A Method of Identifying Thunderstorm Clouds in Satellite Cloud Image Based on Clustering

Lili He [1, 2], Dantong Ouyang[1, 2], Meng Wang[1, 2], Hongtao Bai[1, 2], Qianlong Yang[1, 2], Yaqing Liu[3, 4] and Yu Jiang[1, 2, *]

Abstract: In this paper, the clustering analysis is applied to the satellite image segmentation, and a cloud-based thunderstorm cloud recognition method is proposed in combination with the strong cloud computing power. The method firstly adopts the fuzzy C-means clustering (FCM) to obtain the satellite cloud image segmentation. Secondly, in the cloud image, we dispose the 'high-density connected' pixels in the same cloud clusters and the 'low-density connected' pixels in different cloud clusters. Therefore, we apply the DBSCAN algorithm to the cloud image obtained in the first step to realize cloud cluster knowledge. Finally, using the method of spectral threshold recognition and texture feature recognition in the steps of cloud clusters, thunderstorm cloud clusters are quickly and accurately identified. The experimental results show that cluster analysis has high research and application value in the segmentation processing of meteorological satellite cloud images.

Keywords: Cloud computing, cluster analysis, FCM, DBSCAN, thunderstorm clouds, satellite cloud image.

1 Introduction

As a powerful engine for artificial intelligence, machine learning is widely favored by researchers. Machine learning has considerable application prospects as an interdisciplinary subject of computational statistics and computer science. The more common applications include, but are not limited to, speech recognition, natural language processing, face recognition, driverlessness, recommendation systems, and risk assessment of the financial system. The clustering analysis is an important analysis method for non-supervised learning in machine learning, which has a wide range of applications in the fields of biomedicine, data mining, and image segmentation. MacQueen [MacQueen (1967)] proposed K-means clustering with the idea of partition. Ruspini [Ruspini (1977)] proposed a fuzzy K-means clustering algorithm by introducing the fuzzy theory. Zhang et al. [Zhang, Ramakrishnan and Livny (1996)] used B-tree to store clustering features and achieved BIRCH

[1] College of Computer Science and Technology, Jilin University, Changchun, 130012, China.

[2] Key Laboratory of Symbolic Computation and Knowledge Engineering, Jilin University, Changchun, 130012, China.

[3] School of Information Science & Technology, Dalian Maritime University, Dalian, 116026, China.

[4] School of Informatics, University of Edinburgh, Edinburgh, EH8 9AB, UK.

[*] Corresponding Author: Yu Jiang. Email: jiangyu2011@jlu.edu.cn.

CMC. doi:10.32604/cmc.2018.03840

hierarchical clustering, an incremental clustering algorithm, through hierarchical methods. CURE [Guha, Rastogi and Shim (1998)] is a clustering that is unaffected by the size and shape of the data. Fully considering the connectivity and similarities between classes, Chameleon [Karypis, Han and Kumar (1999)] (Chameleon Clustering) is divided into two stages of splitting and merging, which can obtain the satisfactory clustering results in terms of small-scale data. Ester et al. proposed DBSCAN clustering [Ester, Kriegel, Sander et al. (1996)]; Ankerst et al. [Ankerst, Breunig, Kriegel et al. (1999)] put forward the OPTICS algorithm and realized density clustering through inter-class sorting; STING algorithm [Wang, Yang and Muntz (1997)] is representative of grid-clustering method. Shekholeslami et al. [Sheikholeslami, Chatterjee and Zhang (2000)] presented WaveCluster algorithm for processing large-scale multidimensional data based on the wavelet transform model; In recent years, many mathematical models have also been applied to cluster analysis. Specifically, Juang et al. [Juang and Rabiner (1990)] established a dynamic Markov chain model for data, and then proposed a dynamic clustering method as the center. Zhang et al. [Zhang, Wang and Zhang (2007)] used a combination of self-organizing neural network and K-center clustering to obtain a new type of integrated clustering algorithm. With the cross-fusion of clustering theory, some new clustering methods are generated, such as Agrawal et al. combined with density clustering and grid clustering to propose the CLIQUE algorithm [Parsons, Haque and Liu (2004)].

The earliest proposed clustering ideas for image segmentation are Coleman and Anderews [Zhang, Ramakrishnan and Livny (1996)]. Up to now, spectral clustering and Mean-Shif algorithms have been successfully used in image segmentation. It is common to apply the graph theory and related heuristic algorithms to image segmentation in recent years. The idea of graphic cutting is used to segment images and an early minimum cutting algorithm is proposed. The Ncut (normalized cut) criterion improves the applicability of spectral clustering in image segmentation. In order to fully consider the relationship between the inner region and the edge of the image, Cox et al. [Cox, Rao and Zhong (1996)] proposed a ratio-domain cutting algorithm. Wang et al. [Wang and Siskind (2001)] optimized the graph cutting rule and proposed a mean-cut algorithm that minimizes the average weight of cutting edges. Since the optimal solution of the objective function needs to be obtained by calculating the feature quantity of the similarity or correlation matrix, the spectral clustering algorithm has high time and space complexity, which is not suitable for processing complex large-scale images. Two improvements have been proposed to solve the above-mentioned drawbacks of spectral clustering. On one hand, the large image is first divided into several small images (the most commonly used is an 8×8 pixel matrix) and then segmented according to the similarity between these small images. On the other hand, for each pixel, only the approximate degree of neighboring pixels in a certain area is determined. Although these two methods increase the computational speed of spectral clustering, the results of image segmentation are relatively rough. For some noise interferences in image segmentation, clustering can also play a good role in elimination and suppression. Krinidis et al. [Krinidis and Chatzis (2010)] improved the FCM algorithm for images containing Gaussian noise or impulse noise and also increased its stability during image processing. The data compression function in clustering algorithm can also be used to remove Gaussian noise, such as K-SVD [Aharon, Elad and Bruckstein (2006)] denoising. Cour et al. [Cour, Benezit and Shi (2005)] implemented the multi-scale parallel solution in the Ncut algorithm and improved the efficiency of large-scale image segmentation; Dhillon et al. [Dhillon, Guan and Kulis (2007)] implemented the multi-layer image segmentation algorithm, which can greatly shorten the processing time.

Although the clustering-based image segmentation algorithm has made great progress both in theory and practice, how to select an appropriate cluster analysis technique to achieve accurate and fast image segmentation based on actual images is a problem in image segmentation research. In the process of exploring the practical application of clustering intelligence algorithms, we found that traditional methods for cloud classification, such as the threshold method and fuzzy C-means method, are used to identify thunderstorm clouds by the characteristics of each pixel. It does not consider the adjacent relationship between pixels in the same cloud area, and can only determine the cloud in the region qualitatively. It is difficult to quantitatively estimate the cloud volume and the cloud area, which has great limitations in practical applications. Therefore, we started with the spectral characteristics, texture features of the thunderstorm cloud, and subsequently identified thunderstorm clouds in satellite cloud image based on clustering. This method selects the stationary meteorological satellite cloud image as a research data object. Because of the complexity and ambiguity of cloud maps, we first utilize the fuzzy C-means (FCM) method to separate the surface and cloud regions. Then we use DBSCAN algorithm to get the basic unit of cloud analysis - cloud cluster. At last, step-by-step spectral threshold recognition and texture feature recognition are used in the cloud cluster as a unit. A cloud-based thunderstorm cloud identification method is proposed to quickly and accurately identify thunderstorm cloud clusters. The results show that the image segmentation method based on clustering can fully extract the cloud information from the satellite cloud image, which can play a better role in the complex large-scale image processing.

2 Cloud-based thunderstorm identification algorithm

2.1 Cloud-ground separation method based on FCM

Some lands or bodies of water (called surface) are directly “naked” in the satellite imagery due to no or less clouds in the sky. However, researchers mainly study the cloud classification, cloud recognition, cloud guidance, etc. in the cloud maps. Therefore, image segmentation in the cloud region and the surface region is essential. In the study of clustering, we found that the use of fuzzy C-means clustering (FCM) in cloud image segmentation can achieve better effect. In this condition, this paper proposes a cloud-ground separation method based on FCM.

For the set of target data objects, FCM considers that each object has a parameter that measures the membership relationship with all the clusters, which is called membership degree. Each object belongs to a cluster with different degrees of membership, and these clusters are fuzzy subsets of the object set. FCM uses a fuzzy classification matrix to store each clustering result.

