



*Research article*

## **A two-branch cloud detection algorithm based on the fusion of a feature enhancement module and Gaussian mixture model**

**Fangrong Zhou<sup>1</sup>, Gang Wen<sup>1</sup>, Yi Ma<sup>1</sup>, Yutang Ma<sup>1</sup>, Hao Pan<sup>1</sup>, Hao Geng<sup>1</sup>, Jun Cao<sup>1</sup>, Yitong Fu<sup>2</sup>, Shunzhen Zhou<sup>2</sup> and Kaizheng Wang<sup>2,\*</sup>**

<sup>1</sup> Joint Laboratory of Power Remote Sensing Technology (Electric Power Research Institute, Yunnan Power Grid Company Ltd.), Kunming 650217, China

<sup>2</sup> Faculty of Electric Power Engineering, Kunming University of Science and Technology, Kunming 650000, China

\* **Correspondence:** Email: [kz.wang@foxmail.com](mailto:kz.wang@foxmail.com).

**Abstract:** Accurate cloud detection is an important step to improve the utilization rate of remote sensing (RS). However, existing cloud detection algorithms have difficulty in identifying edge clouds and broken clouds. Therefore, based on the channel data of the Himawari-8 satellite, this work proposes a method that combines the feature enhancement module with the Gaussian mixture model (GMM). First, statistical analysis using the probability density functions (PDFs) of spectral data from clouds and underlying surface pixels was conducted, selecting cluster features suitable for daytime and nighttime. Then, in this work, the Laplacian operator is introduced to enhance the spectral features of cloud edges and broken clouds. Additionally, enhanced spectral features are input into the debugged GMM model for cloud detection. Validation against visual interpretation shows promising consistency, with the proposed algorithm outperforming other methods such as RF, KNN and GMM in accuracy metrics, demonstrating its potential for high-precision cloud detection in RS images.

**Keywords:** cloud detection; Himawari-8; Gaussian mixture model; probability density function; Laplacian operator

---

### **1. Introduction**

In recent years, remote sensing (RS) imagery has been widely applied in various fields, with the

continuous development of RS technology. The fields include environmental monitoring [1], natural disaster monitoring [2–4], mineral resource development [5] and cartography [6]. However, a large part of Earth's surface is covered by clouds [7], which results in a significant amount of ground information being inaccessible to satellites. Additionally, the texture and spectral information of images are inconstant. There are substantial challenges in subsequent object detection and analysis tasks. Therefore, accurate and effective cloud detection is crucial due to being a preprocessing step for getting over the challenges before the analysis of satellite images [8].

Currently, cloud detection methods can be categorized into multi-spectral threshold methods and machine learning-based methods. Multi-spectral threshold methods use spectral differences between clouds and the underlying surface in different wavelength bands to select thresholds for cloud detection. Rossow et al. used spectral thresholding based on the 11  $\mu\text{m}$  brightness temperature data of cloud tops in the international satellite cloud climatology project [9]. Xiong et al. proposed a cloud detection approach based on hybrid multi-spectral features with dynamic thresholds [10]. The multi-spectral threshold methods have a simple structure and are easy to implement. However, it is challenging for them to distinguish between clouds and the underlying surface for them when the ground is covered with ice, snow, desert or a cloud cover that is thin.

Many cloud detection algorithms based on machine learning have emerged with the rapid development of computer science, including the support vector machine (SVM) [11–13], random forest (RF) [14,15], convolutional neural network (CNN) [16,17] and others. Li et al. [11] utilized brightness temperature and texture features, employing SVM as a classifier for cloud detection. Zhang et al. proposed a cloud detection method based on the multi-feature embedded learning SVM [12]. Fu et al. used the results of the ensemble threshold as training samples for the RF classifier [15]. Zhang et al. designed a cascaded feature attention module based on the encoder-decoder structure to obtain and enhance useful feature information [16]. Xie et al. proposed a CNN with two branches to detect different types of clouds [17]. However, all the machine learning methods used by them encounter the following challenges:

(1) Cloud edges are often semi-transparent, making it difficult for them to separate the underlying surface based solely on visual features. Therefore, it is essential to make full use of the multi-channel information from RS satellites.

(2) These machine learning methods require a sufficient amount of training data to achieve reliable performance [18]. However, fragmented cloud samples are often sparse and scattered, demanding a significant amount of effort and time for manual labeling.

The GMM, as an unsupervised learning method, does not require manual sample labeling. Therefore, it requires less preparation work before detection [19]. Furthermore, visual features can have difficulty in distinguishing between cloud edges and the underlying surface during manual labeling, and incorrect labeling may result in model inaccuracies. In contrast, the GMM algorithm use satellite multi-channel reflectance and brightness temperature as clustering features and can effectively leverage the spectral differences between clouds and the underlying surface for classification.

In summary, we propose a cloud detection algorithm based on a feature enhancement module and GMM clustering. The proposed method utilizes clustering-based feature analysis to select appropriate feature schemes for different time periods, enabling cloud detection throughout the day. Additionally, we employ the Laplacian operator to enhance the spectral characteristics of the cloud edges and broken cloud regions and improve the accuracy of the GMM classifier for cloud detection.

The paper is structured as follows: Section 2 presents the various applications of cloud detection.

Section 3 discusses data preprocessing, with a focus on the selection of clustering features. Section 4 presents a modular description of the proposed algorithm, highlighting its key components and functionalities. Section 5 provides a comprehensive account of all experiments conducted and their corresponding observations, shedding light on the algorithm's performance and efficacy. Section 6 discusses the ethical considerations and potential societal impacts of enhanced cloud detection. Finally, Section 7 offers a summary of the paper, summarizing the major contributions.

## 2. Relevant applications of cloud detection

### (1) Wildfire detection

When utilizing satellite data for wildfire monitoring, clouds' mid-infrared reflectance might cause them to be mistakenly identified as fire. Hence, cloud detection is crucial. Zhang et al. set different thresholds during daytime and nighttime to detect clouds [20]. Kang et al. employed an algorithm utilizing Imager's reflectance and  $BT$  to eliminate clouds in aerosol optical depth retrieval [21].

### (2) Cartography in agriculture

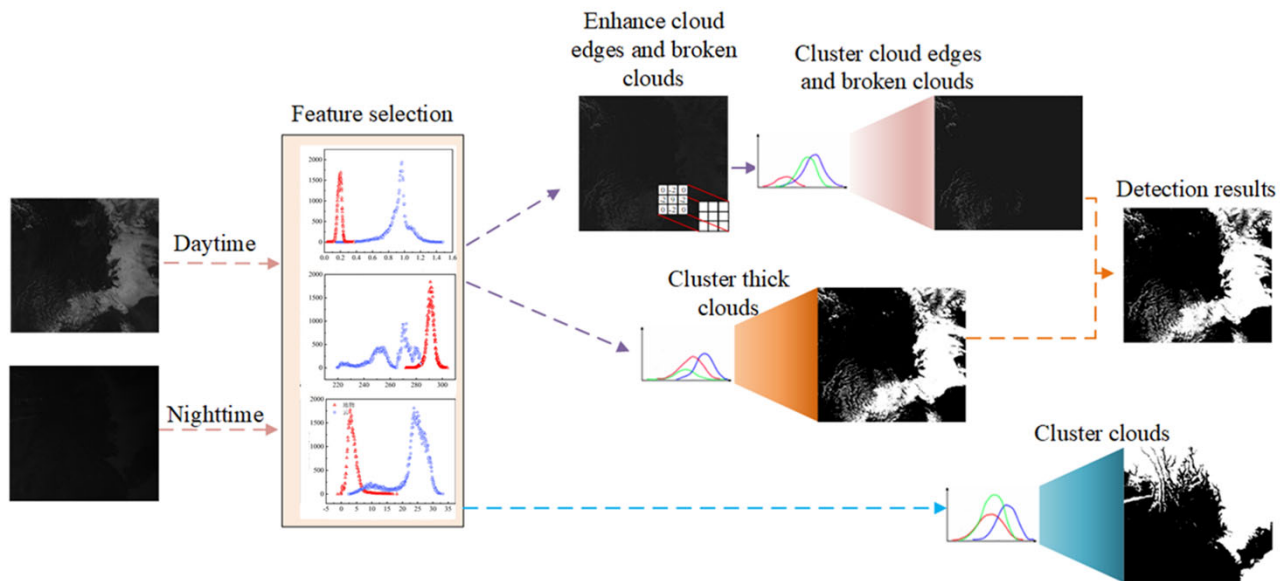
To reduce spectral redundancy in images and improve method efficiency, cloud detection is often a crucial step in data preprocessing during agricultural mapping. Xia et al. utilized cloud masks generated by the C Function of Mask algorithm [22]. Li et al. calculated cloud scores according to the spectral data from six bands to achieve cloud removal [23].

### (3) Vegetation change analysis

Vegetation change analysis is particularly sensitive to cloud contamination because unmarked clouds above the vegetation can potentially be mistaken for change. Even minor errors in cloud detection can lead to significant inaccuracies in downstream vegetation change analysis. Huang et al. used clear view forest pixels as references to define cloud boundaries, separating clouds from the clear view surface in the spectral-temperature space [24]. Long et al. employed cloud identification algorithms provided by Google Earth Engine, utilizing the QA band to identify clouds and shadows in the images [25].

## 3. Data preprocessing

The cloud detection algorithm in this paper is based on multi-spectral data from the Himawari-8 satellite. As shown in Figure 1, the first step involves conducting a statistical analysis using probability density functions (PDFs) of pixel spectral data from clouds and underlying surfaces to select feature schemes for different time periods. The second step involves cloud detection based on the different feature schemes selected. During nighttime, the raw data of  $BT_{15}$  and  $BTD_{7-14}$  are directly subjected to GMM clustering to obtain the detection results. During the daytime, the detection process is divided into three parts. First, a feature enhancement module is applied to  $R_{3+4}$ , and the enhanced data is clustered to detect cloud edges and broken clouds. Furthermore, the raw data is clustered to detect thick clouds. Finally, the results from the dual branches of clustering are combined to obtain the detection results.



