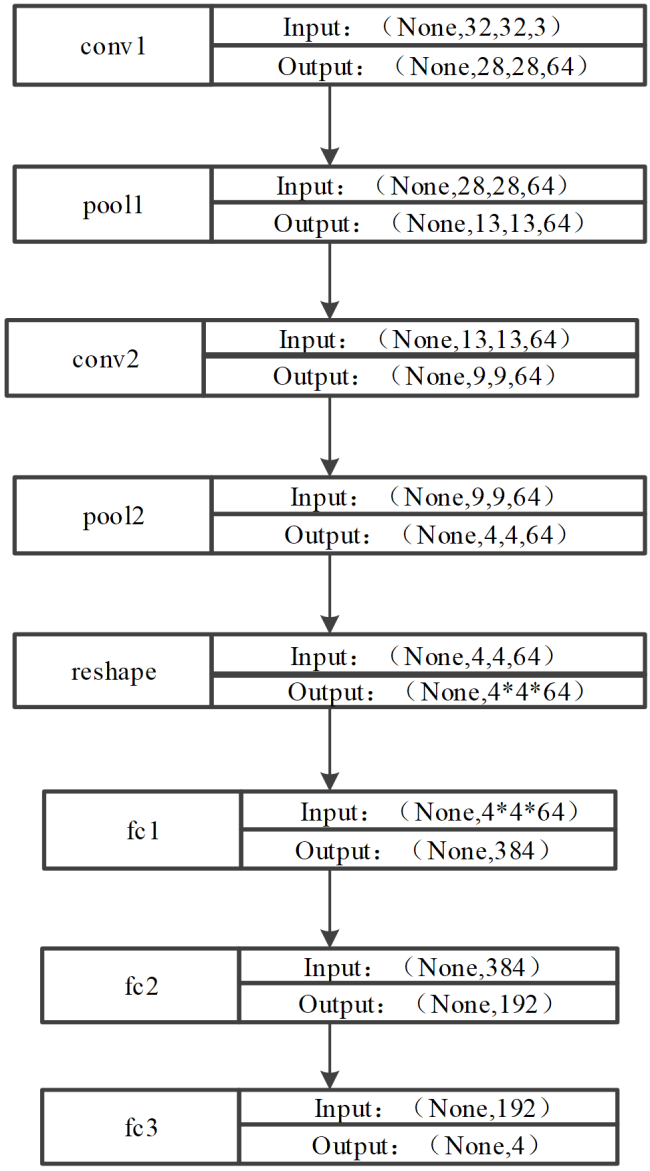

FIGURE 2. AlexNet network parameters of this study

2.3. **Training process in Deep learning.** The above network will now be described. This network uses two convolutional layers and three fully connected layers. The process of convolution pooling and full connection of each layer is as follows: The input data of the first layer is the original $32 \times 32 \times 3$ image. This image is convolved by the$5 \times 5 \times$ 3convolution kernel. The convolution kernel generates a new pixel for each convolution of the original image. The convolution kernel moves along the x-axis direction, and the y-axis direction of the original image and the step size of the movement is 1 pixel. Therefore, the convolution kernel generates 28 pixels in the process of moving, and the 28*28 pixels in the rows and columns form the pixel layer after convolving the original image. There are 64 convolution kernels, which will generate $28 \times 28 \times 64$ convolutional pixel layers. These pixel layers are processed by the pool operation (pooling operation). The scale of the pooling operation is 3×3, and the step size of the procedure is 2.

Then the size of the pooled image is 13. That is, the scale of the pixel after the pooling is $13 \times 13 \times 64$; then, after the normalization process, the range of the normalization operation is 5×5; when the image formed after the first convolution layer operation is back-propagated, each The convolution kernel corresponds to a bias value. That is, the 64 convolution kernels of the first layer correspond to the input layer of the upper layer, and the input data of the second layer is the $13 \times 13 \times 64$ pixel layer output by the first layer. To facilitate subsequent processing, the left and right sides and the upper and lower sides of each pixel layer must be the scale of filling the one-pixel element layer is $13 \times 13 \times 64$. Corresponding to 64 convolution kernels.

