remote sensing

Article

Fine-Tuning Self-Organizing Maps for Sentinel-2 Imagery: Separating Clouds from Bright Surfaces

Viktoria Kristollari * and Vassilia Karathanassi

Lab of Remote Sensing, School of Rural and Surveying Engineering, National Technical University of Athens, 15780 Zographos, Greece; karathan@survey.ntua.gr

* Correspondence: vkristoll@central.ntua.gr

Received: 4 May 2020; Accepted: 10 June 2020; Published: 14 June 2020

Abstract: Removal of cloud interference is a crucial step for the exploitation of the spectral information stored in optical satellite images. Several cloud masking approaches have been developed through time, based on direct interpretation of the spectral and temporal properties of clouds through thresholds. The problem has also been tackled by machine learning methods with artificial neural networks being among the most recent ones. Detection of bright non-cloud objects is one of the most difficult tasks in cloud masking applications since spectral information alone often proves inadequate for their separation from clouds. Scientific attention has recently been redrawn on self-organizing maps (SOMs) because of their unique ability to preserve topologic relations, added to the advantage of faster training time and more interpretative behavior compared to other types of artificial neural networks. This study evaluated a SOM for cloud masking Sentinel-2 images and proposed a fine-tuning methodology to separate clouds from bright land areas. The fine-tuning process which is based on the output of the non-fine-tuned network, at first directly locates the neurons that correspond to the misclassified pixels. Then, the incorrect labels of the neurons are altered without applying further training. The fine-tuning method follows a general procedure, thus its applicability is broad and not confined only in the field of cloud-masking. The network was trained on the largest publicly available spectral database for Sentinel-2 cloud masking applications and was tested on a truly independent database of Sentinel-2 cloud masks. It was evaluated both qualitatively and quantitatively with the interpretation of its behavior through multiple visualization techniques being a main part of the evaluation. It was shown that the fine-tuned SOM successfully recognized the bright non-cloud areas and outperformed the state-of-the-art algorithms: Sen2Cor and Fmask, as well as the version that was not fine-tuned.

Keywords: self-organizing maps; artificial neural networks; cloud masking; Sentinel-2; bright surfaces; fine-tuning

1. Introduction

Optimized processing of acquired optical satellite images requires removal of cloud interference prior to atmospheric correction. Ruled based classification through the application of static or dynamic thresholds is the most common cloud masking approach [1–3]. This approach is derived by the assumption of higher reflectance and lower brightness temperature in clouds compared to other types of surfaces [4–6]. Most widespread threshold methods are Automatic Cloud Cover Assessment (ACCA) [7] and Function of mask (Fmask) [5,6], which was originally designed for Landsat imagery. A threshold based method is also used for the development of the Sentinel-2 cloud masks provided by the level 2A product [8]. Multi-temporal methods based on the idea that abrupt changes in image time series are mainly caused by the presence of clouds have been extensively implemented as well [9–11]. MAJA, which was designed for Sentinel-2 images, is among the most well-known in this category [12].

Attention has also been drawn to cloud masking applications that are based on machine learning approaches. Methods that make use of artificial neural network (ANN) architectures are the most recent in this category thanks to the efficient exploitation of the increasing computational power. Multi-layer perceptrons (MLPs) and convolutional patch-to-pixel or encoder-decoder segmentation architectures are the most common types used in current research. MLPs have been used by Hughes and Hayes [13] for Landsat 7 and by Taravat et al. [14] for Landsat 7 and MSG/SEVIRI. Patch-to-pixel convolutional neural network (CNN) approaches have been applied by Mateo et al. [15] for Proba-V and Le Goff et al. [16] for Spot 6 and have been also combined with random forest [17]. These approaches have been further proposed for the adaptation between different satellite platforms by Segal et al. [18] for WV-2 and Sentinel-2, and by Mateo et al. [19] for Landsat-8 and Proba-V. Concerning encoder-decoder segmentation approaches, architectures based on U-Net, Alexnet-FCN, ResNet-50 and Segnet models have been implemented in Landsat 7,8 [20–22], Sentinel-2 [23], ZY-3 [24,25], Gaofen-1 [26] and high resolution [27,28] satellites.

For the separation of clouds from bright surfaces, use of thermal bands is supposed to improve non-cloud bright object commission error, since they lead to the estimation of cloud height [5,20,29,30]. However such kind of bands are unavailable in Sentinel-2. Rule based cloud masking approaches usually attempt to distinguish bright surfaces either through the use of texture operators as a pre-processing step or through the use of morphology and geometry operators as a post-processing step. Nonetheless, the results most often need to be improved as shown from several current research studies implemented in Landsat [11,31,32], Gaofen-1 [33] and Proba-V [34] satellites that reported misclassification of bright built-up areas, soils and water bodies. A methodology designed for Sentinel-2 by Frantz et al. [35] who used a cloud displacement index based on the parallax effects of three highly correlated near infrared (NIR) bands, has shown most promising results until now.

Convolutional patch-to-pixel and encoder-decoder segmentation architectures have produced in general more successful and more effortless results for the separation of clouds from bright surfaces due to their inherent ability to perceive spatial information. Such a conclusion was reached in studies conducted in WV-2 [18], Sentinel-2 [18,36], Landsat [37] and Gaofen-1 [26] where bright non-cloud object misclassification was not observed. Satisfactory results were also produced by an artificial neural network architecture (ANN) that managed to separate sunglint and noise in Sentinel-2 ocean images [38].

Self-organizing maps (SOMs) [39,40] are a type of competitive ANN that projects data of high dimensionality to a space of low dimensionality by simultaneously preserving topology relations. SOMs are related to vector quantization (VQ), with the difference of conservation of topologic information that makes them suitable for the organization and visualization of complex datasets. Their concept is based on the associative neural properties of the brain where neurons are operating on a localized manner [41]. Contrary to other types of ANNs, SOMs do not perform error-correction learning but interpret the input information by the location of the response in the low-dimensional space without taking into account its magnitude. Even though SOMs are an unsupervised learning method, the produced clusters can be labeled given that available ground-truth data exists, and consequently the clusters can be converted to classes. Majority voting is the common approach to define the labels of the classes, represented by the neurons of the produced map [39,42,43]. SOMs are weakly represented in current machine learning cloud masking research even though recent studies report successful results with the additional advantage of faster training/fine-tuning time and more interpretative behavior compared to other types of ANNs. This fact led to their inclusion in the creation of the operational cloud masking products of Sentinel-2 [8] and Proba-V [44] satellites. Relative studies have been also implemented for Landsat 7 and MODIS [45–48].

The term "fine-tuning" for other types of neural networks (e.g., CNNs) refers to the use of pre-trained neural networks for different applications [19,24,26] than those that they were originally trained for. During fine-tuning, the pre-trained weights are used as initial weights and the network is further trained on the new dataset. As for SOMs, in the cases where fine-tuning is performed, the weights of the map neurons are updated through further training by taking into account the correctness or incorrecteness of the prediction [39,43]. Fine-tuning is supposed to highly increase classification accuracy in SOMs [39].

This study evaluates a SOM for cloud masking Sentinel-2 images and proposes a fine-tuning methodology based on the output of the non-fine-tuned network. The fine-tuning process manages to correct the misclassified predictions of bright non-cloud spectra without applying further training. It is important to note that the fine-tuning method follows a general procedure, thus its applicability is broad and not confined only in the field of cloud-masking. The study takes direct advantage of the similarities of the SOM to a brain map. In more detail, it is based on the fact that a detailed topographical map of the cerebral cortex of the brain can be deduced by various functional or behavioral impairments, or through stimulation of a particular site which leads to the disruption of a cognitive ability [39]. A SOM is trained on a spectral database created by Hollstein et al. [49] which is the largest publicly available for Sentinel-2 cloud applications and is tested on a truly independent (non-overlapping) database of Sentinel-2 cloud masks recently created by Baetens et al. [50] which is also publicly available. The trained SOM neurons are labeled through majority voting by use of the ground truth labels provided in the training database. Finding the neuron with the minimum Euclidean distance from each pixel of the images of the test database, leads to the production of the predicted cloud masks. In a next step, after observation of the cloud masks produced by the non-fine-tuned SOM, the fine-tuning process is applied. During fine-tuning, the SOM neurons that correspond to the bright misclassified non-cloud areas are detected by feeding the corresponding incorrectly classified spectral signatures into the network and consequently identifying the stimulated neurons. Then, the incorrect labels of the respective neurons are directly altered without applying further training. The network is evaluated both qualitatively and quantitatively with the interpretation of its behavior through multiple visualization techniques being a main part of the evaluation. The cloud masks are not only compared with ground truth data, but also with results produced by two state-of-the-art algorithms: Sen2Cor, Fmask. It is noted that in the context of this study, the term "bright non-cloud areas" refers to built-up areas, soils (e.g., desert) and coastal surfaces. It is also mentioned that the fine-tuning methodology proposed in this study was also applied in experiments that specifically targeted incorrectly classified snow pixels. Yet, the results were considered unsatisfactory since the correct classification of the snow pixels led to a large omission error of clouds. These experiments are not presented in the study.

2. Materials and Methods

2.1. Data Description

2.1.1. Training Set

The training set consisted of 8,799,998 Sentinel-2 reflectance spectra in total which form the database created by Hollstein et al. [49]. Sentinel-2 images contain 13 bands with spatial resolution 60 m (three bands), 10 m (four bands) and 20 m (six bands). The wavelengths of the 3 spatial resolutions of the Sentinel-2 instruments are shown in Table 1. The satellite images that correspond to these spectra were collected in 2016 (5,647,725 spectra) and 2017 (3,152,273 spectra) around the globe. Figure 1 depicts their location with black circles. Their processing level is 1C which denotes that they are not atmospherically corrected, thus they are suitable for the application of cloud masking methods. This spectral database is manually created and is the largest publicly available which is designed specifically for cloud masking applications. It contains six classes ("opaque cloud", "cirrus", "snow", "shadow", "water", "land") and the selection process was performed with 20 m resolution. The selection process included the usage of spectral tools such as image enhancements and false-color composites, as well as the observation of the spectral signatures. Table 2 presents the number of spectra for each class. The data are stored in two separate .hdf5 files according to the collection year of the images.

Table 1. Wavelengths of the three spatial resolutions of Sentinel-2.

Spatial Resolution (m)	Band Number	S2A	S2B
		Central Wavelength (nm)	Central Wavelength (nm)
10	2	496.6	492.1
	3	560	559
	4	664.5	665
	8	835.1	833
20	5	703.9	703.8
	6	740.2	739.1
	7	782.5	779.7
	8A	864.8	864
	11	1613.7	1610.4
	12	2202.4	2185.7
60	1	443.9	442.3
	9	945	943.2
	10	1373.5	1376.9

Table 2. Spectra comprising the training set.

Class	Coverage	Number of Spectra
opaque cloud	opaque clouds	1,500,202
cirrus	cirrus and vapor trails	1,205,979
snow	snow and ice	1,271,143
shadow	shadows from clouds, cirrus, mountains, buildings, etc.	1,113,066
water	lakes, rivers, sea	1,435,003
land	remaining: crops, mountains, urban, etc.	2,274,605

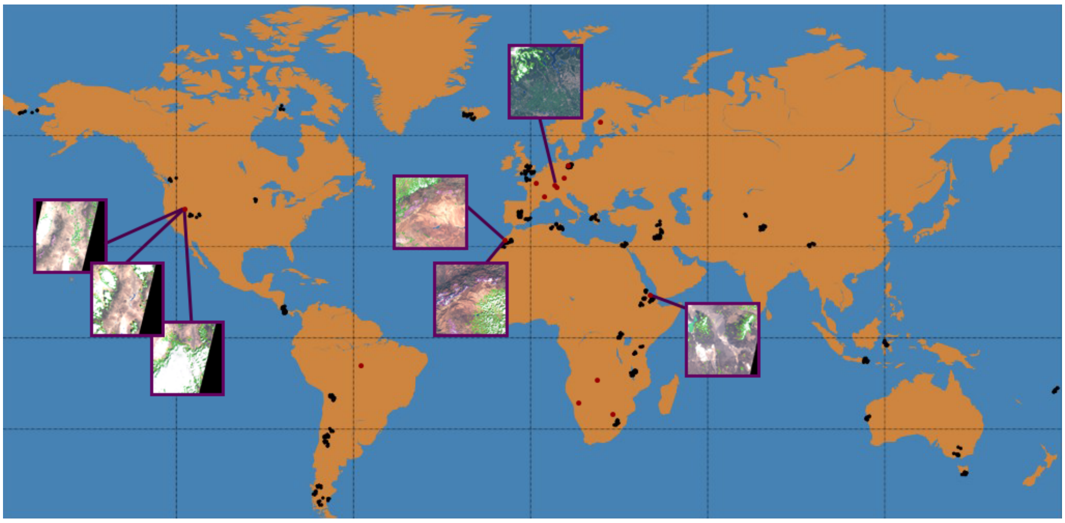


Figure 1. Location of the images of the training set (black circles) and the test set (red circles). The thumbnails depict cases with bright non-cloud objects.

