Research in Astronomy and Astrophysics, 24:045025 (9pp), 2024 April https://doi.org/10.1088/1674-4527/ad339e

Machine Learning-based Identification of Contaminated Images in Light Curve Data Preprocessing

Hui Li[1,2], Rong-Wang Li[1,3], Peng Shu[1], and Yu-Qiang Li[1,3]

[1] Yunnan Observatories, Chinese Academy of Sciences, Kunming 650216, China; lirw@ynao.ac.cn

[2] University of Chinese Academy of Sciences, Beijing 100049, China

[3] Key Laboratory of Space Object and Debris Observation, Chinese Academy of Sciences, Nanjing 210023, China

Received 2023 December 1; revised 2024 February 3; accepted 2024 March 7; published 2024 April 24

Abstract

Attitude is one of the crucial parameters for space objects and plays a vital role in collision prediction and debris removal. Analyzing light curves to determine attitude is the most commonly used method. In photometric observations, outliers may exist in the obtained light curves due to various reasons. Therefore, preprocessing is required to remove these outliers to obtain high quality light curves. Through statistical analysis, the reasons leading to outliers can be categorized into two main types: first, the brightness of the object significantly increases due to the passage of a star nearby, referred to as “stellar contamination,” and second, the brightness markedly decreases due to cloudy cover, referred to as “cloudy contamination.” The traditional approach of manually inspecting images for contamination is time-consuming and labor-intensive. However, we propose the utilization of machine learning methods as a substitute. Convolutional Neural Networks and SVMs are employed to identify cases of stellar contamination and cloudy contamination, achieving F1 scores of 1.00 and 0.98 on a test set, respectively. We also explore other machine learning methods such as ResNet-18 and Light Gradient Boosting Machine, then conduct comparative analyses of the results.

Key words: techniques: image processing – methods: data analysis – light pollution

1. Introduction

Light curves refer to the curves depicting changes in luminosity. The photometry of space objects is influenced by various factors, including the geometry involving the Sun, the space object, and the observer, as well as the object’s shape, orientation, and surface reflectance characteristics. Light curves are essential for studying the rotation state and characteristics of space objects. However, before obtaining light curves, it is necessary to preprocess the source data, which includes the removal of outliers and data contaminated by either stars or a cloud. This preprocessing step ensures the acquisition of high-quality observational data.

The usual preprocessing method often requires manual judgment. But when dealing with a large volume of data, this judgment is time-consuming and labor-intensive. Using machine learning for pattern recognition can significantly improve efficiency and save a substantial amount of time and effort. Machine learning has widespread applications in astronomy, including but not limited to predicting atmospheric seeing in optical observations (Ni et al. 2022), identifying active galactic nucleus (AGN) and pulsar candidates (Zhu et al. 2021), detecting outliers in astronomical images (Han et al. 2022), and classifying Gaia data (Bai et al. 2018).

Hinton & Salakhutdinov (2006) published a paper with two main points: (1) Artificial neural networks with multiple hidden layers exhibit exceptional feature learning capabilities. (2) Effectively overcoming training difficulties in deep neural networks can be achieved through “layerwise pre-training,” which introduced the field of deep learning (Zhou et al. 2017). In fact, there were even highly efficient deep learning models proposed before 2006, such as Convolutional Neural Networks (CNNs). In the 1980s and 1990s, some researchers published studies on CNNs in the field of pattern recognition, showing excellent performance in handwritten digit recognition (Lawrence et al. 1997; Neubauer 1998). However, at that time, CNN still performed poorly with large-scale data. It was not until 2012 that Krizhevsky et al. (2012) used an extended deep CNN to achieve the best classification results in the ImageNet Large Scale Visual Recognition Challenge (LSVRC), which brought CNNs into the limelight and gained increasing attention from researchers.

Cloud detection method has made significant breakthroughs in remote sensing and meteorology, with various theories and approaches proposed. Existing algorithms primarily focus on utilizing cloud spectral information, frequency data, spatial textures, and combining methods such as thresholding, clustering, artificial neural networks, and support vector machines (SVMs) for cloud detection (Zhang 2018). In the field of astronomy, Mommert (2020) used 18-layer ResNet (ResNet-18) and Light Gradient Boosting Machine (lightGBM) to determine the presence of clouds, and Wang et al. (2019) employed SVMs to assess a cloud's presence. Both lightGBM and SVMs demonstrated high discrimination accuracy. Accuracy denotes the proportion of correctly predicted samples to the total number of samples in classification tasks in this paper.