Let $X = (x_1, x_2, \cdots, x_n) \subseteq R^p$ be the set of target data objects, $c(2 \leq c \leq n)$ be the number of classes divided by the set, and $\mathrm{Y} = \{X_i | i = 1,2,\cdots,c\}$ be the partition in the clustering process, which satisfies the conditions: $X_1 \cup X_2 \cup \cdots X_c = X$, $X_i \cap X_j = \emptyset$, $1 \leq i \neq j \leq c$. Then the fuzzy classification matrix of Y is divided into:

$$U_Y = \begin{bmatrix} U_1 \\ U_2 \\ \vdots \\ U_c \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ u_{21} & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ u_{c1} & u_{c2} & \cdots & u_{cn} \end{bmatrix}$$

The degree of membership of the object x_j with respect to the cluster X_i is denoted by u_{ij}, and u_{ij} satisfies:∀j， $0 \leq u_{ij} \leq 1$， $\sum_{i=1}^{c} u_{ij} = 1$， $0 \leq \sum_{j=1}^{n} u_{ij}$.

Let $V = \{V_1, V_2, \cdots, V_n\}$ denote the clustering center, $V_i \in R^p$, $1 \leq i \leq c$, then FCM clustering objective function formula is formula (1).

$$J_\lambda(U,V) = \sum_{j=1} \sum_{i=1} u_{ij}^{\lambda} d_{ij}^{\lambda}, \quad 1 \leq \lambda \leq c \tag{1}$$

Where λ represents the degree of blur index. By changing the size of λ, the ambiguity of the clustering result can be controlled. If λ is reduced, the division of the class and the boundary will become more obvious. Regarding the data object with the form of a super-spherical shape, λ generally takes 2 to represent the Euclidean distance between the sample point Xj and the i-th center Vi.

In order to obtain the best fuzzy c-segmentation of the sample set X, it is necessary to cyclically adjust the cluster center and the membership matrix continuously to finally get the solution (U,V) under $min\{J_\lambda(U,V)\}$ constraint.

Usually, the Lagrange multiplier method is used to solve the optimal problem of the objective function $J_\lambda(U,V)$. The membership degree and the cluster center can be obtained according to formulas (2) and (3) respectively.

$$u_{ij} = 1 / \sum_{k=1} \left(d_{ij}^2 / d_{kj}^2 \right)^{1/(\lambda-1)}, \quad 1 \leq i \leq c, \quad 1 \leq j \leq n \tag{2}$$

$$V_i = \sum_{k=1} u_{ik}^{\lambda} X_k / \sum_{k=1} u_{jk}^{\lambda}, \quad 1 \leq i \leq c \tag{3}$$

The choice of the number of clusters will affect the division criteria based on FCM image segmentation, resulting in different cloud segmentation effects. After reviewing a large amount of relevant data, it is found that researchers generally classified the object types in the cloud image into eight categories: land, water, cumulonimbus, concentrating cloud, cirrus cloud, mid-cloud, pale cloud, and low cloud. Hence the number of divisions of FCM clustering is set to 8.

Improper selection of cluster centers will not only reduce the efficiency of clustering, but also lead to more rough the clustering results. The initial cluster centers of FCM in this paper adopts the general values of the above eight surface types and cloud types (The actual value fluctuates around this value). The grayscale eigenvalues of infrared (IR), water vapor (WV), and visible light (VS) are selected to establish the statistical characteristics of grayscale characteristics of these eight types of samples (the average of several samples), as shown in Tab. 1.

Table 1: Statistical characteristics of grayscale features of cloud samples

	Water	Land	Low Cloud	Pale Cloud	Mid-cloud	Cirrus Cloud	Concentr-ating Cloud	Cumuloni-mbus
IR	96.855	96.972	134.76	116.05	157.14	1727	191.44	216.61
WV	194.17	182.02	214.90	193.52	214.63	2168	223.53	237.36
VS	18.313	41.212	43.539	50.189	54.161	60.8	64.757	75.827

The number of clusters is determined to be 8 and the sample grayscale feature statistics are used as the initial center of FCM clustering to constrain and guide the clustering process. The clustering center and the membership matrix are continuously adjusted during the clustering iteration process, and the clustering result of the two-dimensional feature space is finally obtained. The specific FCM algorithm is as follows:

Algorithm 1 FCM algorithm

Input: number of clusters **c=8**, number of data: **n**; fixed value **λ**,1⩽**λ**⩽∞; threshold ε

Output: membership matrix ***U'***

Step 1: The average feature values of the surface and typical cloud samples are assigned to $\boldsymbol{V_0}$;

Step 2: Let the cycle number ***b*=0**, and use the formula in Step 5 to assign the initial value to the membership matrix $\boldsymbol{U}^{(0)}$;

Step 3: ***b***++;

Step 4: Calculate *c* cluster centers $V_i^{(b)}$(i= $1,2,\cdots,c$). The formula is (4).

$$V_i = \left(\sum_{j=1}^{n} u_{ij}^{\lambda} X_j\right) / \left(\sum_{j=1}^{n} u_{ij}^{\lambda}\right) \tag{4}$$

Step 5: For *j*=1~n, recalculate $\boldsymbol{U}^{(b)}$ and make $\boldsymbol{U}^{(b+1)} = \boldsymbol{U}^{(b)}$;

1. Calculate $I_j$and $\overline{I_j}$

$$I_j = \left\{i | 1 \leq i \leq c; d_{ij} = \left||X_j - V_i|\right| = 0\right\} \tag{5}$$

$$\overline{I_j} = \{1,2,\cdots,C\} - I_j \tag{6}$$

2. For data $\boldsymbol{X_j}$, its degree of membership is updated. If $\boldsymbol{I_i} \neq \emptyset$, then for all $\boldsymbol{i} \in \overline{\boldsymbol{I_j}}$, let $\boldsymbol{u_{ij} = 0}$,

$\sum_{i\in I_j} u_{ij} = 1, j$=$j$+1; otherwise:$u_{ij} = 1/\sum_{k=1}^{c}\left(d_{ij}/d_{kj}\right)^{2(\lambda-1)}$

Step 6: Compare $\boldsymbol{U}^{(b)}$ and $\boldsymbol{U}^{(b+1)}$. If $||\boldsymbol{U}^{(b)} - \boldsymbol{U}^{(b+1)}|| < \boldsymbol{\varepsilon}$, perform Step 7; otherwise, return 3;

Step 7: Let $\boldsymbol{U}' = \boldsymbol{U}^{(b+1)}$;**Ending**.

The FY-2E meteorological satellite has three spectral cloud patterns of visible light, infrared light, and water vapor. Therefore, each pixel in the cloud image has three grayscale characteristic attributes: Visible light, infrared light, and water vapor. FCM has both intuitive effects and high-execution efficiency in two-dimensional plane space. Accordingly, we will specifically analyze how to select the most suitable two from the three attributes.

Using the grayscale eigenvalues of the three channels to construct the two-dimensional orthonormal spectral feature spaces (see Tab. 2), and the pixels on the image are mapped one by one with the points in the two-dimensional feature space.

Table 2: Pixel to feature space mapping

Cloud image element	Two-dimensional spectral feature space points
$R_I(m,n)$, $R_V(m,n)$	$S_{IV}(R_I,R_V)$
$R_I(m,n)$, $R_W(m,n)$	$S_{IW}(R_I,R_W)$
$R_V(m,n)$, $R_W(m,n)$	$S_{VW}(R_V,R_W)$

Among them, SIV (RI, RV), SIW (RI, RW), and SVW (RV, RW) represent the points of IR-VIS, IR-WV, and VS-WV in the two-dimensional feature space respectively.

Algorithm 1 is used to perform FCM clustering on three kinds of two-dimensional grayscale feature spaces in Tab. 2, respectively. Analysis of the clustering results shows that the infrared-visible two-dimensional feature space has the best clustering effect. Therefore, the gray features of infrared and visible light are selected in this paper, that is, FCM clustering is performed on the infrared-visible two-dimensional gray feature space.

