Cloud and Cloud Shadow Segmentation for Remote Sensing Imagery Via Filtered Jaccard Loss Function and Parametric Augmentation

Sorour Mohajerani, *Student Member, IEEE*, and Parvaneh Saeedi, *Member, IEEE*

***Abstract*—Cloud and cloud shadow segmentation are fundamental processes in optical remote sensing image analysis. Current methods for cloud/shadow identification in geospatial imagery are not as accurate as they should, especially in the presence of snow and haze. This article presents a deep learning-based framework for the detection of cloud/shadow in Landsat 8 images. Our method benefits from a convolutional neural network, Cloud-Net+ (a modification of our previously proposed Cloud-Net [1]) that is trained with a novel loss function [filtered Jaccard loss (FJL)]. The proposed loss function is more sensitive to the absence of foreground objects in an image and penalizes/rewards the predicted mask more accurately than other common loss functions. In addition, a sunlight direction-aware data augmentation technique is developed for the task of cloud shadow detection to extend the generalization ability of the proposed model by expanding existing training sets. The combination of Cloud-Net+, FJL function, and the proposed augmentation algorithm delivers superior results on four public cloud/shadow detection datasets. Our experiments on Pascal VOC dataset exemplifies the applicability and quality of our proposed network and loss function in other computer vision applications.**

***Index Terms*—38-Cloud, convolutional neural network (CNN), cloud detection, deep learning, image segmentation, landsat 8, loss function, remote sensing.**

I. Introduction

CLOUD and cloud shadow detection, along with cloud coverage estimation, are among the most critical processes in the analysis of remote sensing imagery. On the one hand, transferring remotely sensed data from air/space-borne sensors to ground stations is an expensive process from time, bandwidth, storage, and computational points of view. On the other hand, no useful information about the earth's surface can be extracted from optical images that are heavily covered by clouds and their shadows. Since, on average, 67% of the earth surface is covered by clouds at any given time [2], it seems that a considerable amount of resources can be saved by transferring only images with no/minimum amount of cloud and shadow coverage.

Cloud coverage by itself could provide useful information about climate and atmospheric parameters [3], as well as natural disasters such as hurricanes and volcanic eruptions [4], [5]. As a result, the identification of clouds and cloud shadows in images is an essential preprocessing task for many applications. Cloud and cloud shadow detection is even more challenging when only a limited number of spectral bands are available. Many air/spaceborne systems such as ZY-3, HJ-1, and GF-2 are equipped only with visible and near-infrared bands [6]. Therefore, algorithms that can identify clouds and their shadows from those few spectral bands become more essential.

In recent years, many cloud/shadow detection algorithms have been developed. These methods can be divided into three main categories: threshold-based [7]–[10], handcrafted [11]–[14], and deep learning-based methods [15]–[20].

Function of mask (FMask) [21]–[23] and automated cloud-cover assessment (ACCA) [24] algorithms are among the most well-known threshold-based algorithms for cloud identification. They construct a decision tree to label each pixel as cloud or other non-cloud classes. In each branch of the tree, a decision is made based on a thresholding function that utilizes one or more spectral bands of the data.

A group of handcrafted methods isolated haze and thick clouds from other pixels using the relationship between spectral responses of the red and blue bands. Some examples of these algorithms include [25], [26], which are derivatives of the haze optimized transformation [11]. Xu *et al.* [27] proposed a handcrafted approach, which involved a Bayesian probabilistic model incorporated with multiple spectral, temporal, and spatial features to separate cloud from noncloud regions.

With recent advances in deep-learning algorithms for image segmentation, several methods have been developed to detect cloud/shadow using deep learning. Xie *et al.* [19] trained a convolutional neural network (CNN) from multiple small patches. Their network classified each patch into one of the three classes of thin cloud, thick cloud, and clear. A major issue in cloud/shadow detection approaches using deep learning is the lack of accurately annotated training images since creating ground truths (GTs) for remote sensing imagery is time consuming and tedious [28]. In addition, default cloud masks provided in remote sensing products are mostly obtained through automatic/semiautomatic thresholding-based approaches, which often make them less accurate. Mohajerani *et al.* [29] removed wrongly labeled icy/snowy regions in default cloud masks in

Manuscript received January 29, 2021; revised March 18, 2021; accepted March 30, 2021. Date of publication April 2, 2021; date of current version April 29, 2021. This work was supported in part by the Government of Canada and MacDonald, Dettwiler, and Associates Ltd., through the Technology Development Program and in part by the Compute Canada. *(Corresponding author: Sorour Mohajerani.)*

The authors are with the School of Engineering Science, Simon Fraser University, Burnaby, BC V5A 1S6, Canada (e-mail: smohajer@sfu.ca; psaeedi@sfu.ca).

Digital Object Identifier 10.1109/JSTARS.2021.3070786

their dataset. They showed that the performance of a well-known semantic segmentation model (U-Net [30]) is considerably improved by training on those corrected GTs. Although existing methods deliver promising results, many of them do not provide robust and accurate cloud/shadow masks in scenes where bright/cold noncloud regions exist alongside clouds [15], [22].

Here, inspired by advances made in deep learning techniques, we propose a new algorithm to identify cloud/shadow regions in Landsat 8 images. Our proposed algorithm consists of a fully convolutional neural network (FCN), which detects cloud and cloud shadow pixels in an end-to-end manner. Our network, Cloud-Net+, is a modified version of our previously open-sourced model, Cloud-Net [1]. In the process of optimizing Cloud-Net+ model, a novel loss function named filtered Jaccard loss (FJL) is developed and used to calculate the error. As a result, Cloud-Net+ is trained more accurately. FJL makes a considerable difference in the performance of systems, especially for images with no cloud/shadow regions.

Many of the threshold based and handcrafted algorithms for cloud shadow detection utilize geometrical relationships between the illumination source, clouds, and shadows [21], [31], [32]. However, such relationships are not taken advantage of in deep learning-based approaches, where shadows are considered independent foreground objects. We incorporate this information through a meaningful parametric data augmentation approach to have a greater variety of shadows in each scene. As a result, newly generated images resemble original images as if they were captured at different times of the day with different sunlight directions. Our experiments show that such systematic augmentation works very well for shadow detection and simultaneous multiclass segmentation of clouds, shadows, and clear areas.

Unlike FMask and ACCA, the proposed approach is not blind to the existing global and local cloud/shadow features of an image. Also, since only four spectral bands—red, green, blue, and near-infrared (RGBNir)—are required for the system's training and prediction, the proposed method can be easily utilized for images obtained by most of the existing satellites as well as airborne systems. Unlike multitemporal-based methods such as [33], the proposed method does not require prior knowledge of the scene, e.g., cloud-free images. The only data required for the parametric augmentation are solar angles, which exist in the metadata accompanying each image. Moreover, such information is used for training purposes only and not for prediction. In addition, to being simple and straightforward, the proposed network can be used in other image segmentation applications.

In summary, the contributions of this article are as follows.

1) Proposing a novel loss function (FJL), which not only penalizes a model for poor prediction of clouds and shadows but also *fairly* rewards the correct prediction of clear regions. Our experiments show that FJL outperforms other commonly used loss functions. In addition, when Cloud-Net+ is trained with it, its results outperform state-of-the-art cloud/shadow detection methods over four public datasets of 38-Cloud, 95-Cloud, Biome 8, and SPARCS.
2) Proposing a sunlight direction-aware augmentation (SDAA) technique to boost the shadow detection performance. Unlike commonly used transformation-based augmentation techniques such as rotation and flipping, which are blind to the scene's geometry, SDAA generates synthetic shadows with various lengths, directions, and levels of shade.
3) Extending our public dataset, 38-Cloud [1], for cloud detection in Landsat 8 imagery to a new dataset (95-Cloud). This new dataset—which is made public—includes 57 more Landsat 8 scenes (than 38-Cloud) along with their manually annotated GTs. It will help researchers improve their cloud detection algorithms using more training and evaluation data.

The remainder of this article is organized as follows. In Section II, a summary of related works in the cloud detection field is reviewed. In Section III, our proposed method is explained. In Section IV, experimental results are discussed. Finally, Section V concludes this article.

II. Related Works

Cloud and shadow detection in remotely sensed images have been an active area of research for many years. One of the first attempts to distinguish between clouds and cloud-free areas was through a probabilistic classifier model [34]. However, it was limited to scenes with clouds over open sea surfaces. One of the first successful and more general automatic cloud detection methods was FMask (first version) [21]. FMask used seven bands of Landsat images to classify each pixel of an image into one of the five classes of land, water, cloud, shadow, and snow. Another version of it (FMask V3) [22] utilized cirrus band to distinguish cirrus clouds along with low altitude clouds. In the last version of it (FMask V4) [23], auxiliary data such as digital elevation maps (DEM), DEM derivatives, and global surface water occurrences (GSWO) were used in addition to the other usual bands for better performance on water, high altitude regions, and Sentinel-2 images.

Several multitemporal methods were developed for cloud/shadow detection [27], [35]–[37]. Mateo-García *et al.* [33] used cloud-free images of Landsat 8 scenes to identify potential clouds. A clustering and some threshold-based postprocessing steps then helped to generate final cloud masks. Zi *et al.* [38] combined a threshold-based method with a classical machine learning approach to segment superpixels and classify them into one of three classes of cloud, potential cloud, and noncloud.

Recently, several FCNs have been developed for cloud and cloud shadow detection tasks [28], [39]–[43]. Yang *et al.* [44] proposed an FCN, which detects clouds in ZY-3 thumbnail satellite images. A built-in boundary refinement approach was incorporated into their network (CDnetV1) to avoid further post-processing. The second version of that network, CDnetV2 [45], was equipped with two new feature fusion and information guidance modules to extract cloud features more accurately. Chai *et al.* [46] proposed Segnet Adaption (a modified version of a well-known FCN for semantic segmentation) to match with remotely sensed images. Segnet Adaption has shown promising results on Biome 8 and Landsat 7 cloud/shadow detection datasets. Recently, Jeppesen *et al.* [47] introduced RS-Net to identify clouds in Landsat 8 images. RS-Net was an FCN inspired by U-Net and was trained with both automatically (via FMask) and manually generated GT images of two public datasets. The authors showed that results obtained by a model trained with FMask's outputs outperform the FMask directly obtained results.