The purpose of this study is to use machine learning models to identify stellar contamination images and cloudy contamination images. A CNN is employed for the binary classification of stellar contaminated images and normal images in this study. Three methods including the lightGBM model, SVM, and ResNet-18 are utilized for the classification of "cloudy contamination." These classification techniques have been adapted to suit the requirements of our study.

2. Data

Data on space objects, including satellites and space debris, were observed at Yunnan Observatories (YNAO)' 1.2 m telescope. The telescope's field of view is $36' \times 36'$, and the CCD model is [Andor] DU888_BV(10687) with a size of 1024×1024 pixels.

Approximately 1,000,000 Flexible Image Transport System (FITS) images, taken of space objects, were obtained at YNAO during 2022. Manual labeling was performed on part of the observational data from 2022, resulting in around 30,000 images labeled as "normal," 1756 images labeled as "cloud," and 582 images labeled as "star." The label "star" indicates stellar contamination, while the label "cloud" indicates cloudy contamination. These labeled images are stored in a Structured Query Language (SQL) file, and they can be directly retrieved by using the corresponding label.

By removing images contaminated by a star and cloud, the data quality can be improved for further research and analysis. The following three images, Figures 1, 2, and 3, represent the three classes mentioned above.

A CNN was employed as a binary classification approach to classify stellar contamination images. This model achieved a classification success rate of 92.21% on the test set. Analysis suggests that image features are related to a space object and elongated star, which make it more reasonable to perform classification in a partial region that includes the space object. Therefore, an attempt was made to process the data in the "star" class to obtain stamp images with a size of 200×200 pixels. The moving target is located at the center of the stamp image (as shown in the Figure 4 captured from Figure 2). A CNN model trained on stamp images achieved 100% accuracy on the test set.

It was found that images' overall standard deviation increased when images contained very bright objects or stars, during the labeling process. This could lead to misclassifying normal images as having cloudy contamination. To ensure the quality of the data set and prevent such misclassifications during labeling, images were directly captured by the telescope in moving-target-tracking mode during nighttime. Cloud-contaminated images during two nights (2023 October 4th and 12nd) were captured.

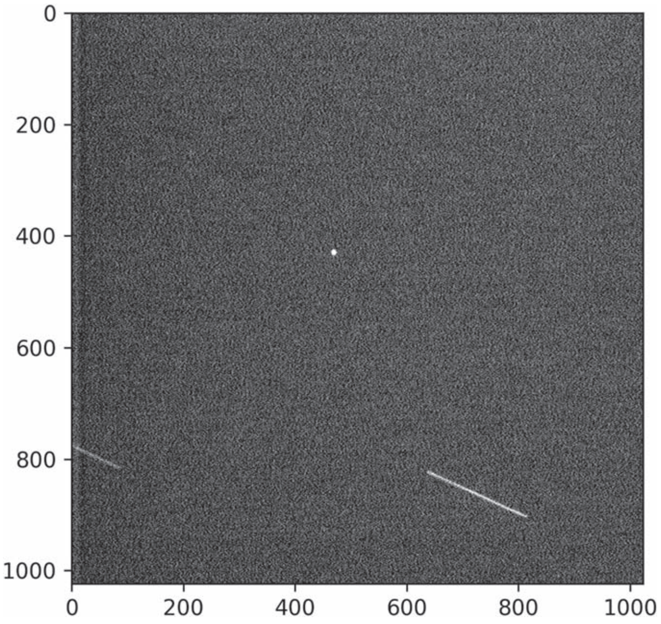

Figure 1. Normal.