It is assumed that the clustered regions of land and water in the two-dimensional gray feature space are SIV (land) and SIV (water body), respectively. Set the grayscale values of pixels (x, y) in the infrared and visible cloud images as PI(x, y) and PV(x, y), respectively. The gray values of the points (x, y) after the cloud image segmentation of the infrared and visible light are GI(x, y), GV(x, y), respectively. Combined with the threshold-based image segmentation, criteria for image segmentation through the FCM method can be established in formulae (7) and (8):

$$G_I(x,y) = \begin{cases} P_I\ (x,y)\ , & \big(P_I(x,y), P_V(x,y)\big) \in S_{IV}(Land) \cup S_{IV}(Water) \\ \quad 0 \quad , & other \end{cases} \tag{7}$$

$$G_V(x,y) = \begin{cases} P_V\ (x,y)\ , & \big(P_I(x,y), P_V(x,y)\big) \in S_{IV}(Land) \cup S_{IV}(Water) \\ \ 0\ , & other \end{cases} \tag{8}$$

In summary, this paper proposes an image segmentation method based on FCM, which can be utilized to perform the first image segmentation processing on the satellite cloud image to achieve cloud separation. The cloud-ground separation method based on FCM flow chart is shown in Fig. 1:

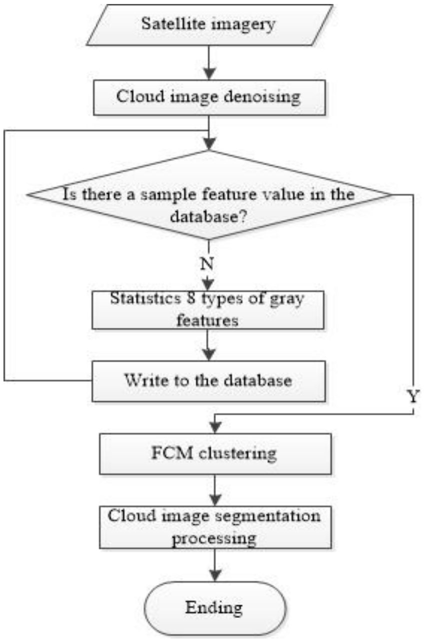

Satellite imagery
Cloud image denoising
Is there a sample feature value in the database?
N
Statistics 8 types of gray features
Write to the database
Y
FCM clustering
Cloud image segmentation processing
Ending

Figure 1: The flow chart of cloud-ground separation method based on FCM

First, the median image filtering method is used to preprocess the input of satellite image to eliminate the noise and interference in the image. The second step is to build the FCM target dataset. The infrared visible light cloud map are selected to construct two-dimensional spectral feature space of IR-VIS, and the grayscale feature values of each pixel in infrared and visible light images are stored by using the two-dimensional array data [256, 2] respectively. The third step is to calculate the general characteristic value of the sample. If there are no eight kinds of grayscale features of land surface and cloud in the database, the cloud sample database will be analyzed, and the gray features of eight kinds of surface and cloud types will be statistically recorded in the database. Otherwise, this step should be ignored and continue to the next step. In the fourth step, the initial cluster center is set as the sample feature value for FCM. Eight classes of surface and cloud grayscale features were obtained from the database as the initial clustering center of the FCM and clustered in the IR-VIS two-dimensional spectral feature space. The fifth step is to perform image segmentation on the cloud image. Using the cluster partition map obtained in the fourth step, all pixels are judged. If its two-dimensional grayscale value is within the surface grayscale feature region, it is a surface pixel point, and the mark is black; otherwise, it is a cloud pixel point and is reserved.

2.2 Cloud identification method based on DBSCAN

After the first step, the cloud map is actually a distribution map of cloud clusters surrounded by a black background. Some of these cloud clusters are interconnected to form a cloud system, and some of them independently exist. Starting from image pixels, DBSCAN connects the cells belonging to the same cloud cluster into a single area through neighbourhood search, which fully embody the basic idea of regional growth in image segmentation processing. In this paper, the DBSCAN clustering algorithm is used in the second image segmentation processing of cloud image to realize cloud cluster recognition.

Here are some definitions based on density clustering:

Definition 1 Given the data object a, we have an n-dimensional hypersphere with an a-center point and Eps as the radius called the Eps-neighborhood of a, which is: $N_{Eps}(a) = \{b \in D | dist(a, b) \leq Eps\}$

Where dist is the distance function.

Definition 2 For data object a, if there are $\left|N_{Eps}(a)\right| \geq Minpts$, then we call a the core under (Eps, Minpts) condition; otherwise, if a is inside the Eps-neighbors of other core points, we call a the border.

Definition 3 For data object a, if it is neither a core nor a border, then we call a as a noise.

Definition 4 Given (Eps, Minpts) if data objects *a* and *b* satisfy:

(1) $a \in N_{Eps}(b)$
(2) $|N_{Eps}(b)| \geq Minpts(b$ is the core)
Then the direct density from *b* to *a* is directly density-reachable.

Definition 5 For the data object set *A*, if there is a series of objects$a_1, a_2, \cdots, a_n(a_i \in A, 1 \leq i \leq n-1)$, where a_i to a_{i+1} are directly density-reachable for Eps and Minpts, then we think a_1 to a_n are density-reachable about Eps and Minpts.

Definition 6 For the data object set A, $a \in A$, $b \in A$, $c \in A$, if *a* and *b* to *c* are all about Eps and Minpts density reachable, then we think *a* and *b* are density-connected about Eps and Minpts as Fig. 2 shown.

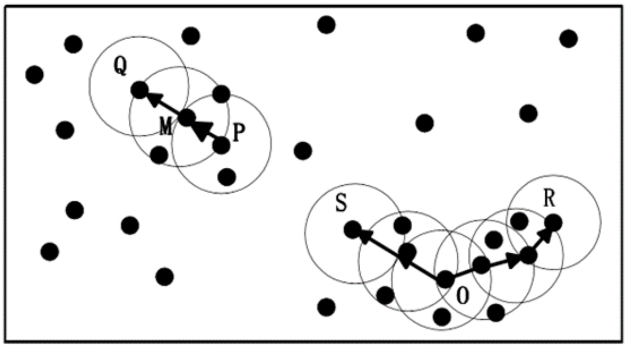


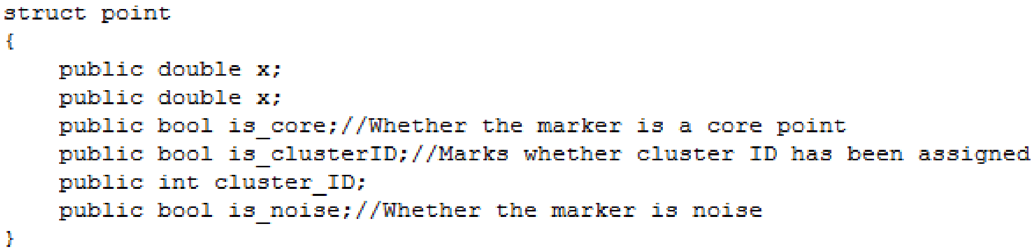
Figure 2: density-reachable and density-connected conceptual diagram

Definition 7 Given a set of data objects *A* and a subset *D* of it, *D* must satisfy the following conditions before determining whether *D* is a cluster:

(1) D $\neq \emptyset$
(2) For $\forall a, b$,if $a \in D$,and from a to b is density-reachable about Eps and Minpts, then $b \in D$
(3) For $\forall a, b$,if $a \in D$ and $b \in D$,a and b are density-connected about Eps and Minpts.

The basic idea of the DBSCAN algorithm is: for the target data set A, randomly select an object ***a***, and search for all objects with a density up to Eps and Minpts starting from ***a***. If a is the core object, a cluster D containing all the points in the Eps-neighborhood of a is generated; if ***a*** is a border or noise, ***a*** is marked as a noise momentarily. Then searching for point ***b*** in D that is not marked. If you have a core that is not in *D*, then add it to *D* and continue to check the Eps-neighborhood of the midpoint. The process is performed cyclically until no new object is added to the current cluster *D*. The above process is the entire generation process of the cluster *D*. The cluster generation process is repeated below until all the points in *A* are classified into a certain category or marked.