2.1.2. Test Set

The test set consisted of 34 Sentinel-2 level 1C images. Their corresponding cloud masks were provided by the database created by Baetens et al. [50]. This database is the only publicly available source of Sentinel-2 ground truth cloud masks at the moment. The creation of the masks was based on the application of random forest and their accuracy is reported to be 98%. The images cover different areas around the world with various land cover and cloud properties: three images were collected in North America, four in South America, nine in Africa and 18 in Europe. The collection dates cover all seasons of the year: seven images were collected in winter (December, January, February), eight in spring (March, April, May), 11 in summer (June, July, August) and eight in fall (September, October, November) between seven a.m. and six p.m. UTC. Before feeding the spectra of the images into the SOM, the bands with spatial resolution 10 and 20 m were resampled to 60 m, since cloud masking applications do not require higher spatial resolution. It is noted that the effect of the spatial resolution difference between the training set (20 m) and the test set (60 m) is expected to be insignificant. In addition, the lower resolution (60 m) significantly improves the inference time of the SOM network because the final size of the images is 1830 × 1830 pixels instead of 5490 × 5490. Figure 1 depicts the location of the images with red circles. It is mentioned that each image covers an area of 109.8 × 109.8 km^2. It is also worth noting that the test set is truly independent from the training set because the spectra do not overlap.

2.2. Theoretical Background

The SOM was introduced by Kohonen [39,40]. It is a shallow ANN architecture which consists of an input layer and an output layer depicted as a 2-dimensional (2D) grid. The output layer is fully connected to the input layer and is made up of a set of neurons (nodes) that represent feature vectors. The neurons are interconnected through receptive fields called neighborhoods and the coefficients of the vectors represent the weights of the network. The size of the vectors of the output layer is the same as the size of the vectors of the input layer. Before starting the training process, random initialization of the feature vectors of the neurons is commonly performed. During training, the network reads a random datapoint and the distance (e.g., Euclidean) of the input with all feature vectors (neurons) is computed. The neuron that presents the minimum distance is the winning neuron which is called the Best Matching Unit (BMU). The neurons of the neighborhood are also activated and the distance of their feature vectors from the input datapoint is reduced. The above process is repeated until a pre-defined stopping criterion (e.g., number of iterations) is met. In each iteration, both the learning rate and the neighborhood are decreased to ensure convergence. A graphical representation of the SOM training process is illustrated in the lower left part of Figure 2 (stage 1 of methodology).

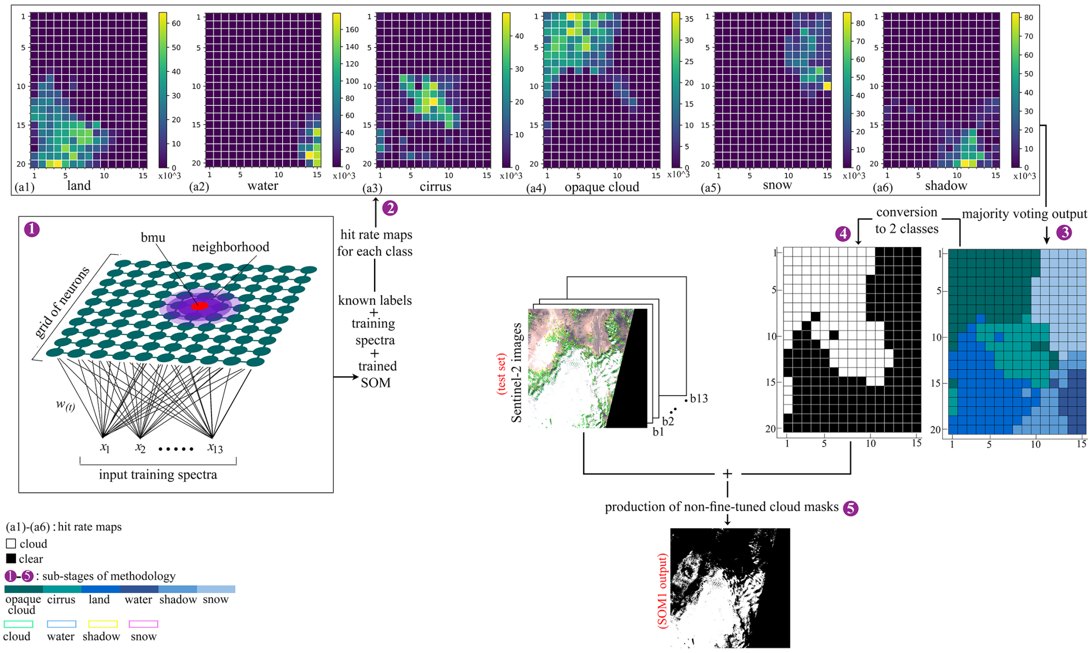


Figure 2. Sub-stages of the methodology of the study: self-organizing map (SOM) training (Stage 1) and production of non-fine-tuned cloud masks (Stage 2: sub-stages 2–5).

2.3. Proposed Methodology

The proposed method consists of three main stages. During the first stage, the SOM is trained on the spectra of the training set. During the second stage, the non-fine-tuned cloud masks of the test set are created. This process involves the calculation of the hit rate maps for each class of the training set and the labeling of the SOM neurons through majority voting. The trained SOM before applying the fine-tuning process will be called "SOM1". Finally, during the third and final stage, the fine-tuning process is applied. It includes the manual sampling of incorrectly classified bright non-cloud pixels by SOM1 and the correction of the labels of their corresponding neurons. By use of the corrected SOM, the temporary fine-tuned cloud masks are produced. Applying a median and a dilation filter leads to the creation of the final fine-tuned cloud masks. We will refer to the final fine-tuned cloud masks as "SOM2" output. The analysis of the three main stages is written in Sections 2.3.1, 2.3.2 and 2.3.3. In addition, the sub-stages of the methodology are depicted in Figures 2 and 3 and are also listed below. The second stage includes the sub-stages represented by the numbers 2–5 of the list and the third stage includes the sub-stages represented by the numbers 6–9.

1. Training of the SOM with spectra randomly selected from the training set.
2. Production of hit rate maps for each class by feeding the complete training set with known spectra labels into the trained SOM and detecting the BMUs.
3. Labeling of the SOM neurons through majority voting.
4. Conversion of the six classes to two, which define the cloud and non-cloud classes.
5. Production of non-fine-tuned cloud masks (SOM1 output) for the entire test set.
6. Observation of the temporary non-fine-tuned cloud masks and manual sampling of incorrectly classified bright non-cloud pixels.
7. Detection of the neurons that correspond to the bright non-cloud pixels and correction of their labels.
8. Production of the temporary fine-tuned cloud masks.
9. Application of a median and a dilation filter and production of the final cloud masks (SOM2 output).

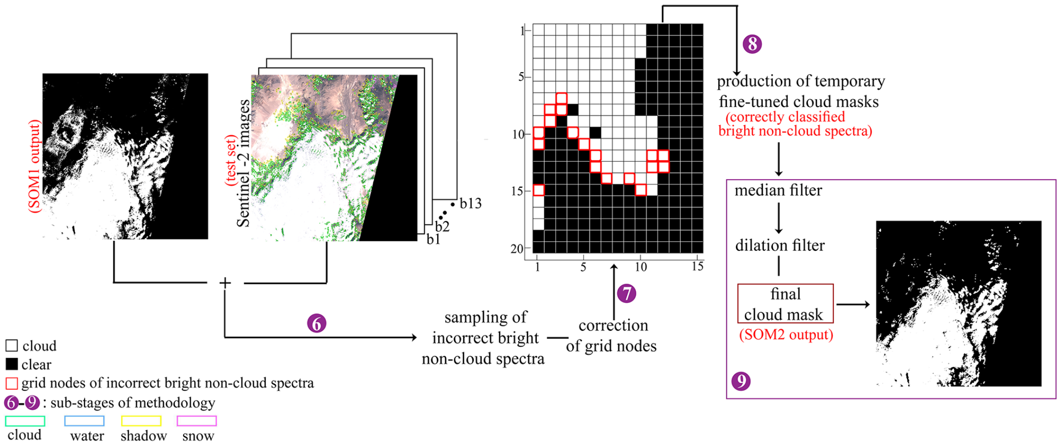


Figure 3. Sub-stages of the methodology of the study: Fine-tuning process (Stage 3: sub-stages 6–9).

2.3.1. Training Process

This section presents the training process of the SOM which represents the first stage of the methodology (Figure 2). The study implemented a SOM according to the Python code for unsupervised learning created by Riese et al. [51]. Before starting the training of the network, a feature scaling process was applied on every wavelength of the Sentinel-2 signatures of the training set (min-max normalization) (Equation (1)) in order to impede variables of higher magnitude to prevail over variables of lower magnitude.

$$x_{\text{scaled}} = \frac{x - x_{\min}}{x_{\max} + x_{\min}} \tag{1}$$

where x: value of a signature in a given band, x_{min}: minimum value of all signatures of the dataset in a given band and x_{max}: maximum value of all signatures of the dataset in a given band.

The training was performed by using all categories of the training set (six) and all Sentinel-2 bands (13). The size of the 2D rectangular grid was selected to be $20_{rows} \times 15_{columns}$ nodes. There is no rule about the selection of the grid size, but it is noted that it is the SOM parameter that mostly affects the processing time. The values of the feature vectors of the neurons were initialized by taking random samples from a uniform distribution in the interval $[0, 1)$. The distances between the input datapoints and the feature vectors of the neurons were calculated according to the Euclidean distance (Equation (2)) and the learning rate according to the equation proposed by Barreto and Araujo [52] (Equation (3)).The start value of the learning rate was set to 0.5 and the end value to 0.05.

$$d(\boldsymbol{x}, \boldsymbol{w}) = \sqrt{\sum_{j=1}^{n} (x_j - w_j)^2} \tag{2}$$

where $\boldsymbol{x}$: input datapoint, $\boldsymbol{w}$: SOM node and n: the dimension of the vectors $\boldsymbol{x}, \boldsymbol{w}$

$$a(t) = a_0 \cdot (\frac{a_{\text{end}}}{a_0})^{t/t_{\text{max}}} \tag{3}$$

where a_0: start value of the learning rate, a_{end}: end value of the learning rate, t: number of current iteration and t_{max}: number of maximum iterations.

As for the number of iterations the "rule of thumb" proposed by Kohonen [39] stating that "the number of maximum iterations should be at least 500 times the number of network units" was followed ($\geq (15 \times 20 \times 500)$, i.e., $\geq$ 150,000) and 1,000,000 iterations were performed. Decreasing learning rates are often implemented in ANNs to increase convergence and prevent oscillations [51].

The neighborhood radius was calculated according to the equation proposed by Matsusita et al. [53] (Equation (4)). The start value of the neighborhood function (radius) was chosen to be $\max(\frac{n_{rows}}{2}, \frac{n_{columns}}{2})$ as usually suggested.

$$\sigma(t) = \sigma_0 \cdot \left(1 - \frac{t}{t_{\text{max}}}\right) \tag{4}$$

where σ_0: start value of the neighborhood radius (t, t_{max}: as defined in Equation (3)).

The neighborhood distance weight which is dependent of the neighborhood radius and the Euclidean distance between the BMU and every other node on the SOM grid was calculated by Equation (5), proposed also by the researchers mentioned above. This equation is called "Pseudo-Gaussian".

$$h_{c,i}(t) = \exp\left(-\frac{d(c,i)^2}{2\sigma(t)^2}\right) \tag{5}$$

where $d(c, i)$: distance between the BMU c and node i on the SOM grid ($\sigma(t)$: as defined in Equation (4)).

Finally, the weights of the SOM after each iteration were updated according to Equation (6).

$$\boldsymbol{w}_i(t+1) = \boldsymbol{w}_i(t) + a(t) \cdot h_{c,i}(t) \cdot (\boldsymbol{x}(t) - \boldsymbol{w}_i(t)) \tag{6}$$

where $\boldsymbol{w}_i(t)$: vector of the weights of node i at iteration t and $\boldsymbol{x}(t)$: datapoint at iteration t ($a(t)$:as defined in Equation (3), $h_{c,i}(t)$: as defined in Equation (5)).

The steps of the training process are depicted in Algorithm 1. After defining the parameters of the network, the training was performed on a CPU (i7-8th generation, 3.7 GHz). It was a rapid process that lasted approximately two minutes.