**Figure 1.** Algorithm flowchart.

### 3.1. Cloud definition

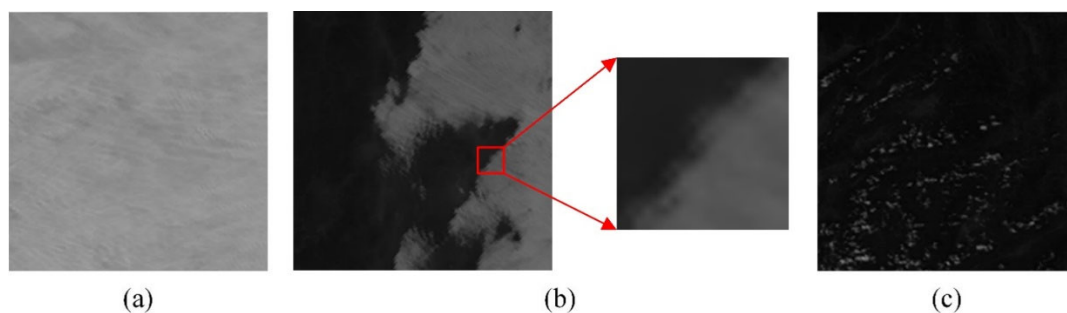
The definition of clouds is crucial, and many scholars currently use a subjective approach to define different types of clouds. In this paper, cloud transparency and distribution are used as criteria for subjective definition.

(1) Thick clouds are opaque and typically appear over large areas.

(2) Cloud edges refer to clouds with transparency at the cloud boundaries. Cloud edges do not abruptly separate but gradually transition between the cloud layer and the non-cloud layer.

(3) Broken clouds exhibit varying degrees of transparency, with their most noticeable characteristic being their distribution in small patches across the sky.

Figure 2 illustrates various cloud types, with some cloud edges highlighted using red boxes.



**Figure 2.** (a) Thick clouds; (b) Cloud edges; (c) Broken clouds.

### 3.2. Himawari-8/AHI data

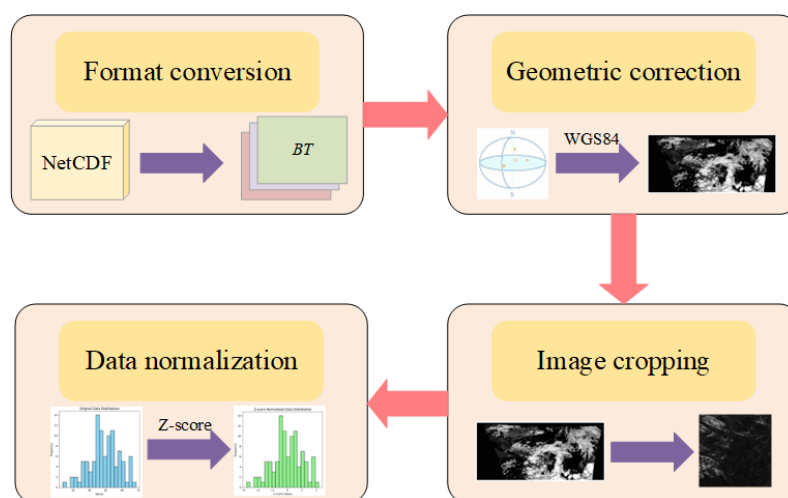
The synchronous meteorological satellite Himawari-8 was officially launched in July 2015. The satellite is equipped with the advanced Himawari imager (AHI), which has 16 observation channels covering visible, near-infrared and thermal infrared ranges [26]. The central wavelengths of these channels for detecting various objects are shown in Table 1. Additionally, AHI can provide high-resolution observation data for the entire disk at intervals of 10 minutes, and its high temporal resolution data is advantageous for target detection and analysis [27]. The open accessibility of Himawari-8/AHI data, provided by the Japan Aerospace Exploration Agency, empowers researchers to extensively leverage this data for scientific inquiry and innovation. This ease of access and the widespread availability of this data have propelled its applications across the fields of Earth sciences, meteorology and environmental studies. For instance, it plays a pivotal role in monitoring natural disasters such as storm tracking, wildfire detection and climate change observations. Moreover, these datasets offer rich information for fields like agriculture, water resource management and urban planning.

As shown in Figure 3, the data retrieved from the Himawari-8 satellite comes in the network common data form (NetCDF) format, requiring preprocessing before utilization. Initially, the NetCDF format will undergo conversion into physical variables. Channels 1 to 6 will denote reflectance data, while channels 7 to 16 will contain brightness temperature data. Following this, a sequence of geometric corrections will be applied, opting for the Geographic WGS84 Projection, succeeded by cropping procedures.

Moreover, different features may have varying measurement units or numerical ranges. Normalization can alleviate these effects, enabling the model to treat each feature more evenly. The normalization formula is as follows:

$$Y = \frac{X - X_{\text{mean}}}{X_{\text{std}}} \quad (1)$$

Where the mean and standard deviation of  $X$  are indicated by  $X_{\text{mean}}$  and  $X_{\text{std}}$ , respectively.  $X$  and  $Y$  represent the initial and the normalized feature values, respectively.



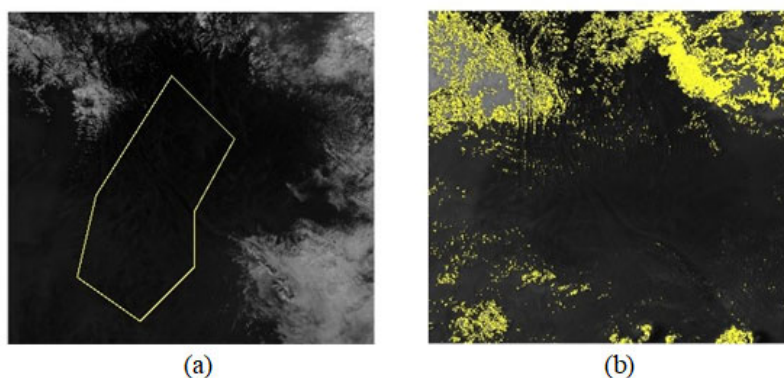
**Figure 3.** Data preprocessing.

**Table 1.** AHI channel characteristics.

Channel	Central wavelength/ $\mu\text{m}$	Detection targets
1	0.46	aerosol, coastline
2	0.51	phytoplankton
3	0.64	vegetation, aerosol
4	0.86	stratus cloud
5	1.6	cloud top phase, snow
6	2.3	surface, cloud, snow
7	3.9	surface, wildfire
8	6.2	upper atmospheric water vapor, rainfall
9	7.0	mid-level atmospheric water vapor, rainfall
10	7.3	sulfur dioxide
11	8.6	water, sulfur dioxide, rainfall
12	9.6	ozone, airflow, wind
13	10.4	surface, cloud
14	11.2	surface, cloud
15	12.4	water, sea surface temperature
16	13.3	temperature, cloud altitude, cloudiness

### 3.3. Cluster feature analysis

The spectral data from the Himawari-8 satellite is analyzed using PDF to generate PDF distributions for reflectance, brightness temperature and brightness temperature difference of pixels corresponding to both clouds and the underlying surface. The selection of cluster features is completed based on the analysis of these PDF distributions. As shown in Figure 4, a random sample of cloud and underlying surface pixels from a series of RS images is used for statistical analysis.



**Figure 4.** (a) Selection of cloud pixels; (b) selection of underlying surface of pixels.

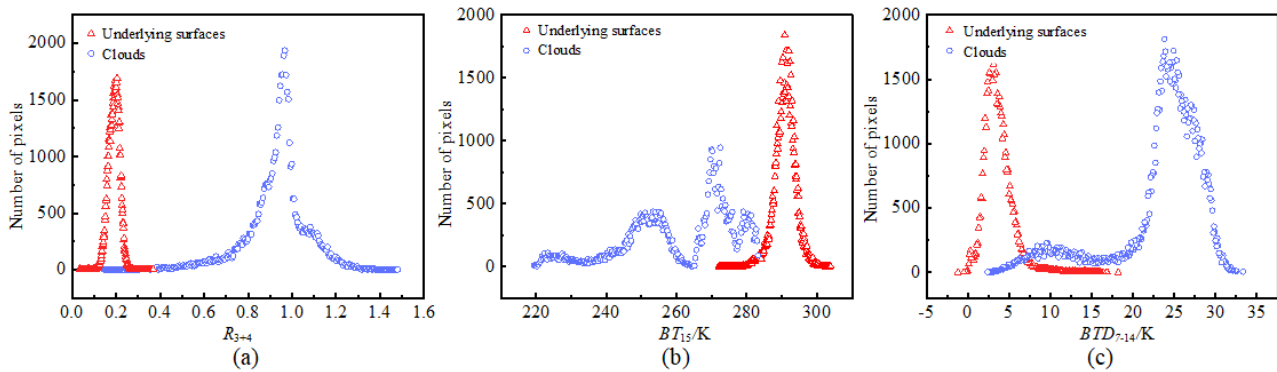
Figure 5 shows the PDF distributions of different cluster features for cloud and underlying surface pixels. The PDF distributions for three cluster features,  $R_{3+4}$ ,  $BT_{15}$  and  $BTD_{7-14}$ , exhibit some overlap for both cloud and underlying surface pixels. As shown in Figure 5(a), the PDF distribution of  $R_{3+4}$  for both cloud and underlying surface pixels roughly follows a normal distribution. Cloud PDF mainly ranges from 0.8 to 1.3, with a peak around 1.2, while the underlying surface PDF predominantly ranges

from 0.2 to 0.35, with a peak around 0.3.

