
METHODS AND PROCESSING TOOLS
AND INTERPRETATION OF SPACE INFORMATION

Cloud Detection from the Himawari-8 Satellite Data Using a Convolutional Neural Network

A. I. Andreev^{a, b, *} and Yu. A. Shamilova^a

^a Far Eastern Center, State Research Center for Space Hydrometeorology Planeta, Khabarovsk, Russia

^b Computer Center, Far Eastern Branch, Russian Academy of Sciences, Khabarovsk, Russia

*e-mail: a.andreev@dvrcpod.ru

Received July 28, 2020; revised November 18, 2020; accepted November 27, 2020

Abstract—The methodology of cloud detection from the data of the Himawari-8 geostationary satellite using a convolutional neural network is considered. The model of the cloud classifier has been tested in various scenarios, including winter and summer at night and daytime, as well as at day and night change. According to the test results, it has been found that, even in complex scenarios, the classifier has minimal errors when compared to the cloud-detection algorithms used in global operational practice.

Keywords: neural network, cloud mask, cloud detection, CNN, multispectral image classification, Himawari-8

DOI: 10.1134/S0001433821090401

INTRODUCTION

Real-time information acquisition and relatively high spatial resolution allow geostationary satellites to be considered the primary tool for global environmental, climate, and atmospheric monitoring. Instruments mounted on geostationary satellites can take images in a wide range of wavelengths, from visible to longwave infrared, which makes it possible to use them to solve various problems related to detecting dangerous weather phenomena, monitoring volcanic activity, detecting fires, etc. These and many other problems imply the use of a cloud mask, which has a number of requirements. The mask must be calculated in real time taking into account the high frequency of imaging. The mask must be calculated for different climatic conditions. In addition, some problems, e.g., monitoring forest fires or analyzing cloud formations, assume mask calculation not only in day time, but also at night.

Based on the problems currently being solved in the Far-Eastern Center of the State Research Center for Space Hydrometeorology Planeta, there is a need to create a cloud mask that takes into account the above requirements. This article proposes an algorithm for daytime and nighttime cloud detection specifically designed for geostationary satellite data using the Himawari-8 satellite data. The approach used in this work is based on the application of a convolutional neural network-based texture classifier (LeCun and Bengio, 1995). The classifier uses spectral and spatial (textural) features, which makes it possible to extract almost the maximum amount of information useful for cloud detection from satellite images. The classifi-

cation method is based on previous research in this field (Mahajan and Fataniya, 2019; Ganci et al., 2011; Dronner et al., 2018; Andreev et al., 2019) and is optimized for a higher data processing rate with minimal loss in classification quality, which makes it possible to use it in real time for geostationary satellites with high acquisition frequency.

CLOUD DETECTION METHODS

Rapid and accurate cloud detection is a challenge taking into account all the variety of external factors, and although various techniques for cloud detection have been presented over the past few decades, the development of more sophisticated and accurate methods is still ongoing (Mahajan and Fataniya, 2019). To date, several approaches can be conventionally identified: threshold-based methods, statistical methods, and machine learning and neural network approaches (Sun et al., 2016).

The threshold-based approach is the most common. It is based on the spectral analysis of the underlying surface and cloud cover in each pixel of the image. Thresholding algorithms are simple to implement and are characterized by a low computational complexity, and the classification results are easy to physically justify. Nevertheless, the quality of these algorithms depends to a large extent on the accuracy of the selection of threshold coefficients. The process of selecting these coefficients is very time-consuming for territories with different climatic conditions. In addition, in complex scenes where there are snow or opti-

cally thin cirrus clouds, the quality of classification is significantly reduced due to the similarity of the spectral characteristics of snow and clouds containing ice crystals (Chen et al., 2018; Stillinger et al., 2019), as well as due to the spectral distortion which occurs when radiation passes through clouds.

The use of radiative transfer models made it possible to improve the threshold algorithms. Simulating all kinds of spectral brightness coefficients under various combinations of parameters, such as solar and satellite zenith angles, atmospheric aerosol content, etc., makes it possible to significantly improve the accuracy of thresholding (Imai and Yoshida, 2016). This method is highly efficient, especially when combined with neural network algorithms (Chen et al., 2018), but currently has a significant limitation, i.e., the simulation of reflectance values at the upper boundary of the atmosphere is only done for pixels, but not for textures. As a consequence, texture information about the underlying surface and cloud cover becomes unavailable.

Statistical methods of cloud detection are based on regression equations obtained using statistical analysis of the values of spectral reflectance and brightness temperature among cloudy and cloudless pixels. In practice, these methods are most often used for preliminary data analysis and have the same drawbacks as the threshold methods, i.e., low efficiency of separating snow and clouds, as well as errors in detecting optically thin clouds. However, this approach can serve as the basis for constructing a cloud classifier (Amato et al., 2008).

Building classifiers based on machine learning algorithms is also a common approach in cloud detection. This approach consists of an automated selection of thresholds based on statistical data using the feature sets of each object being classified, thus combining the advantages of the techniques described above. Neural networks are a special case of machine learning algorithms. Practice shows that the neural network approach combined with texture and spectral features shows the highest accuracy in cloud detection (Droner et al., 2018; Mahajan and Fataniya, 2019).

The cloud detection method proposed in this article is based on the use of a convolutional neural network whose architecture is optimized for the fast processing of geostationary satellite images. In general, the method is universal for low-resolution satellite instruments and can be applied to both geostationary and polar orbiting satellites. As with any method based on machine learning algorithms, the convolutional neural network-based classifier model needs to be trained on preformed data.