Algorithm 1 : SOM training process

```
Input: training set
Input: start value of the learning rate $a_0$
Input: end value of the learning rate $a_{\text{end}}$
Input: start value of the neighborhood function $\sigma_0$
Input: number of maximum iterations $t_{\max}$
Output: trained SOM
Generate random weights $\boldsymbol{w}_i(t)$
Set number of current iteration equal to 1 ($t = 1$)
while $t < t_{\max}$ do
    Get random input datapoint $\boldsymbol{x}(t)$
    Find BMU $c(\boldsymbol{x})$
    Calculate learning rate $a(t)$
    Calculate neighborhood function $\sigma(t)$
    Calculate neighborhood distance weights $h_{c,i}(t)$
    Modify weights $\boldsymbol{w}_i(t+1)$
    $t \leftarrow t + 1$
end while
```

2.3.2. Production of Non-Fine-Tuned Cloud Masks

This section presents the second stage of the methodology (Figure 2) which leads to the production of the non-fine-tuned cloud masks. The non-fine-tuned cloud masks of the test set were produced after the training process was completed. Labeling of the SOM neurons is the condition that needs to be fulfilled before the creation of the masks. The labeling was accomplished through majority voting which was applied on the hit rate maps of each class. The hit rate map (Figure 2(a1–a6)) is a visualization technique that denotes the number of times a neuron was detected as a BMU. For their computation, every single signature of the training set (∼nine million signatures) was fed into the network and the BMU was detected. As already explained in Section 2.2, the BMU that corresponds to a spectral signature is the node that presents the minimum distance. Then, the "hits" were computed for each of the six classes. The creation of the hit rate maps for each class was possible because the labels of the training data were known. The computation of the BMUs for the training data lasted approximately 18 min. After the calculation of the hits, the majority voting was applied where each neuron was assigned the label of the class that corresponded to its maximum hits, e.g., in case the neurons were classified as opaque cloud, the following condition was true:

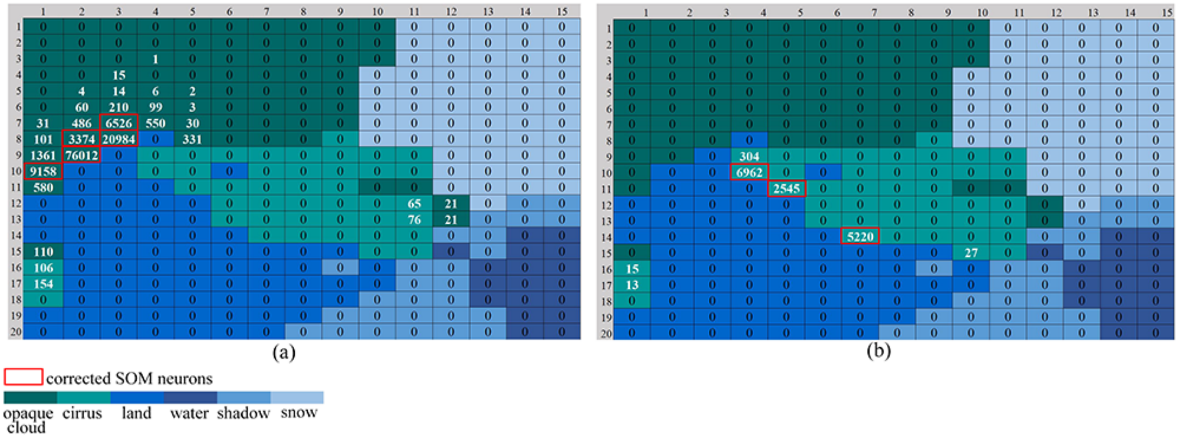
$$\begin{gathered} N_{\text{opaque cloud}} > N_{\text{cirrus}} \quad \& \quad N_{\text{opaque cloud}} > N_{\text{land}} \quad \& \quad N_{\text{opaque cloud}} > N_{\text{water}} \quad \& \\ N_{\text{opaque cloud}} > N_{\text{shadow}} \quad \& \quad N_{\text{opaque cloud}} > N_{\text{snow}} \end{gathered} \tag{7}$$

where N: Number of times a neuron was detected as a BMU for a class.

As a final step, the classes: opaque cloud and cirrus were joined to a class that will be called "cloud" and the classes: land, water, shadow and snow were joined to a class that will be called "clear", to produce the final labeled SOM which contains two classes. The cloud masks of the test set were created by locating the BMU that corresponded to each signature (presented the minimum Euclidean distance) and retrieving the respective label. The process of locating the BMUs for all 1830 × 1830 = 3,348,900 pixels for each Sentinel-2 image (inference time) lasted ∼six min.

2.3.3. Production of Fine-Tuned Cloud Masks

This section presents the third and final stage of the methodology (Figure 3) which leads to the production of the fine-tuned cloud masks. The fine-tuning process aimed at correcting the incorrectly predicted labels of bright non-cloud objects by the trained SOM. The fine-tuning process was applied after the non-fine-tuned cloud masks of the entire test set had been created and observed. For its implementation, after the observation of the non-fine-tuned cloud masks, a sample of misclassified bright pixels (305,228 pixels in total) was selected from four images, and the corresponding BMUs were detected. It is noted that the images that presented high number of misclassified bright non-cloud spectra, were seven in number i.e., ∼20% ($\frac{7}{34}$) of the global test set, which is a high percentage. The output of this process was the detection of the BMUs that represent the bright non-cloud object signatures of the sampled pixels. These BMUs were 18 in number, thus 6% of the total SOM neurons ($\frac{18}{15\times20}$). After the detection of the location of these 18 neurons, their labels were altered from cloud to clear and the temporary fine-tuned cloud masks were produced. It is noted that only the BMUs that corresponded to a number of hits larger than ∼5% of the maximum number of hits for each image, were taken into account. This threshold was derived through a trial and error process which was based on evaluation of the temporary fine-tuned cloud masks. Since these 18 neurons did not exclusively represent the bright non-cloud objects but also a few cloud pixels, a median and a dilation filter of size 3 × 3 were implemented on the temporary fine-tuned cloud masks in order to compensate both for remaining omission and commission errors. Figure 3 depicts the locations of the altered BMUs. As expected they are located in the borders of the classes. In addition, Figure 4 depicts the BMUs and the number of hits for the sampled non-cloud spectra regarding two of the four images that were used in the sampling process. It can be observed that the number of the activated neurons (BMUs) differs, a fact that can be explained by the different spectral variability of the classes. This Figure also illustrates the corrected nodes in red rectangles. The thumbnails of the images where misclassified bright non-cloud pixels occurred are depicted in Figure 1.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
4	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0
5	0	4	14	6	2	0	0	0	0	0	0	0	0	0	0
6	0	60	210	99	3	0	0	0	0	0	0	0	0	0	0
7	31	486	6526	550	30	0	0	0	0	0	0	0	0	0	0
8	101	3374	20984	0	331	0	0	0	0	0	0	0	0	0	0
9	1361	76012	0	0	0	0	0	0	0	0	0	0	0	0	0
10	9158	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	580	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	65	21	0	0	0
13	0	0	0	0	0	0	0	0	0	0	76	21	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	110	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	106	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	154	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	304	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	6962	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	2545	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	5220	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	27	0	0	0	0	0
16	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(b)

Figure 4. Number of hits for the sampled non-cloud spectra. (**a**) Image with high number of activated neurons, (**b**) image with low number of activated neurons.

3. Results

The non-fine-tuned network was at first evaluated (a) by employing several visualization techniques and (b) by calculating the confusion matrix on the training set. Then, evaluation of the fine-tuned (SOM1) and the non-fine-tuned (SOM2) versions was performed on the cloud masks of the test set.

3.1. Visualization Techniques

The trained SOM (SOM1) was evaluated through several visualization techniques which include: (a) the U-matrix [54], (b) the hit rate map, (c) the component planes and (d) 2D scatterplots. These techniques are useful for the interpretation of the behavior of the network.

3.1.1. U-Matrix

The U-matrix, introduced by Ultsch [54], is a very widespread method for visualizing the SOM clusters. It is obtained by calculating the distance (Euclidean in this case) between the neurons that are neighbors. Small distances denoted by low U-matrix values are interpreted as similar data while high values occur in clusters of higher variability or in borders of clusters. The U-matrix produced by the SOM implemented in this study is depicted in Figure 5a. Figure 5b presents the U-matrix with a logarithmic scale with the borders of the classes of the majority voting output overlayed for easier visual interpretation. By observing simultaneously the U-matrix and the clusters of the majority voting output (Figure 5c), it can be concluded that by visual observation the borders between three clusters could probably be deduced (from the yellow colored nodes): (1) snow on the upper right corner, (2) water on the lower right corner and (3) the rest of the classes: land, opaque cloud, cirrus and snow represented by the remaining area. The water class appears to have the smallest variability.

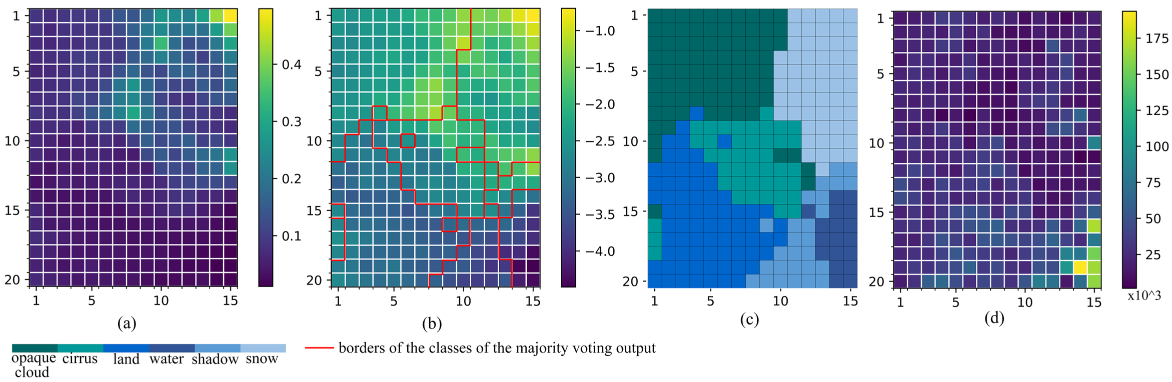


Figure 5. (**a**) U-matrix, (**b**) U-matrix with logarithmic scale, (**c**) majority voting output (**d**) hit rate map.

3.1.2. Hit Rate Map

The hit rate map (Figure 5d) is a way to assess the success of the training process which is considered to be satisfactory when the majority of the cells of the hit rate map depict similar values. Such a scenario indicates that the SOM neurons were uniformly activated. In the case of this study, the neurons seem to have been uniformly activated (as BMUs) by the training data with the exception of the neurons that correspond to the water class (lower right corner). These neurons present higher hit rate because as already observed by the U-matrix, the water spectra show lower variability than the rest of the classes, thus they can be represented by a lower number of neurons.

3.1.3. Component Planes

The component planes depict the coefficients of the feature vectors of the SOM neurons. Each coefficient stands for the spectral value of a Sentinel-2 band, thus their number is equal to the number of Sentinel-2 bands (13). For the purpose of the study, the component planes (Figure 6) were visualized and observed in synergy with the U-matrix and the majority voting output (Figure 5). In Figure 6 they are grouped according to visual similarity. Thus, only one component plane is shown for bands 2–3, 4–8A and 11–12, respectively because based on visual observation they presented similar spectral behavior. Since the component planes are essentially a quantized depiction of the training data with meaningful spatial relations, they are a useful and convenient way to extract the spectral properties of the training data. From their observation it can be seen that the bands that correspond to the blue (B1 (442 nm), B2 (492 nm)) and green (B3 (559 nm)) part of the spectrum, present higher spectral values in the classes: snow, opaque cloud and cirrus. The bands that correspond to the red (B4 (664 nm)–B7 (782 nm)) and NIR (B8 (835 nm), B8A (864.8 nm)) part of the spectrum, present higher spectral values in a small number of neurons representing the snow class. Similar behavior appears in the water absorption band (B9 (945 nm)), with the difference that high values are distributed to more neurons. Concerning the short wave infrared (SWIR) bands, the cirrus band (B10 (1373 nm)) where incident and reflected light are highly absorbed, presents very low values in most neurons. Slightly higher values are presented in many of the neurons that correspond to the cirrus class. In addition, an area that forms the border between the snow and the opaque cloud class presents the highest values, probably because these pixels appear in high altitudes. As for bands 11, 12 (1613, 2202 nm) the neurons that correspond to the snow class present low values, and that is the reason why thresholds in these bands are commonly performed for the separation of snow from clouds.