Figure 5(b) displays the PDF distribution of  $BT_{15}$  for clouds and underlying surface pixels. Cloud  $BT_{15}$  PDF is mainly distributed between 220 K and 285 K, displaying two peaks near 250 K and 270 K, respectively. In contrast, the underlying surface  $BT_{15}$  PDF mainly ranges from 280 K to 300 K, with a peak of around 290 K.

Figure 5(c) demonstrates that the PDF distributions of  $BTD_{7-14}$  for both cloud and underlying surface pixels also roughly follow a normal distribution. Cloud  $BTD_{7-14}$  PDF mainly ranges from 20 K to 30 K, with a peak around 23 K, while the underlying surface  $BTD_{7-14}$  PDF mainly ranges from 0 K to 7 K, with a peak around 2 K.

Although there are significant differences in the  $R_{3+4}$  PDF distributions between clouds and the underlying surface, during nighttime periods, the visible light band of AHI struggles to capture clear visible light images. Therefore, in this work, we use different cluster feature schemes for cloud detection during day and night periods:  $R_{3+4}$  during the day and  $BT_{15}$  and  $BTD_{7-14}$  at night.



**Figure 5.** (a) PDF distribution of cloud and underlying surface pixel  $R_{3+4}$ ; (b) PDF distribution of cloud and underlying surface pixel  $BT_{15}$ ; (c) PDF distribution of cloud and underlying surface pixel  $BTD_{7-14}$ .

## 4. Cloud detection algorithm

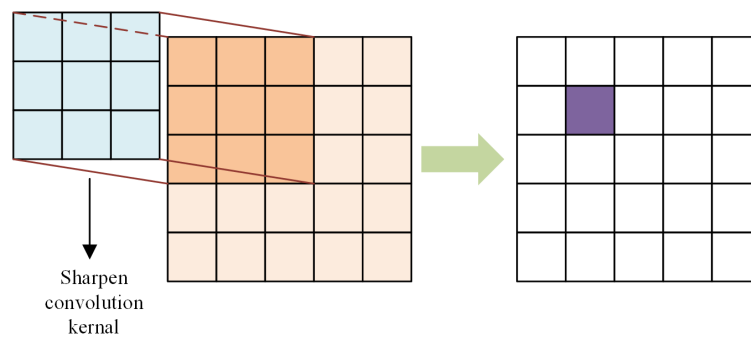
### 4.1. Feature enhancement module

During the process of cloud detection, cloud edges (the overlapping parts of PDFs for various cluster features in Figure 5) are prone to being misclassified as non-cloud pixels. This is primarily due to the minimal spectral feature differences between cloud edges and the underlying surface. Additionally, broken clouds are often sparse and scattered, causing their detection to be challenging. Therefore, as shown in Figure 6, to enhance the algorithm's efficiency in detecting cloud edges and broken clouds, convolution operations are employed to enhance the spectral features of cloud edges and broken clouds. The convolution operation involves sliding the sharpening convolution kernel over each pixel of the original image and computing the weighted sum of the convolution kernel with the pixel values. The formula for this enhancement is as follows:

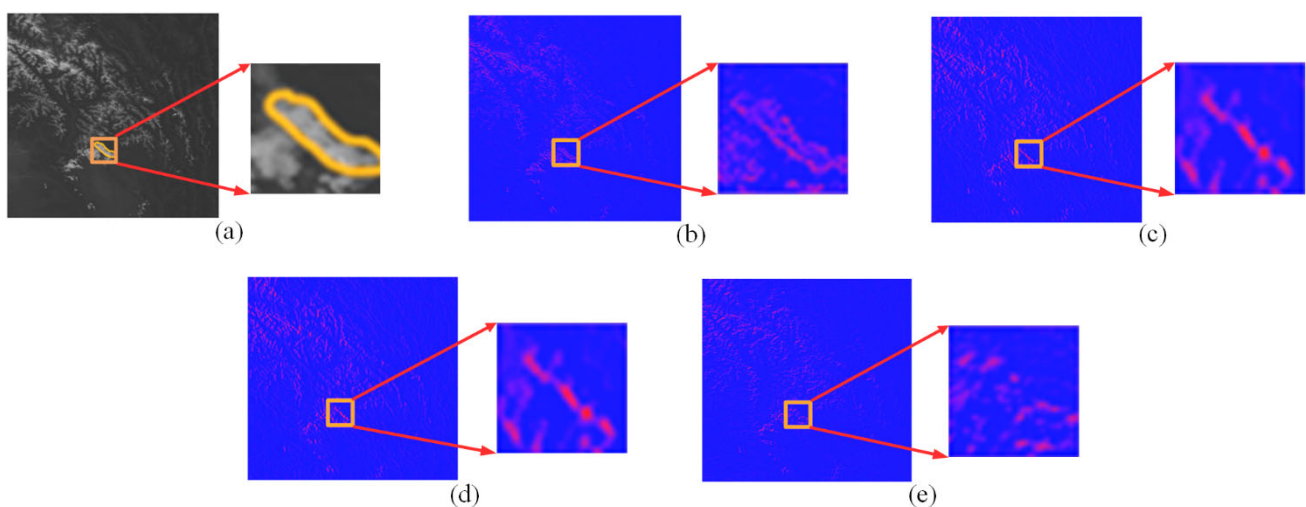
$$g(x, y) = f(x, y) * w(i, j) \quad (2)$$



In the equation,  $f(x, y)$  represents the pixel value at  $(x, y)$ , and  $w(i, j)$  represents the convolution kernel.



**Figure 6.** Convolution operation.



**Figure 7.** (a) Original image; (b) Laplacian operator; (c) Roberts operator; (d) Prewitt operator; and (e) Sobel operator.

The Laplacian operator, often used in image processing, plays a crucial role in extracting gradient information from images [28,29]. It accentuates abrupt changes in pixel values, making it particularly useful in highlighting boundaries and fine details, such as edges of clouds or broken clouds. It is essentially a filter that amplifies high-frequency components, aiding in the detection and emphasis of subtle variations in the image. Compare the sharpening effects of the Laplacian operator with commonly used operators. As depicted in Figure 7, the sharpening results obtained using Roberts, Prewitt and Sobel operators exhibit missing cloud edges, whereas the sharpening results at the edges using the Laplacian operator demonstrate continuity. Therefore, the Laplacian operator is chosen to enhance the features of cloud edges and broken clouds. The definition of the Laplacian operator is as follows:

$$\nabla^2 f = \frac{\partial^2 f}{\partial^2 x} + \frac{\partial^2 f}{\partial^2 y} \quad (3)$$

By separately taking the second derivatives of the Laplacian operator in the  $x$  and  $y$  directions through differencing, we obtain:



$$\nabla^2 f = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y) \quad (4)$$

The above formulas can be transformed into the form of convolution kernels:

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad (5)$$

After the Laplacian operator processing, the spectral features of cloud edges and broken clouds are enhanced. However, the extensive areas of thick clouds that manifest as low-frequency signals are removed. This can lead to sample imbalance and affect clustering results. Therefore, overlaying the image processed with the Laplacian operator onto the original image is crucial. The resulting feature-enhanced image can highlight cloud edges and broken cloud details while preserving the thick cloud portions from the original image. The overlay formula is as follows:

$$g(x, y) = f(x, y) * w(i, j) = f(x, y) - [\nabla^2 f] \quad (6)$$

Finally, based on Eq (6), we obtain the convolution kernel used in this paper:

$$w(i, j) = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix} \quad (7)$$

Figure 8 provides a comparison of the results before and after feature enhancement. From Figure 8(a),(b), a noticeable improvement in the clarity of cloud edges and broken clouds that is within the RS image is observed after the feature enhancement. Additionally, the  $R_{3+4}$  values for cloud edges and broken clouds are extracted from the image for PDF comparison. Figure 8(c) shows that feature enhancement significantly increases the  $R_{3+4}$  values for cloud edges and broken clouds. This will effectively reduce the occurrence of false negatives in the algorithm for cloud edge and broken cloud pixels.

## 4.2. GMM cloud detection algorithm

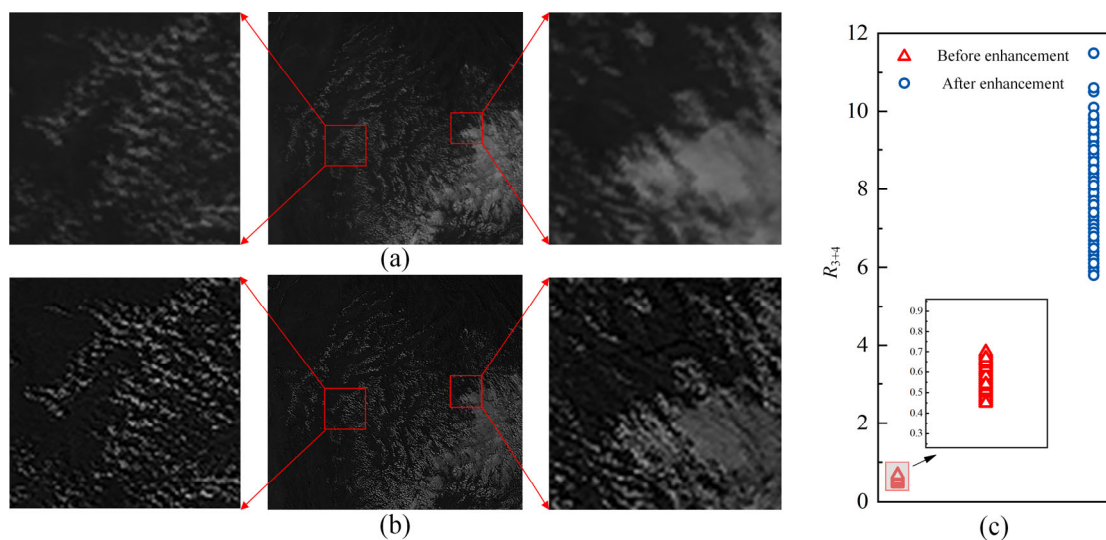
### 4.2.1. GMM clustering basic principle

