

## Ground-Based Cloud Recognition Based on Dense\_SIFT Features

Zhizheng Zhang<sup>1</sup>, Jing Feng<sup>1,\*</sup>, Jun Yan<sup>2</sup>, Xiaolei Wang<sup>1</sup> and Xiaocun Shu<sup>1</sup>

**Abstract:** Clouds play an important role in modulating radiation processes and climate changes in the Earth's atmosphere. Currently, measurement of meteorological elements such as temperature, air pressure, humidity, and wind has been automated. However, the cloud's automatic identification technology is still not perfect. Thus, this paper presents an approach that extracts dense scale-invariant feature transform (Dense\_SIFT) as the local features of four typical cloud images. The extracted cloud features are then clustered by *K*-means algorithm, and the bag-of-words (BoW) model is used to describe each ground-based cloud image. Finally, support vector machine (SVM) is used for classification and recognition. Based on this design, a nephogram recognition intelligent application is implemented. Experiments show that, compared with other classifiers, our approach has better performance and achieved a recognition rate of 88.1%.

**Keywords:** Feature extraction, Dense\_SIFT, nephogram recognition, BoW.

### 1 Introduction

With the development and popularization of smartphones in recent years, the demand for diverse meteorological information by the public has continuously increased. However, the current identification of the ground-based cloud still uses empirical manual qualitative judgment. The ability to use the mobile terminal to automatically recognize the ground-based cloud image at any time not only helps improve the automation of meteorological observations, but also provides diverse meteorological information to general public and enhances the society's popularity of meteorological knowledge.

Scholars have done a lot of research on identification and classification of ground-based cloud image. The main idea is to obtain various feature information from nephogram so that it can be used as the input data to train a proper classifier. To name a few, Singh [Singh and Glennen (2005)] used gray-level co-occurrence matrix (GLCM), LAWS energy and other methods to extract cloud textural features, and achieved five kinds of cloud classification. Lu et al. [Lu and Li (2015)] used textural feature training selective neural network to achieve cloud shape recognition. Liu et al. [Liu, Sun and Chen (2011)] classified nephogram by geometrical features such as cloud proportions and cloud seams which extract infrared nephogram. Zhang [Zhang (2017)] used the migration learning method to train the convolutional neural network and the convolution depth belief

<sup>1</sup> National University of Defense Technology, Nanjing, 21101, China.

<sup>2</sup> University of Wollongong, Wollongong NSW, 2522, Australia.

\* Corresponding Author: Jing Feng. Email: jfeng@seu.edu.cn.

network with the full sky cloud image. He et al. [He and Dantong (2018)] applied clustering analysis to satellite image segmentation and proposed a cloud-based thunderstorm cloud recognition method.

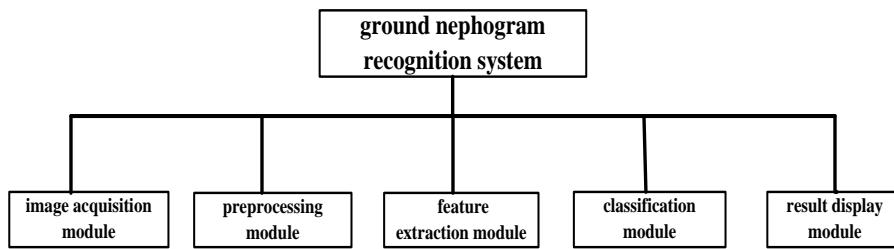
However, most of the above methods make use of the global characteristics of the image, local features have hardly been used to develop ground-based cloud image recognition on mobile terminals. Based on the Android platform, this paper presents an approach that uses four typical clouds as the recognition target. After extracting Dense\_SIFT as the local features of the ground-based cloud, the  $K$ -means algorithm is used to cluster the extracted features to generate a visual dictionary. Finally, the generated dictionary is sent to support vector machine (SVM) for training to obtain a cloud classifier. The experimental results show that the proposed approach achieves a recognition rate of 88.1%.

The rest of paper is organized as follows. Section 2 introduces the system structure and the algorithm flow. Description of the functions and design of the major modules are presented in Section 3, followed by discussion of experimental results in Section 4. Finally, Section 5 concludes this paper and outlines the future work.