Some attributes of the DBSCAN data structure mainly contain core points, noise, coordinates of data object positions, identification of the cluster to which they belong, and whether or not they are to be processed. The data structure can generally be represented by a structure, as shown in Fig. 3.



```
struct point
{
    public double x;
    public double x;
    public bool is_core;//Whether the marker is a core point
    public bool is_clusterID;//Marks whether cluster ID has been assigned
    public int cluster_ID;
    public bool is_noise;//Whether the marker is noise
}
```

Figure 3: The general form of the DBSCAN data structure

When the amount of data is relatively small, using this kind of storage structure will not have a great impact on the efficiency of the algorithm. However, when using DBSCAN to process images containing millions or even millions of pixels, this storage structure greatly increases the memory’s searching and writing burden. In this paper, we use multiple two-dimensional arrays to save the image's pixel information. For example, we can use the integer arrays data_id [W, H] and data_type [W, H] to save all the information:

(1) The position of the pixel in the image can be calculated by subscripting two arrays;
(2) data_id [x, y] can be used to determine whether the point (x, y) has been classified, and save the category of the point;
(3) data_type [x, y] can use different values to represent the type of point (x, y)-core point, noise, invalid point, or unprocessed point.

The bottleneck that limits the efficiency of DBSCAN is the neighborhood lookup algorithm. Normally, DBSCAN completes a neighborhood search for a point by traversing all points. The time complexity is O(n^2). After using the two-dimensional array to store the pixel information in the image, we can use the array index to speed up the neighborhood search. Assuming that we want to find the Eps-neighborhood of p(x, y), we only need to find the ordinate of the Eps-neighborhood according to the subscript of p (see Fig. 4). Then we can judge all the points in this range. The time complexity of this scheme is O(n).

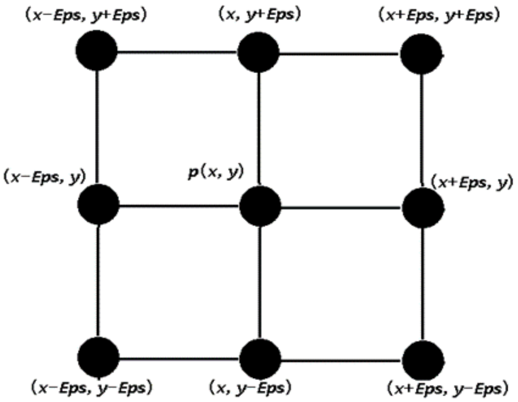


Figure 4: Eps neighborhood lookup in 2D space

Since the Eps neighborhood of *p* is located in two adjacent grids of 4×Eps centered on *p*, it is only necessary to calculate the coordinates of the four vertices of the square with *p* as the center point and Eps as the radius. The optimized implementation of DBSCAN-based cloud image segmentation is as follows:

Algorithm 2 Cloud Image Segmentation Based on DBSCAN

Input: Cloud **D** (Surface Area $\mathbf{D_1}$ and Cloud Region $\mathbf{D_2}$), Parameters **Eps, Minpts**

Output: Processed cloud image

Assuming that the cloud map **D** is separated into the surface area $\mathbf{D_1}$ and the cloud area D_2 after being separated by the cloud, the cloud cluster identification algorithm is implemented as follows:

Step 1: The pixels in the surface area $\mathbf{D_1}$ are marked as invalid points, and the points in the cloud area $\mathbf{D_2}$ are marked as unprocessed points.

Step 2: Determine the type (core point, boundary point, noise) of each point in $\mathbf{D_2}$ and mark it.

Step 3: starting from any point ***p***, if ***p*** is the core point and is not processed, a new cloud $\mathbf{C_i}$ is generated, containing the point ***p*** and all the points in its neighborhood. For point ***q*** in the p-neighborhood, if ***q*** is not processed and is a core point, all unprocessed points in the ***q-neighborhood*** are added to the cloud $\mathbf{C_i}$. Mark all points in $\mathbf{C_i}$ as processed points.

Step 4: Repeat **Step 3** until all the points in the cloud area $\mathbf{D_2}$ have been processed.

Step 5: For each cloud group $\mathbf{C_i}$ in the cloud diagram, it is represented by a single color.

Step 6: Output Displays the processed cloud image.

2.3 Cloud-based thunderstorm identification

The multi-spectral data provided by the cloud map reflects the physical properties of the target object and is the main source of information for most thunderstorm cloud identification methods. The spectral features of the cloud are reflected in the cloud diagram as brightness temperature characteristics, local standard deviation, and the difference of brightness temperature.

The analysis of thunderstorms that occurred on August 6, 2015 in the central and eastern parts of the Qinghai-Tibet Plateau and the north of Hengduan were used for analysis, as shown in Fig. 5. There is a high-pressure cold vortex on the Tibetan Plateau, where the flow in front of the trough is southerly and the flow in the north of the trough is increased. As a result, the cloud system gradually evolves into cyclonic bending, forming a cold front cloud zone. The large amount of convective clouds in front of the front easily makes the ground appear thunderstorms.

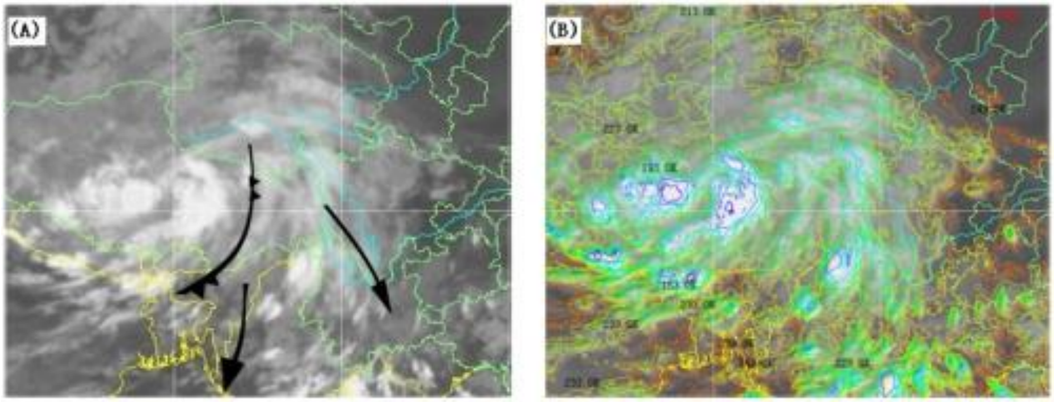


Figure 5: Infrared cloud map (A) and temperature contour map (B)

We analyze the thunderstorm weather from the three perspectives of brightness temperature characteristics, local standard deviation, and bright temperature difference. The results and analysis are as follows: (1) It can be seen from Fig. 6 that the average and maximum temperature change trends are basically the same. The average value is between 210~220 K, and the brightness temperature reaches the maximum value of 233 K around 17:00. (2) In Fig. 7, the fluctuation of the local standard deviation is larger, because the uniformity of the thunderstorm cloud changes greatly. Thunderstorm clouds developed steadily most of the time, reaching a maximum at 14 o’clock, which indicates that thunderstorm clouds have developed more strongly at this time. (3) According to Fig. 8 and Fig. 9, the brightness temperature difference of IR1-IR2 in Thunderstorm Cloud fluctuates from -5 K to -10 K, and the brightness temperature difference of IR1-WV fluctuates from -17 K to -25 K. Usually the emergence of IR1-IR3 and IR1-WV brightness temperature extreme values also indicates the development of thunderstorm clouds.