During detection, the diversity of cloud and the underlying surface results in complex distributions of reflectance and brightness temperature. GMM can represent the PDF of data by the way of using a convex combination of multiple Gaussian functions (a linear combination of multiple Gaussian functions). Given a sufficient number of samples, GMM can model distributions of any shape [30]. Therefore, GMM can accurately describe the probability distributions of reflectance and brightness temperature for both clouds and the underlying surface. Additionally, some spectral characteristics of clouds and underlying surfaces may overlap (as shown in Figure 5). The advantage of GMM compared to clustering algorithms like K-means and mean-shift lies in its more flexible modeling capability. Unlike K-means and other hard clustering methods, GMM has the ability to perform soft clustering which allows data points to belong to different clusters with certain probabilities. This flexibility helps

in better understanding and representing relationships between data points, making the model more adaptable to clusters that exhibit overlap or have fuzzy boundaries. Additionally, GMM uses probability distributions to describe clusters, which makes it better suited for handling complex data distributions, especially when dealing with non-spherical distributions where GMM often outperforms other algorithms [31]. Its PDF formula is as follows:

$$P(x|\theta) = \sum_{j=1}^K \beta_j f(x|\mu_j, \sigma_j^2) \quad (8)$$

In the equation,  $x$  represents a vector  $X = \{x_1, x_2, x_3, \dots, x_n\}$  consisting of  $n$  sample values, where  $X$  contains  $K$  categories. The weight of the  $j$ th class is denoted by  $\beta_j$ , and the function  $f(x)$  is the PDF of a single Gaussian model with mean  $\mu_j$  and variance  $\sigma_j$  for the  $j$ th class.



**Figure 8.** (a) Original image; (b) sharpened image; and (c) comparison of  $R_{3+4}$  before and after sharpening.

#### 4.2.2. EM algorithm for GMM parameter estimation

The set of data for the study area,  $R_{3+4}$ ,  $BT_{15}$  and  $BT_{D7-14}$ , is denoted as  $X = \{x_1, x_2, x_3, \dots, x_n\}$ , where  $x_i$  is a sample drawn from the set  $X$ . The elements in the set are independent of each other and follow a GMM. As shown in Eq (7), the GMM density function contains a parameter set  $\theta = \{\beta, \mu, \sigma^2\}$  and this parameter set is defined as  $\theta$ . For a single Gaussian model, the parameters  $\theta$  can be estimated using the Maximum Likelihood (ML) method, where the likelihood function is given by the PDF:

$$L(\theta) = \prod_{i=1}^n P(x|\theta) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2}\right) \quad (9)$$

Similarly, the likelihood function can be derived from the GMM PDF as follows:

$$L(\theta) = \prod_{i=1}^n P(x_i | \theta) = \sum_{i=1}^n \log \left[ \sum_{j=1}^K \beta_j f(x_i | \mu_j, \sigma_j^2) \right] \quad (10)$$

In the equation, there is a complex nonlinear relationship between the likelihood function  $L(\theta)$  and the parameter set  $\theta$ , making it impractical to directly maximize the likelihood function through differentiation. The Expectation-Maximization (EM) algorithm is a common method for maximum likelihood estimation and is suitable for data with hidden, unknown variables. It can appropriately estimate the optimal parameters of probability models [32,33]. The specific steps of the EM algorithm are as follows:

Step 1: Initialize the parameter set  $\theta = \{\beta, \mu, \sigma^2\}$  for each category (hidden variables) based on prior knowledge.

Step 2: Calculate the probability that the data is generated by a Gaussian distribution as follows:

$$\gamma(i, j) = \frac{\beta_j P(x_i | \mu_j, \sigma_j^2)}{\sum_{j=1}^K \beta_j P(x_i | \mu_j, \sigma_j^2)} \quad (11)$$

Step 3: Update the  $\mu$  and  $\sigma^2$  for each Gaussian model as follows:

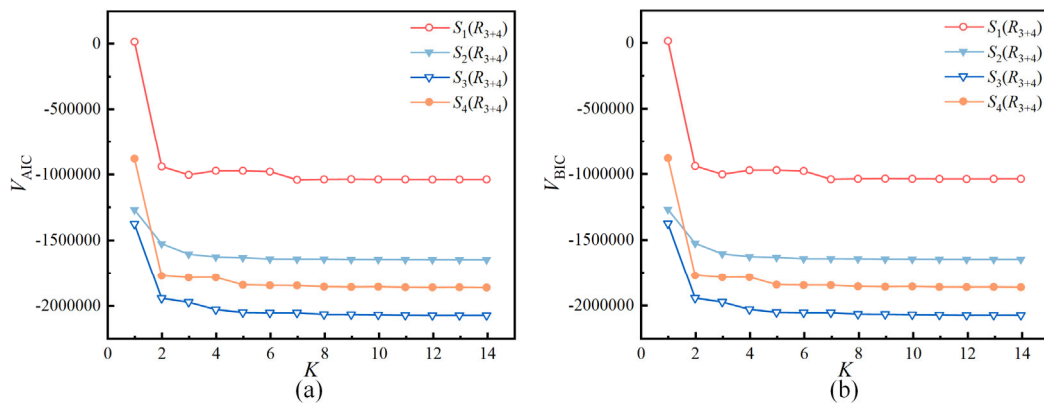
$$\mu_j = \frac{\sum_{i=1}^n \gamma(i, j) x_i}{\sum_{i=1}^n \gamma(i, j)} \quad (12)$$

$$\sigma_j^2 = \frac{\sum_{i=1}^n \gamma(i, j) (x_i - \mu_j)(x_i - \mu_j)^T}{\sum_{i=1}^n \gamma(i, j)} \quad (13)$$

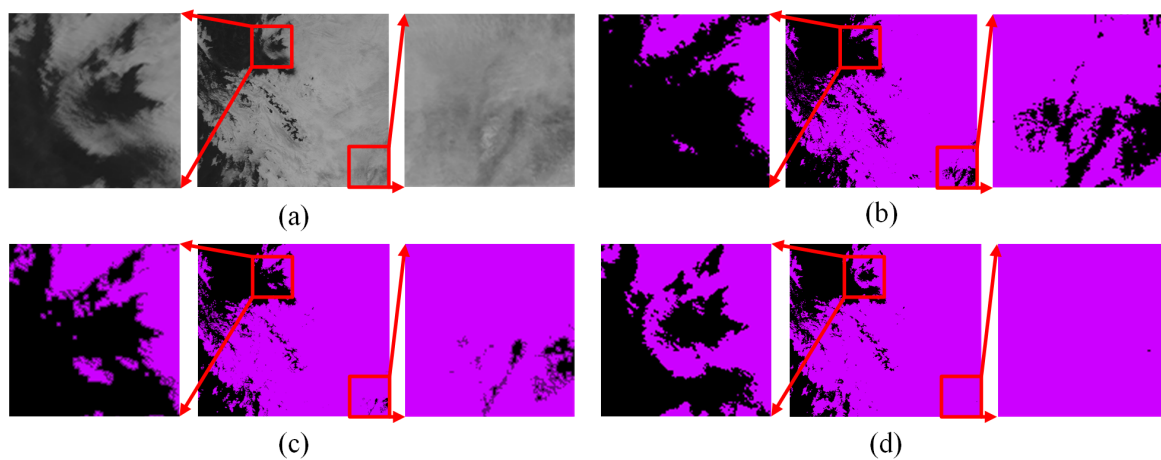
Step 4: Update the  $\beta$  for each Gaussian model as follows:

$$\beta_j = \frac{\sum_{i=1}^n \gamma(i, j)}{n} \quad (14)$$

Repeat Steps 2 through Step 4 until the maximum likelihood value  $L(\theta)$  approaches stability. Once this condition is met, the algorithm is complete.



**Figure 9.** (a) Variation curve of  $V_{AIC}$  with increasing  $K$ ; (a) Variation curve of  $V_{BIC}$  with increasing  $K$ .



**Figure 10.** (a) Visible light image; (b) cloud detection effect when  $K = 5$ ; (c) cloud detection effect when  $K = 6$ ; and (d) cloud detection effect when  $K = 7$ .



**Figure 11.** (a) Cloud detection effect when  $K = 7$ ; (b) Cloud detection effect when  $K = 8$ .

#### 4.2.3. Setting the value of $K$

Before applying the GMM algorithm for clustering, the value of  $K$  needs to be determined. In practical situations, the types of distributions contained in the data samples are unknown. As the number of sub-models in the observed samples increases, the mixture model better reflects the characteristics of the data samples. However, this also leads to a more complex model. A balance point should be established to maintain the simplicity of the mixture model and ensure the accuracy of the

model fit. Therefore, how to determine the value of parameter  $K$  to ensure that the mixture model is both accurate and concise is a primary goal to optimize the model. We use the AIC and BIC criterions from information statistics to determine the parameter  $K$  and their expressions are as follows:

$$\begin{aligned} V_{\text{AIC}} &= 2m - \ln L(x|K, \beta, \mu, \sigma^2) \\ V_{\text{BIC}} &= m \ln n - \ln L(x|K, \beta, \mu, \sigma^2) \end{aligned} \quad (15)$$

In the equations,  $m$  is the number of parameters in the mixture model,  $n$  is the number of samples, and  $\ln L(x|K, \beta, \mu, \sigma^2)$  represents the maximum likelihood value. The minimum value of  $V_{\text{AIC}}$  and  $V_{\text{BIC}}$  is taken as the criteria for selecting the optimal model.