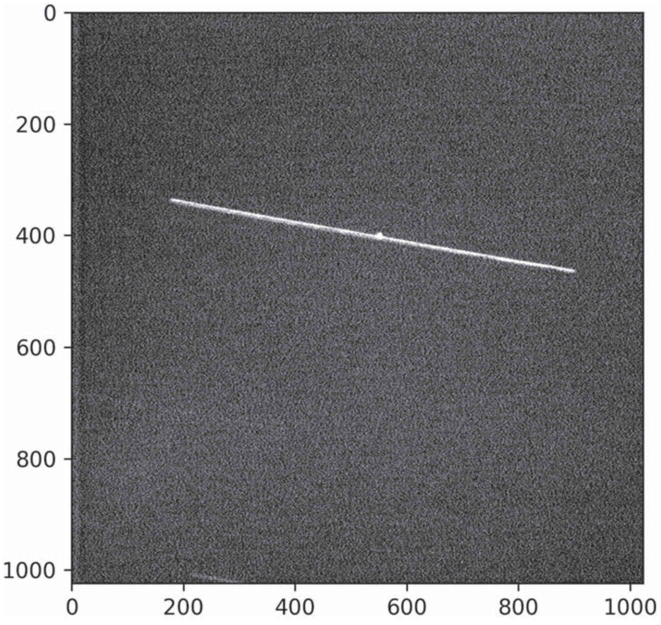

Figure 2. Star.

Finally, a machine learning data set was established, which included "normal" labeled data, "cloudy contamination" data, and the "stellar contamination" data (augmented by rotating

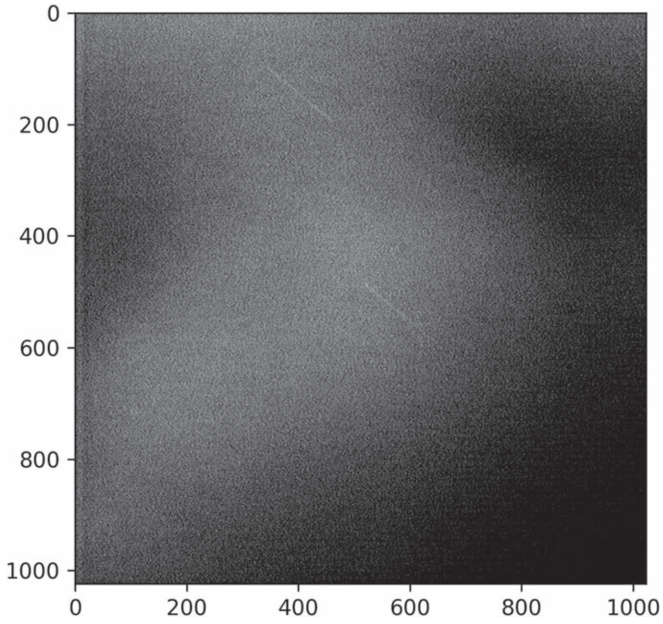

Figure 3. Cloud.

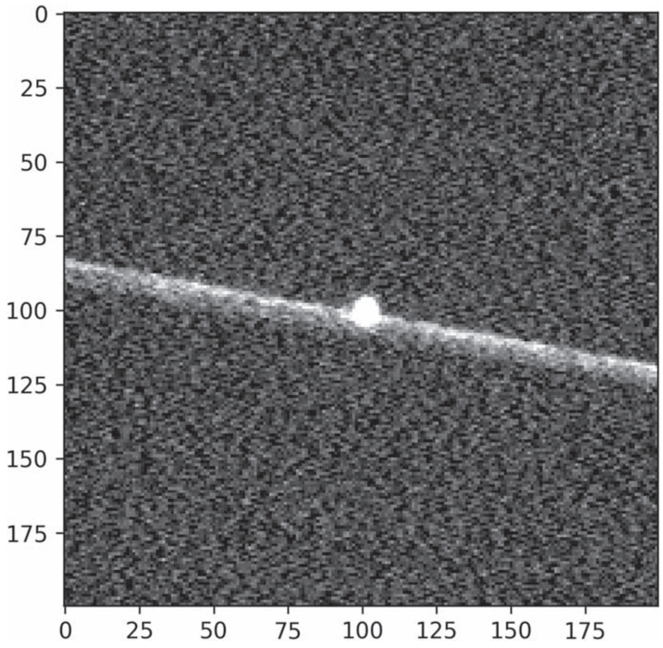

Figure 4. Stamp image captured from Figure 2.

and flipping). The data set sizes and how many images were used in each model are shown in Table 1. This process was unable to extract the required features from some images during data preparation, resulting in differences in the data set sizes used by different models.