The authors of SPARCS CNN [48] also proposed a model to distinguish cloud, shadow, water, snow, and clear regions in Landsat 8 images. They used a pretrained VGG16 as the backbone of their fully convolutional network. They succeeded in reaching human interpreter accuracy with their model. RefUnet v1 [49] was another method, which focuses on retrieving fuzzy boundaries of clouds and shadows. The authors used a UNet for extracting course/rough cloud and cloud shadow in small patches of images. Then, boundaries of clouds/shadows in complete Landsat 8 masks were refined/sharpened using a dense conditional random field (CRF). In this method, the CRF refinement was applied as a postprocessing step. In the second version of RefUNet (RefUNet v2) [50], a simultaneous joint pipeline for detecting and refining edges was utilized. Therefore, the UNet network, which identified course masks, was concatenated with the proposed guided Gaussian filter-based CRF to refine boundaries in an end-to-end manner. In addition, refinements were done on small patch masks rather than large masks of the entire Landsat 8 scenes. In another work, the authors of Cloud-AttU [51] employed a UNet model that is enriched with a specific attention module in its skip connections. Multiple attention modules enabled the model to learn proper features by paying attention to the most relevant locations in input training images or feature maps. CloudFCN [28] was based on a UNet model with the addition of inception modules between convolution blocks in the contracting and expanding arms. In addition to pixel-level segmentation, the authors incorporated a feature into their algorithm for estimating cloud coverage in an image using a regression-friendly loss function. They demonstrated that CloudFCN could achieve high accuracy in the task of cloud screening under various conditions, such as the presence of white noise and using different quantization methods.

Image augmentation has been proven to be effective in increasing the accuracy of deep learning models. There are only a few offline augmentation algorithms reported for remote sensing imagery. Ma *et al.* [53] proposed a generative adversarial network (GAN) to synthesize images for scene labeling. Howe *et al.* [54] introduced another GAN-based approach for earth's surface object detection in airborne images. Zheng *et al.* [55] proposed a method for generating synthetic vehicles in aerial images. For the specific task of cloud/shadow segmentation, only basic geometric and color space transformations in the training phase are used for data augmentation [16], [56]. Mohajerani *et al.* [57], however, developed a GAN (CloudMaskGAN) to convert snowy scenes to nonsnowy and vice versa for augmentation of Landsat 8 images. To the best of our knowledge, CloudMaskGAN is the only reported nontransformation-based augmentation approach for cloud detection purposes.

III. Proposed Method

In this section, the proposed methodology for cloud/shadow detection is described. We focus on using Landsat 8 images. Landsat 8 is equipped with two optical sensors, which together collect eight spectral and two thermal bands. Only four spectral bands—Bands 2 to 5 (RGBNir)—are used (all with 30 m spatial resolution). To keep the proposed method more general, no thermal band is utilized.

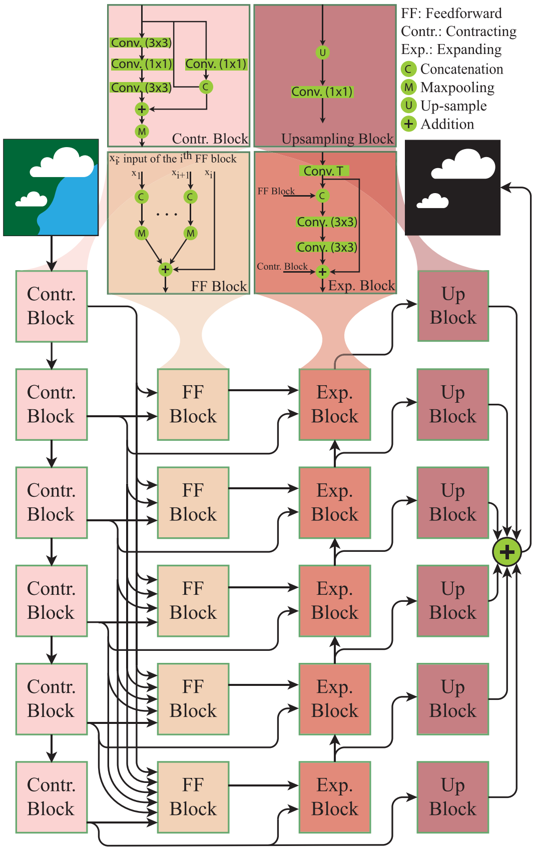


Fig. 1. Cloud-Net+'s architecture.

A. Architecture of the Segmentation Model

Similar to other FCNs, the proposed network is made of two main arms: a contracting arm, and an expanding arm. In the training phase, the contracting arm extracts important high-level cloud/shadow attributes. It also downsamples the input while increasing its depth. On the other hand, the expanding arm utilizes those extracted features to build an image mask—with the same size as the input. The network's input is a multispectral image, and its output is a grayscale cloud/shadow probability map for the input. In the case of multiclass segmentation, the output has multiple channels, one corresponding to each class.

We modified the network's architecture in our previous work [1] to develop a more efficient model that is more sensitive to clouds/shadows. Cloud-Net+, consists of six contracting and five expanding blocks (see Fig. 1). Successive convolutional layers are the heart of the blocks in both arms. The kernel size and the order of these layers play crucial roles in the quality of activated features, and therefore, they affect the final segmentation outcome directly. On the one hand, it seems that as the number of convolutional layers in each block increases, the distinction of the captured context improves. On the other hand, utilizing more of such layers explodes the complexity of the model. To address such tradeoff, we remove the middle 3×3 convolution layer in the last two contracting blocks and the first expanding block of Cloud-Net. Since those layers are dense and contain thousands of parameters, such removal decreases the number of network's parameters significantly. Then, in all contracting arm blocks, a 1×1 convolution layer is added between each two adjacent 3×3 convolution layers. Since each 1×1 convolution layer contains a small number of parameters, the total number of parameters of Cloud-Net+ (32.9 M) is 10% less than that of Cloud-Net (36.4 M). Utilizing the 1×1 kernel size in convolutional layers was suggested initially in [58], and its effectiveness is proven in works such as [59]. Employing such a kernel in expanding blocks does not yield a better recovery of the low-resolution feature maps. Therefore, we do not add it to the expanding blocks. Instead, an aggregation branch (AB) is added to combine all feature maps of the expanding blocks. AB consists of six up-sampling layers (bilinear interpolation) followed by a 1×1 convolution. It helps our model to retrieve cloud/shadow boundaries in the generated masks.

B. Loss Function

Soft Jaccard/Dice loss function has been widely used to optimize many image segmentation models [60]–[62]. The formulation of soft Jaccard loss for two classes of "0" and "1" is as follows:

$$J_L(t,y) = 1 - \frac{\sum_{i=1}^{N} t_i y_i + \epsilon}{\sum_{i=1}^{N} t_i + \sum_{i=1}^{N} y_i - \sum_{i=1}^{N} t_i y_i + \epsilon}. \tag{1}$$

Here, t represents the GT, and y is the output of the network. N denotes the total number of pixels in t. $y_i \in [0,1]$ and $t_i \in \{0,1\}$ are the ith pixel value of y and t, respectively, and ϵ is 10^{-7} to avoid division by zero. Soft Jaccard loss function, however, has a defect: overpenalization of images with no class "1" in their GTs.

Let us consider a small 2×2 input image with $t_0 = [0,0;0,0]$ and two possible predictions of $y_1 = [0.01, 0.01; 0.01, 0.01]$ and $y_2 = [0.99, 0.99; 0.99, 0.99]$. It is clear that y_1 would be a better prediction than y_2 since it can be interpreted as having no class "1" in the input image. However, soft Jaccard losses obtained by y_1 and y_2 are the same: $J_L(t_0, y_1) = J_L(t_0, y_2) \approx 1$. Consequently, the network penalizes y_1 as much as y_2, even though y_1 represents a better prediction. Indeed, the major problem with soft Jaccard loss function is that whenever there is no class "1" in the GT, the numerator of (1) equals to ϵ (which is a small number) and, as a result, the value of the loss approximates 1.

Observing this behavior, we propose a modified soft Jaccard loss function, FJL, with two different versions. The main idea behind FJL is to compensate for unfair values of soft Jaccard loss and replace them with proper values whenever there is no class "1" in the GT. We can summarize the goal of the FJL as follows:

$$\text{FJL}(t,y) = \begin{cases} G_L(t,y), & t_i = 0, \forall i \in \{1,2,3,\ldots,N\} \\ J_L(t,y), & \text{Otherwise} \end{cases} \tag{2}$$

where G_L represents a compensatory function. The condition in the first line of (2) indicates that when all pixels of GT are equal to zero, FJL is similar G_L. Clearly, this condition can be rephrased as follows:

$$\text{FJL}(t,y) = \begin{cases} G_L(t,y), & S = 0 \\ J_L(t,y), & S > 0 \end{cases} \tag{3}$$

where $S = \sum_{i=1}^{N} t_i$. This formulation can be rewritten—in a more general form—as the combination of two functions of J_L and G_L, in which each function is multiplied by, respectively, ideal highpass and lowpass filters

$$\text{FJL}(t,y) = k_G G_L(t,y) LP_{p_c}(S) + k_J J_L(t,y) HP_{p'_c}(S). \tag{4}$$

Here, LP_{p_c} denotes a lowpass filter with the cutoff point of p_c and $HP_{p'_c}$ denotes a highpass filter with the cutoff point of p'_c. k_G and k_J represent coefficients of compensatory and Jaccard losses, respectively. The magnitude of both filters is limited in the [0,1] range. In (4), the value of LP_{p_c} is 0 when $S > p_c$, so the value of $G_L(t,y)LP_{p_c}$ becomes zero, and as a result, FJL only has contributions from J_L. On the other hand, when $S < p'_c$, $HP_{p'_c}$ becomes 0, and FJL is only from the G_L part. This behavior can be simply described as a toggle switch between J_L and G_L.