First, four sets of RS images of  $R_{3+4}$  are selected as sample data, and the range for selecting  $K$  is set between 1 and 14.  $V_{\text{AIC}}$  and  $V_{\text{BIC}}$  are calculated for different values of  $K$ . As shown in Figure 9, with the increase of  $K$ , both  $V_{\text{AIC}}$  and  $V_{\text{BIC}}$  show an overall decreasing trend, and the trend gradually stabilizes when  $K > 8$ . To further determine the specific value of  $K$ , values of  $K$  are set to 5, 6, 7 and 8, and the GMM cloud detection results for different  $K$  values are compared to determine the appropriate value of  $K$ . Figure 10 shows the comparison of cloud pixel omissions for  $K = 5, 6$  and  $7$ . As the increase in the value of  $K$ , the number of omitted cloud pixels gradually decreases. When  $K = 7$ , the algorithm produces detection results for various types of clouds that closely align with the actual cloud coverage. Furthermore, as shown in Figure 11, there is tiny little difference in the detection results of cloud edge details between  $K = 7$  and  $K = 8$ . Therefore,  $K = 7$  ensures that the mixture model is both accurate and concise.

## 5. Results and discussion

### 5.1. Datasets and experimental platform

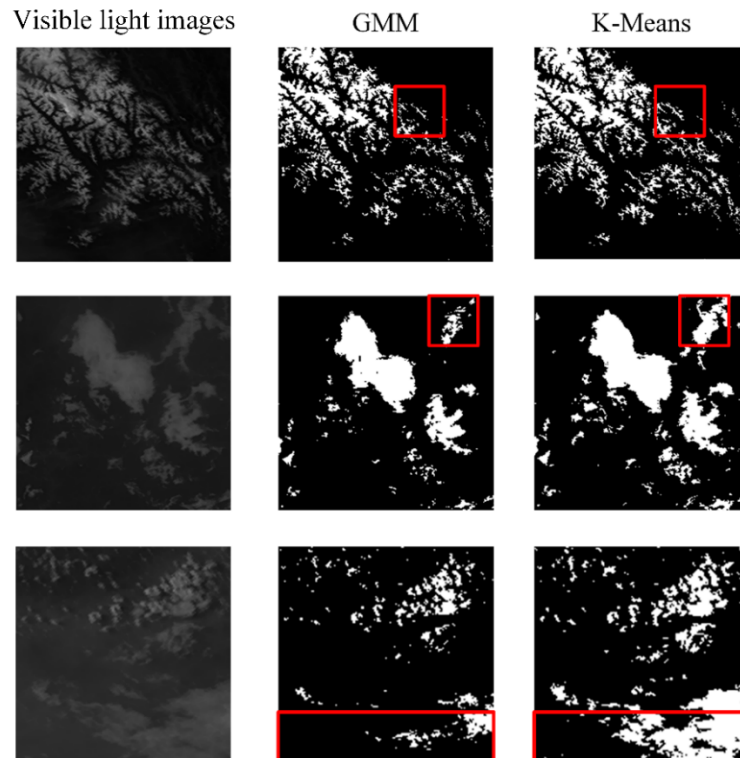
The Himawari-8 data was provided by the Japan Aerospace Exploration Agency (<https://www.eorc.jaxa.jp/ptree/>). In this study, a dataset was constructed for AHI, encompassing partial data from 2021 and 2022. This dataset includes both full disk images and study area images, with over 1,400,000 pixels annotated.

The computer used for the experiments has an AMD R5 5600G processor with a base frequency of 2900 MHz, a 1 TB solid-state drive and 16 GB of RAM. The experiment uses the OpenCV and Scikit-Learn packages to implement the proposed algorithm.

### 5.2. Comparison of clustering algorithms

The GMM algorithm selected in this study requires pre-determination of the number of clusters, similar to the K-means algorithm. Therefore, to demonstrate the advantage of GMM's soft clustering capability over K-means, a comparison was made. The selected images were taken during early morning or evening when the reflectance of clouds is lower compared to daytime, posing a significant challenge to the clustering algorithms. As shown in Figure 12 within the red box, K-means wrongly classified many low-reflectance clouds as underlying surfaces. This is because K-means tends to rigidly assign data points to the nearest cluster center when dealing with non-convex shapes or overlapping regions. Consequently, some low-reflectance clouds were erroneously categorized as land features by K-means. On the other hand, GMM is capable of better reflecting the probability of data

points belonging to different clusters. Therefore, in such scenarios, GMM might more accurately identify areas with ambiguous boundaries. It excels in capturing complex data distributions and delineating regions with unclear boundaries, which might lead to a more precise identification of low-reflectance clouds compared to K-means.



**Figure 12.** Comparison of clustering algorithms.

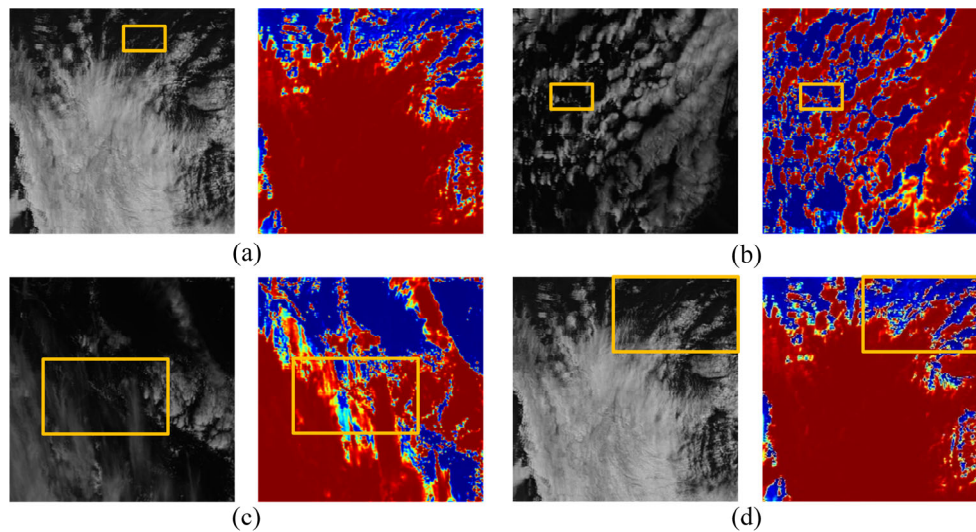
### 5.3. Analysis of cloud edges and broken clouds detection results

The proposed algorithm uses a feature enhancement module to enhance the features of cloud edges and broken clouds, and it integrates the detection results of cloud edges and broken clouds in the final detection results, thereby improving the overall accuracy of the algorithm. Figure 13 shows the heatmap of the proposed algorithm, which reflects the aggregation of broken clouds and cloud edges with their respective categories. In Figure 13(a),(b), the model's attention to broken clouds is evident (highlighted in yellow boxes). Figure 13(b),(c) depict the attention to cloud edges, and it can be observed that the cloud edges of thin clouds also receive significant attention (highlighted in yellow boxes).

### 5.4. Cloud detection results comparison

Traditional methods such as the multi-spectral threshold method are widely used, and machine learning methods like RF, KNN and GMM have efficient classification capabilities. Therefore, this paper implemented the proposed algorithm and the aforementioned algorithm and implemented cloud detection experiments on Himawari-8 images from different time phases and regions to validate the

applicability of the proposed algorithm.



**Figure 13.** (a) and (b) Analysis of broken clouds detection results; (c) and (d) analysis of cloud edges detection results.

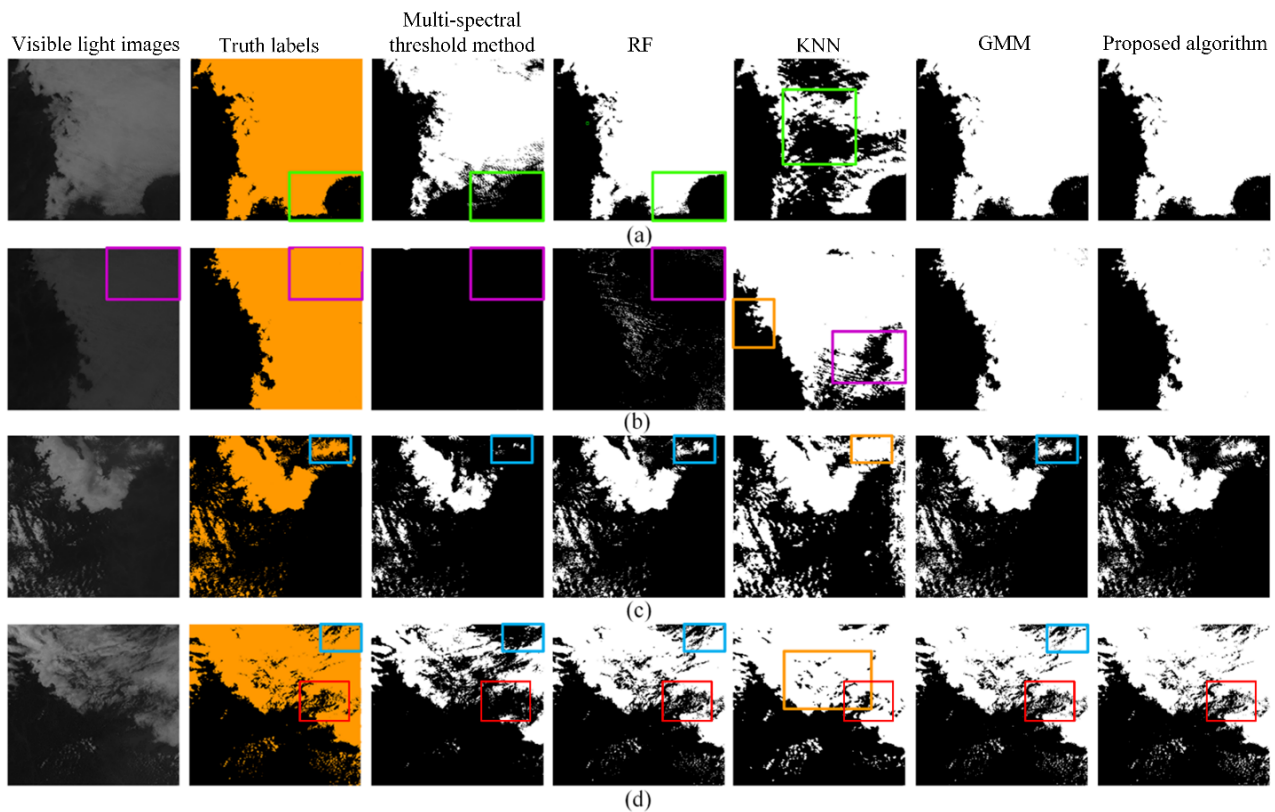
Figure 14 presents the comparison of cloud detection results using the multi-spectral threshold method [10], RF [15], KNN [34], GMM and the proposed algorithm. In the figure, the true labels obtained through visual interpretation are depicted in yellow, while the detection results are shown in white rendering. Professionally trained experts or analysts, following the cloud definition method outlined in Section 3.1, utilized Photoshop software to meticulously annotate the cloud's location pixel by pixel.