The size of the input data of the third layer is $4 \times 4 \times 64$, and a filter of size $4 \times 4 \times 64$ is used to perform convolution operation on the input data of the third layer; The convolution operation of the input data of the layer generates an operation result, which is output by a neuron; there are 384 $4 \times 4 \times 64$ size filters for convolution operation on the input data, and the operation result is output by 384 neurons; The 384 operation results generate 384 values through the Relu activation function, and 384 output result values of this layer are output after the drop operation. Because the size of the filter ($4 \times 4 \times 64$) used in the calculation of the third layer is the same as the size of the feature map to be processed ($4 \times 4 \times 64$), that is, each coefficient in the filter is only equal to the feature One-pixel value in the map is multiplied; while in other convolution layers, the coefficients of each filter are multiplied by the pixel values in multiple feature maps; therefore, the third layer is called a fully connected layer.

The $4 \times 4 \times 64$ scale pixel layer data output by the second layer is fully connected with the 384 neurons in the third layer, and then processed by the Relu function to generate 384 data, and then processed by the dropout function to output 384 data.

The 384 data output from the third layer is fully connected to the 384 neurons of the fourth layer and then processed by Relu to generate 384 data, and then output 384 data after dropout processing.

The 384 data output from the fourth layer is fully connected to the 192 neurons in the last layer, and the trained values are output after training. Next, the layer number parameters of the network are explained as follows: The first layer weights are initialized to generate 64 convolution kernels of 3 channels (RGB pictures) with a size of 5*5, without L2 regularity (wl = 0.0), and then Input the original image for convolution operation, the step size is [1, 1, 1, 1], that is, each pixel is calculated, and the zero-padded mode is 'SAME' (blocks that are not enough for the convolution kernel size are supplemented with 0). Then define the bias parameters for the first layer. Since there are 64 convolution kernels, the bias size is 64. The convolution result is biased and activated using the Relu activation function. In order to improve the training speed, it can train faster, and at the same time, solve the problem of gradient disappearance or gradient dispersion in sigmoid in deeper training networks.

After the convolution, the first layer of the pooling operation is performed. The pooling layer with a size of 3×3 and a step size of 2×2 is used to operate. The result is processed by LRN so that the value becomes larger or become smaller, mimicking the lateral inhibition mechanism of the nervous system. Similarly, the second convolutional layer has the same architecture as the first fully connected layer. Next is a three-layer fully-connected layer. The output of the previous layer is flattened to obtain the flattened length of the data. A fully-connected layer with 384 hidden nodes and a fully-connected layer with 192 hidden nodes are established. Finally, an output layer is created (because the data has a total of 4 categories of labels, the number of output nodes here is 4).

3. **Experimental results.**

3.1. **Data preprocessing.** Satellite data uses two data sets, namely PlanetScope and Sentinel-2 satellite images. In this study, we used the data from Tropics PlanetScope and Sentinel-2 as the data sets T-PS and T-S2, respectively. The remote sensing image was segmented to generate 128×128 pixels, and random scene samples were extracted from its image, which finally obtained 4943 PS image scenes and 4993 S2 image scenes ($400 \times 400m$). This data set was manually divided into 12 labels, which were divided into three groups. The first group was the cloud label, the second group was the shadow label, and the third group was the land cover label. In order to be suitable for the study in this paper, it was manually remade labels. There are 9936 data sets used in this study. They were labeled as cloudy unshaded as label 1 (S2:1519 types, PS: 807 types), and partly cloudy unshaded is labeled as label 2 (S2: 771, PS: 840), partly cloudy partly shaded is labeled as label 3 (S2: 557, PS: 257), and all other scene images that do not contain these three types are labeled as label 0.