To satisfy (3), the cutoff point of the two ideal filters should be equal, so they become complementary. Note that, in signal processing context, the cutoff usually refers to the "frequency" at which the magnitude of a filter changes. However, in this article, cutoff is the "value" of S at which the magnitude of a filter alters. Since this transition should occur when $S = 0$, the cutoff points are set to 0.

To have lowpass and highpass filters with smooth gradient characteristics, inspiring by [63], we have used the sigmoid function as follows:

$$LP_{p_c}(S) = \frac{1}{1 + \exp\left(m(S - p_c)\right)}, HP_{p'_c}(S) = \frac{1}{1 + \exp\left(m(-S + p'_c)\right)} \tag{5}$$

where m denotes the steepness of the sigmoid transition. Choosing sigmoid solves the problem of overpenalizing without adding any nondifferentiable elements or piecewise conditions to the loss function. This leads to a smooth switch between soft Jaccard and compensatory loss in FJL. Since HP and LP filters are not functions of y, the gradient of FJL is still continuous (without any jumps). By substituting (5) in (4), FJL is described as

$$\text{FJL}(t,y) = \frac{k_G G_L(t,y)}{1 + \exp\left(m(S - p_c)\right)} + \frac{k_J J_L(t,y)}{1 + \exp\left(m(-S + p'_c)\right)}. \tag{6}$$

To keep these filters close to ideal, m is required to be a large number. We have set m to 1000 in our experiments to have fast transitions from 0 to 1 and vice versa. In addition, p_c and p'_c are set to 0.5 to leave a safety margin and ensure that $LP_{0.5}(S = 0) = 1$ and $HP_{0.5}(S = 0) = 0$. Also, by setting $k_G = k_J = 1$, the magnitude of the loss remains in [0,1] range.

We can have different versions of FJL, by selecting different functions as G_L. Here, we investigate two different candidates G_{L1} and G_{L2}, to form two corresponding FJL versions FJL$_1$ and FJL$_2$. Our first candidate is the inverted Jaccard function, which is calculated by the complements of GT and prediction arrays. The second one is the normalized version of a common loss function in segmentation tasks, cross entropy (CE). The formulation of these candidates are as follows:

$$G_{L1}(t,y) \equiv \mathrm{InvJ}_L(t,y) = J_L(\bar{t},\bar{y}) \tag{7}$$

$$G_{L2}(t,y) \equiv \mathrm{CE}_{\mathrm{norm}}(t,y) = \frac{\mathrm{CE}(t,y)}{\mathrm{Max}} = \frac{\frac{-1}{N}\sum_{i=1}^{N}(t_i \log(y_i+\epsilon))}{\mathrm{Max}}$$

where $\bar{t}$ and $\bar{y}$ denote the *complements* of t and y, respectively. CE$_{\mathrm{norm}}$ denotes normalized CE. To make sure that the range of FJL$_2(t,y)$ is bounded to [0,1], CE values are normalized by division by their maximum possible value, Max $= \frac{-N}{N}\log(\epsilon) = 16.1180$. Max is obtained when all pixels in the predicted array are complements of GT.

C. Parametric Augmentation (SDAA)

When an image is captured by Landsat 8, the solar azimuth (θ_A) and the zenith (θ_Z) angles at the time of acquisition are recorded in a metadata file. We use these angles to generate different synthetic images by hypothetically changing the sun's location in the sky and, therefore, creating shadows under different illumination directions. This could be interpreted as capturing the same scene at different times of the day. Each image can be augmented unless it is free of shadow.

The main steps of the proposed SDAA are as follows: 1) removing original shadows; 2) changing the direction of illumination; 3) generating the locations of synthetic shadows by projecting clouds using the synthetic illumination direction of step 2; 4) recoloring synthetic shadow locations to generate different levels of shades via gamma transformation. Details of these steps are explained next.

First, a Landsat 8 image (from the training set), its shadow GT, and its cloud GT are selected. To generate realistic images, original shadows should be removed from the image, otherwise, the synthetic image will have two cast shadows. For the shadow removal step, a histogram matching between shadow regions and their shadow-free neighborhood is performed. The result of this step is a shadow-free image.

To create new shadows in the shadow-free image, we generate synthetic solar angles by adding offsets to the original solar angles of θ_A and θ_Z. Using these new angles, the direction of synthetic sunlight (in form of a two-dimensional vector projected on the scene's image plane) is calculated. By moving cloud pixels on the image plane and in the direction of the new sunlight, synthetic shadows of those clouds are generated. We made the following assumptions to simplify this process: all clouds have the same height, and the scene is located on a flat plane. The following equations are used to obtain the new locations of the cast shadows:

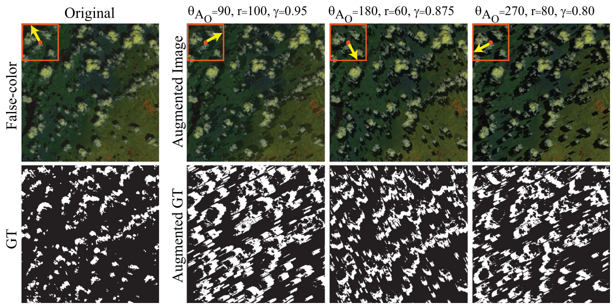

Original
θ$_{A_O}$=90, r=100, γ=0.95
θ$_{A_O}$=180, r=60, γ=0.875
θ$_{A_O}$=270, r=80, γ=0.80
False-color
GT
Augmented Image
Augmented GT

Fig. 2. Examples of SDAA obtained from one of the SPARCS images. The yellow arrow shows the sunlight direction in each image.

$$\begin{aligned} y_{sh} &= y_{cl} + r\sin(\theta_Z+\theta_{Z_O})\cos(\theta_A+\theta_{A_O}) \\ x_{sh} &= x_{cl} + r\sin(\theta_Z+\theta_{Z_O})\sin(\theta_A+\theta_{A_O}) \end{aligned} \tag{8}$$

where y_{cl}, y_{sh}, x_{cl}, and x_{sh} denote the location of a cloud pixel and its corresponding cast shadow along the $y-$ and x-axes. θ_{Z_O} and θ_{A_O} represent solar angle offsets. r (in pixels) is the shifting factor, which defines the length of a cast shadow. The greater the r is set, the more a piece of cloud is shifted, and therefore, the longer its shadow becomes. Since both r and θ_Z could affect the length of the cast shadow, instead of altering both, we keep θ_Z intact and only alter r. Therefore, θ_{Z_O} is set to 0.

The resultant synthetic shadow mask (SSM) will be used as the shadow GT of the augmented image in the training phase. In the next step, SSM is used to adjust the brightness of the synthetic shadow regions in the augmented image using gamma (γ) transformation ($i' = i^{\gamma}$). Pixel values of shadow-free regions in the image—including cloudy and clear areas—are not modified.

Multiple candidates have been considered for hyperparameters of SDAA: θ_{A_O}, r, and γ. These candidates consist of: $\theta_{A_O} = \{90, 180, 270\}^{\circ}$, $r = \{20, 40, 60, 80, 100\}$, and $\gamma = \{0.8, 0.825, 0.85, 0.875, 0.9, 0.925, 0.95, 0.975\}$. Note that θ_{A_O} is selected to cover all possible ranges of the solar azimuth angle—[0360]. Values of r greater than 100 generate too long and unrealistic shadows. In addition, choosing γ values smaller than 0.8 and greater than 0.975 leads to too dark and unnatural-looking bright shadows. The entire process is repeated for all spectral bands (R, G, B, and Nir). Fig. 2 displays some of the augmented images generated by SDAA.

IV. Experimental Settings and Results

A. Training Details

The size of the input data for Cloud-Net+ is $192 \times 192 \times 4$. Four spectral bands of each patch are stacked up in the following order: R, G, B, Nir, to create such input. Before training, input patches are normalized through division by 65 535. 20% of training data are used for validation during the training.

Input patches are randomly augmented using simple online geometric translations such as horizontal flipping, rotation, and zooming. Note that to make the proposed loss functions compatible with multiclass segmentation experiments, in each iteration of the training, loss values obtained for each class are averaged (with weights proportional to the inverse number of pixels in each class). The activation function in the last convolution layer of the network is a sigmoid. In the case of multiclass segmentation, softmax function is used in the last layer. Adam method [64] is utilized as the optimizer.

The initial weights of the network are obtained by a Xavier uniform random initializer [65]. The initial learning rate for the model's training is set to 10^{-4}. If the validation loss does not drop for more than 15 successive epochs, the learning rate is reduced by 70%. This policy is continued until the learning rate reaches 10^{-8}. The batch size is set to 12. The proposed network is developed using Keras deep learning framework with a single GPU.

B. Datasets

1) 38-Cloud Dataset: 38-Cloud dataset, which has been introduced in [1], consists of 8400 nonoverlapping (NOL) patches of 384×384 pixels extracted from 18 Landsat 8 Collection 1 Level-1 scenes as the training set. 9201 patches of the same spatial size obtained from 20 Landsat 8 scenes represent the test set. Scenes are mainly from North America and their GTs are manually extracted. This dataset includes only four spectral bands of R, G, B, and Nir.

All Landsat 8 scenes have black (empty) regions around them. These regions are created when the acquired images are aligned to have the geodetic north at their top. We eliminated training patches with more than 80% empty pixels. The number of nonempty training patches decreases to 5155, resulting in a significant decrease in training time.