In Figure 14(a), the detection results of GMM and the proposed algorithm are the most accurate, while the multi-spectral threshold method and RF exhibit a slight underestimation of cloud pixels, primarily at the cloud edges. KNN shows the highest number of cloud pixel omissions (highlighted within the green bounding boxes in the images). In Figure 14(b), GMM and the proposed algorithm provide the best detection results, with the multi-spectral threshold method and RF exhibit the greatest number of cloud pixels omission. KNN exhibits a large area of omissions in the lower right corner of the image and it also misclassifies some non-cloud pixels as clouds at the cloud edges (highlighted within the purple bounding boxes for omissions and the yellow bounding boxes for misclassifications). In Figure 14(c),(d), the cloud edges in the upper right corners of the images are somewhat blurred. Upon comparison, it is evident that the proposed algorithm produces the most accurate detection results. Moreover, the multi-spectral threshold method, RF and GMM all exhibit varying degrees of omissions at the cloud edges. Multi-spectral threshold method has the most pronounced omission phenomenon, followed by RF (highlighted within the blue bounding boxes for omissions and the yellow bounding boxes for misclassifications).

For the detection of broken clouds, as depicted within the red box in Figure 14(d), the multi-spectral threshold method exhibits the highest number of missed broken cloud pixels, followed by RF and GMM. Additionally, KNN misclassifies numerous pixels as clouds. Analysis of the above comparison results reveals that in sufficient sunlight conditions during the day, the reflectance of clouds in the visible light spectrum is relatively high. During this time, the multi-spectral threshold method is more suitable. However, as it approaches evening, decreasing light levels can lead to higher



threshold values, which is not conducive to cloud detection. RF and KNN, as supervised learning algorithms, have high requirements for sample size and the balance of samples among different categories. However, the small and dispersed nature of scattered clouds poses challenges in creating training samples for RF and KNN, making it difficult for these algorithms to learn the necessary features effectively.



**Figure 14.** (a) 2021-01-04 15:20; (b) 2021-01-26 17:20; (c) 2021-01-19 14:30; (d) 2021-04-21.

In contrast, as an unsupervised learning algorithm, GMM can automatically learn the inherent relationships between different categories and perform classification without the need for an extensive set of training samples. Because the spectral characteristics of cloud edges are somewhat similar to those of underlying surfaces, the threshold set by the multi-spectral threshold method is relatively high for cloud edge detection. While GMM has several advantages in detecting large-area thick clouds, it does not consider the spectral feature similarity between cloud edges and underlying surfaces. Therefore, the proposed algorithm achieves the best detection results for various cloud types.

Additionally, as shown in Figure 15, full disk RS data from September 24, 2022, at UTC 03:00 are selected for cloud detection. Among the five algorithms, the multi-spectral threshold method and GMM achieve relatively good detection results, but there are instances of cloud underestimation in some regions (highlighted within the green boxes). RF and KNN show varying degrees of cloud false positives (highlighted within the red boxes). The proposed algorithm, an enhancement of GMM, reduces the underestimation of cloud pixels, resulting in the best detection performance among the five algorithms, demonstrating that this algorithm provides reliable results even for large-scale areas.

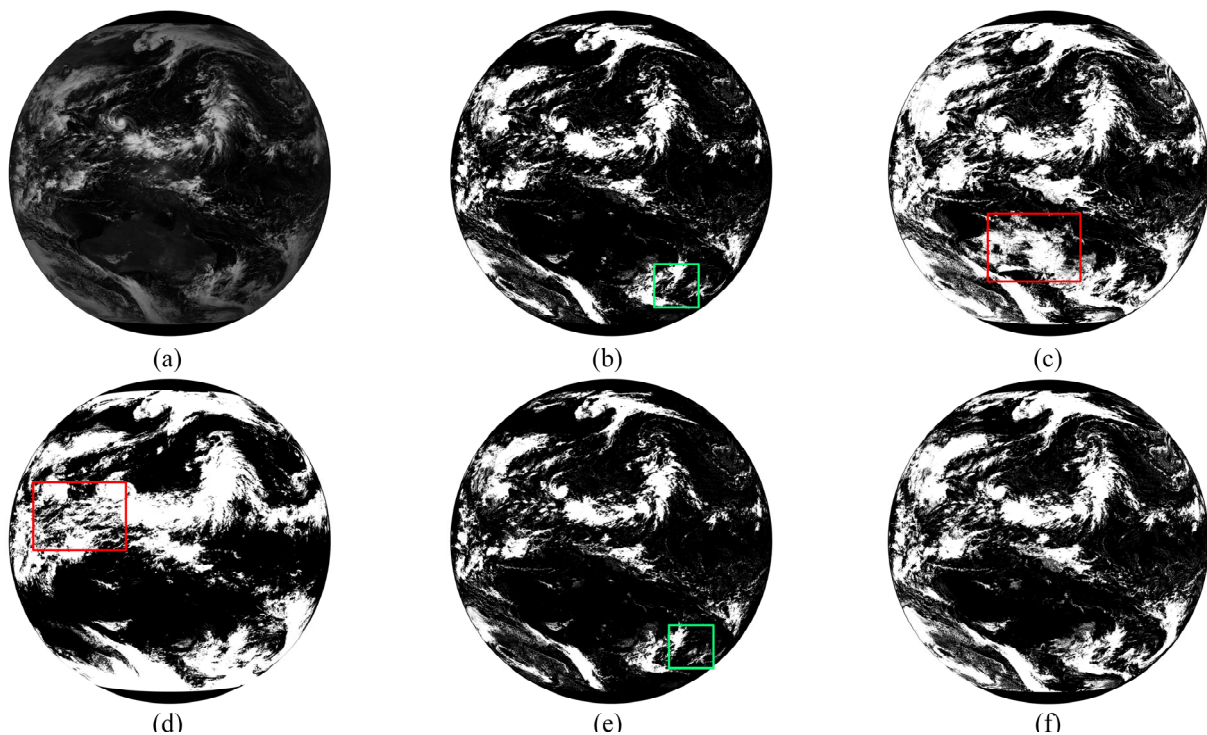
To quantitatively evaluate these methods, this study uses two evaluation metrics, the hit rate ( $HR$ ) and the Hansen-Kuiper skill score ( $KSS$ ), to analyze 15 RS images from different times and regions.

The  $HR$  represents the proportion of correctly identified cloud and clear-sky pixels to the total number of pixels. The  $KSS$  score is used to provide a comprehensive overall measure. The formulas for calculating  $HR$  and  $KSS$  are as follows:

$$HR = \frac{TC + TU}{PA} \quad (16)$$

$$KSS = \frac{TC \times TU - TF \times FT}{(TC + FT)(TU + TF)} \quad (17)$$

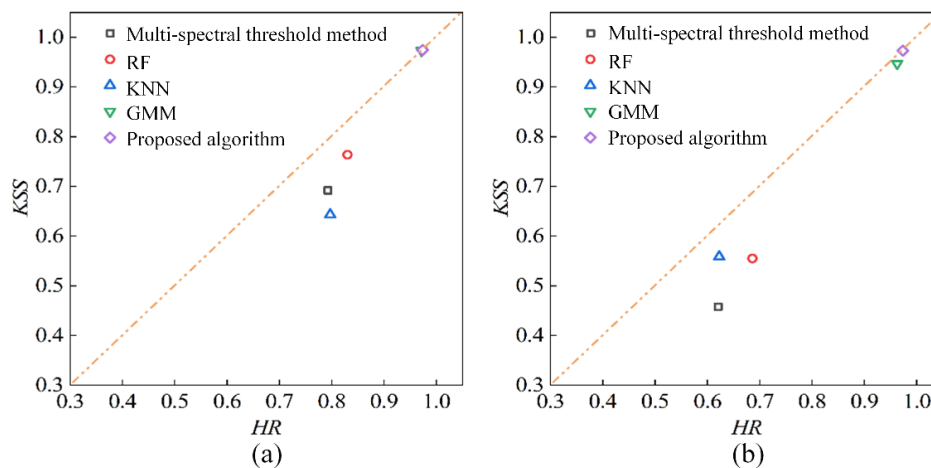
In the formulas,  $TC$  represents the number of pixels correctly detected as clouds,  $TU$  is the number of pixels that are actually non-cloud but are detected as clouds,  $FT$  is the number of pixels that are actually clouds but are detected as non-cloud and  $PA$  is the total number of pixels in the image.