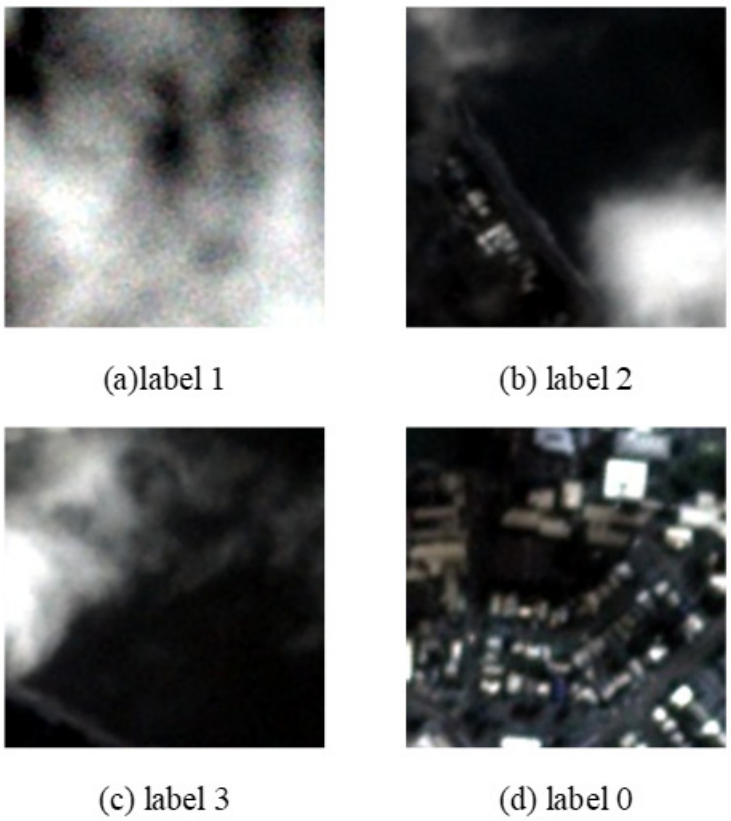

FIGURE 3. Four labels of images of PlanetScope dataset

The manual label image is shown in Figure 3(a) and (b). There are too many data scenarios representing label 0, of which only a portion of label 0 is shown. Data sets T-PS and T-S2 were manually labeled for this study. They are divided into four categories. Images containing cloudy unshaded are labeled as label 1 (S2: 1519, PS: 807), and those marked as partially cloudy unshaded are labeled 2 (S2: 771, PS: 840), and part cloudy partly shaded is marked as 3 (S2: 557, PS: 257). All other combined labels are labeled as label 0. There are 2326 pieces of image data for label 1, 1611 pieces of image data for label 2, 814 pieces of image data for label 3, and 5185 pieces of data for label 0. It can be understood that, since images other than the above three types of labels are all set to label 0, consequently, the data of label 0 is much larger than the data of other labels. It

should be noted that the definition of cloudy is greater than 90% of cloud pixels, and the definition of a slice of clouds is 10-90% of cloud pixels. Similarly, the definition of a slice of shadows is 10-90% of shadow pixels, and the definition of no shadow is less than 10% of pixels.