2) 95-Cloud Dataset: To improve the generalization ability of deep neural networks trained on 38-Cloud dataset, we have extended it by adding 57 new Landsat 8 scenes to the training scenes of 38-Cloud dataset. Therefore, in total, the new training set consists of 75 scenes. 38-Cloud test set has been kept intact in 95-Cloud dataset for evaluation consistency. The GTs for the new scenes are manually extracted. Different images in 95-Cloud are selected to include various land cover types such as soil, vegetation, urban areas, snow, ice, water, haze, and different cloud patterns. The average cloud coverage percentage in 95-Cloud dataset images is kept around 50%. Following the same pattern as in 38-Cloud dataset, the total number of patches for training is 34 701 and for the test is 9201. Removing empty patches from 95-Cloud training set reduces the number of patches to 21 502. We made this dataset publicly available to the community at https://github.com/SorourMo/95-Cloud-An-Extension-to-38-Cloud-Dataset https://github.com/SorourMo/95-Cloud-An-Extension-to-38-Cloud-Dataset.

3) Biome 8 Dataset: Biome 8 dataset [66] is a publicly available dataset consisting of 96 Landsat 8 scenes with their manually generated GTs with five classes of cloud, thin cloud, clear, cloud shadow, and empty. We generated a binary cloud GT out of Biome 8 GTs by merging both thin cloud and cloud classes into one cloud class. We marked the rest of the classes as clear. For the shadow segmentation task, all classes, except shadow, are combined under a clear class. Following the same pattern from 38-Cloud and 95-Cloud datasets, the total number of cropped patches extracted from the entire Biome 8 dataset is 44 327. Removing empty patches reduces this number to 27 358.

4) SPARCS Dataset: SPARCS dataset [48], [67] consists of 80 patches of 1000×1000 extracted from complete Landsat 8 scenes. The GTs of these patches are manually generated. Each pixel is classified into one of the cloud, shadow, snow/ice, water, land, and flooded classes. For the binary segmentation of clouds/shadows, we have combined all noncloud and nonshadow classes under the clear class. In multiclass segmentation, three classes of cloud, shadow, and clear have been kept. Following the same pattern as previous datasets, the number of extracted NOL patches for this dataset is 720.

C. Model Evaluation

To evaluate the performance of the proposed algorithm, various experiments are performed. For 38-Cloud and 95-Cloud datasets, our model is trained and tested on each of those datasets' explicitly defined training set and testing set, respectively. For SPARCS dataset, as suggested in [47], we randomly extract five folds of images (each fold consists of 16 complete images or 144 NOL patches). Folds 2 to 5 are considered for training and fold 1 for testing in our experiments. We have also conducted 5-fold cross-validation (5CV) over the extracted folds to compare obtained results with those from RS-Net [47].

For Biome 8 dataset, we follow the instructions described in [47] and extract two folds from Biome 8 dataset. We randomly select two cloudy, two mid-cloud, and two clear scenes for each biome category. Therefore, 48 scenes are extracted for fold 1 and 48 scenes for fold 2. In our experiments, fold 2 is used for training and fold 1 for testing. To compare against RS-Net, 2-fold cross validation (2CV) is conducted, and the obtained numerical results are averaged.

Evaluation of the SDAA method requires different arrangements. For SPARCS, multiple sets of augmented training images are generated using various combinations of hyperparameters described in Section III-C. NOL patches of 384×384 pixels are extracted from those images and are added to the original training patches of SPARCS dataset one set at a time. Then, to find the best combination of hyperparameters, shadow detection training and testing are performed for each of those expanded training sets. The numerical results of these experiments are calculated and compared against those obtained from training with original patches only. Those combinations of hyperparameters that resulted in a higher Jaccard index than the training with original patches are identified (11 combinations). SDAA patches are generated by those combinations and added all together to the original training patches of a dataset for a final evaluation.

Considering SPARCS's folds 2 to 5 as the training set, we use the best hyperparameters confirmed in the previous steps to generate SDAA patches, leading to 6812 training patches in total. For Biome 8 dataset, we consider 70% of the patches (extracted from 32 scenes with shadow) for training and the rest for testing to have a fair comparison against Segnet Adaption [46]. Generating SDAA patches for this training set and then adding them to the original training patches results in 17 155 patches in total.

TABLE I
QUANTITATIVE RESULTS FOR THE PROPOSED LOSS FUNCTIONS OBTAINED BY CLOUD-NET+ MODEL FOR CLOUD DETECTION (IN %)

Training/Test	Loss	Jaccard	Precision	Recall	Accuracy
38-Cloud training set /test set	J_L	88.41	97.44	90.51	96.21
	proposed FJL_1	**88.85**	97.39	**91.01**	**96.35**
	CE	87.71	97.25	89.94	95.97
	proposed FJL_2	88.26	**97.70**	90.13	95.91
SPARCS folds 2-5 /fold 1	J_L	81.77	92.78	87.32	94.69
	proposed FJL_1	83.99	92.23	90.37	95.30
	CE	81.55	92.25	87.55	94.60
	proposed FJL_2	**85.11**	**93.43**	**90.52**	**95.68**

After training the proposed model with training patches of the abovementioned datasets, the obtained weights are saved and used for the evaluation of the model by prediction over unseen test scenes. Test patches of each scene are resized to $384 \times 384 \times 4$ and fed to the model. Then, the cloud/shadow probability map corresponding to each patch is generated. Next, probability maps (grayscale images) are stitched together to build a prediction map for a complete scene. A 50% threshold is used to get a final binary mask. In multiclass segmentation, an argmax operation is applied on the output probability map to produce a mask for each class. Cloud prediction by Cloud-Net+ takes about 30 s for all patches of a typical Landsat 8 scene with a P100-PCIE-12 GB GPU, half of which is for reading and preparing, and the other half is for inference.

D. Evaluation Metrics

Once cloud masks of complete Landsat 8 scenes (including empty patches) are obtained by our algorithm, they are compared against their corresponding GTs. Then, the performance is quantitatively measured by Jaccard index (or mean intersection over union), precision, recall, and accuracy. These metrics are commonly used in state-of-the-art segmentation algorithms and are defined as follows:

$$\text{Jaccard Index} = \frac{\sum_{i=1}^{M} tp_i}{\sum_{i=1}^{M}(tp_i + fp_i + fn_i)},$$

$$\text{Precision} = \frac{\sum_{i=1}^{M} tp_i}{\sum_{i=1}^{M}(tp_i + fp_i)}$$

$$\text{Recall} = \frac{\sum_{i=1}^{M} tp_i}{\sum_{i=1}^{M}(tp_i + fn_i)},$$

$$\text{Accuracy} = \frac{\sum_{i=1}^{M}(tp_i + tn_i)}{\sum_{i=1}^{M}(tp_i + tn_i + fp_i + fn_i)}$$

where tp, tn, fp, and fn are the numbers of true positive, true negative, false positive, and false negative pixels for each class in each test set scene, respectively. M is the total number of scenes in the test data.

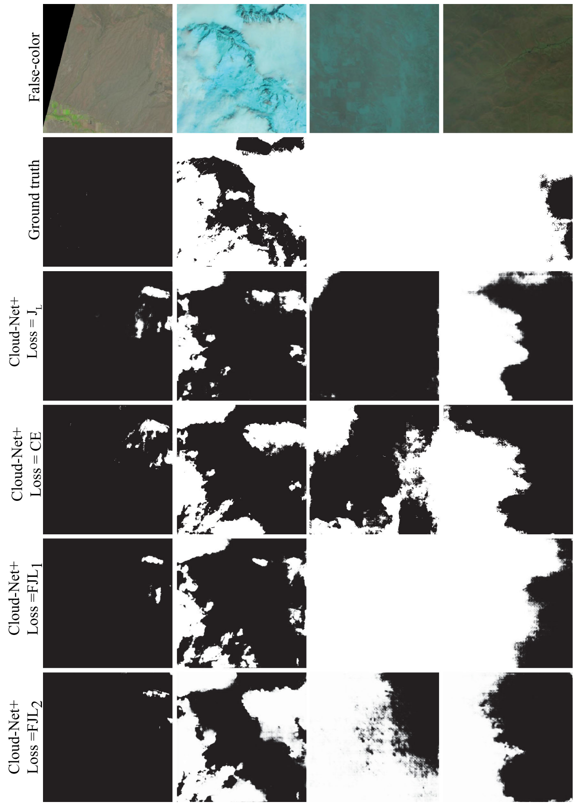


Fig. 3. Visual examples of the cloud detection results over patches of 38-Cloud test set obtained by different loss functions.

TABLE II
QUANTITATIVE RESULTS OF SHADOW DETECTION USING SDAA OBTAINED WITH CLOUD-NET+ AND DIFFERENT LOSS FUNCTIONS (IN %)

Training/Test	Loss	Jaccard	Precision	Recall	Accuracy
SPARCS folds 2-5 / fold 1	J_L	58.99	76.82	71.76	95.97
	CE	57.55	79.06	67.90	95.95
	proposed FJL_1	60.59	75.09	75.84	96.02
	proposed FJL_2	60.52	74.91	75.91	96.00
SPARCS folds 2-5 + SDAA patches / fold 1	J_L	**62.38**	**79.05**	**74.73**	**96.36**
	CE	**61.66**	**81.09**	**72.01**	**96.38**
	proposed FJL_1	**63.44**	**79.20**	**76.12**	**96.45**
	proposed FJL_2	**63.22**	**78.41**	**76.54**	**96.40**
Biome 8 70% / 30%	proposed FJL_1	53.79	58.45	**87.09**	98.21
	proposed FJL_2	52.11	56.31	**87.47**	98.07
Biome 8 70%+SDAA / 30%	proposed FJL_1	**54.71**	**64.05**	78.94	**98.43**
	proposed FJL_2	**54.41**	**59.67**	86.06	**98.27**

E. Quantitative and Qualitative Results

1) Evaluation of the Proposed Loss Functions: Table I demonstrates experimental results for evaluating the proposed loss functions. From Table I, we conclude that for both 38-Cloud and SPARCS datasets, Cloud-Net+ trained with both versions of FJL performs better than the original loss functions that they are derived from. A comparison of different loss functions over 38-Cloud dataset is displayed in Fig. 3. Note the better performance of both FJL loss functions on detecting thick and thin clouds (haze) over snow (two middle columns).