## 2 System structure analysis and design

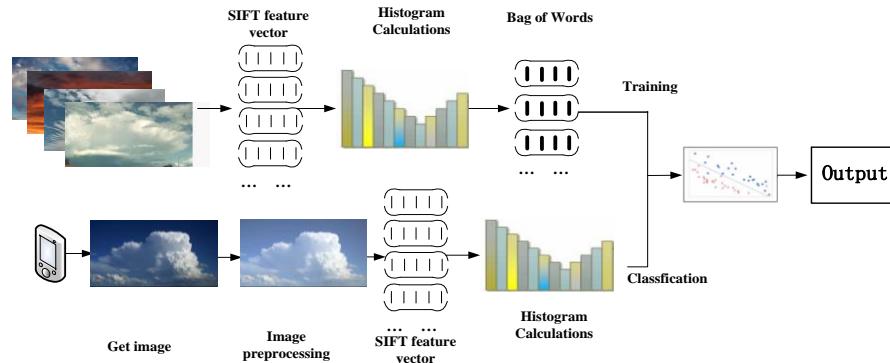
In order to realize the identification and classification of the ground-based cloud image, this application needs to complete the shooting and calling of the camera, the feature extraction of the image, and the training of cloud classifiers, etc. We use Android as the application development platform because of its popularity [Daofeng and Mingxing (2018)].

As shown in Fig. 1, our proposed ground-based cloud image recognition application consists of five sub-modules: the image acquisition module, the image preprocessing module, the feature extraction module, the classification and recognition module, and the result display module.



**Figure 1:** System architecture of the ground-based cloud image recognition application

Fig. 2 depicts the algorithm flow chart of this application. The image acquisition module sends the nephogram to the image preprocessing module for illumination correction and grayscale processing. Then the 128-dimensional Dense\_SIFT features obtained by the feature extraction module construct a visual vocabulary. Finally, the visual vocabulary is sent to trained Nephogram SVM classifier for recognition.

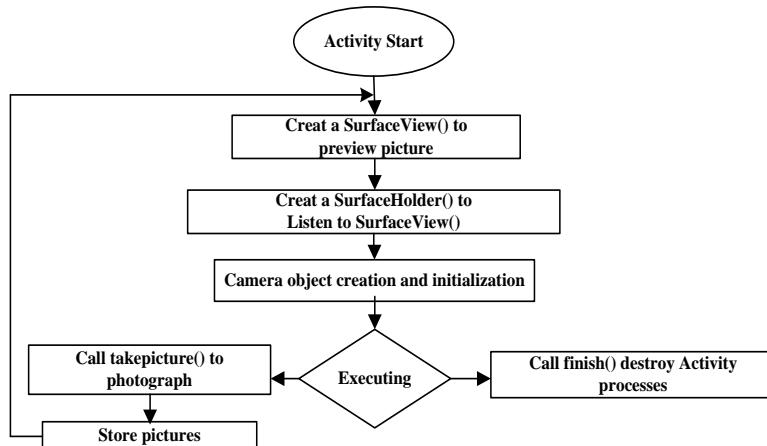


**Figure 2:** Flow chart of ground-based cloud image recognition application

### 3 Technical details of the application

#### 3.1 Image acquisition module

This module mainly performs image acquisition by calling the Android Camera library. The library's methods can be used to read images, start/stop preview images, take photos, get video frames, etc., and manage the camera hardware on the device [Zhang, Liu and Xiang (2015)]. The program workflow of the image acquisition module is shown in Fig. 3.



**Figure 3:** Flow char of image acquisition module

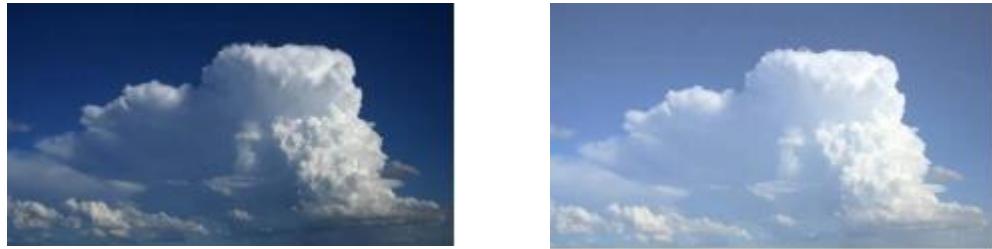
In the process of shooting, the cloud should cover the screen as much as possible to minimize obstructions from other objects.