DATASET FORMATION

The first step in developing a classification algorithm is to collect, label according to class affiliation, and preprocess training and test data of the required

size. Let the necessary sample volume be such that, if it is doubled, the overall level of classification accuracy changes insignificantly (less than 1%) while observing the condition of proportionality of the test and training sample volumes (the approximate volume ratio of 1 : 4, respectively, is accepted in this work). Note also that the training sample cannot contain samples from the test sample for the same time of satellite imaging. Multispectral images from the Advanced Himawari Imager (AHI) instrument mounted on the Himawari-8 geostationary satellite (Da, 2015) were used in this work. The data cover the time interval from January 2016 to July 2019 in a total of 302 images over the Asia-Pacific region (30° – 65° N and 105° – 180° W).

In order to solve the cloud detection problem, spectral channels previously successfully used in the works of other authors (Wang et al., 2019; Afzali Goroooh et al., 2020) and known to be effective in similar tasks were applied. These include visible and infrared wavelength channels reduced to a spatial resolution of 2 km: 0.64, 0.86, 1.6, 2.3, 3.9, 6.9, 7.3, 8.6, 11.2, and 12.4 μm , from which the training sample was subsequently formed.

Each sample from the dataset is a third-order tensor $X_{i,j,k}$, where the indices i and j correspond to the row and column of the image and k is the spectral channel number of the device. Each sample was assigned a class label $\{0, 1\}$ depending on whether the central pixel $p(i_c, j_c)$ ($i_c = T/2$, $j_c = T/2$, $k = 1 \dots C$) is cloudy or a cloudless underlying surface, respectively, where T is the texture size, C is the number of spectral channels, and i_c and j_c are coordinates of the texture center. The central pixel can be located not only in the center of the object in question, but also at the boundary of two or more classes (provided that this boundary is visually distinguishable).

The classification algorithm speed largely depends on the texture size. Therefore, in this work, the texture size T was assumed to be five pixels for the best use of computational resources. In (Ganci et al., 2011), this size was previously successfully applied to the problem of cloud detection from the SEVIRI data. In addition, as will be found later, a smaller texture size contributes to the increased detail of the resulting cloud mask in comparison with the results of the previous study (Andreev et al., 2019), where a size of 32 pixels was used.

The training datasets were formed in a sequential manner. The first stage of training sample formation involved the manual classification of points on satellite images by experienced decoders using RGB spectral channel synthesis of 0.64, 0.86, and 1.6 μm for daytime and 3.9, 11, and 12 μm for nighttime. Classified points totaling 5000 examples included samples of snow, ice, water surface, soil, and various types of cloud cover. Subsequently, these points became texture centers that were used to train the algorithm. After the pre-training, a test classification was performed on new test images, which had not previously been encountered in

the training or test sample. The classification results were analyzed, in the process of which errors were detected. Then, for these test images, the data in the error areas were relabeled and the new labeled data were added to the training and test samples, after which the algorithm was retrained. This recursive process continued until the number of classification errors was reduced to a minimum (in this work, this procedure was repeated up to ten times). This approach reduces the amount of data used in training and focuses on problematic cases of classification making adjustments to the data collection process. As a result of this work, the total amount of data was approximately 62000 texture samples.

In order to increase the number of training textures, they were augmented by rotating around the center every 90° and reflecting horizontally, which increased the original sample size to 495000.

The texture data obtained in the daytime and nighttime were combined into a single sample and subjected to preprocessing. Reflectance values at the upper boundary of the atmosphere were normalized by the solar zenith angle:

$$Ref_{corr} = \frac{Ref}{\cos SZA},$$

where Ref_{corr} is the corrected reflectance value, Ref is the original reflectance value, and SZA is the solar zenith angle value per pixel. Each spectral channel was then normalized between 0 and 1. Reflectance values at solar zenith angles greater than 85° were equated to zero. This value of the zenith angle was selected based on the results of (Godin, 2014), where the problem of matching the daytime and nighttime cloud masks was also solved. The data generated in this way were further used to train and test the classification algorithm.

CLASSIFICATION ALGORITHM

The classification algorithm is based on a convolutional neural network which was previously successfully used for object recognition on satellite images (Francis et al., 2019). The convolutional neural-network algorithm in the classification problem consists in the sequential transformation of the tensor of the source image (texture) using the convolution operation by matrix kernels to the output vector, in which one of the possible classes is encoded. The coefficients of matrix kernels are selected automatically in the learning process while encoding certain features of the image (straight lines, meshes, angles, etc.). When a sliding window passes over a certain region of the image and the convolution operation is applied, a response is generated in the form of a probability value that there is a certain feature in this region. The combination of such features serves as an indicator for a particular texture class (e.g., snow in the mountains). In addition, the values of the features (brightness tem-

perature and spectral brightness coefficients) also carry a significant amount of information.

The MetNet3 neural network model built in this work is a further development of the architecture presented in (Kramareva et al., 2019) and (Andreev et al., 2019). A number of improvements were made to this architecture in order to reduce its computational complexity with a slight deterioration in classification accuracy (not more than 1.5%), taking into account the results of recent research in this area (Szegedy et al., 2016). One such improvement is the transition to a fully convolutional architecture by replacing the fully connected output neural layers with a combination of convolutional layers, which reduced the computational complexity. Another architectural solution is the introduction of skipping connections between layers in order to extract texture features at different scales and reduce the phenomenon of model retraining. Subsequently, the two architecture branches that have output tensors A and B are combined into a single tensor C through the \oplus concatenation operation (which consists in alternate adding elements of the B tensor to the end of the A tensor):

$$C_{ijk} = A_{ijm} \oplus B_{ijn},$$

where i, j, k, m , and n are tensor indices, $k = m + n$. A similar approach in the architecture was successfully tested in (Dröner et al., 2018; Mateo-Garcia et al., 2019) on cloud detection by segmenting the entire multispectral image from geostationary satellite data, which is generally similar to the approach used in this article, where classification is performed using the sliding-window method for each pixel separately.