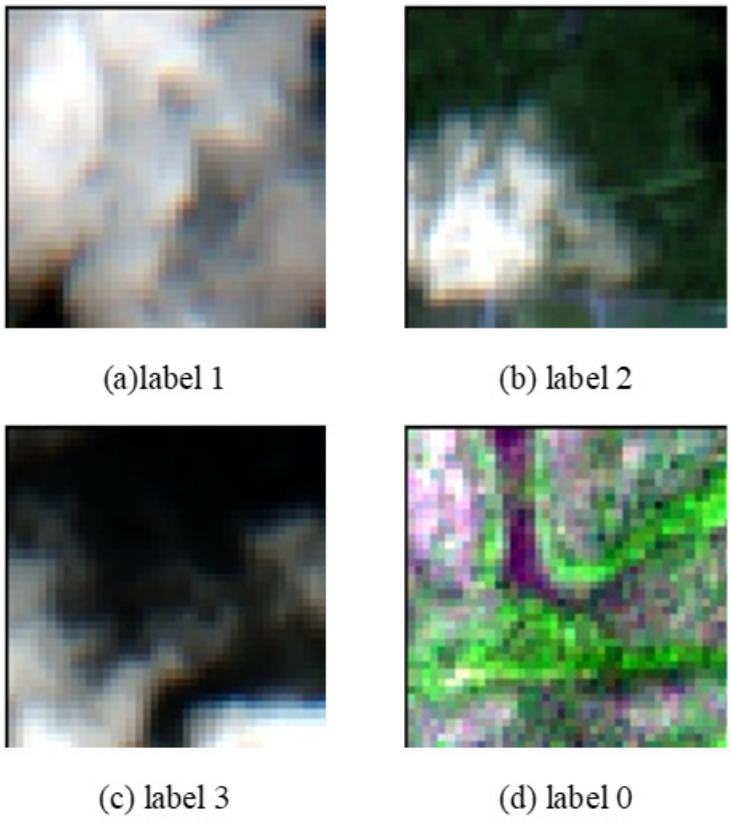

FIGURE 4. Four labels of images of Sentinel-2 dataset

In addition, the data set provided by Yuri is four-channel and 12 kinds of labels. In this paper, the third-party library gdal in Python is used to convert the remote sensing image into a three-channel RGB image, which is convenient for subsequent experiments. The images and the labels we have manually modified have been uploaded to the link(https://pan.baidu.com/s/1U_s2SEPGqskysi9eYfGqgQ). and researchers who need it can download it by themselves. Thank you Yuri for sharing his dataset, the link is as follows(https://github.com/yurithefury/ChipClassification).

3.2. **Classification results and index evaluation.** Commonly used classification models have the following detection accuracy: F-score, ROC curve, kappa coefficient.

Take the binary classification as an example, 1 represents a positive category, 0 represents a negative category, and the instances are divided into positive (native) or negative (negative). But in practice, there are four situations:

1) If a case is positive and predicted to be positive, it is a true post (TP)

2) If an instance is a positive class but is predicted to be a negative class, it is a false negative class (False Negative FN)

3) If a case is a negative class but is predicted to become a positive class, it is a false positive class (False Postive FP)

4) If a case is a negative class but is predicted to become a negative class, it is a true negative class (True Negative TN)

The calculation formulas for the horizontal and vertical axes can be obtained from the above table, as shown in Table 1.

Table 1. Confusion matrix of two classification

class	1(Predicted)	0(Predicted)
1(Labeled)	TP	FN
0(Labeled)	FN	TP

(1) True Positive Rate (TPR): TP / (TP + FN), which represents the ratio of actual positive instances to all actual positive cases in the positive class predicted by the classifier.

(2) False Positive Rate (FPR): TP / (TP + FP), which represents the ratio of actual positive instances to all predicted positive instances in the positive class predicted by the classifier. For multi-classification problems, accuracy is no longer the only evaluation indicator, as well as accuracy and recall. F-score is a comprehensive indicator that reconciles these two parameters:

$$F_1 = \frac{2PR}{P+R} \tag{1}$$

According to the prediction results of the learner, the samples are sorted, and the positive samples are predicted one by one in this order. Each point on the ROC curve reflects the sensitivity to the same signal stimulus.

The horizontal axis generally represents the specificity of the negative-positive rate (FPR), which divides the proportion of all negative examples in the negative cases; (1-Specificity), the larger the FPR, the more negative classes are predicted in the positive class.

The vertical axis represents a true positive rate (TPR) sensitivity. Sensitivity (positive class coverage), and the larger the TPR, the more actual positive classes are predicted in the positive class. The ideal target is TPR = 1 and FPR = 0, that is, the (0,1) point. The closer the ROC curve is to the (0,1) point, the better off the 45-degree diagonal. Figure 4 is a four-category ROC curve. It can be seen intuitively that the TPR of each category follows the FPR curve. From the figure, it can be seen that the ratio of label 0 is 0.95, the ratio of label 1 is 0.95, and the ratio of label 2 is 0.86, and the ratio of label 3 is 0.88. The classification of labels 0 and 1 is better, and the ranking of labels 2 and 3 is next. Kappa coefficient is a method used to evaluate consistency in statistics. We can use it to evaluate the accuracy of multi-class models. The value range of this coefficient is [-1,1]. In practical applications, it is [0,1] is similar to the principle that a convex curve generally does not appear in the ROC curve. The higher the value of this coefficient is, the higher the classification accuracy achieved by the model is. The calculation method of the kappa coefficient can be expressed as follows:

$$k = \frac{p_0 - p_e}{1 - p_e} \tag{2}$$

Where P_o is expressed as the total classification accuracy; P_e is expressed as:

$$p_e = \frac{a_1 \times b_1 + a_2 \times b_2 + ... + a_c \times b_c}{n \times n} \tag{3}$$