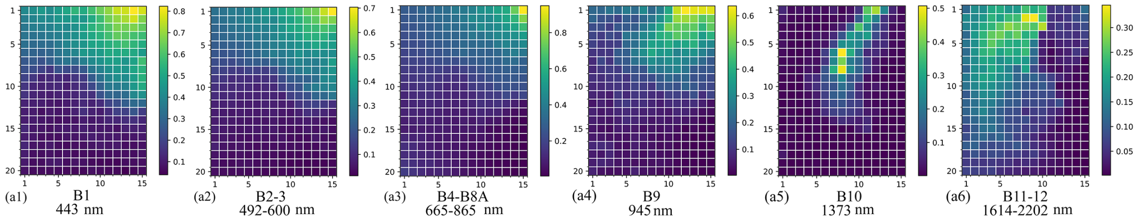


Figure 6. Component planes.

3.1.4. Scatterplots

2D scatterplots are commonly visualized as a straightforward means to evaluate the distribution of the feature vectors of the SOM neurons among the training data. The better the shape of the envelope formed by the neurons simulates the shape of the training data, the greater the accuracy of the SOM training process. For the purpose of the study the 2D scatterplots between several Sentinel-2 bands were observed both for the six classes of the training data in total, and for each class separately. Figure 7 depicts the 2D scatterplots for the six classes between bands 3(559 nm)–8(835 nm) and 3–11(1613 nm). Figure 8 depicts respective 2D scatterplots for each class separately. As it can be seen in Figure 7a the spectral values of the SOM nodes for bands 3 and 8 simulate very well the distribution of the training data. As for bands 3 and 11 (Figure 7b), the distribution of the nodes is less dispersed and thus fails to simulate the peripheral areas of the opaque cloud and snow classes. It also does not reach some extreme sparse data of the land class.

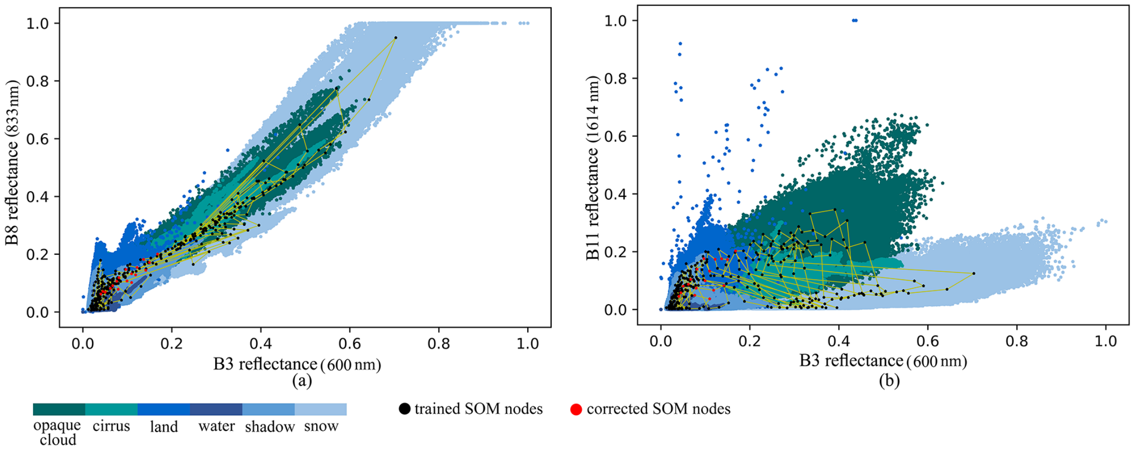


Figure 7. Scatterplots of the six classes of the training set.

Similar conclusions are derived by observing Figure 8 which provides a clearer image for the interpretation of the distribution of the SOM nodes. As shown in Figure 8(a1), it is clear that the opaque cloud class is well represented as far as the spectral properties of the green (B3) and the NIR (B8) bands are concerned. However, in Figure 8(b1), as already mentioned, there is a spectral area at the top right that is not fully covered. Similar behavior is also observed for the snow class (Figure 8(a6, b6)). Regarding the land class (Figure 8(a3, b3)), even though the SOM neurons are distributed among the majority of the training data, their dispersion does not reach a portion on the top left. The cirrus (Figure 8(a2, b2)), shadow (Figure 8(a5, b5)) and water (Figure 8(a4, b4)) classes appear to be very well delineated by the scattering of the SOM neurons. It is noted that resembling behavior was shown for the other visible, NIR and SWIR Sentinel-2 bands.

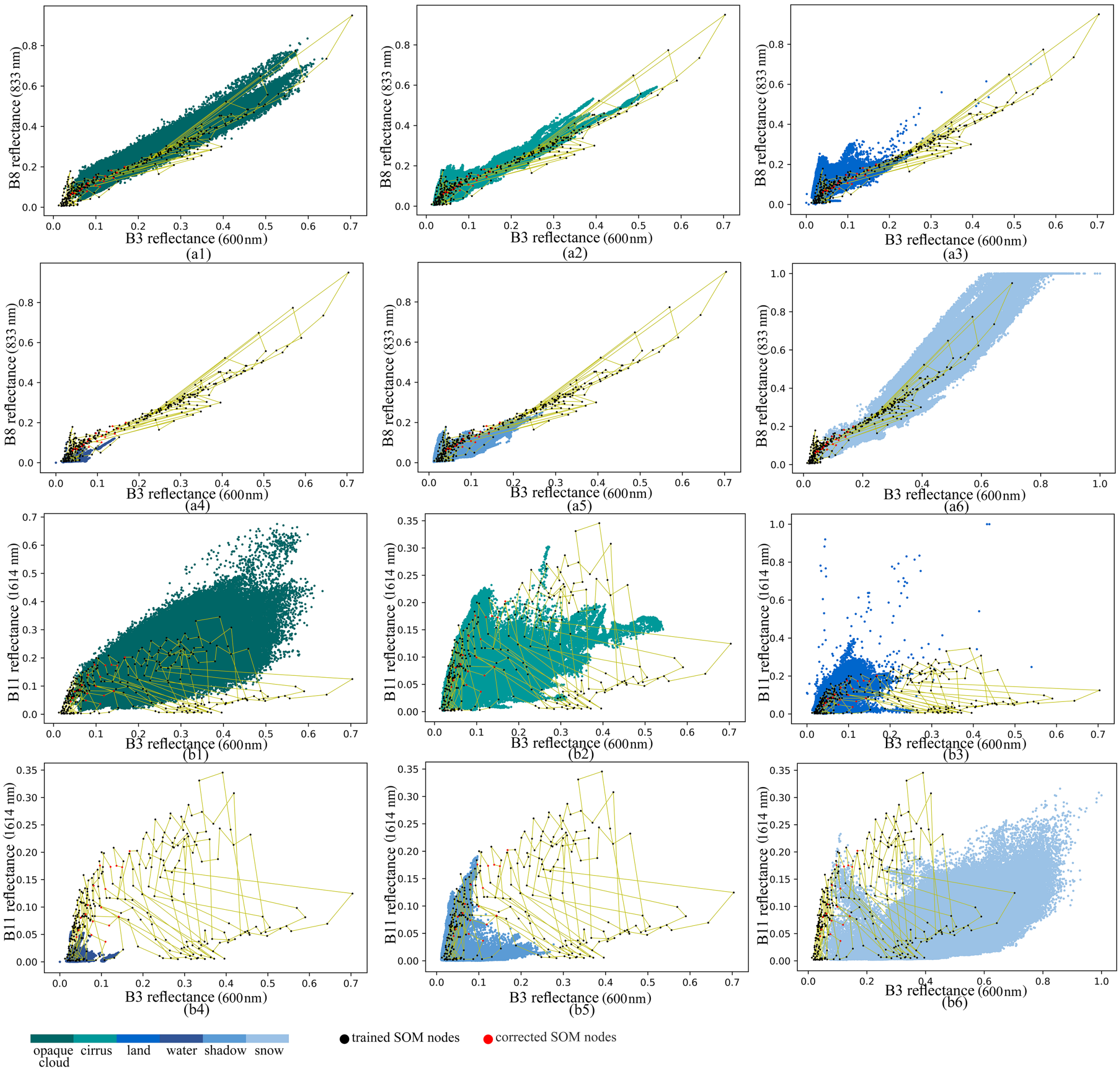


Figure 8. Separate scatterplots of the six classes of the training set.

3.2. Training Set

The confusion matrix was created (Table 3) in order to evaluate the performance of the trained network (SOM1) on the training spectra. It was created by feeding into the non fine-tuned trained network all the training spectral signatures and predicting their class. Producer's accuracy (recall) (Equation (8)) corresponds to omission error (100%—omission error) and is calculated by a ratio where the nominator is the number of correctly classified signatures for a class, and the denominator is the nominator plus the number of the class signatures that were omitted. The user's accuracy (precision)

(Equation (9)) corresponds to commission error (100%—commission error) and is calculated by a ratio where the nominator is the number of correctly classified signatures of a class and the denominator is the nominator plus the number of misclassified signatures to this class.

$$\text{Recall} = \frac{\text{TP}}{\text{TP + FN}} \tag{8}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP + FP}} \tag{9}$$

$$\text{Accuracy} = \frac{\text{TP + TN}}{\text{TP + FN + FP + TN}} \tag{10}$$

where TP: True positive, TN: True negative, FP: False positive, FN: False negative.

Table 3. Confusion matrix of trained SOM.

	Other	**Cloud**	**Snow**	**Producer's Accuracy**
Other	4,742,750	74,068	5856	0.983
Cloud	222,145	2,463,208	20,828	0.910
Snow	1048	8847	1,261,248	0.992
User's accuracy	0.955	0.967	0.979	

The confusion matrix presented an overall accuracy (Equation (10)) of ∼96%. By observing the values of the table it can be observed that the snow class presents low omission and commission errors (<1%, ∼2%). As for the "other" class that includes the classes land, shadow and water, the commission error is ∼4% and the omission error is ∼2%. Higher omission error is presented for the cloud class (∼9%), while the commission error is close to the formerly mentioned classes.

3.3. Cloud Masks of the Entire Test Set

Several evaluation metrics were calculated for the quantitative evaluation of the cloud masks produced by the trained SOM before and after applying the fine-tuning process. The evaluation metrics that were computed were accuracy (Equation (10)), recall (Equation (8)), precision (Equation (9)) and fscore (Equation (11)) which combines recall and precision metrics.

$$\text{Fscore} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision + recall}} \tag{11}$$

where TP, TN, FP, FN: as defined in Equations (8)–(10).

These metrics were also computed for two state-of-the-art algorithms: Sen2Cor and Fmask. Their average values are presented in Table 4. By observing the table values it can be deduced that the two SOM versions as well as Sen2Cor and Fmask perform similarly with differences often less than 1%. The average values of the evaluation metrics are: accuracy: ∼93%, recall: ∼92%, precision: ∼98% and fscore: ∼95%.

The similar behavior is also shown in the plots of Figure 9 where the evaluation metrics for each of the images of the test set are presented, as well as in the box plots depicted in Figure 10. A box plot is a diagram that illustrates the variance of the data. It consists of two boxes. The lower side of the lower box corresponds to the first quartile and the upper side to the second quartile. The vertical lines crossing the boxes denote the distance of the maximum or minimum value in comparison to the second quartile. The box plots of this study indicate slightly greater variance of recall values for SOM1 and SOM2 with lower values of the first quartile. In addition, these algorithms present slightly higher precision values and smaller distance of the minimum value from the second quartile. It is noted that Sen2Cor shows the highest average recall values (lowest omission error) but also the lowest average precision values (highest commission error).

Table 4. Evaluation metrics of Sentinel-2 cloud masks (entire test set).