#### 3.2 Image preprocessing module

The image preprocessing module mainly uses the Bitmap class of Android system to realize image processing function. Nowadays, the resolution of the mobile phone screen is generally high. In order to ensure the speed and efficiency of the operation, the size of

the captured image is transformed and the resolution of the image is reduced. This process calls upon the Bitmap class to convert each cloud image into a  $512 \times 512$  image. In order to avoid the influence of illumination on the cloud image, Gamma correction is performed on the cloud image [Rahman and Rahman (2016)]. The transformation formula is constructed as Eq. (1), where  $I$  is the image.

$$f(I) = I^\gamma \quad (1)$$



**Figure 4:** Original image and corrected nephogram

As shown in Fig. 4, gamma correction can effectively reduce the impact of illumination on the image. Then, we use the weighted average method to gray the image to provide grayscale images for extracting Dense\_SIFT features.

### 3.3 Feature extraction module

This application uses the Dense\_SIFT [Olgun (2016)] feature as the cloud shape feature descriptor to represent the local feature description of the image, and then discretizes the obtained cloud features. After that, the BoW (Bag of Words) model [Escalante, Víctor and Escalera (2017)] is used to express the ground-based cloud image. The ground-based cloud image completes the feature extraction of this module.

#### 3.3.1 Dense\_SIFT feature extraction

The SIFT feature [Lowe (2004)] is a robust descriptor that characterizes image gradient change information. Represented by a 128-dimensional vector, each feature is very stable and maintains invariance to rotation, scale scaling, and brightness variations. Presented by David Lowe in 1999, it is now widely used in image processing.

Considering the complexity of the actual cloud, when the feature extraction of the cloud image is performed, the number of SIFT feature points is uncertain or the number is small, which improves the complexity of the subsequent classifier construction process. This application uses Dense\_SIFT instead of the traditional SIFT feature to ensure the stability of the number of the feature points. Moreover, Dense\_SIFT reduces the feature point detection process and improves the calculation speed compared with the traditional SIFT.

Dense SIFT uses an intensive sampling algorithm to extract features into a  $16 \times 16$  rectangular pixel area with a step size of 8 pixels. The gradient magnitude  $m(x, y)$  and direction  $\theta(x, y)$  of each pixel in the graph are calculated using the formulas below.

$$m(x, y) = \sqrt{[L(x+1, y) - L(x-1, y)]^2 + [L(x, y+1) - L(x, y-1)]^2} \quad (2)$$

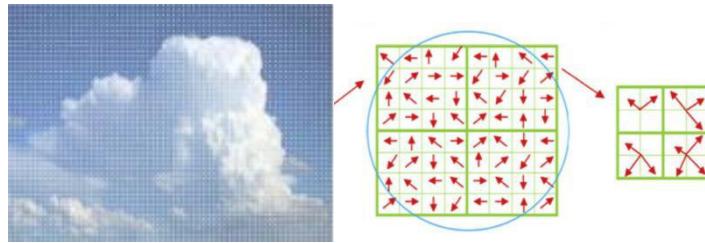
$$\theta(x, y) = \tan^{-1}[L(x, y+1) - L(x, y-1)]/[L(x+1, y) - L(x-1, y)] \quad (3)$$

In Eqs. (2) and (3):  $L$  is a scale space function, and  $(x, y)$  is the pixel point coordinate. The scale space is a description of the image on multiple scales. The scale space function is:

$$L(x, y, \sigma) = G(x, y, \sigma) \otimes I(x, y) \quad (4)$$

Among them,  $G(x, y, \sigma)$  is the Gaussian function and  $I(x, y)$  is the original image.

The feature points are gradually extracted from the cloud map. Each  $16 \times 16$  rectangular pixel area is divided into  $4 \times 4$  sub-areas, and a gradient histogram of each sub-area is calculated. The horizontal axis of the gradient histogram is the direction angle of the gradient, which is divided into 8 intervals, and the vertical axis is the cumulative value of the gradient angles in each direction. A gradient histogram that contains eight directional gradient information for each sub-region is called a seed point. The feature descriptor of such a key point is a 128-dimensional vector of length  $4 \times 4 \times 8$ . Fig. 5 shows a feature extraction effect diagram.



**Figure 5:** The process of generating the Dense\_SIFT features

### 3.3.2 Create a visual vocabulary

The basic idea of the bag of word (BOW) [Peng and Wang (2016); Yi, Zhili, Xingming et al. (2018)] model is to treat the image as text and the features of the image as visual vocabulary. In other words, the image is treated as an irregular feature set, and a description amount is constructed for each image to complete the classification.

The extracted Dense SIFT feature vectors of each nephogram image are collected together and stored in a 128-dimensional space to form a feature vector library of the nephogram image. Then the  $K$ -means algorithm is used to cluster and construct a visual vocabulary with  $K$  cluster centers. The value of  $K$  is determined as 800. By calculating the Euclidean distance between the feature vector and the cluster center, the feature vector is mapped to the cluster center closest to it. After that, the application calculates word frequency statistics, counts the number of occurrences of each feature vector in the image, and represents the image in a  $K$ -dimensional visual vocabulary histogram, which is the BoW description vector. The height of the histogram represents the number of feature vectors at the center of the cluster.