2) Evaluation of the Proposed SDAA Method: Numerical results for cloud shadow detection with and without SDAA using different loss functions are reported in Table II. From this table, the addition of the augmented patches obtained by SDAA improves all evaluation metrics over SPARCS dataset compared to using original training patches only. Using SDAA

TABLE III
COMPARISON OF THE PROPOSED CLOUD DETECTION METHOD WITH OTHER METHODS (IN %)

Train/Test	Method	Jaccard	Precision	Recall	Accuracy
38-Cloud training set /test set	Fmask V3 [22]	85.91	88.65	**96.52**	94.94
	U-Net [29]	85.03	96.15	88.02	95.05
	Cloud-Net [1]	87.32	97.60	89.23	95.86
	Cloud-AttU [51]	88.72	97.16	91.30	**97.05**
	Cloud-Net+ + FJL_1 (ours)	**88.85**	97.39	91.01	96.35
	Cloud-Net+ + FJL_2 (ours)	88.26	**97.70**	90.13	95.91
95-Cloud training set /test set	Fmask V3 [22]	85.91	88.65	**96.52**	94.94
	Cloud-Net [1]	90.83	**97.67**	92.84	97.00
	Cloud-Net+ + FJL_1 (ours)	**91.57**	96.94	94.28	**97.23**
	Cloud-Net+ + FJL_2 (ours)	91.01	97.49	93.19	97.06
SPARCS folds 2-5 /fold 1	Fmask V3 [22]	73.01	78.37	**91.44**	90.79
	SegNet [68]	75.80	83.35	89.33	92.23
	PSPNet [69]	78.91	90.11	86.39	93.71
	Cloud-Net [1]	81.36	90.81	88.66	94.46
	Cloud-Net+ + FJL_1 (ours)	84.00	92.23	90.37	95.30
	Cloud-Net+ + FJL_2 (ours)	**85.11**	**93.43**	90.52	**95.68**
SPARCS 5CV	RS-Net [47]	-	88.92	83.89	94.85
	Cloud-Net+ + FJL_1 (ours)	83.11	90.75	**90.82**	96.20
	Cloud-Net+ + FJL_2 (ours)	**83.14**	**90.77**	90.81	**96.23**
Biome 8 fold 2 /fold 1	Fmask V3 [22]	79.81	82.38	**96.24**	93.05
	Cloud-Net [1]	84.84	93.92	89.78	95.48
	Cloud-Net+ + FJL_1 (ours)	**85.44**	**92.88**	91.42	**95.55**
	Cloud-Net+ + FJL_2 (ours)	85.15	91.43	92.52	95.39
Biome 8 2CV	CloudFCN [28]	-	-	90.57	91.00
	RS-Net [47]	-	**92.15**	91.31	92.10
	Cloud-Net+ + FJL_1 (ours)	**85.15**	90.80	93.28	**95.27**
	Cloud-Net+ + FJL_2 (ours)	83.71	88.69	**93.79**	94.67

to train over Biome 8 dataset also results in a higher Jaccard index and accuracy than without it.

3) Comparison With State-of-the-Art Cloud Detection Methods: Table III compares cloud detection results obtained by the proposed method with other methods on various datasets. From this table, the combination of the proposed Cloud-Net+ with FJL_1 delivers a higher Jaccard index over 38-Cloud dataset than the U-Net, Cloud-AttU, and FMask V3 methods.

Four visual examples of the predicted cloud masks are displayed in Fig. 4. The second and third columns show scenes in the presence of snow. FMask and U-Net call many fps where land is covered with snow. The image of the third column does not include any clouds. The only method that successfully identifies all clear pixels is our method. The fourth column of this figure is captured over a region with bare soil—another difficult case for detecting clouds. The proposed method can distinguish those regions from clouds more accurate than the other methods.

Although it is not fair to compare our method (which only uses four spectral bands) with FMask V4 (which uses seven spectral bands, one thermal band, DEM, and GSWO), we obtained the numerical results of FMask V4 over 38-Cloud test set. FMask V4 shows lower accuracy (96.23%) compared to Cloud-Net+ with FJL_1. Note that FMask V4 has limited applicability, as not many remote sensing platforms are equipped with more than four spectral bands of RGBNir.

The numerical results over 95-Cloud test set (which has the same test set as 38-Cloud) are improved since they are obtained by a network trained with more training images (95-Cloud training set is larger than 38-Cloud). Fig. 5 highlights the improvement of results over this dataset.

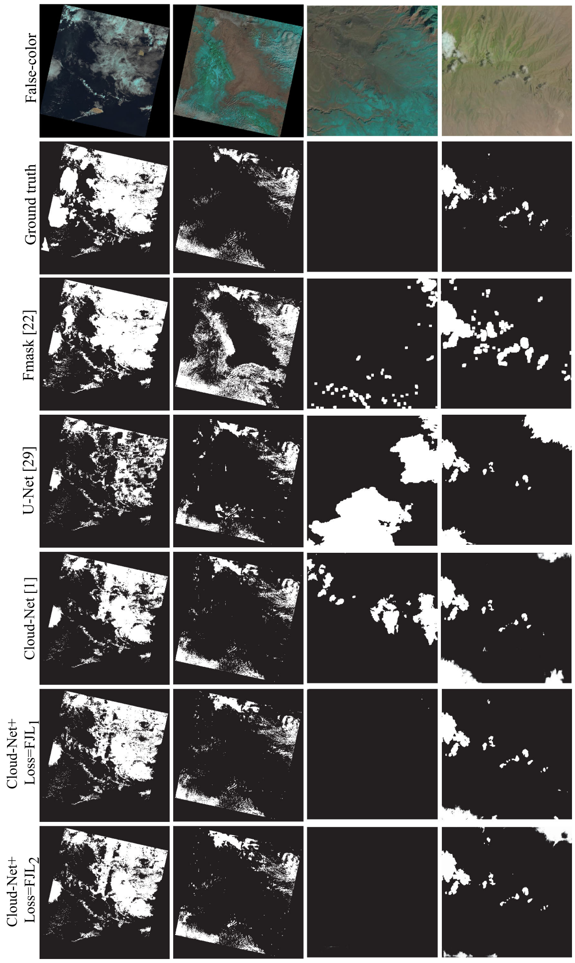


Fig. 4. Visual examples of cloud detection over 38-Cloud dataset.

For SPARCS dataset, Cloud-Net+ outperforms SegNet [68] and PSPNet [69], two common semantic segmentation methods. In addition, Cloud-Net+ outperforms RS-Net over two datasets of SPARCS and Biome 8. On Biome 8 dataset, our method, which utilizes only four spectral bands, shows a higher recall and accuracy than that of CloudFCN that uses eleven spectral bands. Note that Jaccard index and accuracy take into account both fp and fn in a predicted mask and penalize for all of those falsely labeled pixels, as opposed to precision and recall, which only penalize for one of the fp and fn. That is why Jaccard index—which is not reported in some papers—and accuracy are the most important metrics when one judges different methods. Visual results over Biome 8 dataset are shown in Fig. 6.

4) Comparison With State-of-the-Art Cloud and Shadow Detection Methods: The numerical results for simultaneous segmentation of cloud, shadow, and clear regions over two datasets are reported in Table IV. The reported Average Jaccard in this table is the average of Jaccard indices over three classes. Two settings were considered for evaluation by the authors of SPARCS CNN: a) using 72 images from the SPARCS dataset for training and the rest of SPARCS images for testing; b) utilizing the same 72 SPARCS images for training and 24 scenes from the Biome 8 dataset for testing (names of these 24 scenes are

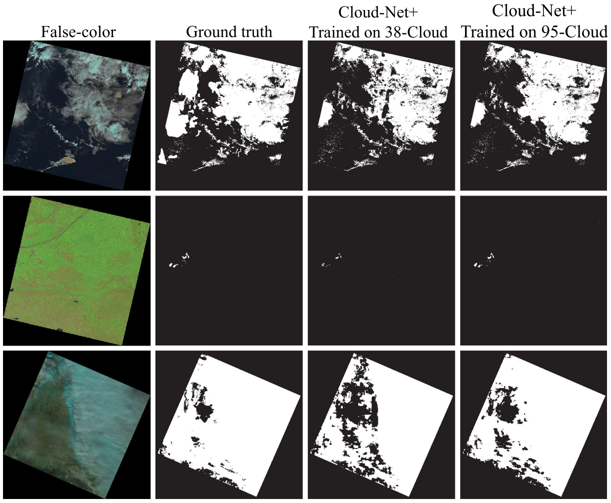

False-color
Ground truth
Cloud-Net+ Trained on 38-Cloud
Cloud-Net+ Trained on 95-Cloud

Fig. 5. Visual examples of cloud detection over 95-Cloud dataset.

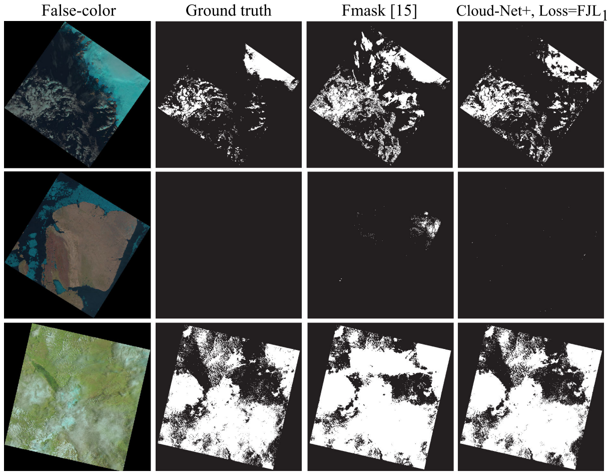


Fig. 6. Visual examples of cloud detection over Biome 8 dataset.