Before starting the training process, a small (15% size of the source dataset) validation sample was extracted from the training dataset, intended to evaluate the algorithm in the training process and to adjust its parameters. In the final evaluation of the classification results, this set was subsequently not used.

The neural network was trained using the Adam algorithm (Kingma and Ba, 2014), for which the standard learning rate was set to 10^{-4} . The cross-entropy formula for the case of binary classification (Sadowski, 2016) was used as the target loss function. The classification model converges after approximately 400 iterations of the training algorithm (the convergence character is exponential) in the considered problem with the specified training parameters.

The algorithm for implementing the classifier model was developed in the Python 3.7 programming language using the TensorFlow 1.14 and Keras 2.2.4 neural network modules. The maximum calculation time for a 2300×7500 -pixel cloud mask (region 30° N 65° W to 75° N 200° W) takes up to 4 min using the following hardware configuration: an Intel Core i7-5820K CPU, an NVIDIA GTX 1060 GPU, 16 GB of CPU RAM, and 3 GB of graphics card RAM. Therefore, the specified calculation time makes it possible to

Table 1. Results of a classification precision evaluation for the validation dataset

Time of day	Precision, %	Recall, %	F1, %
Daytime	96.14	97.86	96.99
Nighttime	98.87	98.59	98.73

apply this classifier in the real-time mode on this hardware.

ANALYSIS OF RESULTS

The quality of cloud masks obtained by the classifier presented in this article was evaluated using a validation dataset, as is common in machine learning problems, and by comparing the results with the performance of cloud detection algorithms used in global operational practice. Precision, recall, and the F1-measure indicator, which is a harmonic average of precision and recall (Friedman et al., 2001), were used as evaluation metrics:

$$Precision = \frac{TP}{TP + FP}, \quad (1)$$

$$Recall = \frac{TP}{TP + FN}, \quad (2)$$

$$F1 = 2 \frac{Precision \times Recall}{Precision + Recall} = \frac{2TP}{2TP + FP + FN}, \quad (3)$$

where:

- TP (True Positive) is the number of examples where there is cloudiness in the example under consideration and in the reference.
- FP (False Positive) is the number of examples where there is cloudiness in the examined example but no cloudiness in the reference.
- FN (False Negative) is the number of examples where there is no cloudiness on the examined example but there is cloudiness in the reference.

The validation sample for daytime and nighttime included about 14000 examples of cloudy and cloudless textures. This sample was generated using 76 observation terms from January 2016 to July 2019. Following formulas (1), (2), and (3), the results presented in Table 1 were obtained for the validation sample.

The results presented in Table 1 indicate a high generalizability of the classifier under consideration, since the validation sample data are not contained in the training sample data. A slightly higher precision of the classification at night is due to a slightly less representative sample of textures. This is caused by the fact that manual classification using only IR channel data is much more labor-intensive, because there are often situations when it is problematic to unambiguously distinguish the presence or absence of cloud cover in the pixel in question. When auxiliary information such

as ground-station data was not available, such cases were not included in the training and test datasets.

Another approach was also considered for a more comprehensive evaluation of the classifier. The main idea of this approach is a pixel-by-pixel comparison of cloud masks, with the reference ones obtained using known valid algorithms. In this work, the NOAA JPSS_GRAN product containing a binary cloud mask based on data from the VIIRS SC NOAA-20 instrument (<https://www.bou.class.noaa.gov>) was selected as the reference. This cloud mask is generated by an algorithm based on a series of threshold tests described in detail in (Godin, 2014). This algorithm takes into account the underlying surface type, data on wind speed and direction near the sea surface, water content in the atmospheric column, and air temperature in the surface layer.

Due to errors observed in the cloud masks from the VIIRS instrument data, the cloud mask from the 2B-CLDCLASS-LIDAR product (<http://www.cloudsat.cira.colostate.edu>) generated from CloudSat and CALIPSO satellite data is used as a reference in winter time. This cloud mask is a track with a width of 1 pixel (approximately 1.4 km) (Sassen, 2008), along which the vertical sounding of the atmosphere was carried out in order to detect cloudiness and subsequently classify it. The threshold algorithm used to process this data uses data on cloud height, temperature, reflectance, and optical thickness.

The mask comparison used VIIRS instrument data from August 1–5, 2019, and February 1–5, 2020, as well as CloudSat and CALIPSO data from February 1–8, 2017. The maximum difference in acquisition time between the reference instruments (VIIRS, CloudSat, and CALIPSO) and AHI is not more than 5 min. On the cloud masks in the places of their intersections, polygons were cut in the total number of 32 pcs., examples of which are shown in Figs. 4–8. The polygons are distributed throughout the Asia-Pacific region. Table 2 shows the average validation results for the different scenarios.

The precision evaluation results showed a slightly higher result for CloudSat data when compared to VIIRS. This fact is explained by the fact that CloudSat takes measurements along the track, so the number of points related to cloud boundaries is small. In the case of VIIRS cloud masks in the form of images, the contribution of boundary values is taken into account to a greater extent, which leads to a slight decrease in precision when compared with these masks.