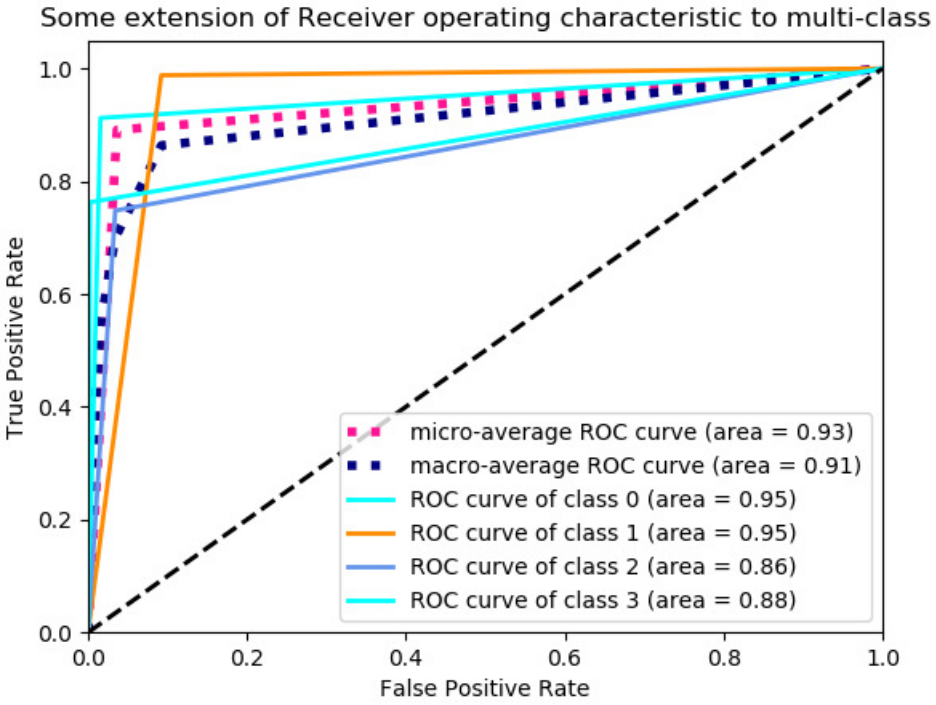

FIGURE 5. ROC curve of four categories

Where a_i represents the number of true samples of the i category, and b_i represents the number of samples predicted by the i category.

Figure 6 and Figure 7 are the curves of the loss rate and accuracy rate of the training set and test set with the number of iterations during training. The training iterations are performed 50 times for about 25 minutes, and the speed of more than ten frames in 1s is predicted. It can be seen from the figure that the accuracy rate and loss rate of the test and training sets gradually approach 1 and 0 with the number of iterations. In our model detection, the final classification accuracy reached 0.99.

The four-class confusion matrix is shown in Table 2(we use pre instead of predicted):

TABLE 2. Confusion matrix of four-class

class	0(Pre)	1(Pre)	2(Pre)	3(Pre)	total
0(Labeled)	4730	244	199	12	5185
1(Labeled)	12	2298	15	1	2326
2(Labeled)	43	353	1205	10	1611
3(Labeled)	19	104	70	621	814
total	4804	2999	1489	644	

According to the data in Table II above, it can be obtained that the Kappa coefficient is 0.83, the F-score is 0.86, the recall rate is 0.85, and the accuracy rate is 0.88. The above accuracy evaluation indicators are the average values calculated after each category.

Because the accurate evaluation of the algorithm compared with this study uses overall accuracy (OA) and, where OA is the general evaluation test set based on the accuracy of CNN classification using the following equations:

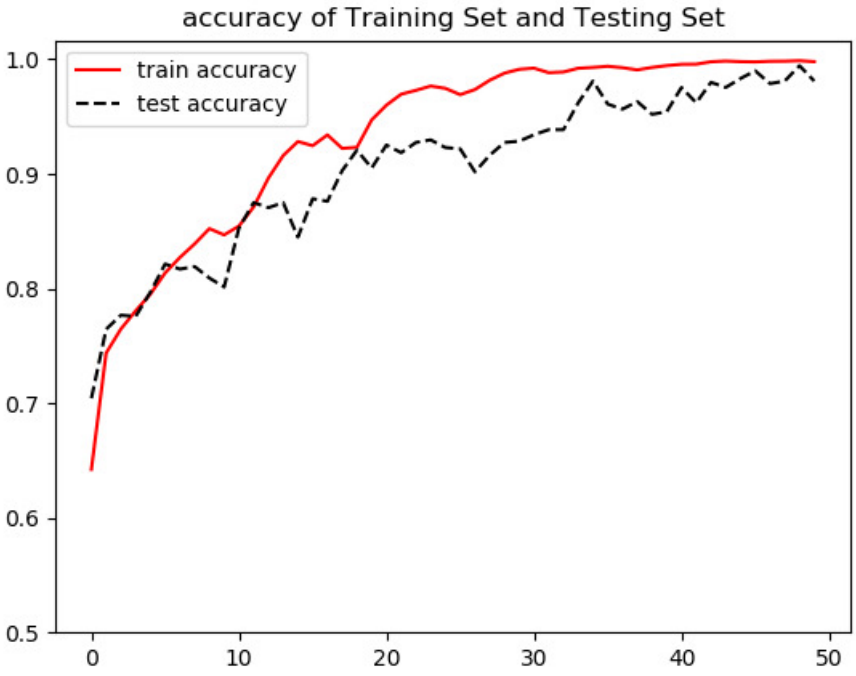

FIGURE 6. Accuracy of training and testing sets

FIGURE 7. Loss of training and testing sets

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

F_2 is the weighted average of recall and accuracy using the following equations:

$$F_2 = \frac{5 \times TP}{5 \times TP + 4 \times TN + FP} \tag{5}$$

As a comparison with Yuri's experiment [31], we investigated the performance of the improved CNN model trained and validated on the combination of T-S2 and T-PS datasets. The comparison of labels F2 and OA generated using individual models, and their ensemble could be seen in Table 3. The ensemble of three individual CNN models generally resulted in the improvement of labels F2 and OA, suggesting that the ensemble approach is effective in selecting the best performing model. However, the magnitude of the improvement in terms of F2 and OA between ensemble and individual models was marginal. In this case, compared with our improved AlexNet model, its total accuracy evaluation index F2 (OA) reaches 0.85 and 0.94, which are improved by 0.03 and 0.14, respectively.

Table 3. Total accuracy evaluation various models F2(OA)

methods	train data	test data	F2	OA
Improved AlexNet	S2-train+PS-train	S2-test+PS-test	0.85	0.94
DenseNet201	S2-train+PS-train	S2-test+PS-test	0.81	0.74
ResNet50	S2-train+PS-train	S2-test+PS-test	0.8	0.74
VGG10	S2-train+PS-train	S2-test+PS-test	0.81	0.73
Ensemble	S2-train+PS-train	S2-test+PS-test	0.82	0.8

It can be seen from the above figures that compared with other CNN models, the accuracy of this algorithm is much higher than the results of other model training. At the same time, compared with the most popular cloud and cloud shadow mask algorithms (i.e. Sen2Cor and MACCS), both algorithms are based on pixel-level classification. Since we manually combined the cloud and shadow labels and only combined two types of cloud labels and two types of shadow labels, we will not compare it with label 0 here, and the remaining labels will be selected from the average accuracy results of other algorithms. Therefore, the accuracy of the three algorithms can be compared with our algorithm, as shown in Table 4. Looking at the accuracy results of each label, the accuracy of our algorithm has improved a lot.

Table 4. Evaluation F2(OA) of each label in four methods

	mproved AlexNet	Sen2Cor	MACCS	CNN ensemble
Label1	0.93(0.92)	0.88(0.85)	0.83(0.80)	0.83(0.86)
Label2	0.76(0.93)	0.77(0.81)	0.57(0.76)	0.80(0.79)
Label3	0.80(0.98)	0.38(0.81)	0.54(0.76)	0.54(0.76)

As can be seen in Table 4. Our improved CNN model was able to detect scenes containing cloud,and shadow cover four types in both PlanetScope and Sentinel-2 imagery quite accurately (Label one F2 (OA) of 0.94 (0.92), Label two F2(OA) of 0.76(0.93), Label

three F2(OA) of 0.80(0.98). As the classification was performed at the scene-level (i.e. $128 \times 128m$ pixels, $400 \times 400m$), there is a substantial loss of information in image scenes labeled as, for example, 'partly cloudy' or 'partly shaded' if these masks were to be used for masking cloudy and shaded areas in high-resolution satellite imagery. However, this is also true for classifications generated using the MACCS algorithm, as it uses a dilatation procedure to enlarge the classification regions by 480m, which is larger than the size of our scenes (i.e. $400 \times 400m$). It is also important to note that while the Sen2Cor algorithm performs classification at the pixel level, it misses a lot of cloud shadows with per-label F2 for partly shaded labels of 0.17.