TABLE IV
QUANTITATIVE RESULTS FOR SIMULTANEOUS CLOUD AND SHADOW DETECTION (IN %). IN COLUMNS 2 TO 4, THE FIRST ROW INDICATES METRICS FOR THE CLOUD CLASS, THE SECOND ROW FOR THE SHADOW CLASS, AND THE THIRD ROW FOR THE CLEAR CLASS

Training/ Test	**Model**	**Jaccard**	**Precision**	**Recall**	**Average Jaccard**	**Accuracy**
SPARCS folds 2-5 /fold 1	Fmask V3 [22]	73.01 32.24 88.41	78.37 52.53 95.58	91.44 45.51 92.18	64.56	92.79
	Cloud-Net [1]	82.94 60.22 93.32	90.75 76.47 96.39	90.60 73.92 96.70	78.83	96.04
	Cloud-Net+ + SDAA + FJL_1 (proposed)	83.96 63.42 93.99	91.93 80.49 96.43	90.63 74.94 97.37	80.45	96.41
	Cloud-Net+ + SDAA + FJL_2 (proposed)	83.77 64.04 94.15	91.97 82.18 96.38	90.38 74.37 97.60	**80.65**	**96.47**
SPARCS 72 / 8	SPARCS CNN [48] w/ 2-px buffer	92.44 87.42 97.92	96.42 94.87 98.75	95.72 91.75 99.14	**92.59**	**100**
	Cloud-Net+ + SDAA + FJL_1 (proposed)	76.81 61.57 95.25	84.67 70.56 98.40	89.21 82.85 96.75	77.88	96.87
	Cloud-Net+ + SDAA + FJL_2 (proposed)	75.56 62.32 95.29	85.33 71.47 98.19	86.83 82.97 96.99	77.72	96.87
SPARCS 72 / Biome 8 24 scenes with shadow	SPARCS CNN [48] w/o 2-px buffer	not repo-rted	not repo-rted	not repo-rted	not reported	91.04
	SPARCS CNN [48] w/ 2-px buffer	not repo-rted	not repo-rted	not repo-rted	not reported	95.40
	Cloud-Net+ + SDAA + FJL_1 (proposed)	75.62 41.92 92.57	82.88 56.70 97.06	89.62 61.66 95.23	**70.04**	**95.62**
	Cloud-Net+ + SDAA + FJL_2 (proposed)	72.48 41.19 92.11	81.75 53.82 96.74	86.47 63.71 95.06	68.59	95.23
Biome 8 scenes with shadow 70% / 30%	Fmask V3 [22]	74.88 30.51 92.51	75.69 41.57 99.18	98.58 53.41 93.22	65.96	95.40
	Segnet Adaption [46]	not repo-rted	93.20 72.20 96.88	93.71 71.48 96.78	not reported	94.93
	Cloud-Net+ + SDAA + FJL_1 (proposed)	85.53 51.20 95.26	93.99 55.10 98.04	90.47 87.85 97.10	77.33	97.29
	Cloud-Net+ + SDAA + FJL_2 (proposed)	85.64 51.70 95.53	94.89 56.00 97.99	89.78 87.07 97.43	**77.62**	**97.40**
RefUNet Landsat 8 training set / test set	RefUNet v1 [49]	not repo-rted	85.49 41.36 89.52	97.17 09.03 84.31	not reported	93.43
	RefUNet v2 [50]	not repo-rted	87.93 40.64 91.99	95.83 39.00 84.51	not reported	93.60
	Cloud-Net+ + SDAA + FJL_1 (proposed)	86.52 36.48 93.06	94.43 58.22 95.37	91.17 49.42 97.46	**72.02**	**95.87**
	Cloud-Net+ + SDAA + FJL_2 (proposed)	86.27 36.63 92.89	94.59 57.01 95.28	90.74 50.60 97.37	71.93	95.79

listed in [48]). We used the same settings in our experiments for a fair comparison. Two pixels were considered by the authors of CNN SPARCS as a leeway buffer at boundaries of predicted clouds and shadows and clear regions, where either class was considered correct within two pixels from the predicted boundary in ground truths. Note that we used no buffer pixels for the evaluation of our method. According to Table IV, when SPARCS CNN is trained on SPARCS dataset and evaluated on Biome 8 (even by considering a two-pixel buffer), it is outperformed by our method in terms of accuracy. However, when this method is trained and evaluated on SPARCS dataset, the numerical results are almost perfect. It should be noted that those perfect results reported for this experiment are obtained by having a two-pixel leeway buffer and using ten spectral bands, as opposed to no leeway buffer and using only four spectral bands in our method. As an example of how much this two-pixel leeway affects numerical results of the SPARCS CNN, notice the improvement in accuracy (from 91.04% to 95.4%), where SPARCS CNN method is trained on the SPARCS dataset and evaluated on the Biome 8. Fig. 7 displays some visual results for simultaneous segmentation of clouds and cloud shadows over SPARCS dataset.

For comparison againt RefUNets, we downloaded the cited the training, validation, and testing Landsat 8 scenes and conducted our experiments on them. In RefUNet v1 and RefUnet

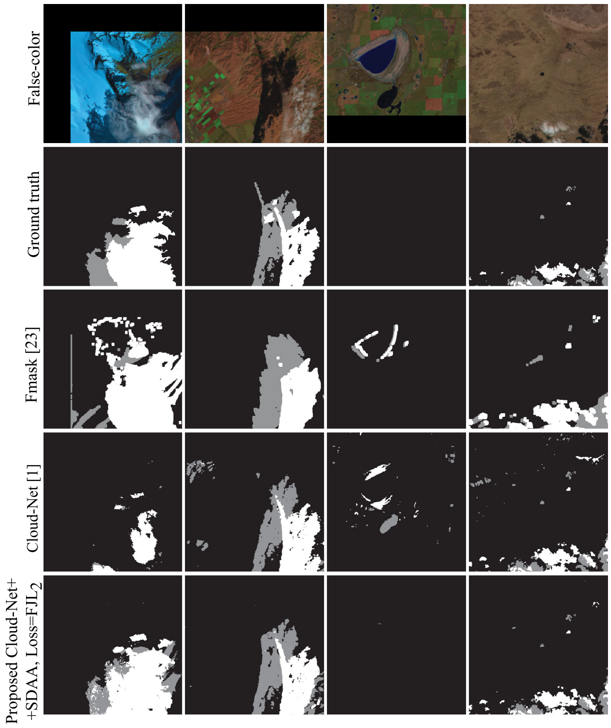


Fig. 7. Visual examples of cloud and shadow detection results over SPARCS dataset. White and gray pixels show cloud and shadow regions.

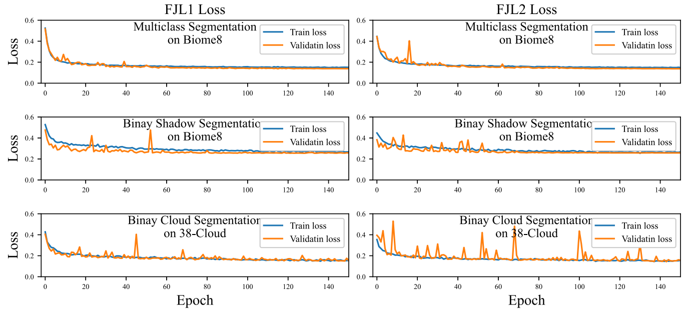


Fig. 8. Training and validation loss trend for three experiments using two versions of the proposed loss functions. For better visualization, only the first 150 epochs are shown.

v2, the input consists of four (RGBNir) and seven spectral bands, respectively, while the default cloud and shadow masks extracted from Landsat 8 products' QA bands are used for ground truths in experiments. Although sophisticated boundary refinements were applied on RefUnet methods's output masks, the obtained accuracy by either of these methods is not as high as the proposed method.

F. Recommendations on the Application of the Proposed Loss Functions

In general, the numerical results for FJL$_1$ and FJL$_2$ are not significantly different in the majority of the experiments conducted. (see Tables III and IV).

Another point is that validation loss values during training seem to be more stable in FJL$_1$ than FJL$_2$, as shown in Fig. 8. We believe that this is because the inverse Jaccard function is used as the compensatory function in FJL$_1$, which is of the same nature as the main J_L function. However, the cross entropy, which is used as a compensatory function in FJL$_2$, is calculated very differently from J_L as it is a log-based function. Although the range of both FJL$_1$ and FJL$_2$ is bound to [0,1], the similarity between the compensatory function and the main J_L function in FJL$_1$ leads to a more stable validation loss trend and fewer abrupt spikes than FJL$_2$.

TABLE V
COMPARISON OF THE CLOUD DETECTION PERFORMANCE OF THE TWO PROPOSED LOSS FUNCTIONS OVER VARIOUS BIOME TYPES (JACCARD/ACCURACY IN%). EACH REPORTED METRIC IS THE AVERAGE OF THE METRIC VALUES OBTAINED OVER FOLD 1 AND 2

Loss	Snow/ Ice	Shrubland	Forest	Water	Urban	Grass/ Crops	Wetland	Barren
FJL_1	**51.8/ 86.1**	**89.7/ 96.8**	84.8/ 94.2	**91.0/ 97.4**	94.4/ **98.4**	**94.2/ 98.1**	90.0/ 96.9	**87.3/ 96.5**
FJL_2	47.4/ 81.8	87.9/ 96.3	**86.1/ 94.6**	90.1/ 97.2	**94.6/ 98.4**	92.1/ 97.3	**93.0/ 97.8**	87.0/ 96.4

TABLE VI
QUANTITATIVE RESULTS FOR FURTHER EXPERIMENTS (IN %)

Training/Test	Model	Jaccard	Precision	Recall	Accuracy
38-Cloud training set /test set	Cloud-Net	87.32	97.60	89.23	95.86
	Cloud-Net+ w/o AB	87.84	**97.84**	89.57	96.04
	Cloud-Net+	**88.41**	97.44	**90.51**	**96.21**
SPARCS folds 2-5 /fold 1	Cloud-Net+ w/o OL	81.77	**92.78**	87.32	94.69
	Cloud-Net+ w OL	**83.35**	92.08	**89.78**	**95.11**

According to Table IV, for multiclass segmentation of remote sensing imagery, FJL$_1$ provides a higher recall for cloud class, which means that it is more conservative in predicting noncloud regions. As a result, for applications in which as many clear pixels as possible are required, FJL$_1$ is recommended. Otherwise, Table IV demonstrates very similar Jaccard and accuracy results for the identification of clouds, shadows, and clear regions in images, suggesting that either of these loss functions is effective and leads to high-quality prediction masks.