Daytime Summer

From the results of the visual analysis (see Fig. 1) and metrics calculation (Table 2), we can note that the VIIRS cloud mask is comparable to the cloud mask obtained by the developed classifier from the AHI data. The VIIRS cloud mask has false positives along

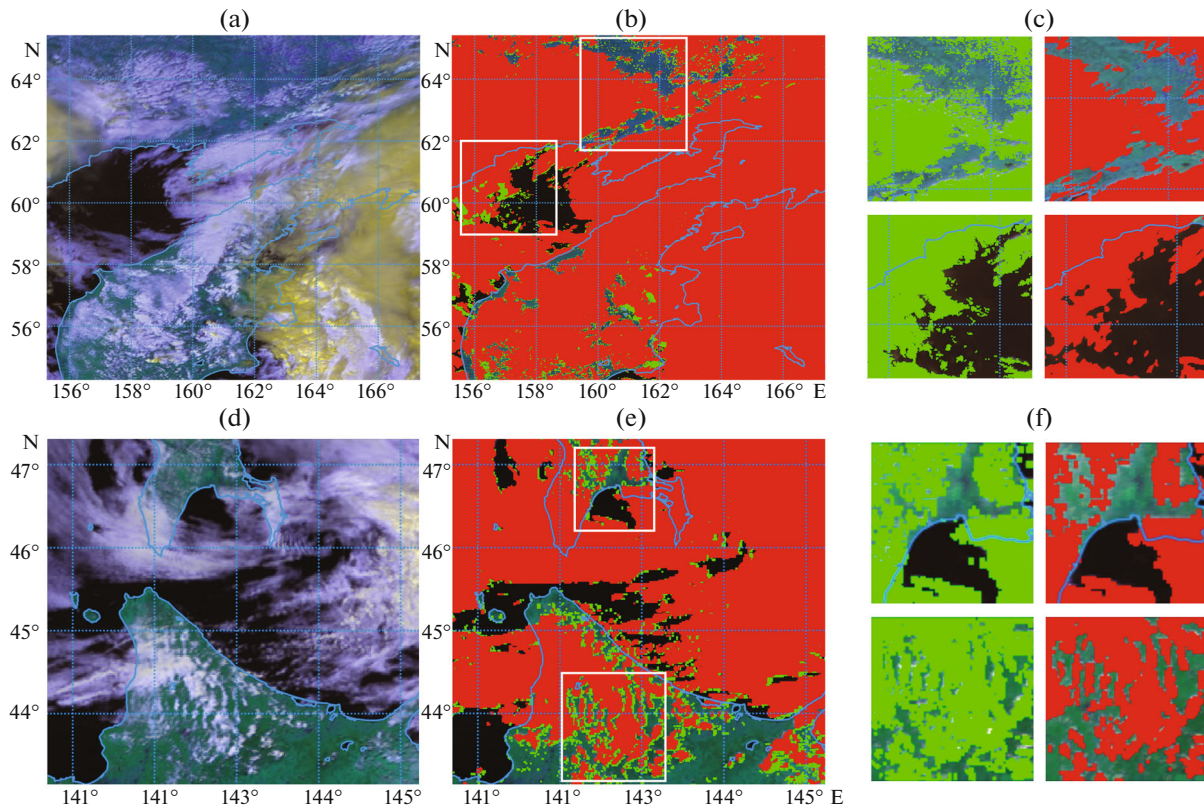


Fig. 1. Examples of cloud masks for 2 h 20 min on Aug. 4, 2019, UTC (top row) and 3 h 30 min on August 5, 2019, UTC (bottom row): (a, c) RGB channel synthesis (R: 0.64, G: 0.86, B: 1.6 μm); (b, d) AHI cloud mask (red) superimposed over the VIIRS mask (green); enlarged mask fragments are shown in the right-hand area of the figure. The blue lines indicate the coastal line (solid) and coordinate grid (dashed).

coastlines, rivers, and lakes. The AHI cloud mask is devoid of these drawbacks, although it tends to fail to fully identify the edges of optically thin clouds (Fig. 1d, enlarged scale in the right region of the figure, bottom row). The incomplete extraction of the optically thin cloudiness can be regarded as an advantage or a disadvantage, depending on the purpose of the mask.

According to the tests, the AHI cloud mask has an average precision of about 96% and recall of 94% when compared to the VIIRS mask, which is a rather high-quality indicator.

Nighttime Summer

The cases where instrument imagery in the visible wavelength range is not available are significantly more

challenging. In this case, the main problem is the detection of lower layer cloudiness (layer, layer-cumulus, etc.), because their temperature is close to the underlying surface temperature, which makes their detection in the infrared range much more difficult. Figure 2 shows examples of polygons at night during summer.

Based on the analysis of formulas (1) and (2) (high precision and low recall) and the results of Table 2, as well as the visual evaluation, it follows that the VIIRS cloud mask often has a large number of false positives (areas highlighted by circles in Fig. 2), which is confirmed by the high precision index (on average, 95%) and low recall (63%). The AHI cloud mask confidently detects convective cloudiness, but omissions can be observed for clouds located at the periphery of a large cloud mass.