4. **Conclusions.** The algorithm in this paper can accurately detect scenes with clouds and shadows in PlanetScope and Sentinel-2 images, and at the same time, the accuracy of image detection for cloudy and shadowless views can reach 92. The detection accuracy of scenes that are partially cloudy without shadows and partly cloudy and partially shadowed is less than 80. The reason is that the number of training samples is not enough. Accordingly, the data set of these two labels can be further expanded to increase its detection accuracy.

In order to highlight the cloud and shadow labels, we manually divide them into four categories. In the next step, we can expand the labels on the underlying surface to transform the four-classification problem into a multi-classification problem, while continuing to improve our algorithm model. The detection accuracy is further enhanced. We can also consider multi-label classification to create more value for the dataset.

The significant contributions of this work are summarized as follows:

1) For multi-scene classification of high-resolution multi-spectral satellite images such as Sentinel-2, the improved AlexNet convolutional neural network algorithm is used, and the total accuracy can reach 99.

2) Compared with Yuri's research, the new method developed for classification of Sentinel-2 images is based on the set-based extended multi-label algorithm, which manually divides 12 categories into four categories, which significantly saves the difficulty of the experimental algorithm. The issue of cloud detection was highlighted. At the same time, compared with its four classification accuracy results, the four classification accuracy of this study is higher than its accuracy.

Acknowledgment. This work is supported by the National Natural Science Foundations of China (61871150), National Key R&D Program of China (2016YFB0502502).We gratefully acknowledge the helpful comments and suggestions of the reviewers,which have improved the presentation.

REFERENCES

[1] E. K. Wang, F. Wang, S. Kumari, J.Yeh, C. Chen, Intelligent monitor for typhoon in IoT system of smart city, *The Journal of Supercomputing*, vol.77, no.3, pp.3024-3043, 2021.

[2] J. F. Pekel, A. CottamA, N.Gorelick, A. S.Belward, High-resolution mapping of global surface water and its long-term changes, *Nature*, vol.540, no.7633, pp.418–422, 2016.

[3] A. Banskota, N. Kayastha , M. J. Falkowski, M. A. Wulder, R. E. Froese, J. C. White, Forest Monitoring Using Landsat Time Series Data: A Review, *Canadian Journal of Remote Sensing*,vol.40, no.5, pp.362–384, 2014.

[4] M.Wieland, M.Pittore, Large-area settlement pattern recognition from Landsat-8 data, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol.119, pp.294–308,2016.

[5] S. Yang, X.Wang, Sparse Representation and SRCNN based Spatio-temporal Information Fusion Method of Multi-sensor Remote Sensing Data, *Journal of Network Intelligence*, vol.6, no.1, pp. 40–53, 2021.