Method	Accuracy	Recall	Precision	Fscore
Sen2Cor	0.920	0.928	0.969	0.943
Fmask	0.922	0.917	0.984	0.945
SOM1	0.928	0.919	0.988	0.949
SOM2	0.928	0.919	0.986	0.949

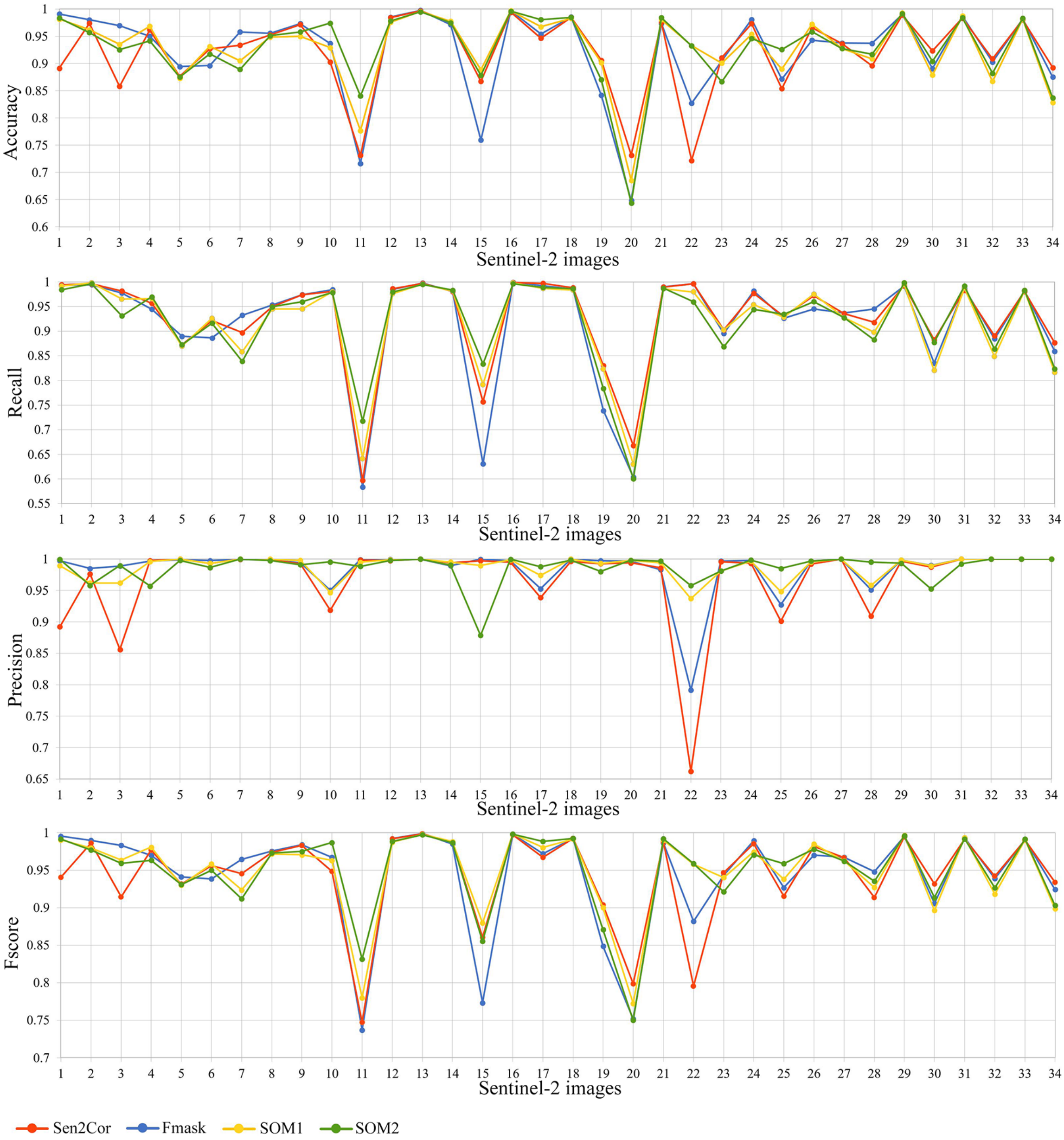


Figure 9. Evaluation metrics of the entire test set.

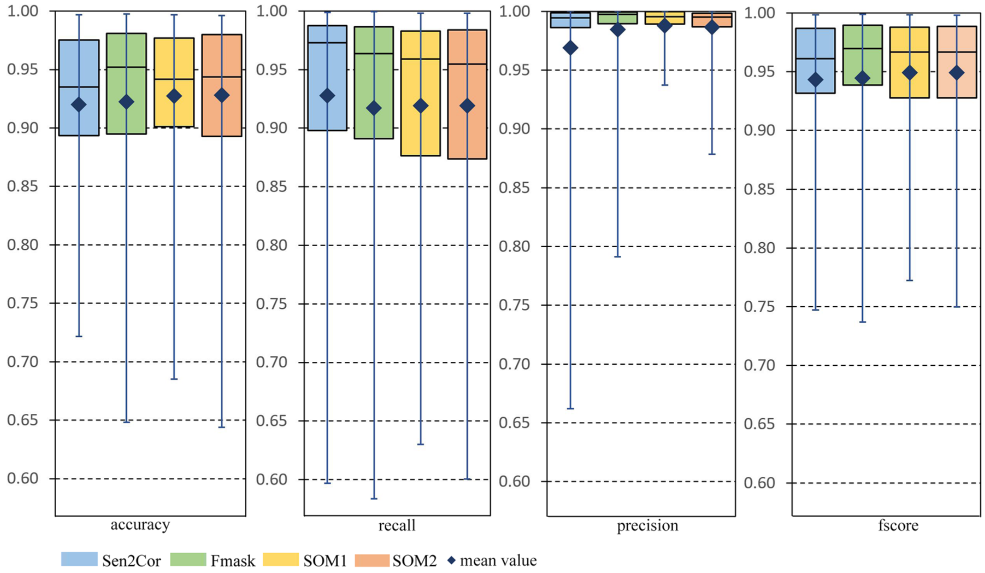


Figure 10. Box plots of the evaluation metrics of the entire test set.

3.4. Cloud Masks of Fine-Tuned Cases with Bright Non-Cloud Objects

3.4.1. Images Used in the Fine-Tuning Process

Figure 11 presents the cloud masks produced by Sen2Cor and Fmask for the four images with bright non-cloud objects which were used for the selection of the sample of incorrectly classified bright non-cloud pixels during the fine-tuning process (Section 2.3.3). Respective results for SOM1 and SOM2 are presented in Figure 12. These figures show the RGB natural composite where the ground truth categories were delineated by Baetens et al. [50], as well as correctly predicted pixels (for cloud (TP) and clear (TN) categories) along with omission (FN) and commission error (FP). The latter figure also shows the neurons that were altered in terms of their label during the fine-tuning process. These neurons as already mentioned corresponded to the signatures of the bright non-cloud objects that were incorrectly classified by SOM1. Based on the four images depicted in Figure 11(a1–a4), the labels of 18 neurons in total were altered from cloud to clear. The evaluation metrics for these images are presented in Tables 5 and 6.

For Figure 11(a1), the cloud masks produced by Sen2Cor (Figure 11(b1)) and Fmask ((Figure 11(c1)) incorrectly classified two large bright areas of land (commission error: ∼8% and ∼5%) with Sen2Cor performing worse. The cloud mask produced by SOM1 (Figure 12(a1)) was similar to the Fmask output but it can also be seen that two small snow areas were incorrectly detected. The cloud mask produced by SOM2 ((Figure 12(c1)) performed significantly better by correctly classifying the majority of the bright land pixels (commission error: <1%). Likewise, for Figure 11(a2), the cloud masks produced by Sen2Cor (Figure 11(b2)) and Fmask ((Figure 11(c2)) misclassified a number of bright land and snow pixels (commission error: ∼10% and ∼7%). SOM1 (Figure 12(a2)) showed lower commission error (∼5%) and SOM2 performed better than the previous methods ((Figure 12(c2)) (commission error: <2%). Regarding the SOM2 result, it should be observed that the misclassification of snow pixels is slightly lower than SOM1. This is due to the fact that a small percentage of snow pixels was selected along with the surrounding soil bright pixels during the fine-tuning process. As already clarified in the Introduction, experimental attempts were also made for the alteration of the labels of more neurons that corresponded specifically to snow pixels. However, the results showed that a large omission error for the cloud class occurred and thus the results were considered unsatisfactory. Taking into account the fact that the training set produced <1% omission error for the snow class, it is safe to assume that the test set includes various snow spectra (e.g., wet, dry) of different thickness that do not appear in the training set.

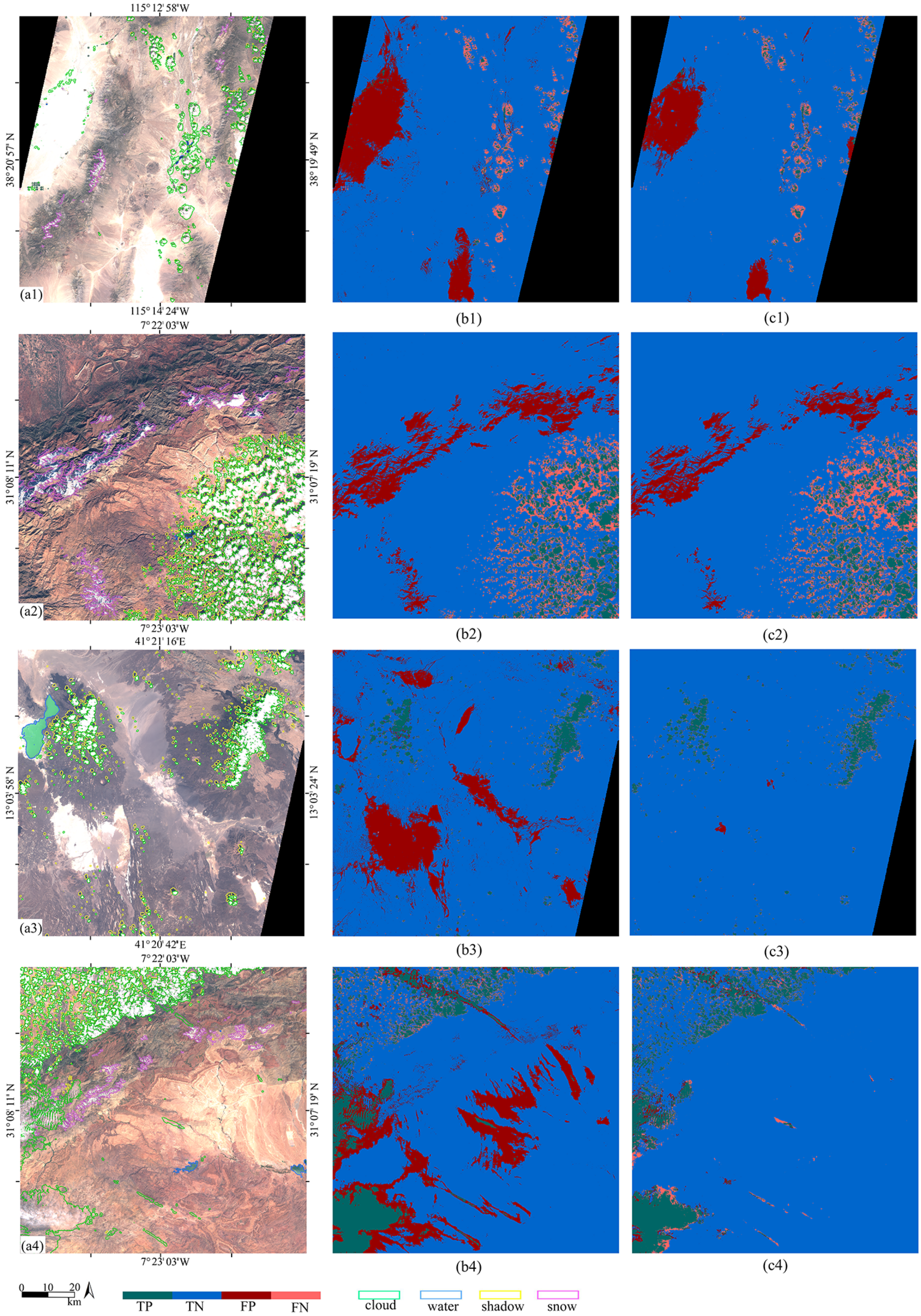


Figure 11. Cloud masks of Sentinel-2 images with bright non-cloud objects. (**a1**–**a4**): RGB composites with delineation of categories, (**b1**–**b4**): Sen2Cor cloud masks, (**c1**–**c4**): Fmask cloud masks.