Table 2. Precision evaluations for tested polygons

Used instrument, scenario	Precision	Recall	F1
VIIRS cloud mask, summer, daytime	95.86	94.25	94.95
VIIRS cloudiness mask, summer, nighttime	95.43	63.07	75.47
CloudSat cloud mask, winter, daytime	97.73	88.63	92.59

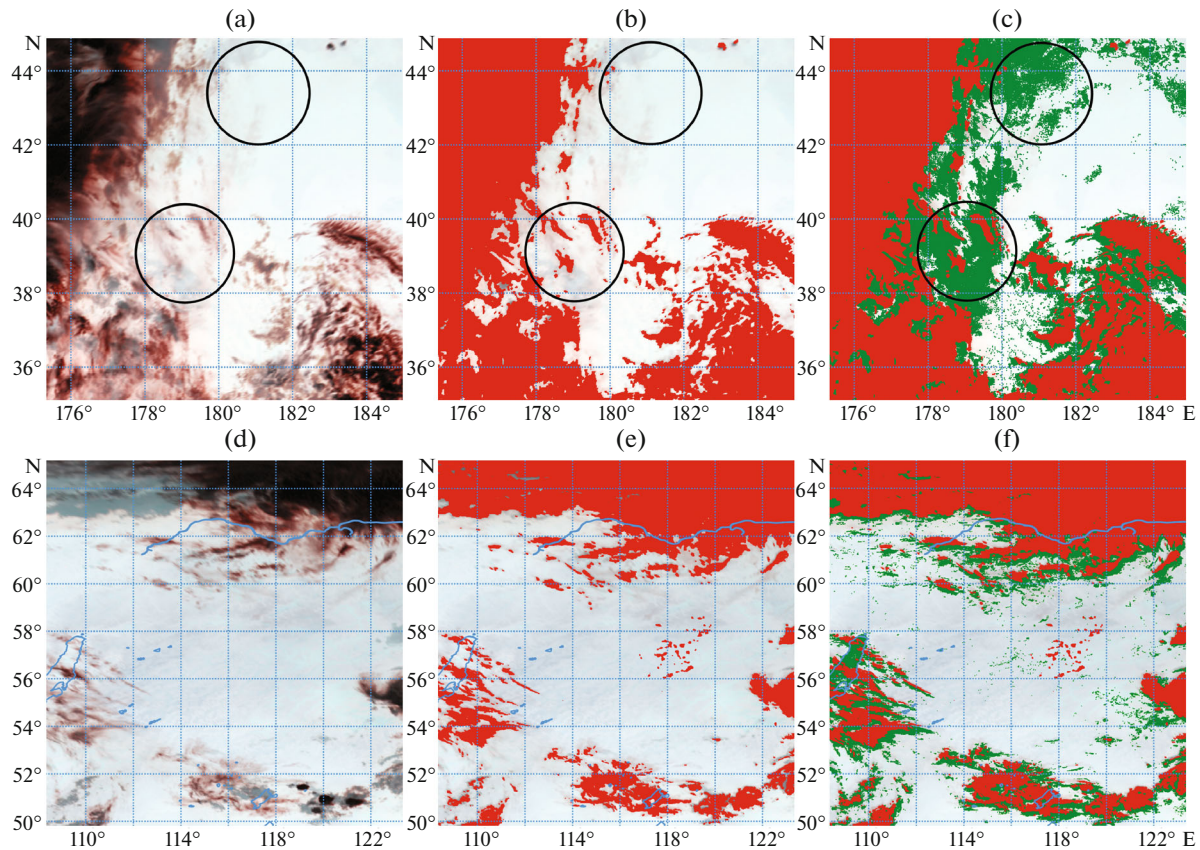


Fig. 2. Examples of cloud masks for Aug. 1, 2019, 13:40 UTC (top row) and August 2, 2019, 18:30 UTC (bottom row): (a, d) RGB channel synthesis (R: 3.9, G: 11.2, B: 12.4 μm), dark regions correspond to lower temperatures; (b, e) AHI cloud mask (red); and (c, f) comparison with VIIRS masks (green). Circles indicate regions of interest. The blue lines show the coastal line (solid) and coordinate grid (dashed).

Daytime Winter

The previously mentioned problem of cloud detection in the presence of snow and ice demonstrates the shortcomings of thresholding techniques well, as can be seen in the VIIRS cloud mask in Figs. 3b and 3e. The algorithm used to compute the VIIRS mask determines the probability of cloud presence for each pixel, but even at a probability value of 0.995 artifacts in the form of a large amount of noise are observed, especially for mountainous areas. On the AHI cloud mask (Figs. 3c, 3f), the cloudiness fields are completely correctly identified and ice and snow are recognized with a high degree of precision.

The CPR (radar) and CALIOP (lidar) instruments of CloudSat and CALIPSO satellites, respectively, were considered alternatives to the VIIRS instrument in this work. Precision estimates using this product are given in Table 2 for winter daylight hours. The precision of the validated cloud mask is quite high here (97% on average), which is confirmed by the visual analysis of the AHI satellite images. The available errors are mainly due to instrument mismatches in time and angle of observation, as well as to the high

sensitivity of microwave instruments to aerosols contained in the atmosphere.

Nighttime Winter

Due to the fact that the 2B-CLDCLASS-LIDAR data are not available at night, a visual quality evaluation was performed for this scenario. Figure 4 shows an example of the AHI cloud mask compared to the VIIRS mask (Fig. 4d). For ease of analysis, a synthesis of visible channels during the daytime is also presented to evaluate the snow cover distribution.

By analyzing the classification results in the winter night time, we can conclude that the cloud mask according to the VIIRS instrument actually has the same drawbacks as in summer: artifacts along the coastline and “cloud overestimation,” but the number of false classifications of snow is low. Ice is observed in the area indicated by the circle, which was also misclassified by the VIIRS Mask as cloud cover. The mask identifies cold convective clouds and detects snow and ice in the AHI instrument data well but still often “underestimates” cumulus and layer clouds.