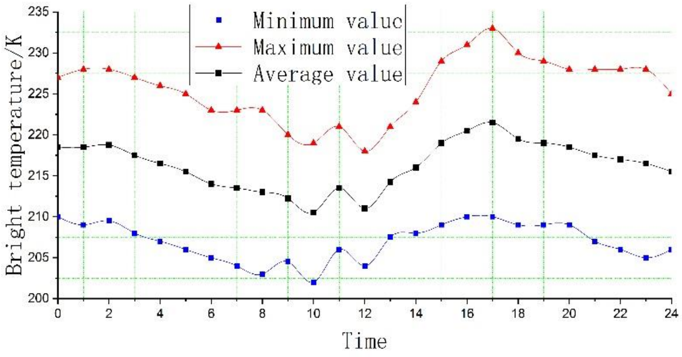


Figure 6: Changes of brightness and temperature of thunderstorm clouds

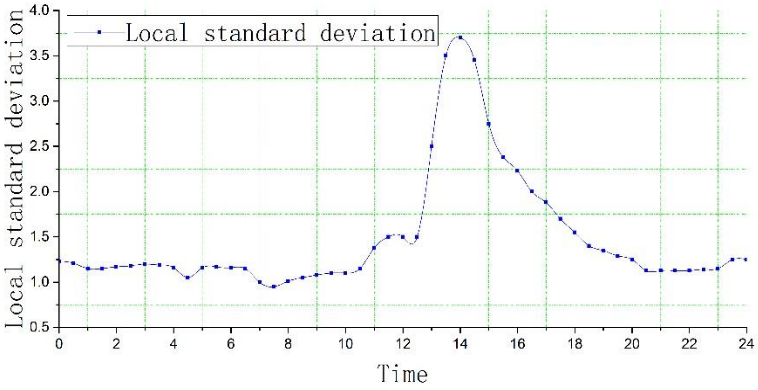


Figure 7: Changes of local standard deviation of thunderstorm clouds

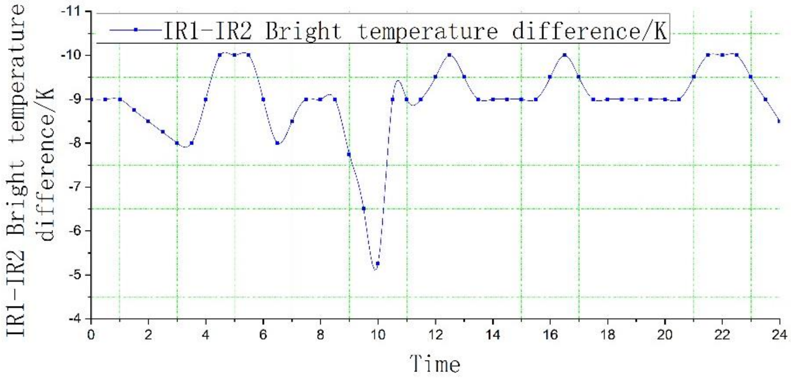


Figure 8: Changes in brightness temperature of thunderstorm cloud IR1-IR2

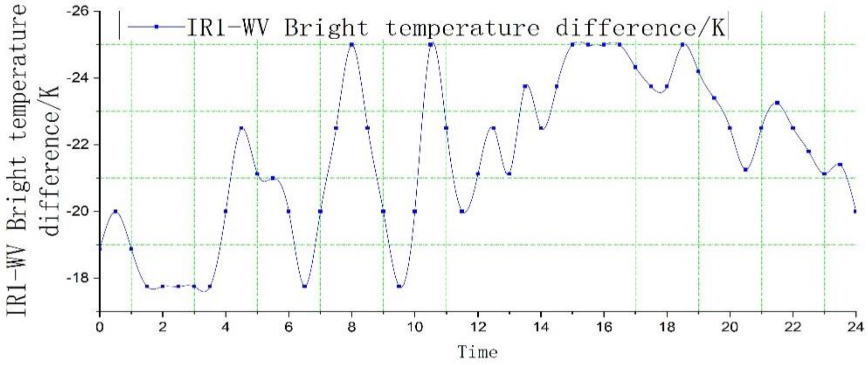


Figure 9: Changes in brightness temperature of thunderstorm cloud IR1-WV

The characteristics of brightness temperature, local standard deviation, and brightness temperature difference can intuitively show changes in the spectral characteristics of a thunderstorm cloud on the map, but it does not apply to the calculation process. In order to quantify the above-mentioned spectral characteristics of thunderstorm clouds, three meteorological parameters-deep convection index DCI, infrared multi-spectral band differences TIR1-IR2 and TIR1-WV-are introduced.

(1) Deep convection index (*DCI*)

$$DCI = \begin{cases} 250 - T_{BB}, T_{BB} < 250K \\ \quad 0 \quad\;\, , T_{BB} \geq 250K \end{cases} \tag{9}$$

Among them, TBB indicates cloud-top brightness. The smaller the TBB, the greater the DCI, which indicates that the area is more prone to thunderstorms and other strong convective weather.

(2) Infrared multispectral band differences TIR1-IR2 and TIR1-WV

The brightness temperature difference of infrared channel TIR1-IR2 can infer the height of the cloud. If the difference between the infrared temperature and brightness temperature of water vapor channel TIR1-WV is less than zero, it indicates that the cloud height in this area is greater than the troposphere, and strong convection may occur at this time.

Through the analysis of thunderstorm clouds, DCI, TIR1-IR2, and TIR1-WV have a good correspondence with the development of thunderstorm clouds. The value range of the spectral characteristics index of thunderstorm cloud inversion is shown in Tab. 3.

Table 3: The index of thunderstorm cloud spectral characteristic

Index	Value
DCI	10~60
$T_{IR1\text{-}IR2}$	-15~-4
$T_{IR1\text{-}WV}$	-30~-15

Some clouds are very similar to thunderstorm clouds in spectral characteristics, such as cirrus clouds. It is not sufficient to use only spectral features as a recognition basis for thunderstorm clouds. Texture analysis can identify thunderstorm clouds from the perspective of texture. Texture is a common visual phenomenon. Thunderstorm cloud is relatively uniform and smooth on satellite image. Gray level co-occurrence matrix is used in this paper to analyze and extract the texture features of cloud clusters.

Let gray value of the pixel $P_1(x,y)$ in the image be i, and move P1 according to the relation $z=(d_x,d_y)$ to obtain the pixel $P_2(x+d_x,y+d_y)$ with the direction θ separated by j. And the gray value is j. The number of such changes is denoted as $P_z(i,j)$. The spacing and azimuth angle between P1 and P2 can be determined by z, and z is determined to obtain the gray level co-occurrence matrix Pz about z:

$$P_z=\begin{bmatrix} P_z(0,0) & P_z(0,1) & \dots & P_z(0,j) & \dots & P_z(0,k-1) \\ P_z(1,0) & P_z(1,1) & \dots & P_z(1,j) & \dots & P_z(1,k-1) \\ \dots & \dots & \dots & \dots & \dots & \dots \\ P_z(i,0) & \dots & \dots & P_z(i,j) & \dots & P_z(i,k-1) \\ \dots & \dots & \dots & \dots & \dots & \dots \\ P_z(k-1,0) & \dots & \dots & P_z(k-1,0) & \dots & P_z(k-1,k-1) \end{bmatrix}$$

Where, k is the gray level of the cloud image. In general, the distance d is 1. The azimuth angle θ in horizontal, vertical, and diagonal linestake 0°, 45°, 90°, and 135°, and these can be changed by setting z, such that a represents 0° and a represents 45°. For ease of calculation, all elements in Pz are generally normalized.

$$\hat{P}(i,j)=\frac{P(i,j)}{\sum_{i=0}^{k-1}\sum_{j=0}^{k-1}P(i,j)} \tag{10}$$

The gray level co-occurrence matrix of the image is constructed, and various parameters can be used to describe the texture features. The four statistical parameters commonly used are Energy (ASM), Contrast (CON), Inverse Moment (IDM), and Entropy (ENT).

Their calculation formulae are (2-9), (2-10), (2-11), and (2-12), respectively.

$$ASM=\sum_{i=0}^{k-1}\sum_{j=0}^{k-1}\hat{P}(i,j)^2 \tag{11}$$

$$CON=\sum_{n=0}^{k-1}n^2\left\{\sum_{|i-j|=n}\hat{P}(i,j)\right\} \tag{12}$$

$$IDM=\sum_{i=0}^{k-1}\sum_{j=0}^{k-1}\frac{\hat{P}(i,j)}{1+(i-j)^2} \tag{13}$$

$$\mathrm{ENT}=-\sum_{i=0}^{k-1}\sum_{j=0}^{k-1}\hat{P}(i,j)\log\hat{P}(i,j) \tag{14}$$

For a given image, the energy reflects whether the grayscale change has regularity, and whether the contrast ratio reflects changes in the brightness of pixels and surrounding pixels. The inverse difference moment reflects the change of local details. And the entropy reflects the complexity and confusion of its texture distribution.