### **3.4 Classification module**

Support Vector Machine (SVM) was proposed in 1995 by VAPNIK based on VC dimension theory and structural risk minimization. Nowadays it is widely used in pattern recognition, classification and other fields, which can better solve the linear indivisible problem in low-dimensional space [Cherkassky and Ma (2004); Ravikumar and Vijayan (2017)]. Our application uses the created visual vocabulary histogram as a training sample to train the SVM classifier.

The choice of penalty factors and kernel functions has a large impact on multi-class SVM classification methods. The classification of ground-based cloud image is mostly nonlinear. In this paper, the histogram cross-core function [Dileep (2012)] is used as the kernel function, and the optimal parameters are selected by cross-validation.

Suppose two histograms are  $x = [x(1), x(2), \dots, x(n)]$ , and  $y = [y(1), y(2), \dots, y(n)]$ , the histogram orthogonal kernel can be defined as:

$$K(x, y) = \sum_{i=1}^n \min[x(i), y(i)] \quad (5)$$

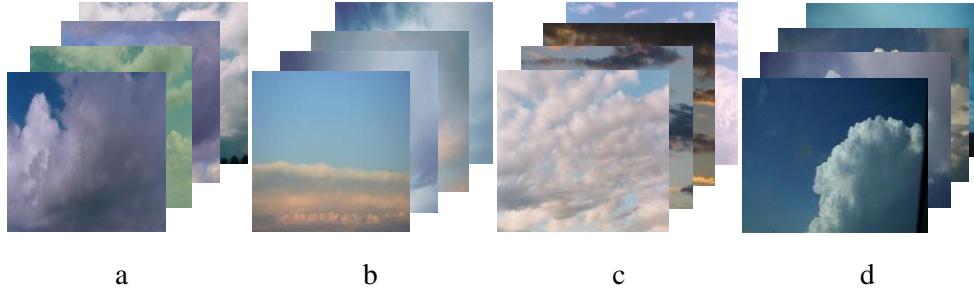
### **3.5 Result display module**

This module obtains the cloud name output by the classification identification module and displays it on the software interface. If the device has stored the detailed information of the cloud (such as cloud description, corresponding weather information, etc.), the information is read and displayed at the same time on the user's screen.

## **4 Experimental results**

This application uses the Glory V9 mobile phone from Huawei as the debugging equipment. The device has a 12-megapixel main camera and is powered by the Android 8.0 operating system. Software development uses Java language and Android Studio as the integrated development environment. The image feature extraction module is implemented by C/C++, and the call of C/C++ library by Android Studio is realized by JNI.

This application selects four typical cloud classes: stratocumulus, altostratus, altocumulus and cumulonimbus, to establish a classification sample library. The training samples are selected from the new version of the International Cloud Atlas from the World Meteorological Organization and the ground-based cloud image data are add. There are 160 training samples for each type of cloud and 40 test samples. Four typical cloud samples are shown in Fig. 6.



**Figure 6:** Four typical cloud samples. (a: stratocumulus; b: altostratus; c: altocumulus; d: cumulonimbus)

In this experiment, we use different classifiers to match the nephogram feature description method proposed in this paper, including SVM, KNN, Decision Tree and Naive Bayesian methods. The experimental results are shown in Tab. 1.

**Table 1:** Experimental results

Classifier	Stratocumulus	Altostratus	Altocumulus	Cumulonimbus	Accuracy
KNN	77.50%	70.00%	75.00%	85.00%	76.90%
Decision Tree	80.00%	77.50%	77.50%	87.50%	80.50%
Naive Bayesian	85.00%	77.50%	80.00%	92.50%	83.70%
SVM	87.50%	82.50%	87.50%	95.00%	88.10%

From Tab. 1 we can see that SVM has better classification performance and lower classification error rate than other classifiers. Analysis of the reasons for the following points: First, the number of cloud image test samples to be identified in this test is small, resulting in a certain degree of error in the final result. In contrast, SVM has the advantage of small-scale sample classification and has better classification accuracy. Second, the cumulonimbus cloud has a higher recognition rate than other types of cloud types because its local features are quite different and easy to distinguish. Last, single feature has limited ability to distinguish cloud images. So local features and global features should be combined for classification.