**Figure 15.** (a) Visible light image; (b) detection results of multi-spectral threshold method; (c) detection results of RF; (d) detection results of KNN; (e) detection results of GMM; and (f) detection results of the proposed algorithm.

Figure 16 shows the evaluation results of different algorithms for detecting thick clouds, cloud edges and broken clouds. The values of  $HR$  and  $KSS$  approaching 1 indicate superior algorithm performance. For thick cloud detection, the  $HR$  values of the multi-spectral threshold method, RF, KNN, GMM and the proposed algorithm are 0.793, 0.832, 0.799, 0.974 and 0.976, respectively, and the  $KSS$  values are 0.692, 0.763, 0.642, 0.973 and 0.974, respectively. It is evident that GMM and the proposed algorithm outperform the other three algorithms in thick cloud detection. For cloud edge and broken cloud detection, the  $HR$  values of the five algorithms are 0.619, 0.687, 0.624, 0.963 and 0.974, respectively, and the  $KSS$  values are 0.458, 0.561, 0.563, 0.947 and 0.973, respectively. Compared to

thick cloud detection, the  $HR$  and  $KSS$  values of the multi-spectral threshold method, RF and KNN significantly decrease in cloud edge and broken cloud detection, while those of GMM and the proposed algorithm remain above 0.96 and 0.94, respectively. Moreover, the proposed algorithm, which incorporates the feature enhancement module, shows an improvement of 0.011 in  $HR$  and 0.026 in  $KSS$  compared to GMM. This indicates that the feature enhancement module aids the classification model in extracting deeper cloud edge and broken cloud features. Additionally, as shown in Table 2, 20 images sized at  $256 \times 256$  were selected for quantitative analysis to demonstrate the overall performance of the algorithm. Compared to GMM, the proposed algorithm shows an improvement of 0.002 in  $HR$  and 0.015 in  $KSS$ .



**Figure 16.** (a) Evaluation metrics for thick cloud detection; (b) Evaluation metrics for cloud edge and broken cloud detection.

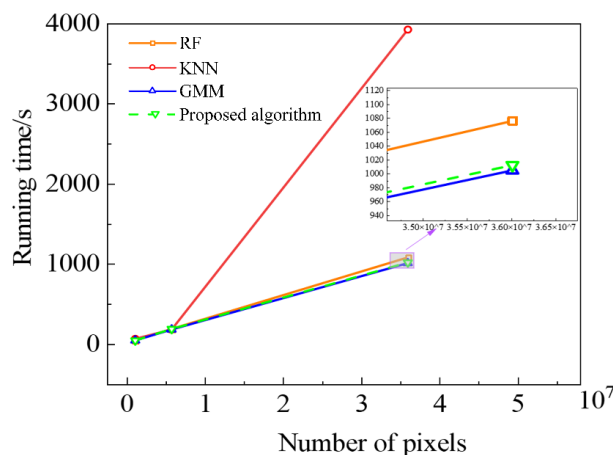
**Table 2.** Evaluation metrics for cloud detection.

Algorithm	$HR$	$KSS$
Multi-spectral threshold method	0.763	0.572
RF	0.811	0.65
KNN	0.753	0.606
GMM	0.973	0.958
Proposed algorithm	<b>0.975</b>	<b>0.973</b>

### 5.5. Algorithm runtime comparison

When performing target detection and task analysis, the vast amount of RS data meant that the runtime of cloud detection algorithms is an important concern. To address this, RS images of sizes  $1000 \times 1100$ ,  $2401 \times 2401$  and  $6001 \times 6001$  are selected to compare the runtime differences between different algorithms. As shown in Figure 17, when the image sizes are  $1000 \times 1100$  and  $2401 \times 2401$ , the runtime differences among the four machine learning algorithms does not vary significantly, with GMM having the shortest runtime, followed by the proposed algorithm, RF and KNN. However, when the image size is  $6001 \times 6001$ , KNN's runtime is significantly longer than the other three algorithms (approximately 3.9 times longer than GMM and 3.65 times longer than RF), while the proposed

algorithm with the feature enhancement module has a runtime only 7.45 seconds longer than GMM. Therefore, for cloud detection in large-sized RS images, the algorithm proposed in this study can maintain high accuracy while having a relatively short runtime.



**Figure 17.** Comparison of running times.

### 5.6. Limitations of the algorithm

While the proposed algorithm generally achieves satisfactory cloud detection results in most scenarios, it encounters errors, particularly when dealing with cloud overlap. This defect is especially obvious when dealing with cloud shadows. These shadows, appearing as dark regions in visible light images, share spectral characteristics similar to the underlying surfaces. The similarity poses a challenge in distinguishing clouds within these shadowed areas, leading to errors in identification. Figure 18 illustrates examples of cloud detection errors specifically within shadowed regions, highlighting numerous omitted pixels in the detection results. In addition, the proposed algorithm is tailored for satellites capable of providing 2 km resolution images. With higher image resolution, the diversity of surface types becomes more intricate, leading to fluctuations in the optimal  $K$  value. In such scenarios, extensive experimental analysis becomes crucial for re-evaluating the selection of the  $K$  value. Even then, pinpointing a stable  $K$  value can be challenging. For instance, in an image observed at 2 km resolution, a pixel  $m(x, y)$  might initially be identified as a cloud. However, as the resolution increases, the area represented by  $m(x, y)$  encompasses more pixels. Among these newly added pixels, some might reveal underlying surfaces that were previously unobservable at the 2 km resolution.

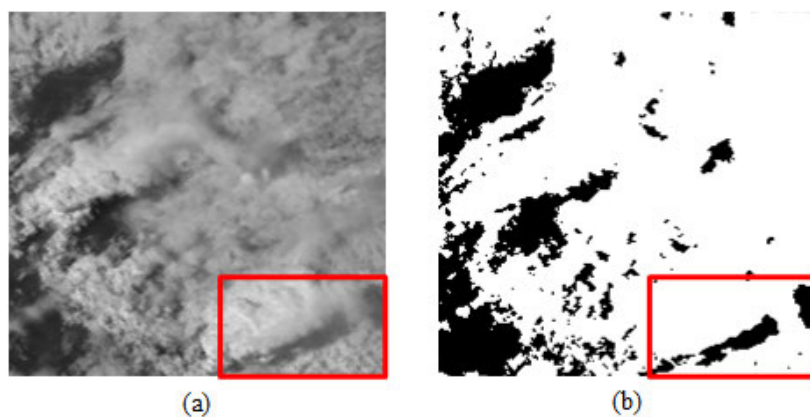
The future research goal is to combine thermal infrared spectroscopy with neural network algorithms to enhance the algorithm's transferability. Moreover, by exploring the imaging correlation between clouds and their shadows, there is potential to calculate the orientation and distance of shadows concerning cloud areas. This computational approach could be helpful in effectively filtering out these shadows from the detection results.

## 6. Ethical considerations and potential societal impacts

The improved cloud detection technology holds significant potential in disaster management. Accurate cloud coverage data is meaningful to monitoring and early warning systems for natural

disasters such as storms and floods, aiding in the proactive implementation of response measures and minimizing potential losses. However, when applying these technologies, the critical importance of data accuracy and timeliness in decision-making must be considered to prevent the potential risks of misinformation or delays.

Furthermore, advancements in cloud detection technology may raise concerns regarding personal privacy, particularly in the monitoring and analysis of extensive areas. Therefore, safeguarding personal information and data privacy should be a crucial consideration when utilizing these algorithms, necessitating the establishment of stringent guidelines for data processing and sharing.



**Figure 18.** Detection errors caused by cloud shadows. (a) Visible light image; and (b) detection result.

## 7. Conclusions

In this paper, we propose a multi-spectral data cloud detection algorithm based on a feature enhancement module and Gaussian mixture model (GMM) clustering. First, the spectral data of clouds and underlying surfaces are statistically analyzed. Based on the analysis results, an appropriate feature scheme is selected to achieve cloud detection throughout the day. Second, the Laplacian operator is used to enhance the spectral features of cloud edges and broken clouds. The enhanced data is then input into a GMM classifier for cloud edge and broken cloud detection. The original RS data is simultaneously subjected to GMM clustering to obtain thick cloud detection results. Finally, the results from the both branches are combined to obtain the final cloud detection result.

Comparing the results of the multi-spectral threshold methods, RF, KNN, GMM and the proposed algorithm, the following conclusions are drawn: The proposed algorithm achieves the highest accuracy with a hit rate ( $HR$ ) of 0.976 for the case of thick cloud detection and a Hansen-Kuiper skill score ( $KSS$ ) of 0.974 for the case of cloud edges and broken cloud detection. This means the proposed algorithm demonstrates significant advantages over other algorithms, with an  $HR$  and  $KSS$  improvement of 0.011 and 0.026, respectively, compared to GMM. Furthermore, the inclusion of the feature enhancement module in the proposed algorithm maintains a relatively short runtime. This further demonstrates that the proposed algorithm can achieve fast and high-precision cloud detection for different types of clouds. Therefore, the proposed algorithm can provide real-time cloud detection results for various tasks such as vegetation change analysis, disaster monitoring and cartography, offering a viable solution to alleviate the challenges posed by cloud interference.



## Use of AI tools declaration

The authors declare that they have not used Artificial Intelligence (AI) tools in the creation of this article.

## Acknowledgments

This work is supported by the Major Scientific and Technological Projects of Yunnan Province: Research on Key Technologies of Ecological Environment Monitoring and Intelligent Management of Natural Resources in Yunnan (202202AD080010), the National Natural Science Foundation of China (52107017) and the Fundamental Research Fund of Science and Technology Department of Yunnan Province (202201AU070172).