3. Classification Method

A CNN was employed for binary classification for the stellar contamination case. Regarding the cloudy contamination situation, three different models, ResNet-18, lightGBM, and SVM, were individually tested for binary classification. The purpose of these four binary classification methods was to distinguish contaminated images from normal images, specifically separating stellar contamination images from normal images and cloudy contamination images from normal images.

3.1. Stellar Contamination

A CNN is utilized to perform binary classification on the pre-annotated data set, allowing the network to learn image features and subsequently classify images. A CNN, through the combination of convolution operations, padding, pooling, fully connected layers, and activation functions, effectively extracts features from images and conducts advanced pattern recognition. This ability has made CNNs widely applicable in computer vision tasks, including image classification, object detection, and image segmentation. In fields such as astronomy, CNNs can also be employed for image processing and pattern recognition tasks, assisting researchers in handling and analyzing celestial images.

For a large volume of data, a CNN can utilize gradient descent to find appropriate parameters, allowing the trained neural network to distinguish specific pattern types effectively.

3.1.1. CNN Model

We imported modules such as `Dense`, `Flatten`, `Conv2D`, `MaxPool2D`, and `Activation` from `tensorflow.keras.layers` and constructed a CNN structure as illustrated in Figure 5 below.

The first convolutional layer comprises five convolution kernels with a size of 4×4 pixels, and the default stride for convolution is one. The second convolutional layer consists of five convolution kernels with a size of 3×3 pixels, and the default stride for a convolution is one. The ReLU function is employed as the activation function. The Dense layer has an output unit count of one, and the activation function used is the sigmoid function. The loss function utilized is BinaryCrossentropy (`tensorflow.keras.losses.BinaryCrossentropy`), and the Adam optimizer is applied.

Ninety percent of the total 8090 images are used as the training set, with the remaining 10% serving as the test set, and a total of 10 epochs were trained. The training results are presented in Section 4.1.

3.1.2. CNN Model with Stamp Images

From the example image Figure 2, it can be observed that the class feature of stellar contamination is entirely determined by moving objects and star trails near the moving objects. The remaining parts of the images are considered invalid.

Table 1
Subtotals of All Classes in the Dataset and Images Used in Each Model

Classification	Count	Used in				
		CNN	CNN (Stamp Images)	ResNet	lightGBM	SVM
normal	31 558	4600	2448	15000	10334	15000
cloudy contamination (Oct 4th)	3199	0	0	3199	2527	3199
cloudy contamination (Oct 12nd)	302	0	0	302	231	302
stellar contamination	3490	3490	0	0	0	0
stellar contamination (stamp)	1566	0	1566	0	0	0

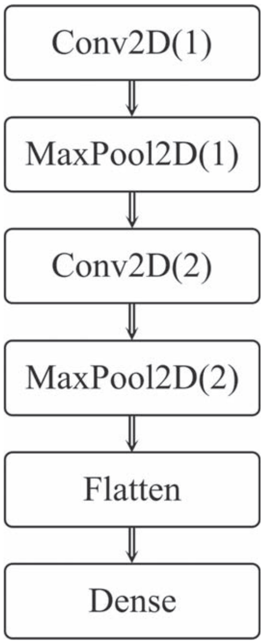

Figure 5. CNN structure.

Therefore, cropping out the region of interest known as the "stamp image" and training only on these stamp images can enhance classification accuracy.

Training the CNN with stamp images, the model parameters are the same as those in Section 3.1.1. As expected, there was a significant improvement in model accuracy, as shown in the training results in Section 4.1.

3.2. Cloudy Contamination

The ResNet model exhibits low accuracy for the classification of cloudy contamination and normal data. The lightGBM model achieves high accuracy, nevertheless with poor generalization capability, but the SVM model achieves high accuracy with strong generalization ability.

3.2.1. ResNet Model

The architecture of deep convolutional neural networks was introduced by He et al. (2016). ResNet primarily addresses the issues of gradient vanishing and exploding in deep neural networks, enabling the training of deeper neural networks. As a result, it achieved significant success in image classification and computer vision tasks. In this study, we attempt a ResNet-18 architecture similar to Michael Mommert's (Mommert 2020). We import modules such as `Resnet` and `BasicBlock` from `torchvision.model.resnet` and used these modules to construct the ResNet-18.