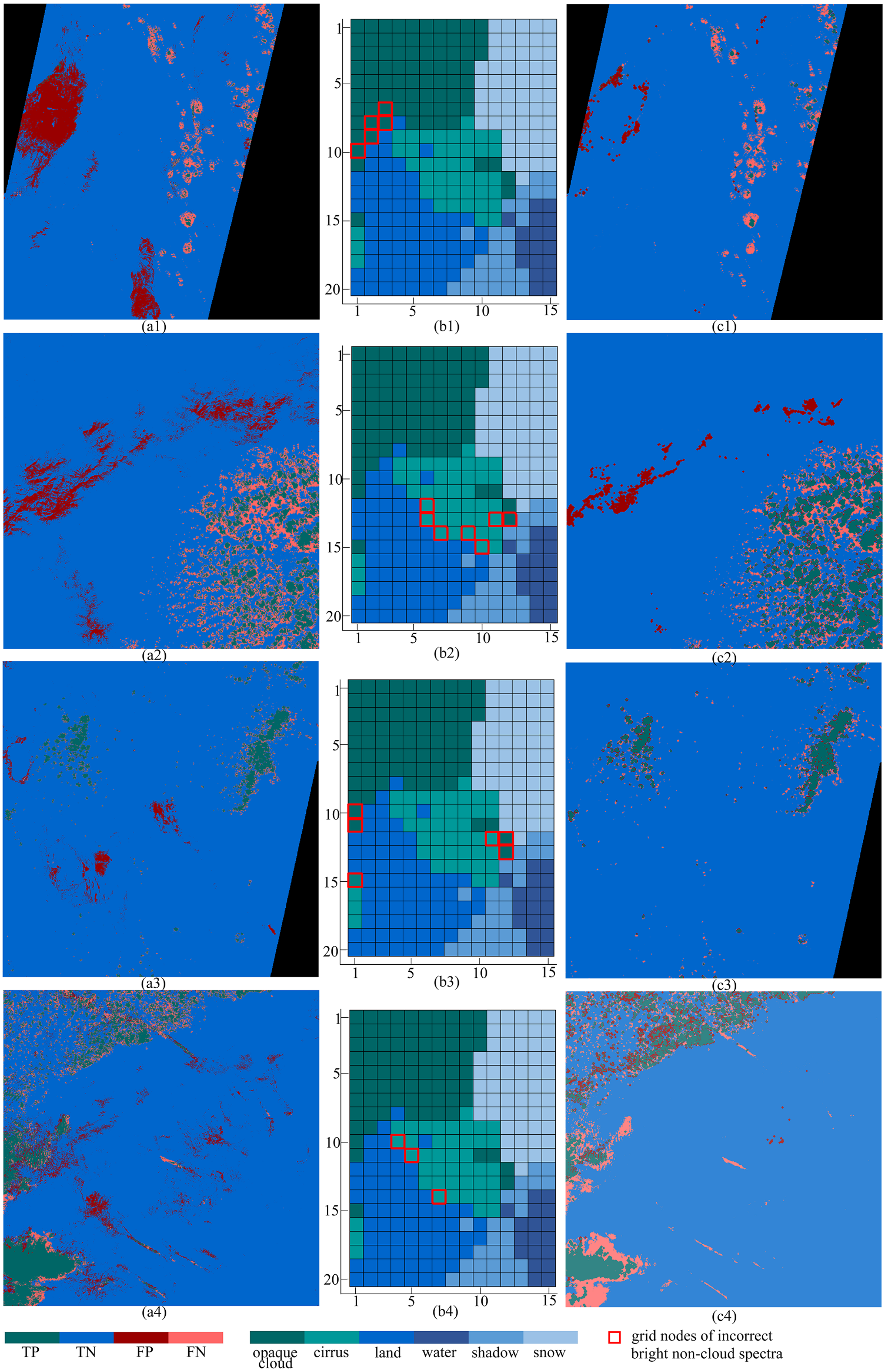


Figure 12. (**a1**–**a4**): SOM1 cloud masks, (**b1**–**b4**): neurons with altered labels, (**c1**–**c4**): SOM2 cloud masks.

Table 5. Evaluation metrics of images with bright non-cloud objects (accuracy, recall).

	Accuracy				Recall			
	Sen2Cor	Fmask	SOM1	SOM2	Sen2Cor	Fmask	SOM1	SOM2
Figure 11(a1)	0.903	0.937	0.929	0.974	0.980	0.984	0.980	0.979
Figure 11(a2)	0.854	0.871	0.890	0.926	0.930	0.926	0.928	0.934
Figure 11(a3)	0.891	0.991	0.982	0.984	0.994	0.993	0.992	0.984
Figure 11(a4)	0.858	0.970	0.935	0.926	0.982	0.978	0.965	0.931
Figure 13(a1)	0.947	0.954	0.967	0.981	0.997	0.992	0.987	0.989
Figure 13(a2)	0.896	0.937	0.909	0.917	0.918	0.945	0.898	0.882
Figure 13(a3)	0.976	0.972	0.977	0.975	0.981	0.981	0.981	0.983
mean	0.903	0.947	0.941	0.954	0.969	0.971	0.961	0.954

Table 6. Evaluation metrics of images with bright non-cloud objects (precision, fscore).

	Precision				Fscore			
	Sen2Cor	Fmask	SOM1	SOM2	Sen2Cor	Fmask	SOM1	SOM2
Figure 11(a1)	0.919	0.950	0.947	0.995	0.949	0.967	0.963	0.987
Figure 11(a2)	0.901	0.927	0.948	0.985	0.916	0.927	0.938	0.959
Figure 11(a3)	0.892	0.997	0.989	0.999	0.941	0.995	0.990	0.992
Figure 11(a4)	0.856	0.989	0.962	0.990	0.915	0.983	0.963	0.959
Figure 13(a1)	0.939	0.953	0.974	0.988	0.967	0.972	0.980	0.988
Figure 13(a2)	0.909	0.951	0.958	0.995	0.914	0.948	0.927	0.935
Figure 13(a3)	0.994	0.989	0.995	0.991	0.987	0.985	0.988	0.987
mean	0.918	0.965	0.968	0.992	0.941	0.968	0.964	0.972

For Figure 11(a3), Sen2Cor (Figure 11(b3)) showed a high commission error (∼11%) by incorrectly classifying several bright non-cloud areas. However, Fmask (Figure 11(c3)) presented satisfactory results as it only misclassified two small bright land areas. Slightly lower performance was depicted by SOM1 (Figure 12(a3)) which failed to correctly detect a few more bright non-cloud pixels. The best cloud mask was produced by SOM2 ((Figure 12(c3)) which managed to successfully predict the class of the bright non-cloud surfaces. Finally, for Figure 11(a4), Fmask (Figure 11(d3)) and SOM2 (Figure 12(c4) presented the most successful results, followed by SOM1 (Figure 12(a4) (commission error: ∼4%) and Sen2Cor which produced the worst cloud mask (commission error (∼14%)).

It is noted that as expected by the SOM theory, the neurons that were altered were located at the borders of the cloud class (opaque cloud + cirrus) with either the land or the snow class.

3.4.2. Images Not Used in the Fine-Tuning Process

Figure 13 shows the cloud masks produced by the four methods for the three of the seven images with bright non-cloud objects which were not used for the selection of the sample of incorrectly classified bright non-cloud pixels during the fine-tuning process. It was observed that results are similar to those of the images presented in Section 3.4.1.

For Figure 13(a1), Sen2Cor (Figure 13(b1)) presents the less satisfactory results by incorrectly classifying two bright land areas (commission error: ∼6%). Fmask (Figure 13(c1)) performs slightly better (commission error: ∼5%) and SOM1 (Figure 13(d1)) appears to be more successful since it misclassifies a lower percentage of pixels (commission error: ∼3%). The SOM2 (Figure 13(e1) cloud mask shows the most satisfactory results (commission error: ∼1%). Likewise, for Figure 13(a2), SOM2 (Figure 13(e2)) illustrates the best performance (commission error: <1%) followed by SOM1 (Figure 13(d2)) (commission error: ∼4%), Fmask (Figure 13(c2) (commission error: ∼5%) and Sen2Cor (Figure 13(b2)) (commission error: ∼9%). Finally, for Figure 13(a3), Sen2Cor (Figure 13(b3)) and Fmask (Figure 13(c3)) fail to correctly detect the bright urban elements like buildings and streets as shown in the zoomed in areas delineated by black circles for easier perception. SOM1 and SOM2 cloud masks (Figures 13(d3,e3)) do not present such an issue.

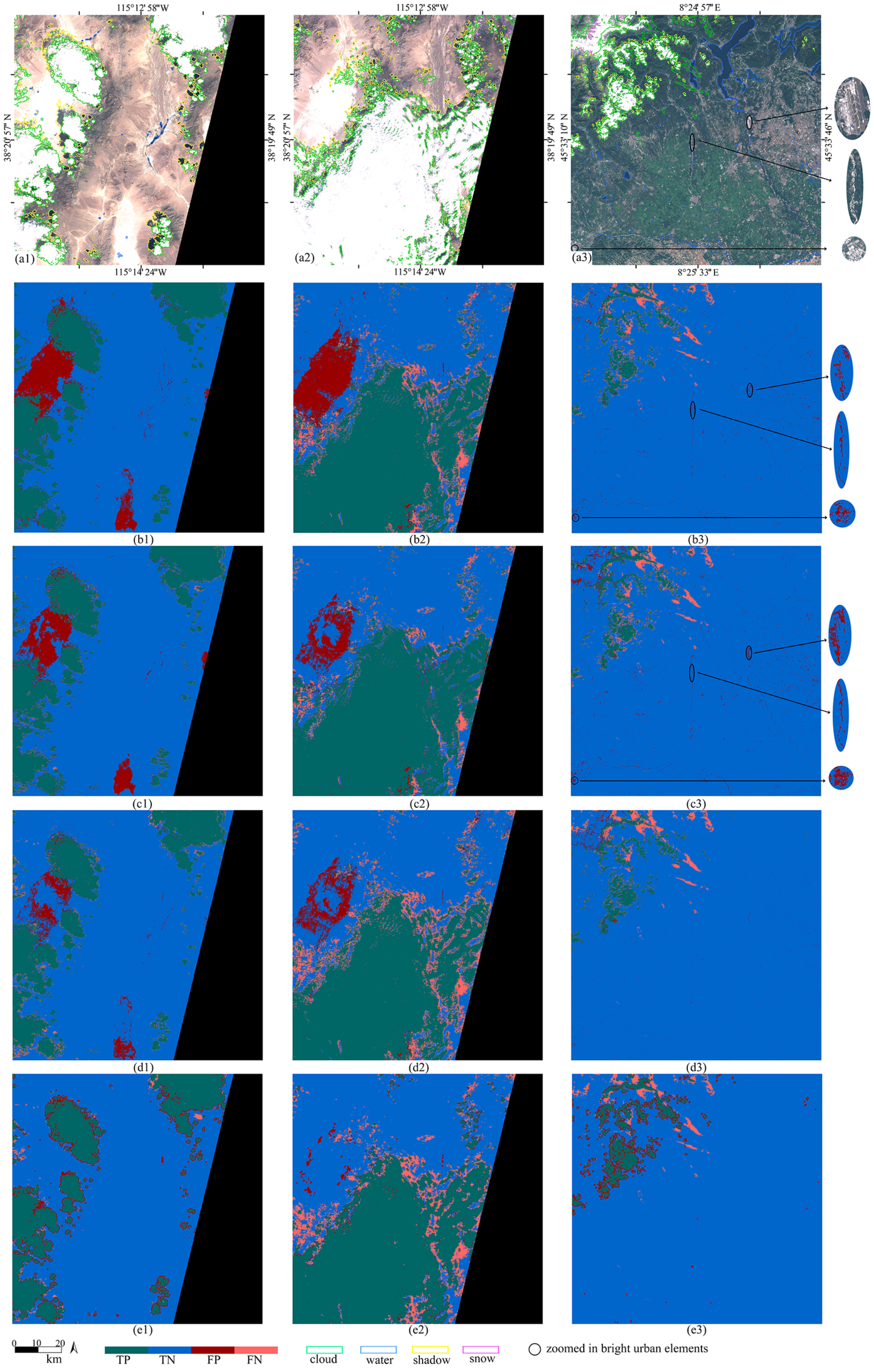


Figure 13. Cloud masks of Sentinel-2 images with bright non-cloud objects. (**a1**–**a3**): RGB composites with delineation of categories, (**b1**–**b3**): Sen2Cor cloud masks, (**c1**–**c3**): Fmask cloud masks, (**d1**–**d3**): SOM1 cloud masks, (**e1**–**e3**): SOM2 cloud masks.

Plots created by the values of the four evaluation metrics which are shown in Figure 14 are useful for the evaluation. By observing the line formed by the precision values, it is easily deduced that SOM2 produced by fine-tuning SOM1 network, outperformed the other algorithms. This conclusion is also reached by observing the box plots of Figure 15. It is also noted that the SOM2 results produce only slight decrease to recall values compared to SOM1.

Concerning the comparison of the time needed to produce a cloud mask for a Sentinel-2 image, as already mentioned in Section 2.3.2 the SOM proposed in this study needs ∼six minutes, while the newest versions of Fmask and Sen2cor which are much faster than the previous ones, when run from command line need ∼four and ∼two min, respectively. Nevertheless, even though Fmask and Sen2Cor run faster than the proposed SOM, they fail to distinguish the bright non-cloud objects of the test set, thus their time-efficiency becomes irrelevant in this case.