When selecting a cloud image texture to analyze a target image, an infrared channel cloud image is generally selected. The infrared cloud map A in Fig. 5 is still used here to perform texture analysis on the thunderstorm cloud. Take the five thunderstorm clouds (1, 2, 3, 4, 5) and 3 non-thunderstorm clouds (6, 7 and 8), as shown in Fig. 10. The texture characteristics of thunderstorm clouds and non-thunderstorm clouds are quite different, so grayscale co-occurrence matrices can be used to find differences in the value of texture statistics for thunderstorm clouds and non-thunderstorm clouds.

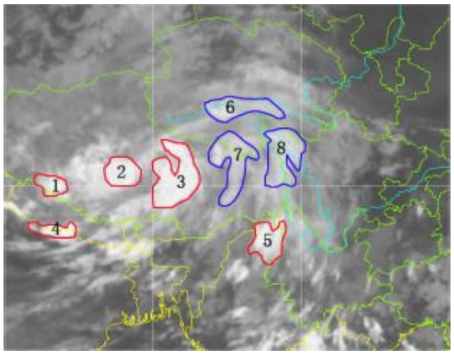


Figure 10: Infrared cloud at 8:00 on August 6, 2015

For each cloud cluster, the four statistical parameters of energy, contrast ratio, inverse difference moment, and entropy are calculated, as shown in Fig. 11. In this figure, the abscissa represents 8 cloud clusters, and each point on the abscissa has 3 sets of statistics (maximum, minimum and average), and the ordinates represent 4 kinds of characteristic statistics.

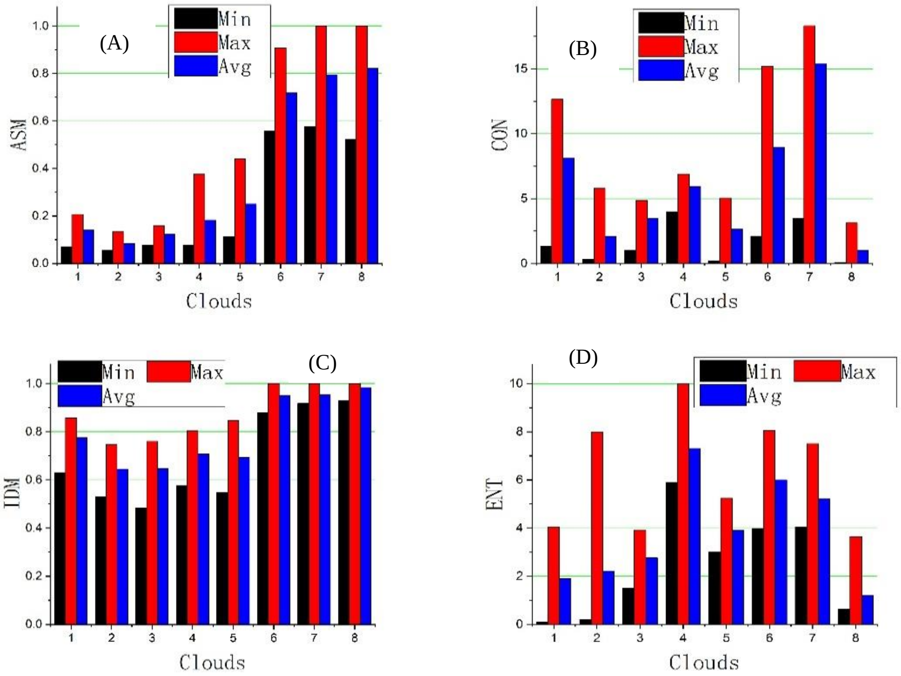


Figure 11: The 8 cloud cluster texture parameter statistics

By analyzing and comparing the statistics of five thunderstorms and three non-thunderstorms, it was found that the ASM values of the five thunderstorms ranged from 0 to 0.4, while the ASM values of the three non- thunderstorms were substantially greater than 0.5. The IDM values of the eight cloud clusters are similar, but thunderstorm clouds and non-thunderstorm clouds can still be distinguished from it. The IDM value of the thunderstorm cloud is about 0~0.8, and the IDM value of the thunderstorm cloud is more than 0.9. The contrast is relatively stable in the clouds 2, 3, 4, and 5, the fluctuation range is 0-7, and the cloud 1 part exceeds 7, so the contrast has a certain reference value. The ENT values of eight cloud groups fluctuate up and down without rules, which makes it difficult to distinguish between cloud groups. Therefore, entropy cannot be used as a criterion for identifying thunderstorm clouds.

According to the above texture analysis, we use energy, inverse difference moment, and contrast ratio to complete the thunderstorm cloud recognition. The range of the inversion thunderstorm cloud texture feature index is shown in Tab. 4:

Table 4: The index of texture feature of thunderstorm cloud

Index	Value
Energy (*ASM*)	0~0.4
Inverse Difference Moment (*IDM*)	0~0.8
Contrast Ratio (*CON*)	0~7

Based on the above analysis, the realization of thunderstorm cloud based on cloud cluster is as follows:

Algorithm 3 The thunderstorm Cloud Recognition Algorithm based on cloud

Input: Cloud D

Output: A cloud map showing a thunderstorm cloud.

Step 1: Preprocess the cloud image D and use the fast median filtering algorithm to eliminate noise and interference;

Step 2: The image segmentation method based on FCM is used for the first processing to realize the separation of cloud area and surface area;

Step 3: Use DBSCAN-based image segmentation method to process cloud image for the second time, realize cloud cluster identification, and get cloud cluster $\mathbf{A_1}, \mathbf{A_2}, \cdots, \mathbf{A_n}$;

Step 4: Calculate the spectral characteristic index **DCI**, **TIR1-IR2** and **TIR1-WV** of all the pixels of the cloud group $\mathbf{A_i}$. The number of pixel points of the conditions in Tab. 3 in the statistical cloud group $\mathbf{A_i}$ is $\boldsymbol{n_i}$. If $n_i \geq N_i \times 0.3$ ($\mathbf{N_i}$ is the total number of pixels of $\mathbf{A_i}$), the $\mathbf{A_i}$ is initially marked as a thunderstorm cloud. So you can get a series of thunderstorm clouds $\mathbf{B_1}, \mathbf{B_2}, \cdots$;

Step 5: The preliminary identified thunder cloud $\mathbf{B_j}$ on texture analysis, the 3 x 3 window respectively to extract energy, deficit moment and contrast the three characteristic parameters, judge whether meet the values in Tab. 4. If the conditions are met, Bj is determined to be a thunderstorm. In this way, all the thunderstorm clouds **C₁**, **C₂**,⋯;

Step 6: Use some color for thunderstorm **C₁**, **C₂**, ⋯The area is filled and marked as a thunderstorm cloud;

Step 7: Output displays.

3 The experimental results

In the experiment, we used two kinds of cloud image segmentation methods-traditional threshold-based image segmentation methods and FCM-based image segmentation methods to achieve cloud separation. In Fig. 12, (A) is the original satellite image, (B) is a threshold-based image segmentation effect, and (C) is the FCM-based image segmentation effect proposed in this paper.

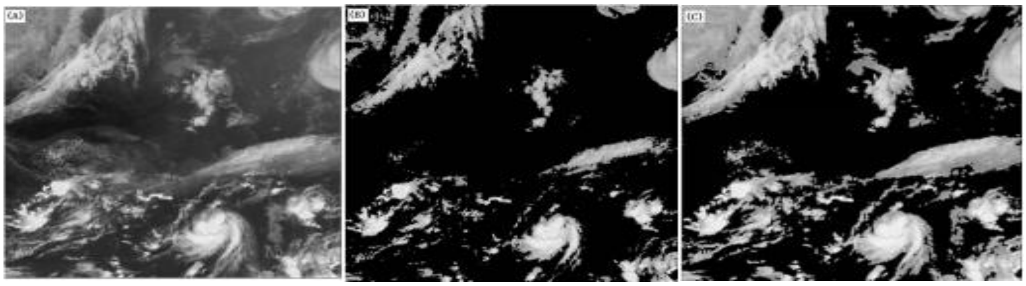


Figure 12: Effect of cloud separation experiment

Traditional threshold-based image segmentation methods use the general gray value of the ground surface (infrared is 97, and visible light is 42 as the threshold, as shown in Tab. 2). Experiments demonstrate that the cloud area distribution obtained by this method is far less than that in the actual situation. As many cloud areas are divided into surface areas, especially in the "transition phase" of the cloud. Although the threshold value can reflect the gray scale feature boundary of the cloud and the surface in most cases the limit is not fixed in practice. Therefore, for the specific cloud image segmentation, the threshold-based image segmentation method is not applicable, which cannot guarantee the accuracy of image segmentation and will reduce the quality of follow-up research. The FCM-based image segmentation method proposed in this paper can obtain a dynamic range of gray scale features based on the characteristics of each cloud image, and can solve the problem of uncertain ownership of certain pixels. Comparing 12(A) and 12(C), it is found that the cloud area obtained by this method is almost the same as that in the actual situation.