## Conflict of interest

The authors declare that there are no conflicts of interest.

## References

1. I. Z. Cetin, H. Sevik, Investigation of the relationship between bioclimatic comfort and land use by using GIS and RS techniques in Trabzon, *Environ. Monit. Assess.*, **192** (2020), 1–14. <https://doi.org/10.1007/s10661-019-8029-4>
2. K. Kak, Satellite remote sensing for disaster management support: A holistic and staged approach based on case studies in Sentinel Asia, *Int. J. Disaster Risk Reduct.*, **33** (2019), 417–432. <https://doi.org/10.1016/j.ijdr.2018.09.015>
3. P. Barmpoutis, P. Papaioannou, K. Dimitropoulos, N. Grammalidis, A review on early forest fire detection systems using optical remote sensing, *Sensors*, **20** (2020), 6442. <https://doi.org/10.3390/s20226442>
4. T. P. P. Sharma, J. Zhang, U. A. Koju, S. Zhang, Y. Bai, M. K. Suwal, Review of flood disaster studies in Nepal: A remote sensing perspective, *J. Disaster Risk Reduct.*, **34** (2019), 18–27. <https://doi.org/10.1016/j.ijdr.2018.11.022>
5. R. R. Girija, S. Mayappan, Mapping of mineral resources and lithological units: A review of remote sensing techniques, *Int. J. Image Data Fusion.*, **10** (2019), 79–106. <https://doi.org/10.1080/19479832.2019.1589585>
6. G. L. Spadoni, A. Cavalli, L. Congedo, M. Munafò, Analysis of normalized difference vegetation index (NDVI) multi-temporal series for the production of forest cartography, *Remote Sens. Appl.*, **20** (2020), 100419. <https://doi.org/10.1016/j.rsase.2020.100419>
7. H. Harde, How much CO<sub>2</sub> and the sun contribute to global warming: Comparison of simulated temperature trends with last century observations, *Sci. Clim. Change*, **2** (2022), 105–133.
8. Q. He, X. Sun, Z. Yan, K. Fu, DABNet: Deformable contextual and boundary-weighted network for cloud detection in remote sensing images, *IEEE Trans. Geosci. Remote Sens.*, **60** (2021), 1–16. <https://doi.org/10.1109/TGRS.2020.3045474>

9. W. Rossow, E. Duenas, The international satellite cloud climatology project (ISCCP) web site: An online resource for research, *Bull. Am. Meteorol. Soc.*, **85** (2004), 167–172. <https://doi.org/10.1175/BAMS-85-2-167>
10. Q. Xiong, Y. Wang, D. Liu, S. Ye, Z. Du, W. Liu, et al., A cloud detection approach based on hybrid multispectral features with dynamic thresholds for GF-1 remote sensing images, *Remote Sens.*, **12** (2020), 450. <https://doi.org/10.3390/rs12030450>
11. P. Li, L. Dong, H. Xiao, M. Xu, A cloud image detection method based on SVM vector machine, *Neurocomputing*, **169** (2015), 34–42. <https://doi.org/10.1016/j.neucom.2014.09.102>
12. W. Zhang, S. Jin, L. Zhou, X. Xie, F. Wang, L. Jiang, et al., Multi-feature embedded learning SVM for cloud detection in remote sensing images, *Comput. Electr. Eng.*, **102** (2022), 108177. <https://doi.org/10.1016/j.compeleceng.2022.108177>
13. H. Ishida, Y. Oishi, K. Morita, K. Moriwaki, T. Y. Nakajima, Development of a support vector machine based cloud detection method for MODIS with the adjustability to various conditions, *Remote Sens. Environ.*, **205** (2018), 390–407. <https://doi.org/10.1016/j.rse.2017.11.003>
14. N. Ghasemian, M. Akhoondzadeh, Introducing two random forest based methods for cloud detection in remote sensing images, *Adv. Space. Res.*, **62** (2018), 288–303. <https://doi.org/10.1016/j.asr.2018.04.030>
15. H. Fu, Y. Shen, J. Liu, G. He, J. Chen, P. Liu, et al., Cloud detection for FY meteorology satellite based on ensemble thresholds and random forests approach, *Remote Sens.*, **11** (2018), 44. <https://doi.org/10.3390/rs11010044>
16. J. Zhang, J. Wu, H. Wang, Y. Wang, Y. Li, Cloud detection method using CNN based on cascaded feature attention and channel attention, *IEEE Trans. Geosci. Remote Sens.*, **60** (2021), 1–17. <https://doi.org/10.1109/TGRS.2021.3120752>
17. F. Xie, M. Shi, Z. Shi, J. Yin, D. Zhao, Multilevel cloud detection in remote sensing images based on deep learning, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, **10** (2017), 3631–3640. <https://doi.org/10.1109/JSTARS.2017.2686488>
18. L. Sun, X. Yang, S. Jia, C. Jia, Q. Wang, X. Liu, et al., Satellite data cloud detection using deep learning supported by hyperspectral data, *Int. J. Remote Sens.*, **41** (2020), 1349–1371. <https://doi.org/10.1080/01431161.2019.1667548>
19. D. A. Reynolds, Gaussian mixture models, *Encycl. Biom.*, **741** (2009), 659–663. [https://doi.org/10.1007/978-0-387-73003-5\\_196](https://doi.org/10.1007/978-0-387-73003-5_196)
20. D. Zhang, C. Huang, J. Gu, J. Hou, Y. Zhang, W. Han, et al., Real-time wildfire detection algorithm based on VIIRS fire product and Himawari-8 data, *Remote Sens.*, **15** (2023), 1541. <https://doi.org/10.3390/rs15061541>
21. Y. Kang, T. Sung, J. Im, Toward an adaptable deep-learning model for satellite-based wildfire monitoring with consideration of environmental conditions, *Remote Sens. Environ.*, **298** (2023), 113814. <https://doi.org/10.1016/j.rse.2023.113814>
22. J. Xia, N. Yokoya, B. Adriano, K. Kanemoto, National high-resolution cropland classification of Japan with agricultural census information and multi-temporal multi-modality datasets, *Int. J. Appl. Earth Obs. Geoinf.*, **117** (2023), 103193. <https://doi.org/10.1016/j.jag.2023.103193>
23. X. Li, Y. Qu, H. Geng, Q. Xin, J. Huang, S. Peng, et al., Mapping annual 10-m maize cropland changes in China during 2017–2021, *Sci. Data*, **10** (2023), 765. <https://doi.org/10.1038/s41597-023-02665-3>



24. C. Huang, N. Thomas, S. N. Goward, J. G. Masek, Z. Zhu, R. G. John, Automated masking of cloud and cloud shadow for forest change analysis using Landsat images, *Int. J. Remote Sens.*, **31** (2010), 5449–5464. <https://doi.org/10.1080/01431160903369642>
25. X. Long, X. Li, H. Lin, M. Zhang, Mapping the vegetation distribution and dynamics of a wetland using adaptive-stacking and google earth engine based on multi-source remote sensing data, *Int. J. Appl. Earth Obs. Geoinf.*, **102** (2021): 102453. <https://doi.org/10.1016/j.jag.2021.102453>
26. B. Chen, J. Hu, Z. Song, X. Zhou, L. Zhao, Y. Wang, et al., Exploring high-resolution near-surface CO concentrations based on Himawari-8 top-of-atmosphere radiation data: Assessing the distribution of city-level CO hotspots in China, *Atmos. Environ.*, **312** (2023), 120021. <https://doi.org/10.1016/j.atmosenv.2023.120021>
27. W. Xu, W. Wang, N. Wang, B. Chen, A new algorithm for Himawari-8 aerosol optical depth retrieval by integrating regional PM<sub>2.5</sub> concentrations, *IEEE Trans. Geosci. Remote Sens.*, **60** (2022), 1–11. <https://doi.org/10.1109/TGRS.2022.3155503>
28. C. Gu, X. Lu, C. Zhang, Example-based color transfer with Gaussian mixture modeling, *Pattern Recognit.*, **129** (2022), 108716. <https://doi.org/10.1016/j.patcog.2022.108716>
29. Y. Zhang, Y. Feng, X. Liu, D. Zhai, X. Ji, H. Wang, et al., Color-guided depth image recovery with adaptive data fidelity and transferred graph Laplacian regularization, *IEEE Trans. Circuits Syst. Video Technol.*, **30** (2019), 320–333. <https://doi.org/10.1109/TCSVT.2018.2890574>
30. P. Johnston, K. Nogueira, K. Swingler, GMM-IL: Image classification using incrementally learnt, independent probabilistic models for small sample sizes, *IEEE Access*, **11** (2023), 25492–25501. <https://doi.org/10.1109/ACCESS.2023.3255795>
31. Y. Li, J. Zhang, Z. Ma, Y. Zhang, Clustering analysis in the wireless propagation channel with a variational Gaussian mixture model, *IEEE Trans. Big Data*, **6** (2018), 223–232. <https://doi.org/10.1109/TBDATA.2018.2840696>
32. Z. Zha, X. Yuan, J. Zhou, C. Zhu, B. Wen, Image restoration via simultaneous nonlocal self-similarity priors, *IEEE Trans. Image Process.*, **29** (2020), 8561–8576. <https://doi.org/10.1109/TIP.2020.3015545>
33. C. Gu, X. Lu, Y. He, C. Zhang, Blur removal via blurred-noisy image pair, *IEEE Trans. Image Process.*, **30** (2020), 345–359. <https://doi.org/10.1109/TIP.2020.3036745>
34. A. Heinle, A. Macke, A. Srivastav, Automatic cloud classification of whole sky images, *Atmos. Meas. Tech.*, **3** (2010), 557–567. <https://doi.org/10.5194/amt-3-557-2010>



AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)