We investigated how each loss function performed over different biomes/land cover types for cloud detection and reported the results in Table V. We used the two-fold cross validation strategy for this investigation on Biome 8 dataset, where eight different biomes are distributed equally across the two folds. According to this table, the numerical results of FJL$_1$ and FJL$_2$ are very similar in most of the biomes. The only biome type in which both Jaccard and accuracy obtained by two proposed loss functions differ by a large gap (more than 5%), is the snow/ice biome. Table V shows that FJL$_1$ is more effective in snowy or icy land types than FJL$_2$. As a result, we recommend using FJL$_1$ for the cloud segmentation of regions including ice, snow, and mountains, especially during the colder seasons.

G. Further Experiments

To further highlight the effect of some of our choices made in this article, some experiments are conducted and summarized in Table VI. All experiments in this table are conveyed using soft

TABLE VII
QUANTITATIVE RESULTS OVER PASCAL VOC DATASET (IN %)

Method	Loss	Jaccard	Accuracy
Cloud-Net+	J_L	58.63	97.19
Cloud-Net+	FJL_1	**63.11**	**97.59**
Cloud-Net+	FJL_2	62.76	97.37

Jaccard loss function for cloud segmentation. First, Cloud-Net+ architecture is compared against Cloud-Net (with and without using the aggregation branch, AB). As Table VI indicates, Jaccard index and accuracy of Cloud-Net+ are higher than those of Cloud-Net, despite Cloud-Net+'s 10% less trainable parameters. Another interesting point is that by removing AB from Cloud-Net+ architecture, its performance deteriorates. This indicates that AB is capable of retrieving more details in the predicted cloud mask.

Due to computational hardware constraints, we perform our experiments with overlapping (OL) patches only on SPARCS dataset, which is the smallest of all datasets used for comparison in this article. For SPARCS, extracting and using patches with 50% overlap (as suggested in [47]) improved the performance of cloud detection.

H. Experiment on Pascal VOC Dataset

To test the proposed FJL functions beyond cloud/shadow detection applications, we have conducted experiments over Pascal VOC 2012 semantic segmentation dataset [70], [71]. This dataset contains 10 582 images for training and 1449 for testing. Each pixel in each image has been assigned to one of the 21 existing classes, including airplane, bicycle, cat, horse, and person, to name a few. To obtain a binary segmentation, we considered pixels of only one of the classes—person class—as the positive class and all other pixels in each image as the negative class. Numerical results in Table VII indicate that both versions of the proposed loss function outperform soft Jaccard loss for other types of images in addition to the remote sensing ones.

V. CONCLUSION

In this article, we have developed a novel system for cloud and cloud shadow detection in Landsat 8 imagery using a deep learning-based approach. The proposed network, Cloud-Net+, benefits from multiple convolution blocks that extract global and local cloud/shadow features. This network, which outperforms its parent network, Cloud-Net, has been optimized using two new and novel loss functions that reduce the number of misclassified pixels. These loss functions can be used for other binary or multiclass segmentation tasks, where the target object exists only in some of the images. In addition, a new augmentation technique, SDAA, is introduced for the task of cloud shadow detection. SDAA takes into account solar angles to generate natural-looking synthetic RGBNir images. We have also released an extension to our previously introduced cloud detection dataset. It will help researchers to improve the generalization ability of their cloud segmentation algorithms.

REFERENCES

[1] S. Mohajerani and P. Saeedi, “Cloud-Net: An end-to-end cloud detection algorithm for Landsat 8 imagery,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 1029–1032.

[2] M. D. King, S. Platnick, W. P. Menzel, S. A. Ackerman, and P. A. Hubanks, “Spatial and temporal distribution of clouds observed by MODIS onboard the terra and aqua satellites,” *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 7, pp. 3826–3852, Jul. 2013.

[3] C. R. Mirza, T. Koike, K. Yang, and T. Graf, “The development of 1-D ice cloud microphysics data assimilation system (IMDAS) for cloud parameter retrievals by integrating satellite data,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2008, vol. 2, pp. II-501–II-504.

[4] L. Zhu, M. Wang, J. Shao, C. Liu, C. Zhao, and Y. Zhao, “Remote sensing of global volcanic eruptions using Fengyun series satellites,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2015, pp. 4797–4800.

[5] R. S. Reddy, D. Lu, F. Tuluri, and M. Fadavi, “Simulation and prediction of hurricane Lili during landfall over the central gulf states using MM5 modeling system and satellite data,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2017, pp. 36–39.

[6] M. Huang, X. Xing, W. Luo, and Z. Wang, “Recalibration of offshore chlorophyll content based on virtual satellite constellation,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 7976–7979.

[7] H. Zhai, H. Zhang, L. Zhang, and P. Li, “Cloud/shadow detection based on spectral indices for multi/hyperspectral optical remote sensing imagery,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 144, pp. 235–253, 2018.

[8] Z. Li, H. Shen, H. Li, G. Xia, P. Gamba, and L. Zhang, “Multi-feature combined cloud and cloud shadow detection in GaoFen-1 wide field of view imagery,” *Remote Sens. Environ.*, vol. 191, pp. 342–358, 2017.

[9] S. Qiu, Z. Zhu, and C. E. Woodcock, “Cirrus clouds that adversely affect Landsat 8 images: What are they and how to detect them?,” *Remote Sens. Environ.*, vol. 246, 2020, Art. no. 111884.

[10] X. Zhu and E. H. Helmer, “An automatic method for screening clouds and cloud shadows in optical satellite image time series in cloudy regions,” *Remote Sens. Environ.*, vol. 214, pp. 135–153, 2018.

[11] Y. Zhang, B. Guindon, and J. Cihlar, “An image transform to characterize and compensate for spatial variations in thin cloud contamination of Landsat images,” *Remote Sens. Environ.*, vol. 82, no. 2, pp. 173–187, 2002.

[12] Z. An and Z. Shi, “Scene learning for cloud detection on remote-sensing images,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 8, pp. 4206–4222, Aug. 2015.

[13] R. Luo *et al.*, “Fusion of hyperspectral and lidar data for classification of cloud-shadow mixed remote sensed scene,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 8, pp. 3768–3781, Aug. 2017.

[14] P. Bo, S. Fenzhen, and M. Yunshan, “A cloud and cloud shadow detection method based on fuzzy c-means algorithm,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1714–1727, 2020.

[15] S. Ji, P. Dai, M. Lu, and Y. Zhang, “Simultaneous cloud detection and removal from bitemporal remote sensing images using cascade convolutional neural networks,” *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 732–748, Jan. 2021.

[16] M. Segal-Rozenhaimer, A. Li, K. Das, and V. Chirayath, “Cloud detection algorithm for multi-modal satellite imagery using convolutional neural-networks (CNN),” *Remote Sens. Environ.*, vol. 237, 2020, Art. no. 111446.

[17] Z. Li, H. Shen, Q. Cheng, Y. Liu, S. You, and Z. He, “Deep learning based cloud detection for medium and high resolution remote sensing images of different sensors,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 150, pp. 197–212, 2019.

[18] G. Mateo-García, V. Laparra, D. López-Puigdollers, and L. Gómez-Chova, “Transferring deep learning models for cloud detection between Landsat-8 and Proba-V,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 160, pp. 1–17, 2020.

[19] F. Xie, M. Shi, Z. Shi, J. Yin, and D. Zhao, “Multilevel cloud detection in remote sensing images based on deep learning,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 8, pp. 3631–3640, Aug. 2017.

[20] Y. Li, W. Chen, Y. Zhang, C. Tao, R. Xiao, and Y. Tan, “Accurate cloud detection in high-resolution remote sensing imagery by weakly supervised deep learning,” *Remote Sens. Environ.*, vol. 250, 2020, Art. no. 112045.

[21] Z. Zhu and C. E. Woodcock, “Object-based cloud and cloud shadow detection in Landsat imagery,” *Remote Sens. Environ.*, vol. 118, pp. 83–94, 2012.

[22] Z. Zhu, S. Wang, and C. E. Woodcock, “Improvement and expansion of the Fmask algorithm: Cloud, cloud shadow, and snow detection for Landsats 4-7, 8, and Sentinel 2 images,” *Remote Sens. Environ.*, vol. 159, pp. 269–277, 2015.

[23] S. Qiu, Z. Zhu, and B. He, “Fmask 4.0: Improved cloud and cloud shadow detection in Landsats 4 and Sentinel-2 imagery,” *Remote Sens. Environ.*, vol. 231, 2019, Art. no. 111205.

[24] R. R. Irish, J. L. Barker, S. Goward, and T. Arvidson, “Characterization of the Landsat-7 ETM automated cloud-cover assessment (ACCA) algorithm,” *Photogrammetry Eng. Remote Sens.*, vol. 72, pp. 1179–1188, 2006.

[25] Y. Zhang, B. Guindon, and X. Li, “A robust approach for object-based detection and radiometric characterization of cloud shadow using haze optimized transformation,” *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 9, pp. 5540–5547, Sep. 2014.

[26] S. Chen, X. Chen, J. Chen, P. Jia, X. Cao, and C. Liu, “An iterative haze optimized transformation for automatic cloud/haze detection of landsat imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 5, pp. 2682–2694, May 2016.

[27] L. Xu, A. Wong, and D. A. Clausi, “A novel Bayesian spatial-temporal random field model applied to cloud detection from remotely sensed imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 4913–4924, Sep. 2017.

[28] A. Francis, P. Sidiropoulos, and J. Muller, “CloudFCN: Accurate and robust cloud detection for satellite imagery with deep learning,” *Remote Sens.*, vol. 11, no. 19, p. 2312, 2019.

[29] S. Mohajerani, T. A. Krammer, and P. Saeedi, “A cloud detection algorithm for remote sensing images using fully convolutional neural networks,” in *Proc. IEEE Int. Workshop Multimedia Signal Process.*, 2018, pp. 1–5.