In the DBSCAN-based cloud image segmentation experiment, we found that different Eps and Minpts parameters directly affect the size of the neighborhood, and then show different cloud segmentation effects. For convenience, these two parameters are expressed as parameter pair form $(Eps, Minpts)$.

This paper uses the four parameters of cloud number (Num), minimum cloud area (MinS, to count the number of pixels contained in the cloud), maximum cloud area (MaxS, counting the number of pixels contained in the cloud), and execution time to compare and analyze different experimental results. The four parameters of the parameter pairs (1, 4), (1, 6), (2, 12), (2, 18), (4, 40), and (4, 60) are respectively counted, as shown in Tab. 5.

Table 5: Cloud cluster identification experiment parameters for different (Eps, Minpts)

Parameter	(*Eps*, *Minpts*)					
	(1, 4)	(1, 6)	(2, 12)	(2, 18)	(4, 40)	(4, 60)
Num	89	80	52	49	27	25
MinS	20	24	31	32	29	35
MaxS	19845	18362	17965	17128	15500	14531
T/ms	103424	88304	81926	76798	53732	43300

It can be seen that: (1) If Eps decreases, Num will increase, MinS will decrease, Max will increase, and T will increase. This shows that the smaller the radius of the neighborhood is, the smaller the resolution of cloud cluster recognition is, and the smaller cloud cluster can be identified, but the excessive execution time leads to long execution time. (2) When Eps is fixed, as Minpts increases, Num decreases, MinS hardly changes, MaxS decreases, and execution time decreases. This shows that the Minpts value limits the number of core points, indirectly reduces the number of iterations of the algorithm, and makes each cloud size difference more even.

It can be seen that before using the DBSCAN algorithm for cloud segmentation experiments, we can reasonably adjust Eps and Minpts according to our own needs so as to obtain cloud clusters of ideal size. For example, in the meteorological research, the size of cloud clusters required are more stringent, and the values of Eps and Minpts can be appropriately reduced when analyzing local areas. When the analysis area is large or the approximate distribution of clouds is needed, Eps and Minpts can be added appropriately to reduce the execution time.

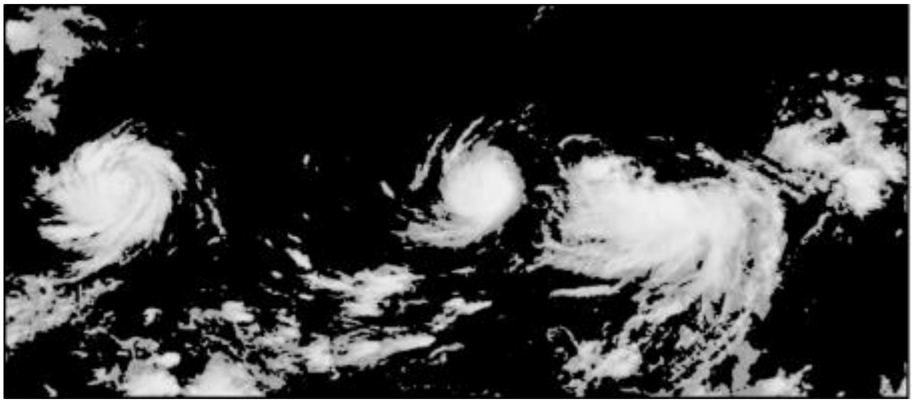


Figure 13: Cloud image

Normally, Eps and Minpts is taken as 4 and 60, respectively, to get the best results. Next, a cloud image with a larger resolution is selected for DBSCAN-based image segmentation to verify the effectiveness of the two parameters. The experimental cloud image is shown in Fig. 13. The effect after the DBSCAN segmentation experiment is shown in Fig. 14.

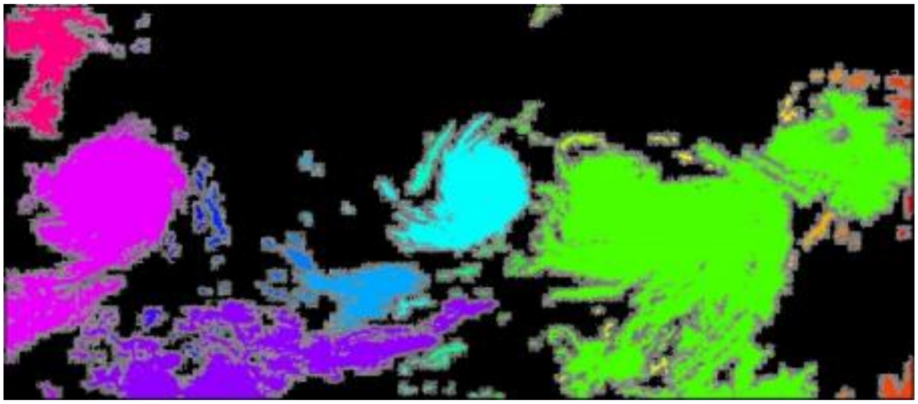


Figure 14: The cloud recognition effect under (4, 60)

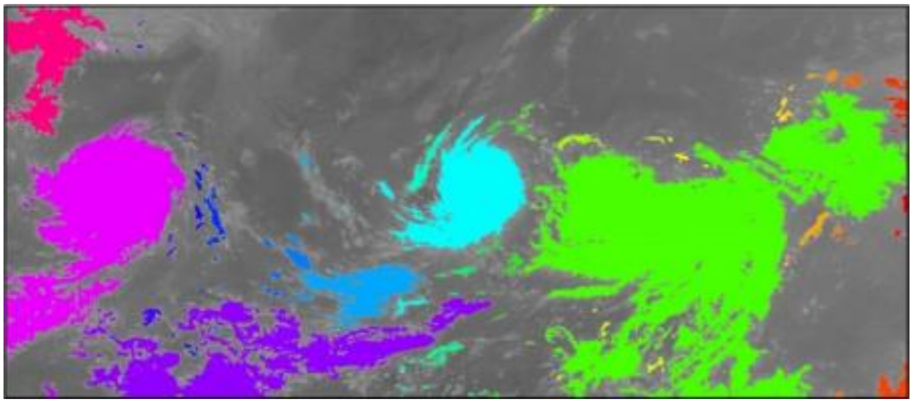


Figure 15: Overlapping diagram after cloud segmentation

The effect of the second image segmentation of the cloud image is superimposed on the original cloud image, resulting in Fig. 15. As can be seen from the figure, the satellite cloud image is processed in two steps based on FCM-based and DBSCAN-based image segmentation, and the results obtained and the actual cloud distribution in the cloud map have a good fitting effect.

The experimental data were selected from the local clouds in two consecutive time periods from August 8 to August 9, 2015, as shown in Fig. 16(A) and 16(B) respectively.

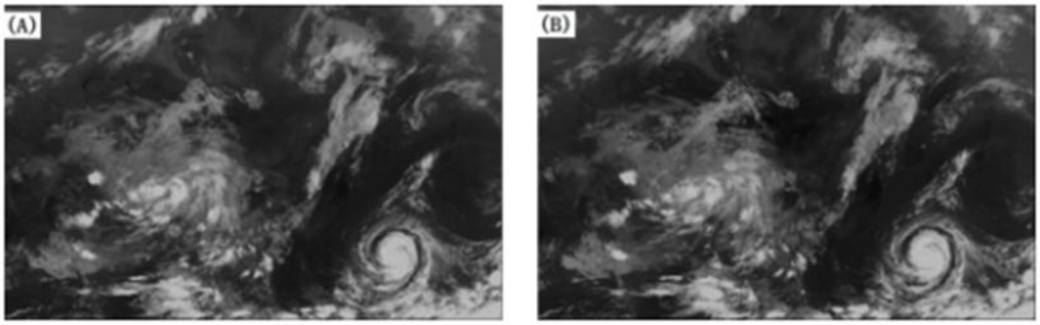


Figure 16: The experimental cloud map

Using the cloud-based thunderstorm cloud recognition method presented in this paper, each cloud in Fig. 16 is processed and the identified thunderstorm cloud is marked in red as shown in Fig. 17.