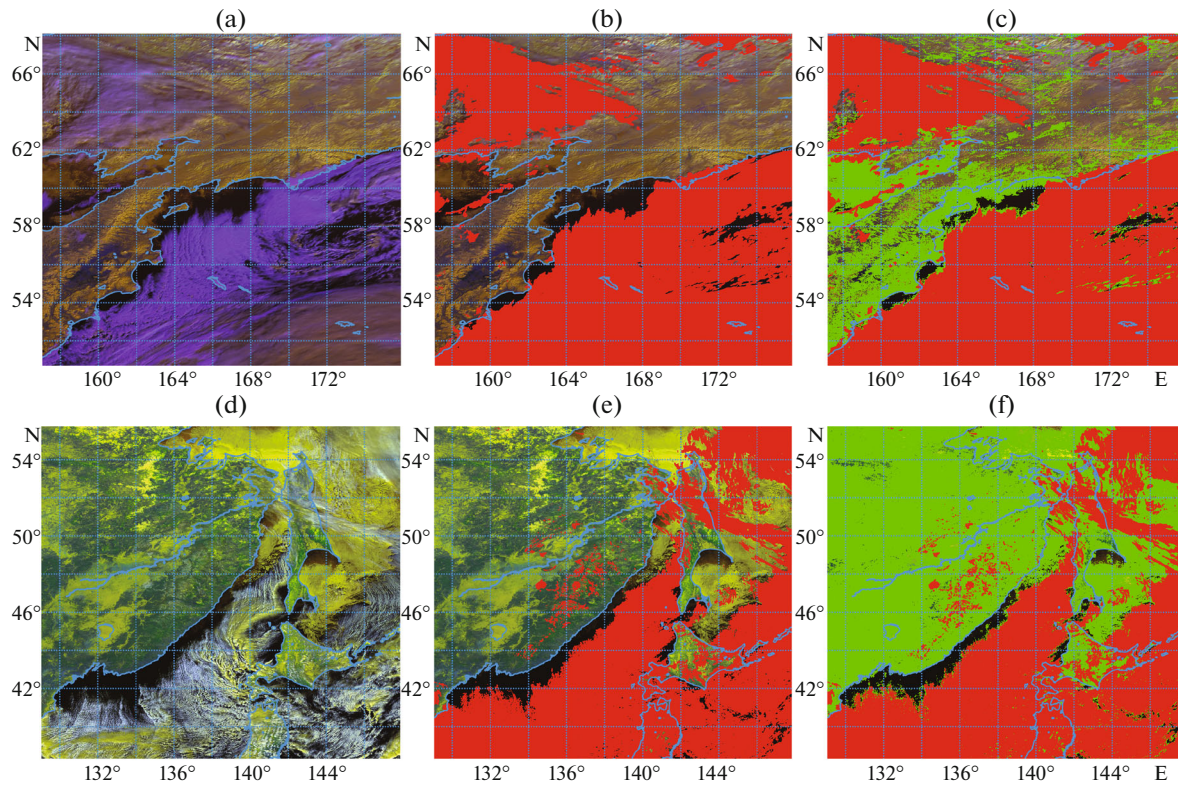


Fig. 3. Examples of cloud masks for Feb. 1, 2020, 0:40 UTC (top row) and Feb. 8, 2020, 3:20 UTC (bottom row): (a, d) RGB channel synthesis (R: 0.64, G: 0.86, B: 1.6 μm); (b, e) VIIRS cloud mask (red); and (c, f) comparison with VIIRS masks (green). The blue lines indicate the coastal line (solid) and coordinate grid (dashed).

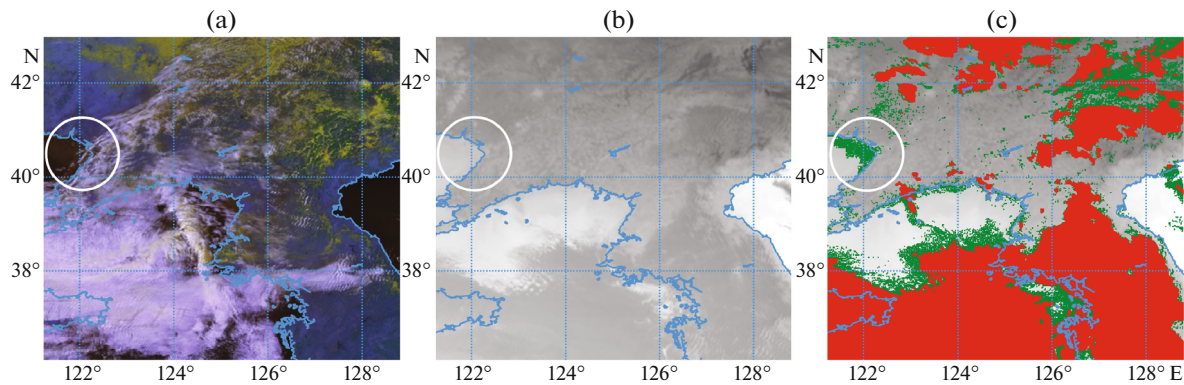


Fig. 4. Example of the cloud mask for Feb. 7, 2020, 17:50 UTC: (a) RGB channel synthesis (R: 0.64, G: 0.86, B: 1.6 μm) for Feb. 7, 2020, 2:00 UTC to estimate snow distribution; (b) IR channel 11 μm , darker areas have lower temperatures; and (c) VIIRS cloud mask (green) compared to the AHI mask (red). Circles indicate regions of interest. The blue lines show the coastal line (solid) and coordinate grid (dashed).

Alternation of Day and Night

Finally, let us consider a scenario in which there is a transition between daytime and nighttime cloud masks. Figure 5 shows a typical example in which a cloud that can be clearly seen in the visible image merges with the underlying surface in the infrared

range (the area is indicated by a white circle in Fig. 5a). Note also that such examples of cloud cover were not included in the training dataset due to difficulties in their manual interpretation by specialists at night. The cloud cover in the lower part of the image shows a smooth transition of the cloud cover mask through the terminator line set at 85° on the zenith angle of the Sun.