[6] Z. Y. Yan, M. L. Yan, and S.Hao, Cloud and Cloud Shadow Detection Using Multilevel Feature Fused Segmentation Network, *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 10, pp. 1600-1604, 2018.
[7] Y. L. Lecun, L. Bottou, Y. Bengio, Gradient-Based Learning Applied to Document Recognition, *Proceedings of the IEEE*, vol.86, no.11, pp.2278–2324, 1998.
[8] J. Deng, W. Dong, R. Socher, ImageNet: a Large-Scale Hierarchical Image Database, *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, Miami, Florida, USA, pp.20–22, 2009.
[9] L. B. Pedro, M. A. Ricardo, On the performance of GoogLeNet and AlexNet applied to sketches, *AAAI'16: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, Phoenix, Arizona USA, pp.1124-1128,2016.
[10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A.Rabinovich, Going deeper with convolutions,*2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, pp.1-5, 2015.
[11] K. He, X. Zhang, and S, Ren, Deep Residual Learning for Image Recognition*2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
[12] J. Wei, L. Sun, and C. Jia, Dynamic threshold cloud detection algorithms for MODIS and Landsat 8 data, *IEEE International Geoscience and Remote Sensing Symposium*, 2016.
[13] O. Yu, I. Haruma, and N. Ryosuke, A new Landsat 8 cloud discrimination algorithm using thresholding tests, *International Journal of Remote Sensing*, pp1–21, 2018.
[14] Y. Luo, A. P. Trishchenko, and K. V. Khlopenkov, Developing clear-sky, cloud and cloud shadow mask for producing clear-sky composites at a 250-meter spatial resolution for the seven MODIS land bands over Canada and North America, *Remote Sensing of Environment*, vol.112, no.12, pp.4167–4185, 2008.
[15] M. Main-Knorn, B. Pflug, and J. Louis, Sen2Cor for Sentinel-2 , 2017
[16] K. Y. Lee, C. H. Lin, Cloud Detection of Optical Satellite Images Using Support Vector Machines, *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp289–293, 2016.
[17] Z. Li, H. Shen, and H. Li, Multi-feature combined cloud and cloud shadow detection in GaoFen-1 wide field of view imagery, *Remote Sensing of Environment*, vol.191, pp.342–358, 2017.
[18] A. Hollstein, K. Segl, and L. Guanter, Ready-to-use methods for the detection of clouds, cirrus, snow, shadow, water, and clear sky pixels in Sentinel-2 MSI images, *Remote Sensing of Environment*, vol.8, no.8, pp.666, 2016.
[19] M. Hughes, H. Daniel, Automated Detection of Cloud and Cloud Shadow in Single-Date Landsat Imagery Using Neural Networks and Spatial Post-Processing, *Remote Sensing* , vol.6, no.6, pp.4907–4926, 2014.
[20] Y. Zi, F. Xie, and Z. Jiang, A cloud detection method for Landsat 8 images based on PCANet, *Remote Sensing of Environment*, vol.10, no.10, pp.887, 2018.
[21] D. U. Ufuk, C. Demirpolat, and M. F. Demirci, Fast cloud detection using low-frequency components of satellite imagery, *2017 25th Signal Processing and Communications Applications Conference (SIU)*, 2017.
[22] Niessen, J. Wiro, Medical image computing and computer-assisted intervention-MICCAI in *14th International Conference*, G. FichtingerAnne, M. Peters (eds.), Toronto, Canada, 2011.
[23] S. Ozkan, M. Efendioglu, and C. Demirpolat, Cloud Detection from RGB Color Remote Sensing Images with Deep Pyramid Networks, *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, 2018.
[24] Y. Zhan, J. Wang, J. Shi, Distinguishing Cloud and Snow in Satellite Images via Deep Convolutional Network, *Geoscience and Remote Sensing Letters*, vol.99, pp.1–5, 2017.
[25] S. Liu, W. Deng, Very deep convolutional neural network-based image classification using small training sample size, *Proc. of the 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pp. 730–734, Kuala Lumpur, 2015.
[26] F. Isikdogan, A. C. Bovik, P. Passalacqua, Surface Water Mapping by Deep Learning, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol.99, pp.1–10, 2017.
[27] S. Santra and K. Mali, Pixel variation problem identification in image segmentation for big image data set in cloud platform, *2016 IEEE International Conference on Advances in Computer Applications*, Coimbatore,India ,pp. 318–320, 2016.

[28] Y. Li, J. Chen, Q. Ma, H. K. Zhang, J. Liu, Evaluation of Sentinel-2A Surface Reflectance Derived Using Sen2Cor in North America *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* vol. 11, no. 6, pp. 1997–2021, 2018.

[29] W. Pramaditya, L. Wahyu. Random Forest Classification Scenarios for Benthic Habitat Mapping using Planetscope Image. *2019 IEEE International Geoscience and Remote Sensing Symposium*, Yokohama, Japan, pp.8245–8248, 2019.

[30] M. Segal-Rozenhaimer, A. Li, K. Das, V. Chirayath, Cloud detection algorithm for multi-modal satellite imagery using convolutional neural-networks (CNN), *Remote Sensing of Environment* ,vol. 2020, no.237, pp.111446,2020.

[31] Y. Shendryk, Y. Rist, C. Ticehurst, P. Thorburna, Deep learning for multi-modal classification of cloud, shadow and land cover scenes in PlanetScope and Sentinel-2 imagery, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 157, pp.124-136, 2019.