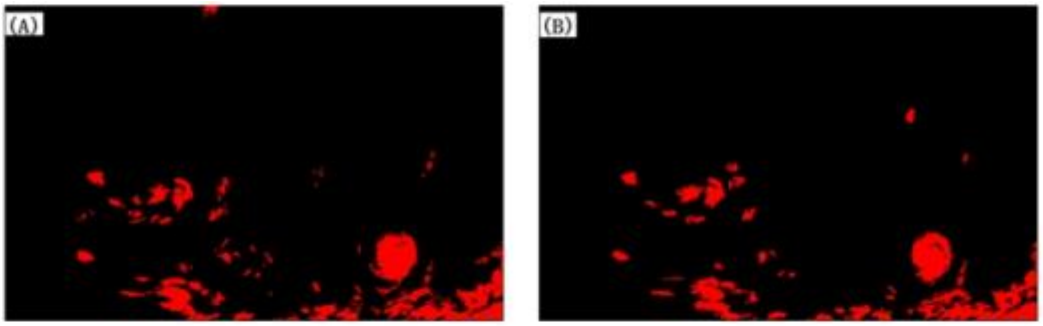


Figure 17: The recognition results of thunderstorm cloud

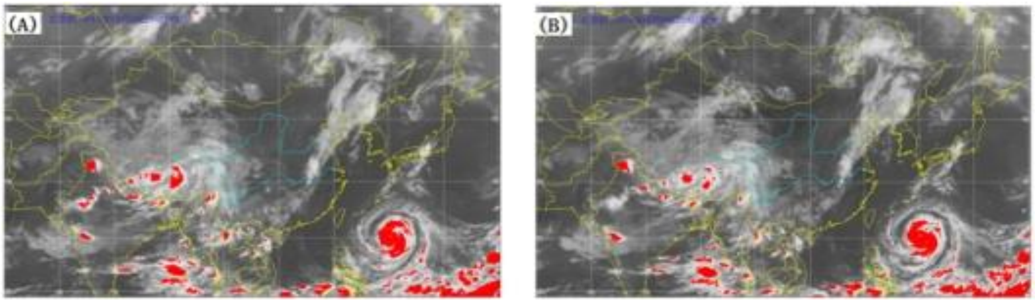


Figure 18: The actual distribution of thunderstorm cloud

Fig. 18 shows the actual thunderstorm cloud distribution area at 8 o’clock to 9 o’clock on August 6, 2015 monitored by a meteorological institute. Through comparison, this article can accurately identify thunderstorm clouds on the basis of clouds. The distribution of thunderstorm clouds is also consistent with the actual situation, which fully demonstrates that the FCM-based image segmentation and the DBSCAN-based image segmentation proposed in this paper are effective for satellite image processing. It is applicable to satellite image processing and can extract the target of interest from the cloud image. The information ensures the efficiency and quality of follow-up analysis and recognition of the target, meeting the requirements of current weather satellite image processing.

5 Conclusion

In this paper, cluster analysis is applied to satellite image segmentation. Firstly, the multi-spectral characteristics of the cloud image are fully utilized. The typical clusters are used as the initial clustering center of the fuzzy C clustering algorithm in the two-dimensional infrared-visible spectral feature space. According to the clustering results, the two-dimensional grayscale feature variation range of the surface and the cloud is obtained. Based on this feature range, the first image segmentation process is performed on the cloud image to eliminate the surface area and preserve the cloud area. Then, the data storage structure of DBSCAN algorithm's and neighborhood search algorithm are adjusted. Combining the image segmentation idea of regional growth, the second step of image segmentation of the satellite cloud image is realized, and the cloud cluster of the whole image is studied conveniently. Finally, the two clustering methods are applied to FY-2E cloud imagery of stationary weather satellites, and a cloud-based thunderstorm cloud identification method is proposed. Experimental results show that image segmentation based on FCM and DBSCAN can accurately and effectively obtain useful targets in complex images. Image segmentation based on clustering analysis theory has good application prospects in image processing and target recognition in meteorological and military fields.

Acknowledgement: This work was supported in part by the National Natural Science Foundation of China (51679105, 61672261, 51409117); Jilin Province Department of Education Thirteen Five science and technology research projects [2016] No. 432, [2017] No. JJKH20170804KJ.

References

Aharon, M.; Elad, M.; Bruckstein, A. (2006): K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, vol. 54. no. 11, pp. 4311-4322.

Ankerst, M.; Breunig, M. M.; Kriegel, H. P.; Sander, J. (1999): OPTICS: Ordering points to identify the clustering structure. *ACM Sigmod Record*, vol. 28, no. 2, pp. 49-60.

Cour, T.; Benezit, F.; Shi, J. (2005): Spectral segmentation with multiscale graph decomposition. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 1124-1131.

Cox, I. J.; Rao, S. B.; Zhong, Y. (1996): "Ratio regions": A technique for image segmentation. *Proceedings of the 13th International Conference on Pattern Recognition*, vol. 2, pp. 557-564.

Dhillon, I. S.; Guan, Y.; Kulis, B. (2007): Weighted graph cuts without eigenvectors a multilevel approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 11, pp. 1944-1157.

Ester, M.; Kriegel, H. P.; Sander, J.; Xu, X. (1996): A density-based algorithm for discovering clusters in large spatial databases with noise. *Kdd*, vol. 96, pp. 226-231.

Guha, S.; Rastogi, R.; Shim, K. (1998): CURE: An efficient clustering algorithm for large databases. *ACM Sigmod Record*, vol. 27, no. 2, pp. 73-84.

Juang, B. H.; Rabiner, L. R. (1990): The segmental K-means algorithm for estimating parameters of hidden Markov models. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 9, pp. 1639-1641.

Karypis, G.; Han, E. H.; Kumar, V. (1999): Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, vol. 32, no. 8, pp. 68-75.

Krinidis, S.; Chatzis, V. (2010): A robust fuzzy local information C-means clustering algorithm. *IEEE Transactions on Image Processing*, vol. 19, no. 5, pp. 1328-1337.

Lupascu, C. A.; Tegolo, D.; Trucco, E. (2010): FABC: retinal vessel segmentation using AdaBoost. *IEEE Transactions on Information Technology in Biomedicine*, vol. 14. no. 5, pp. 1267-1274.

MacQueen, J. (1967): Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, no. 14, pp. 281-297.

Parsons, L.; Haque, E.; Liu, H. (2004): Subspace clustering for high dimensional data: A review. *Acm Sigkdd Explorations Newsletter*, vol. 6, no. 1, pp. 90-105.

Ruspini, E. H. (1977): A theory of fuzzy clustering. *Decision and Control including the 16th Symposium on Adaptive Processes and A Special Symposium on Fuzzy Set Theory and Applications*, vol. 16, pp. 1378-1383.

Sheikholeslami, G.; Chatterjee, S.; Zhang, A. (2000): WaveCluster: A wavelet-based clustering approach for spatial data in very large databases. *International Journal on Very Large Data Bases*, vol. 8, no. 3-4, pp. 289-304.

Wang, S.; Siskind, J. M. (2001): Image segmentation with minimum mean cut. *IEEE International Conference on Computer Vision*, vol. 1, pp. 517-524.

Wang, W.; Yang, J.; Muntz, R. (1997): STING: A statistical information grid approach to spatial data mining. *Proceedings of the 23rd International Conference on Very Large Data Bases*, vol. 97, pp. 186-195.

Zhang, T.; Ramakrishnan, R.; Livny, M. (1996): BIRCH: an efficient data clustering method for very large databases. *ACM Sigmod Record*, vol. 25, no. 2, pp. 103-114.

Zhang, Z.; Wang, S. Z.; Zhang, Y. (2007): New clustering method based on hybrid of SOM and PAM. *Journal of Computer Applications*, vol. 27, no. 6, pp. 1400-1402.