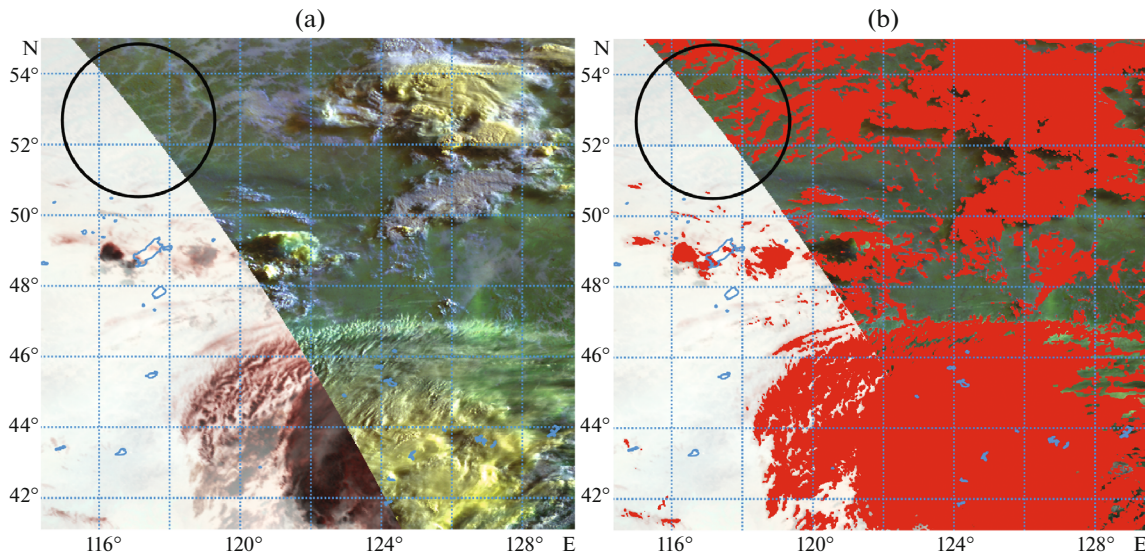


Fig. 5. Example of the cloud mask for Aug. 2, 2019, 21:00 UTC: (a) RGB channel syntheses (R: 3.9, G: 11.2, B: 12.4 μm) and (R: 0.64, G: 0.86, B: 1.6 μm) left and right of the terminator line, respectively; (b) AHI cloud mask. The circle indicates the region of interest. The blue lines show the coastal line (solid) and coordinate grid (dashed).

CONCLUSIONS

This article considered one of the most promising approaches to cloud detection using geostationary satellite data by the example of the Himawari-8 satellite based on the application of a convolutional neural network and taking into account the spectral and textural characteristics of clouds and the underlying surface.

The quality of the cloud masks obtained using the classifier proposed in this work was evaluated using the validation texture set, as well as by a pixel-by-pixel comparison with cloud masks from SC NOAA-20, CloudSat, and CALIPSO data. The validation process included a comprehensive quality evaluation using Precision, Recall, and f1-measure metrics under various scenarios, including winter and summer, nighttime and daytime, and the transient process (dusk/dawn). Due to the labor intensity of the validation process, its results do not claim to be absolutely complete; nevertheless, according to the available data, we can conclude about the high quality of the obtained classifier, the results of which can be used for practical purposes.

The undoubted advantages of the approach used in this work are the high precision of snow and ice detection, no need to use third-party data (except for satellite instrument images), and the universality of the algorithm with respect to different climatic and geographical conditions. The latter is achieved by supplementing the training sample with data obtained for other territories by the additional training of the existing model. In this case, the threshold algorithms require a manual selection of coefficients and consideration of territorial and temporal dependences in the analytical form. At the same time, the approach considered in this work makes it possible to take into account more complex dependences than in the threshold algorithms.

The disadvantages of the classifier model include the incomplete extraction of stratus and cumulus clouds (especially at night) and a slight smoothing of the mask (a consequence of using the texture method for classification), which leads to its lower level of details. In general, the disadvantages of the approach used in this work can also include the high labor intensity of the formation of the training sample, because at this stage a manual markup of large volumes of data (tens of thousands for the most accurate classification) with the involvement of experienced specialists in the decoding of satellite images is required. Note that partly the share of manual labor can be reduced by applying already existing cloud masks when preparing a new training sample. In addition, the possibilities of maximally automating the generation of training texture sets are currently being investigated.

According to the authors, one of the further directions for improving the classifier, in addition to the spectral and spatial (texture) components, is to take into account the temporal dependence, which requires considering a sequence of satellite images. Since a cloud is a dynamic system, the analysis of the temporal component in each pixel of the image can improve the accuracy of cloud detection against the background of a static underlying surface. One promising direction for further research is also to improve the detection accuracy of the lower cumulus and stratus clouds at night.

Based on the results of this work, the classifier model was introduced into the operational work of the Far Eastern Center of the State Research Center for Space Hydrometeorology Planeta as part of the software package for calculating cloud cover parameters.