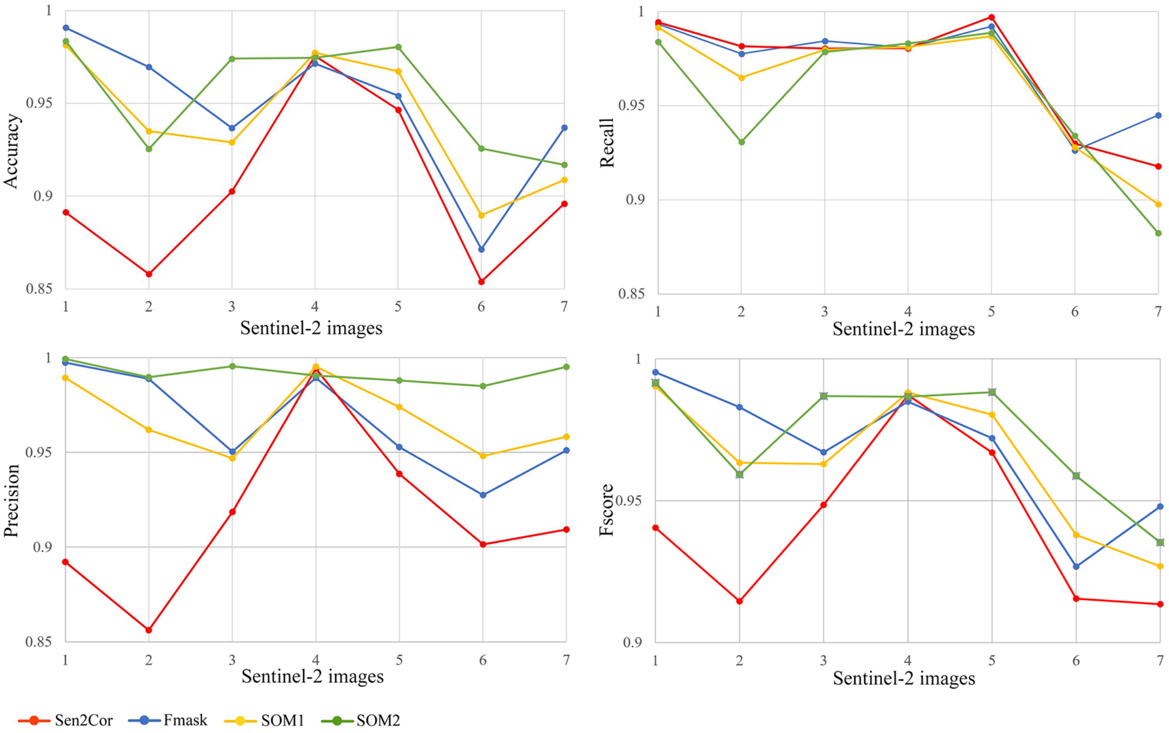


Figure 14. Evaluation metrics of the Sentinel-2 images with bright non-cloud objects.

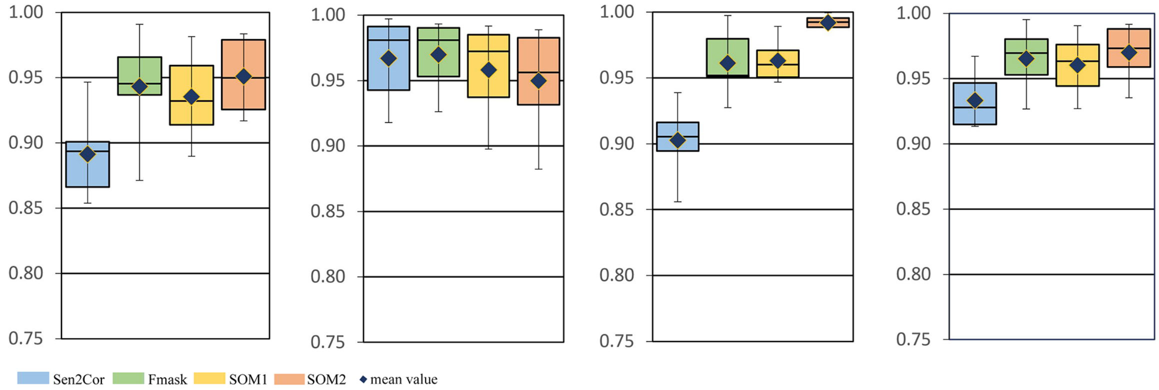


Figure 15. Box plots of the Sentinel-2 images with bright non-cloud objects.

4. Discussion

This section highlights the main points that distinguish the proposed method from: (a) other types of neural networks (MLPs, CNNs), (b) the most commonly applied state-of-the-art algorithms (Sen2Cor, Fmask) and (c) the most widely used unsupervised classification method (k-means). The discussion also includes comments on the potential risk of the proposed fine-tuning approach.

To begin with, time-efficiency is one of their main benefits of SOMs compared to other types of artificial neural networks such as MLPs and CNNs which are widely and most frequently applied in current research and industrial applications. MLPs and CNNs require multiple hours for training while SOMs usually need only a few minutes. As a matter of fact, the SOM proposed in this study was trained in two minutes and required 18 min to acquire its labeling in order to produce cloud masks. Concerning fine-tuning for MLPs and CNNs, the main difference of the fine-tuning stage from the initial training stage is that during fine-tuning a smaller training set and fewer hidden layers are used, but the process is still slower compared to the proposed fine-tuning method (SOM2). In addition, the proposed fine-tuning method required only a small labeled dataset in order to detect the BMUs of the bright non-cloud spectra and then alter their labels.

Another advantage of SOMs is that their behavior is much more interpretative compared MLPs/CNNs where the performance is mainly evaluated through the accuracy of the predictions on the test set. In contrast, in SOMs, the network can also be evaluated by useful visualization techniques that analyze the similarity/dissimilarity of the neighboring nodes, the uniformity of the activation of the neurons, the fast extraction of spectral properties from quantized data and the distribution of the SOM nodes among the data.

As far as comparison with Fmask and Sen2Cor is concerned, the proposed fine-tuning method outperformed them by far in the separation of bright non-cloud objects from clouds. However, the newest versions of Fmask and Sen2cor produce a Sentinel-2 cloud mask faster i.e., ∼four minutes are required for Fmask and ∼two minutes for Sen2Cor against ∼six minutes for the proposed SOM.

Regarding the similarities of SOMs with the k-means, it can be indeed stated that the two methods are very similar with the main difference being that in SOMs the centers (neurons) interact with each other and create neighborhoods that carry topologic information. The centers in k-means do not interact with each other. It is noted that when the SOM grid consists of a small number of neurons, it is equivalent to k-means because the neighborhood concept is invalid for very few neurons. In practice, concerning our study, the main difference of the two methods lies in the fine-tuning process which would not be practically possible to be applied in the k-means algorithm. The reason is that it would require the number of clusters to be the same as the number of the SOM nodes, and thus the training process would be very slow. The common practice of fine-tuning the k-means algorithm is to run the method with an increasing number of classes until satisfying results are produced. This approach is still time-consuming and is even more cumbersome when the training set is different from the test set. In general, training a k-means is much slower than training a SOM, because in k-means every time the centers are updated, the distance between the new centers and all the data points needs to be calculated. In contrast, for the SOM training, data points are fed into the network one by one and every time a data point is fed into the network, only the distance of this data point with the nodes of the SOM grid needs to be calculated.

A final point to be discussed is the potential risk of "altering the incorrect labeling" in our proposed fine-tuning approach. The effect of altering the incorrect labeling in the case study analyzed in our paper is that the altered nodes (as explained in Section 2.3.3) are not only the BMUs of the bright non-cloud spectra, but also of a few cloud pixels. In practice, that means that when we produce a cloud mask by using the fine-tuned SOM version (SOM2) it is probable that there are a few cloud pixels in the image that correspond to the altered neurons, and thus they will be misclassified to the clear class. In this paper, we alleviated this issue by running a median and a dilation filter in order to retrieve the cloud pixels that SOM2 had misclassified. Thus, we have proven, that concerning the 34 images of the test set, the effect of altering the incorrect labeling can be overcome. Since these images were captured around the globe in different seasons and times of the day, and represent a large variety of land cover, we believe that SOM2 would have a similar performance in images that were not included in our test dataset. Our opinion is reinforced by the fact that the proposed fine-tuned method alters the neurons of the borders of the opaque cloud and cirrus class with the land class and not the labels of the neurons that are situated in the center of the classes in the SOM grid.

5. Conclusions

This study evaluated a SOM for cloud masking Sentinel-2 images and proposed a fine-tuning methodology based on the output of the non-fine-tuned network. The fine-tuning process managed to correct the misclassified predictions of bright non-cloud spectra without applying further training. The proposed fine-tuning method is the most important contribution of the study since it follows a general procedure, thus its applicability is broad and not confined only in the field of cloud-masking. It was performed by directly locating the neurons that corresponded to the incorrectly predicted bright non-cloud objects and altering their labels. This process was chosen over the common practice of further training the network by feeding the label data (supervised training) since it was considered faster, simpler and more efficient. Further training would probably also require more data than those available. A median and a dilation filter was performed as the final step of fine-tuning to compensate both for remaining omission and commission errors caused by the fact that the altered neurons represented also a percentage of cloud pixels.

The SOM was trained on approximately nine million spectral signatures extracted from the largest publicly available database of Sentinel-2 signatures for cloud masking applications. After the completion of the training, the non-fine-tuned network was at first evaluated (a) by employing several visualization techniques which illustrated essential spectral properties of the nodes based on topologic information and lead to the interpretation of its behavior and (b) by calculating the confusion matrix on the training set where it produced an overall accuracy ∼96%. Then, evaluation of the non-fine-tuned (SOM1) and the non-fine-tuned (SOM2) versions was performed on a truly independent test set of 34 Sentinel-2 ground truth cloud masks provided by the only publicly available source. By evaluating this entire test set through several evaluation metrics and plots, and comparing them with two state-of-the-art algorithms (Sen2Cor, Fmask), it was deduced that both the two SOM versions and the two state-of-the-art algorithms produced similar results (accuracy: ∼93%, recall: ∼92%, precision: ∼98% and fscore: ∼95%). However, when performing quantitative and qualitative evaluation process for the cases with bright non-cloud objects, it was shown that the fine-tuned version performed more successfully with average commission error less than 1%. The respective values for the SOM before fine-tuning were ∼3%, for Fmask ∼4% and for Sen2Cor ∼8%. The fine-tuning method proposed in this study was also applied in experiments that specifically targeted incorrectly classified snow pixels. However, the results were considered unsatisfactory because a large omission error of clouds was produced. Thus, these experiments were not presented in this study.

As a general conclusion, the study showed that the proposed method for fine-tuning SOMs is very effective for separating bright non-cloud objects from clouds, while the commonly used state-of-the-art algorithms failed in this task. In addition, the method is simple in its implementation and time-efficient, since it only involves the detection of the BMUs of interest and it requires very few data points as input. Thus, in our future work we intend to investigate the potential of the method in different scenarios, especially for big data analysis where processing time is crucial. We will also consider testing the method in datasets with greater availability of ground-truth data.

Author Contributions: Conceptualization, all authors; methodology, all authors; software, V.K. (Viktoria Kristollari); validation, V.K. (Viktoria Kristollari); formal analysis, V.K. (Viktoria Kristollari); investigation, V.K. (Viktoria Kristollari); resources, V.K. (Vassilia Karathanassi); data curation, V.K. (Viktoria Kristollari); writing—original draft, all authors; writing—review and editing, all authors; visualization, V.K. (Viktoria Kristollari); supervision, V.K. (Vassilia Karathanassi); project administration, V.K. (Vassilia Karathanassi);

funding acquisition, V.K. (Vassilia Karathanassi). All authors have read and agreed to the published version of the manuscript.

Funding: This work is part of the HYPERION project. HYPERION has received funding from the European Union's Framework Programme for Research and Innovation (Horizon 2020) under grant agreement no. 821054. The content of this publication is the sole responsibility of NTUA (Work Package 6, Task 6.3) and does not necessarily reflect the opinion of the European Union. For all figures Permission have been obtained from the owners.

Acknowledgments: The authors would like to express their appreciation to the anonymous reviewers for their constructive comments and recommendations.