[30] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2015, pp. 234–241.

[31] B. Zhong, W. Chen, S. Wu, L. Hu, X. Luo, and Q. Liu, “A cloud detection method based on relationship between objects of cloud and cloud-shadow for chinese moderate to high resolution satellite imagery,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 11, pp. 4898–4908, Nov. 2017.

[32] X. Zhang, L. Liu, X. Chen, S. Xie, and L. Lei, “A novel multitemporal cloud and cloud shadow detection method using the integrated cloud Z-scores model,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 1, pp. 123–134, Jan. 2019.

[33] G. Mateo-García, L. Gómez-Chova, J. Amorós-López, J. Muñoz-Marí, and G. Camps-Valls, “Multitemporal cloud masking in the Google Earth Engine,” *Remote Sens.*, vol. 10, no. 7, p. 1079, 2018.

[34] J. Kittler and D. Pairman, “Contextual pattern recognition applied to cloud detection and identification,” *IEEE Trans. Geosci. Remote Sens.*, vol. GE-23, no. 6, pp. 855–863, Nov. 1985.

[35] G. Vivone, P. Addesso, R. Conte, M. Longo, and R. Restaino, “A class of cloud detection algorithms based on a MAP-MRF approach in space and time,” *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 5100–5115, Aug. 2014.

[36] D. S. Candra, S. Phinn, and P. Scarth, “Automated cloud and cloud-shadow masking for Landsat 8 using multitemporal images in a variety of environments,” *Remote Sens.*, vol. 11, no. 17, p. 2060, 2019.

[37] S. Ji, P. Dai, M. Lu, and Y. Zhang, “Simultaneous cloud detection and removal from bitemporal remote sensing images using cascade convolutional neural networks,” *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 732–748, Jan. 2021.

[38] Y. Zi, F. Xie, and Z. Jiang, “A cloud detection method for Landsat 8 images based on PCANet,” *Remote Sens.*, vol. 10, no. 6, p. 877, 2018.

[39] N. Chen *et al.*, “New neural network cloud mask algorithm based on radiative transfer simulations,” *Remote Sens. Environ.*, vol. 219, pp. 62–71, 2018.

[40] Z. Shao, Y. Pan, C. Diao, and J. Cai, “Cloud detection in remote sensing images based on multiscale features-convolutional neural network,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 6, pp. 4062–4076, Jun. 2019.

[41] W. Xie, J. Yang, Y. Li, J. Lei, J. Zhong, and J. Li, “Discriminative feature learning constrained unsupervised network for cloud detection in remote sensing imagery,” *Remote Sens.*, vol. 12, no. 3, p. 456, 2020.

[42] J. Lu *et al.*, “PSegnet and NPSegnet: New neural network architectures for cloud recognition of remote sensing images,” *IEEE Access*, vol. 7, pp. 87323–87333, 2019.

[43] M. Wieland, Y. Li, and S. Martinis, “Multi-sensor cloud and cloud shadow segmentation with a convolutional neural network,” *Remote Sens. Environ.*, vol. 230, 2019, Art. no. 111203.

[44] J. Yang, J. Guo, H. Yue, Z. Liu, H. Hu, and K. Li, “CDNet: CNN-Based cloud detection for remote sensing imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 6195–6211, Aug. 2019.

[45] J. Guo, J. Yang, H. Yue, H. Tan, C. Hou, and K. Li, “CDNetv2: CNN-Based cloud detection for remote sensing imagery with cloud-snow coexistence,” *IEEE. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 700–713, Jan. 2021.

[46] D. Chai, S. Newsam, H. K. Zhang, Y. Qiu, and J. Huang, “Cloud and cloud shadow detection in Landsat imagery based on deep convolutional neural networks,” *Remote Sens. Environ.*, vol. 225, pp. 307–316, 2019.

[47] J. H. Jeppesen, R. H. Jacobsen, F. Inceoglu, and T. S. Toftegaard, “A cloud detection algorithm for satellite imagery based on deep learning,” *Remote Sens. Environ.*, vol. 229, pp. 247–259, 2019.

[48] M. J. Hughes and R. Kennedy, “High-quality cloud masking of Landsat 8 imagery using convolutional neural networks,” *Remote Sens.*, vol. 11, no. 21, p. 2591, 2019.

[49] L. Jiao, L. Huo, C. Hu, and P. Tang, “Refined UNet: UNet-Based refinement network for cloud and shadow precise segmentation,” *Remote Sens.*, vol. 12, no. 12, p. 2001, 2020.

[50] L. Jiao, L. Huo, C. Hu, and P. Tang, “Refined UNet V2: End-to-end patch-wise network for noise-free cloud and shadow segmentation,” *Remote Sens.*, vol. 12, no. 21, p. 3530, 2020.

[51] Y. Guo, X. Cao, B. Liu, and M. Gao, “Cloud detection for satellite imagery using attention-based U-Net convolutional neural network,” *Symmetry*, vol. 12, no. 6, p. 1056, 2020.

[52] D. Ma, P. Tang, and L. Zhao, “SiftingGAN: Generating and sifting labeled samples to improve the remote sensing image scene classification baseline in vitro,” *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 7, pp. 1046–1050, Jul. 2019.

[53] J. Howe, K. Pula, and A. A. Reite, “Conditional Generative Adversarial Networks for Data Augmentation and Adaptation in Remotely Sensed Imagery,” in *Applications of Machine Learning*, M. E. Zelinski, T. M. Taha, J. Howe, A. A. S. Awwal, and K. M. Iftekharuddin, Eds., vol. 11139. Bellingham, WA, USA: SPIE, 2019, pp. 119–131.

[54] K. Zheng, M. Wei, G. Sun, B. Anas, and Y. Li, “Using vehicle synthesis generative adversarial networks to improve vehicle detection in remote sensing images,” *ISPRS Int. J. Geo-Inf.*, vol. 8, no. 9, p. 390, 2019.

[55] D. Marmanis, K. Schindler, J. Wegner, S. Galliani, M. Datcu, and U. Stilla, “Classification with an edge: Improving semantic image segmentation with boundary detection,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 135, pp. 158–172, 2018.

[56] S. Mohajerani, R. Asad, K. Abhishek, N. Sharma, A. v. Duynhoven, and P. Saeedi, “CloudMaskGAN: A content-aware unpaired image-to-image translation algorithm for remote sensing imagery,” in *Proc. IEEE Int. Conf. Image Process.*, 2019, pp. 1965–1969.

[57] M. Lin, Q. Chen, and S. Yan, “Network in Network,” in *Proc. Int. Conf. Learn. Represent.*, Canada, 2014.

[58] S. Zagoruyko and N. Komodakis, “Wide residual networks,” in *Proc. Brit. Mach. Vis. Conf.*, 2016, pp. 87. 1–87. 12.

[59] W. Waegeman, K. Dembczyński, A. Jachnik, W. Cheng, and E. Hüllermeier, “On the bayes-optimality of f-measure maximizers,” *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3333–3388, 2014.

[60] A. A. Novikov, D. Lenis, D. Major, J. Hladůvka, M. Wimmer, and K. Bühler, “Fully convolutional architectures for multiclass segmentation in chest radiographs,” *IEEE Trans. Med. Imag.*, vol. 37, no. 8, pp. 1865–1876, Aug. 2018.

[61] Y. Yuan, M. Chao, and Y. C. Lo, “Automatic skin lesion segmentation using deep fully convolutional networks with Jaccard distance,” *IEEE Trans. Med. Imag.*, vol. 36, no. 9, pp. 1876–1886, Sep. 2017.

[62] X. Lu, C. Ma, B. Ni, X. Yang, I. Reid, and M. Yang, “Deep regression tracking with shrinkage loss,” in *Proc. Eur. Conf. Comp. Vis.*, 2018, pp. 353–369.

[63] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. Int. Conf. Learn. Represent.*, USA, 2015, pp. 1–15.

[64] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proc. Int. Conf. Artif. Intell. Statist.*, 2010, vol. 9, pp. 249–256.

[65] S. Foga *et al.*, “Cloud detection algorithm comparison and validation for operational Landsat data products,” *Remote Sens. Environ.*, vol. 194, pp. 379–390, 2017.

[66] M. J. Hughes and D. J. Hayes, “Automated detection of cloud and cloud shadow in single-date landsat imagery using neural networks and spatial post-processing,” *Remote Sens.*, vol. 6, no. 6, pp. 4907–4926, 2014.

[67] V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: A. deep convolutional encoder-decoder architecture for image segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

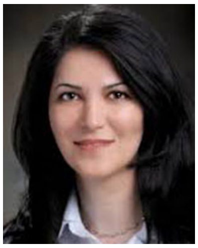
[68] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6230–6239.

[69] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge." *Int. J. of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[70] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik, "Semantic contours from inverse detectors," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 991–998.

Sorour Mohajerani (Student Member, IEEE) received the B.Sc. degree in electrical engineering from Tehran Polytechnic, Tehran, Iran, in 2010, and the M.Sc. degree in electrical engineering from the Iran University of Science and Technology, Tehran, Iran, in 2013.

In 2017, she was with the School of Engineering Science, Simon Fraser University, Vancouver, BC, Canada as a Ph.D. student and Research Assistant. Her current research interests include computer vision, deep learning, and remote sensing data processing



Parvaneh Saeedi (Member, IEEE) received the M.Sc. and Ph.D. degrees in electrical and computer engineering from the University of British Columbia, Vancouver, BC, Canada, in 1998 and 2004, respectively.

From 2004 to 2006, she was a Research Associate with MacDonald Detwiller and Associates Ltd. In 2007, she was with the School of Engineering Science, Simon Fraser University, Burnaby, BC, Canada, where she is an Associate Professor. She is currently the Associate Dean, Research and Graduate Studies with the Faculty of Applied Sciences, Simon Fraser University. Her research interests include the field of signal processing, which is the science behind the digital life and its applications, including artificial intelligence, machine learning, pattern recognition, and machine vision.