CONFLICT OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

- Afzali Goroo, V., Kalia, S., Nguyen, P., Hsu Kuo-Lin, Sorooshian, S., Ganguly, S., and Nemani, R., Deep neural network cloud-type classification (DeepCTC) model and its application in evaluating PERSIANN-CCS, *Remote Sens.*, 2020, vol. 12, no. 2, id 316. <https://doi.org/10.3390/rs12020316>
- Amato, U., Antoniadis, A., Cuomo, V., Cuttillo, L., Franzese, M., Murino, L., and Serio, C., Statistical cloud detection from SEVIRI multispectral images, *Remote Sens. Environ.*, 2008, vol. 112, no. 3, pp. 750–766. <https://doi.org/10.1016/j.rse.2007.06.004>
- Andreev, A.I., Shamilova, Y.A., and Kholodov, E.I., Using convolutional neural networks for cloud detection from Meteor-M no. 2 MSU-MR data, *Russ. Meteorol. Hydrol.*, 2019, vol. 44, no. 7, pp. 459–466.
- Chen, N., Li, W., Gatebe, C., Tanikawa, T., Hori, M., Shimada, R., Aoki, T., and Stamnes, K., New neural network cloud mask algorithm based on radiative transfer simulations, *Remote Sens. Environ.*, 2018, vol. 219, pp. 62–71. <https://doi.org/10.1016/j.rse.2018.09.029>
- Da, C., Preliminary assessment of the Advanced Himawari Imager (AHI) measurement onboard Himawari-8 geostationary satellite, *Remote Sens. Lett.*, 2015, vol. 6, no. 8, pp. 637–646. <https://doi.org/10.1080/2150704X.2015.1066522>
- Dröner, J., Korfhage, N., Egli, S., Mühling, M., Thies, B., Bendix, J., Freisleben, B., and Seeger, B., Fast cloud segmentation using convolutional neural networks, *Remote Sens.*, 2018, vol. 10, no. 11, id 1782. <https://doi.org/10.3390/rs10111782>
- Francis, A., Sidiropoulos, P., and Muller, J.P., CloudFCN: Accurate and robust cloud detection for satellite imagery with deep learning, *Remote Sens.*, 2019, vol. 11, no. 19, id 2312. <https://doi.org/10.3390/rs11192312>
- Ganci, G., Vicari, A., Bonfiglio, S., and Gallo, G., A texon-based cloud detection algorithm for MSG-SEVIRI multispectral images, *Geomatics, Nat. Hazards Risk*, 2011, vol. 2, no. 3, pp. 279–290. <https://doi.org/10.1080/19475705.2011.578263>
- Godin, R., *Joint Polar Satellite System (JPSS) VIIRS Cloud Mask (VCM) Algorithm Theoretical Basis Document (ATBD)*, Greenbelt, Md.: NASA Goddard Space Flight Center, 2014, JPSS Ground Project, Code 474, 474–00033.
- Hastie, T., Tibshirani, R., and Friedman, J., *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, 2009.
- Imai, T. and Yoshida, R., Algorithm theoretical basis for Himawari-8 cloud mask product, *Meteorol. Satell. Cent. Tech. Note*, 2016, vol. 61, pp. 1–17.
- Kingma, D.P. and Ba, J., Adam: A method for stochastic optimization, 2014. <https://arxiv.org/abs/1412.6980>.
- Kramareva, L.S., Andreev, A.I., Simonenko, E.V., and Sorokin, A.A., The use of a convolutional neural network for detecting snow according to the data of the multi-channel satellite device of Meteor-M No. 2 spacecraft, *Procedia Comput. Sci.*, 2019, vol. 150, pp. 368–375. <https://doi.org/10.1016/j.procs.2019.02.065>
- LeCun, Y. and Bengio, Y., Convolutional networks for images, speech, and time series, in *The Handbook of Brain Theory and Neural Networks*, 1995, vol. 3361, no. 10, pp. 255–258.
- Mahajan, S. and Fataniya, B., Cloud detection methodologies: Variants and development—a review, *Complex Intell. Syst.*, 2019, pp. 1–11.
- Mateo-García, G., Adsua, E.J., Pérez-Suay, A., and Gómez-Chova, L., Convolutional long short-term memory network for multitemporal cloud detection over landmarks, *IEEE International Geoscience and Remote Sensing Symposium (IGARSS-2019)*, pp. 210–213. <https://doi.org/10.1109/IGARSS.2019.8897832>
- Sadowski, P., Notes on backpropagation, 2016. <https://www.ics.uci.edu/pjsadows/notes.pdf>.
- Sassen, K., Wang, Z., and Liu, D., Global distribution of cirrus clouds from CloudSat/Cloud–Aerosol lidar and infrared pathfinder satellite observations (CALIPSO) measurements, *J. Geophys. Res.: Atmos.*, 2008, vol. 113, no. D8. <https://doi.org/10.1029/2008JD009972>
- Stillinger, T., Roberts, D.A., Collar, N.M., and Dozier, J., Cloud masking for Landsat 8 and MODIS Terra over snow-covered terrain: Error analysis and spectral similarity between snow and cloud, *Water Resour. Res.*, 2019, vol. 55, no. 7, pp. 6169–6184. <https://doi.org/10.1029/2019WR024932>
- Sun, L., Wei, J., Wang, J., Mi, X., Guo, Y., Lv, Y., Yang, Y., Gan, P., Zhou, X., Jia, C., and Tian, X., A universal dynamic threshold cloud detection algorithm (UDTCDA) supported by a prior surface reflectance database, *J. Geophys. Res.: Atmos.*, 2016, vol. 121, no. 12, pp. 7172–7196. <https://doi.org/10.1002/2015JD024722>
- Szegedy, C., Ioffe, S., and Vanhoucke, V., Inception-v4, Inception-ResNet and the impact of residual connection on learning, 2016. <https://arxiv.org/abs/1602.07261v2>.
- Wang, X., Min, M., Wang, F., Guo, J., Li, B., and Tang, S., Intercomparisons of Cloud Mask Products Among Fengyun-4A, Himawari-8, and MODIS, *IEEE Trans. Geosci. Remote Sens.*, 2019, vol. 57, no. 11, pp. 8827–8839. <https://doi.org/10.1109/TGRS.2019.2923247>

Translated by O. Pismenov