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

References

1. Wilson, M.J.; Oreopoulos, L. Enhancing a simple MODIS cloud mask algorithm for the landsat data continuity mission. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 723–731. [CrossRef]
2. Zhuge, X.Y.; Zou, X.; Wang, Y. A Fast Cloud Detection Algorithm Applicable to Monitoring and Nowcasting of Daytime Cloud Systems. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 6111–6119. [CrossRef]
3. Jedlovec, G.; Haines, S.; LaFontaine, F. Spatial and Temporal Varying Thresholds for Cloud Detection in GOES Imagery. *IEEE Trans. Geosci. Remote Sens.* **2008**, *46*, 1705–1717. [CrossRef]
4. Platnick, S.; King, M.D.; Ackerman, S.A.; Menzel, W.P.; Baum, B.A.; Riédi, J.C.; Frey, R.A. The MODIS cloud products: Algorithms and examples from Terra. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 459–473. [CrossRef]
5. Zhu, Z.; Woodcock, C.E. Object-based cloud and cloud shadow detection in Landsat imagery. *Remote Sens. Environ.* **2012**, *118*, 83–94. [CrossRef]
6. Zhu, Z.; Wang, S.; Woodcock, C.E. 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.* **2015**, *159*, 269–277. [CrossRef]
7. Irish, R.R. Landsat 7 automatic cloud cover assessment. In Proceedings of the SPIE Algorithms for Multispectral, Hyperspectral, and Ultraspectral Imagery VI, Orlando, FL, USA, 24 April 2000; Volume 4049, pp. 348–355. [CrossRef]
8. Richter, R.; Louis, J.; Müller-Wilm, U. *Sentinel-2 MSI—Level 2A Products Algorithm Theoretical Basis Document*; (Special Publication); European Space Agency, ESA SP: Paris, France, 2012.
9. Hagolle, O.; Huc, M.; Pascual, D.V.; Dedieu, G. A multi-temporal method for cloud detection, applied to FORMOSAT-2, VENMS, LANDSAT and SENTINEL-2 images. *Remote Sens. Environ.* **2010**, *114*, 1747–1755. [CrossRef]
10. Karvonen, J. Cloud masking of MODIS imagery based on multitemporal image analysis. *Int. J. Remote Sens.* **2014**, *35*, 8008–8024. [CrossRef]
11. Mateo-García, G.; Gómez-Chova, L.; Amorós-López, J.; Muñoz-Marí, J.; Camps-Valls, G. Multitemporal Cloud Masking in the Google Earth Engine. *Remote Sens.* **2018**, *10*, 1079. [CrossRef]
12. Hagolle, O.; Huc, M.; Desjardins, C.; Auer, S.; Richter, R. *MAJA ATBD Algorithm Theoretical Basis Document*; Technical Report; CNES+CESBIO and DLR: Paris, France; Toulouse, France; Köln, Germany, 2017.
13. Hughes, M.; Hayes, D. Automated detection of cloud and cloud shadow in single-date Landsat imagery using neural networks and spatial post-processing. *Remote Sens.* **2014**, *6*, 4907–4926. [CrossRef]
14. Taravat, A.; Peronaci, S.; Sist, M.; Del Frate, F.; Oppelt, N. The combination of band ratioing techniques and neural networks algorithms for MSG SEVIRI and Landsat ETM+ cloud masking. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; pp. 2315–2318. [CrossRef]
15. Mateo-García, G.; Gómez-Chova, L.; Camps-Valls, G. Convolutional neural networks for multispectral image cloud masking. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 2255–2258. [CrossRef]
16. Le Goff, M.; Tourneret, J.Y.; Wendt, H.; Ortner, M.; Spigai, M. Deep learning for cloud detection. In Proceedings of the 8th International Conference of Pattern Recognition Systems (ICPRS 2017), Madrid, Spain, 11–13 July 2017. [CrossRef]

17. Liu, H.; Zeng, D.; Tian, Q. Super-pixel cloud detection using Hierarchical Fusion CNN. In Proceedings of the 2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM), Xi'an, China, 13–16 September 2018; pp. 1–6. [CrossRef]
18. Segal-Rozenhaimer, M.; Li, A.; Das, K.; Chirayath, V. Cloud detection algorithm for multi-modal satellite imagery using convolutional neural-networks (CNN). *Remote Sens. Environ.* **2020**, *237*, 111446. [CrossRef]
19. Mateo-García, G.; Gómez-Chova, L. Convolutional Neural Networks for Cloud Screening: Transfer Learning from Landsat-8 to Proba-V. In Proceedings of the IGARSS 2018—2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 2103–2106. [CrossRef]
20. Jeppesen, J.H.; Jacobsen, R.H.; Inceoglu, F.; Toftegaard, T.S. A cloud detection algorithm for satellite imagery based on deep learning. *Remote Sens. Environ.* **2019**, *229*, 247–259. [CrossRef]
21. Mohajerani, S.; Krammer, T.A.; Saeedi, P. A Cloud Detection Algorithm for Remote Sensing Images Using Fully Convolutional Neural Networks. In Proceedings of the 2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP), Vancouver, BC, Canada, 29–31 August 2018; pp. 1–5. [CrossRef]
22. Chai, D.; Newsam, S.; Zhang, H.K.; Qiu, Y.; Huang, J. Cloud and cloud shadow detection in Landsat imagery based on deep convolutional neural networks. *Remote Sens. Environ.* **2019**, *225*, 307–316. [CrossRef]
23. Wieland, M.; Li, Y.; Martinis, S. Multi-sensor cloud and cloud shadow segmentation with a convolutional neural network. *Remote Sens. Environ.* **2019**, *230*, 111203. [CrossRef]
24. Yang, J.; Guo, J.; Yue, H.; Liu, Z.; Hu, H.; Li, K. CDnet: CNN-Based Cloud Detection for Remote Sensing Imagery. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 6195–6211. [CrossRef]
25. Sholar, J.M. *Lightweight Deconvolutional Neural Networks for Efficient Cloud Identification in Satellite Images*; Stanford University: Stanford, CA, USA, 2017.
26. Li, Z.; Shen, H.; Wei, Y.; Cheng, Q.; Yuan, Q. Cloud detection by fusing multi-scale convolutional features. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2018**, *IV-3*, 149–152. [CrossRef]
27. Lu, J.; Wang, Y.; Zhu, Y.; Ji, X.; Xing, T.; Li, W.; Zomaya, A.Y. P_Segnet and NP_Segnet: New Neural Network Architectures for Cloud Recognition of Remote Sensing Images. *IEEE Access* **2019**, *7*, 87323–87333. [CrossRef]
28. Yuan, K.; Meng, G.; Cheng, D.; Bai, J.; Xiang, S.; Pan, C. Efficient cloud detection in remote sensing images using edge-aware segmentation network and easy-to-hard training strategy. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 61–65. [CrossRef]
29. Huang, C.; Thomas, N.; Goward, S.N.; Masek, J.G.; Zhu, Z.; Townshend, J.R.; Vogelmann, J.E. Automated masking of cloud and cloud shadow for forest change analysis using Landsat images. *Int. J. Remote Sens.* **2010**, *31*, 5449–5464. [CrossRef]
30. Wu, S.; Zhong, B.; Li, W.; Liu, Q. A new cloud detection method over Tibetan plateau and its surrounding area. In Proceedings of the 2013 IEEE International Geoscience and Remote Sensing Symposium—IGARSS, Melbourne, Australia, 21–26 July 2013; pp. 550–553. [CrossRef]
31. Zhu, X.; Helmer, E.H. An automatic method for screening clouds and cloud shadows in optical satellite image time series in cloudy regions. *Remote Sens. Environ.* **2018**, *214*, 135–153. [CrossRef]
32. Oishi, Y.; Ishida, H.; Nakamura, R. A new Landsat 8 cloud discrimination algorithm using thresholding tests. *Int. J. Remote Sens.* **2018**, *39*, 9113–9133. [CrossRef]
33. Li, Z.; Shen, H.; Li, H.; Xia, G.; Gamba, P.; Zhang, L. Multi-feature combined cloud and cloud shadow detection in GaoFen-1 wide field of view imagery. *Remote Sens. Environ.* **2017**, *191*, 342–358. [CrossRef]
34. Iannone, R.; Niro, F.; Goryl, P.; Dransfeld, S.; Hoersch, B.; Stelzer, K.; Kirches, G.; Paperin, M.; Brockmann, C.; Gómez-Chova, L.; et al. Proba-V cloud detection Round Robin: Validation results and recommendations. In Proceedings of the 2017 9th International Workshop on the Analysis of Multitemporal Remote Sensing Images (MultiTemp), Brugge, Belgium, 27–29 June 2017; pp. 1–8. [CrossRef]
35. Frantz, D.; Haß, E.; Uhl, A.; Stoffels, J.; Hill, J. Improvement of the Fmask algorithm for Sentinel-2 images: Separating clouds from bright surfaces based on parallax effects. *Remote Sens. Environ.* **2018**, *215*, 471–481. [CrossRef]
36. Kristollari, V.; Karathanassi, V. Convolutional neural networks for detecting challenging cases in cloud masking using Sentinel-2 imagery. In Proceedings of the SPIE Eighth International Conference on Remote Sensing and Geoinformation of the Environment (RSCy2020), Paphos, Cyprus, 16–18 March 2020; Manuscript submitted for publication.

37. Zhaoxiang, Z.; Iwasaki, A.; Guodong, X.; Jianing, S. Small satellite cloud detection based on deep learning and image compression. *Preprints* **2018**, 2018020103. [CrossRef]
38. Kristollari, V.; Karathanassi, V. Artificial neural networks for cloud masking of Sentinel-2 ocean images with noise and sunglint. *Int. J. Remote Sens.* **2020**, *41*, 4102–4135. [CrossRef]
39. Kohonen, T. The self-organizing map. *Proc. IEEE* **1990**, *78*, 1464–1480. [CrossRef]
40. Kohonen, T. Essentials of the self-organizing map. *Neural Netw.* **2013**, *37*, 52–65. [CrossRef]
41. Hsu, S.H.; Hsieh, J.P.A.; Chih, T.C.; Hsu, K.C. A two-stage architecture for stock price forecasting by integrating self-organizing map and support vector regression. *Expert Syst. Appl.* **2009**, *36*, 7947–7951. [CrossRef]
42. Hagenbuchner, M.; Tsoi, A.C. A supervised training algorithm for self-organizing maps for structures. *Pattern Recognit. Lett.* **2005**, *26*, 1874–1884. [CrossRef]
43. Ji, C. Land-use classification of remotely sensed data using Kohonen self-organizing feature map neural networks. *Photogramm. Eng. Remote Sens.* **2000**, *66*, 1451–1460.
44. Berthelot, B. *ATBD Cloud Detection for PROBA-V*; Technical Report; European Space Agency, ESA SP: Paris, France, 2017.
45. Charantonis, A.A. Cloud Detection Algorithm Development in Preparation for the Sentinel-2 Mission. Master's Thesis, École Polytechnique, Toulouse, France, 2009.
46. Chabiron, O. Cloud Detection Using SOM. Development of a Generic Method. Master's Thesis, Institut de Mathématiques de Toulouse, Toulouse, France, 2013.
47. Angeli, S.; Quesney, A.; Gross, L. *Image Simplification Using Kohonen Maps: Application to Satellite Data for Cloud Detection and Land Cover Mapping*; IntechOpen: London, UK, 2012; Chapter 14. [CrossRef]
48. Zhang, W.; Wang, J.; Jin, D.; Oreopoulos, L.; Zhang, Z. A deterministic self-organizing map approach and its application on satellite data based cloud type classification. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018; pp. 2027–2034. [CrossRef]
49. Hollstein, A.; Segl, K.; Guanter, L.; Brell, M.; Enesco, M. Ready-to-use methods for the detection of clouds, cirrus, snow, shadow, water and clear sky pixels in Sentinel-2 MSI images. *Remote Sens.* **2016**, *8*, 666. [CrossRef]
50. Baetens, L.; Desjardins, C.; Hagolle, O. Validation of Copernicus Sentinel-2 Cloud Masks Obtained from MAJA, Sen2Cor, and FMask Processors Using Reference Cloud Masks Generated with a Supervised Active Learning Procedure. *Remote Sens.* **2019**, *11*, 433. [CrossRef]
51. Riese, F.M.; Keller, S.; Hinz, S. Supervised and Semi-Supervised Self-Organizing Maps for Regression and Classification Focusing on Hyperspectral Data. *Remote Sens.* **2019**, *12*, 7. [CrossRef]
52. Barreto, G.; Araujo, A. Identification and Control of Dynamical Systems Using the Self-Organizing Map. *IEEE Trans. Neural Netw.* **2004**, *15*, 1244–1259. [CrossRef] [PubMed]
53. Matsushita, H.; Nishio, Y. Self-Organizing Map with Weighted Connections avoiding false-neighbor effects. In Proceedings of the 2010 IEEE International Symposium on Circuits and Systems, Paris, France, 30 May–2 June 2010; pp. 2554–2557.
54. Ultsch, A.; Siemon, H. Kohonen's self organizing feature maps for exploratory data analysis. In Proceedings of the International Neural Network Conference (INNC'90), Paris, France, 9–13 July 1990; Kluwer: Dordrecht, The Netherlands, 1990.