In order to obtain the classification effect of Dense\_SIFT features on the nephogram, we compare the classification results obtained by using the SIFT and Dense SIFT features under the same experimental conditions. Among them, the histogram cross-core function is selected as the SVM kernel, and the processing time is the classification time of all test images, and the results are shown in Tab. 2.

**Table 2:** Feature recognition performance comparison

Eigenvectors	Processing time	Accuracy
SIFT	62.57	82.50%
Dense_SIFT	40.14	88.10%

Clearly, Dense SIFT features a higher recognition rate than SIFT when the training database and the BoW are of the same size. Dense SIFT has shorter recognition time for feature extraction and classification. It is more suitable for occasions when the shorter processing time is required.

## 5 Conclusion

This paper designs and develops a ground-based cloud image recognition application based on the Android platform. The application extracts Dense\_SIFT as a cloud feature descriptor, uses the BoW model to create a visual vocabulary, selects SVM as a classifier, and completes the recognition of four typical cloud shapes. The recognition effect reaches the expected target. The application has the characteristics of convenience, simple operation, etc., which improves the automation degree of meteorological observation and also provides convenience for meteorological science. However, the application is still in the experimental stage, and its function needs to be improved in the future. First, the recognition rate of the cloud is not high enough in the complex background. Second, the recognition algorithm is limited to the four typical types of clouds described in this paper. More clouds identification needs to be investigated in the future.

**Acknowledgement:** We appreciate the valuable comments from the reviewers. The research work presented in this paper is supported by Nature Fund of Science of China grant No. 6137119.

## References

- Cherkassky, V.; Ma, Y.** (2004): Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Networks*, vol. 17, no. 1, pp. 113-126.
- Cao, Y.; Zhou, Z. L.; Sun, X. M.; Gao, C. Z.** (2018): Coverless information hiding based on the molecular structure images of material. *Computers, Materials & Continua*, vol. 54, no. 2, pp. 197-207.
- Dileep, A. D.** (2012): Speaker recognition using pyramid match kernel based support vector machines. *International Journal of Speech Technology*, vol. 15, no. 3, pp. 365-379.
- Escalante, H. J.; Ponce-López, V.; Escalera, S.** (2017): Evolving weighting schemes for the bag of visual words. *Neural Computing & Applications*, vol. 28, no. 5, pp. 925-939.
- He, L. L.; Ouyang, D. T.** (2018): A method of identifying thunderstorm clouds in satellite cloud image based on clustering. *Computers, Materials & Continua*, vol. 57, no. 3, pp. 549-570.

- Liu, L.; Sun, X. J.; Chen, F.** (2011): Cloud classification based on structure features of Infrared Images. *Journal of Atmospheric and Oceanic Technology*, vol. 28, no. 3, pp. 410-417.
- Lowe, D. G.** (2004): Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110.
- Li, D. F.; Luo, M. X.** (2018): Provably secure APK redevelopment authorization scheme in the standard model. *Computers, Materials & Continua*, vol. 56, no. 3, pp. 447-465.
- Lu, G. Y.; Li, T.** (2015): Ground nephogram recognition algorithm based on selective neural network ensemble. *Computer Applications and Software*, vol. 15, no. 3, pp. 185-187.
- Olgun, M.** (2016): Wheat grain classifications by using dense SIFT features with SVM classifier. *Computers & Electronics in Agriculture*, vol. 122, no. C, pp. 185-190.
- Peng, X.; Wang, L.; Wang, X.** (2016): Bag of visual words and fusion methods for action recognition. *Comprehensive Study and Good Practice*, vol. 150, no. C, pp. 109-125.
- Rahman, S.; Rahman, M. M.** (2016): An adaptive gamma correction for image enhancement. *Eurasip Journal on Image & Video Processing*, vol. 2016, no. 1, pp. 1-13.
- Gurusamy, R.; Subramaniam, V.** (2017): A machine learning approach for MRI Brain tumor classification. *Computers, Materials & Continua*, vol. 53, no. 2, pp. 91-108.
- Singh, M.; Glennen, M.** (2005): Automated ground-based cloud recognition. *Pattern Analysis & Applications*, vol. 8, no. 3, pp. 258-271.
- Zhang, Z.; Liu, P.; Xiang, J.** (2015): How Your phone camera can be used to stealthily spy on you: transplantation attacks against android camera service. *IEEE Transactions on Control Systems Technology*, vol. 13, no. 1, pp. 99-110.
- Zhang, Z.** (2017): *The Study of All-Sky Cloud Classification Based on Deep Learning (Ph.D. Thesis)*. Beijing Jiaotong University.