Eighty percent of the data are allocated for the training set, and the remaining 20% are used as the test set. We initiate the optimization process with a learning rate of 0.025, which decreases by a factor of 0.3 every second epoch for a total of 10 epochs. The training and testing results can be found in Section 4.2.

3.2.2. lightGBM Model

This section also draws reference from the work of Michael Mommert (Mommert 2020). The principles and characteristics of the lightGBM model can be found in the paper. The features of *Time derivatives* employed in the lightGBM model in this study differ from those used by Michael Mommert, since the data used in our study were obtained from the telescope in the moving object tracking mode and have a smaller field of view. The features are as follows:

1. Background-related features. Due to the impact of cloudy contamination on the background of FITS images, we utilize three features: median background brightness, mean background brightness, and background brightness standard deviation.

2. Time derivatives. Considering the relative motion of clouds within the field of view and the varying sizes of clouds, we opt to calculate the differences between the above-mentioned features at the current moment and those from 5 s ago, 10 s ago, and 15 s ago. This results in a total of nine new features.

3. Environment features. The analysis suggests that three features—solar elevation angle, lunar elevation angle, and lunar phase—might influence the imaging quality of cloudy layers. Actually, it was observed that these three features have minimal impact on the classification results due to the fact that nearly all images are captured at night, and the observation site experiences strong light pollution. In summary, a total of 15 features are considered.

The decision tree has a maximum depth of 5, with a total of 100 decision trees. Each tree has 20 leaf nodes, and the minimum number of samples on each leaf node is 20. L1 and L2 regularization strengths are set to 30 and 5, respectively. The training results of the model can be found in Section 4.3.

3.2.3. SVM Model

SVM is a supervised learning algorithm based on the principle of finding an optimal separating hyperplane in a high-dimensional space to distinguish data points from different categories. The core objective of SVM is to locate a separating hyperplane in such a way that it maximizes the distance between the nearest points of the two different data categories. These closest points are referred to as support vectors.

In practical applications, data are often not linearly separable. To address nonlinear data, SVM employs kernel functions to map the data into a higher-dimensional space, making it linearly separable. The linear kernel, polynomial kernel, and radial basis function are commonly used kernel functions. SVM also involves a regularization parameter that can adjust the margin of the separating hyperplane to prevent overfitting or underfitting the data. An SVM is used for classifying different types of galaxies based on their features or for identifying and categorizing various celestial objects, such as stars, galaxies, quasars, and other cosmic entities.

The Gray-Level Co-occurrence Matrix (GLCM) is a commonly used tool in image processing and texture analysis to describe the grayscale relationships between pixels in an image. It is possible to obtain textural features, such as *contrast*, *correlation*, *energy*, *entropy*, and other texture features, by calculating the GLCM. The calculation formula for the elements of the GLCM is as follows (Wang et al. 2019)

$$\begin{aligned} & g(i, j) \\ & = \#\{f(x_1, y_1) = i, f(x_2, y_2) = j|(x_1, y_1), (x_2, y_2) \in M \times N\}, \end{aligned} \tag{1}$$

in which x and y are coordinates within the image, i and j are the row and column indexes respectively of the matrix g, M and N are the sums of the rows and columns of image respectively, g is the GLCM of the image f, and # means the number of elements in the set. The distance between (x_1, y_1) and (x_2, y_2) is d, and the angle of the two points with the abscissa axis is q.

Ulaby et al. (1986) found that *1.contrast*, *2.IDM*, *3.energy*, and *4.correlation* are uncorrelated, and these four features are easy to compute and provide high classification accuracy. Baraldi & Panniggiani (1995) conducted a detailed study of six texture features and identified *1. contrast* and *5. entropy* as the two most important features. We selected 10 normal images and 10 cloudy contamination images then calculated these five features for these 20 images after computing the GLCM with $d = 1$, $q = 0$. Additionally, we calculated *6.G*, the gray-value inconsistency of an image. The formulas for these six features are as follows (Liu et al. 2003):

1. Contrast.

$$\mathrm{CON} = \sum_{i=1}^{M_g}\sum_{j=1}^{N_g}(i-j)^2 g(i, j). \tag{2}$$

M_g and N_g represent the total number of rows and columns respectively in matrix g.

2. Inverse Difference Moment.

$$\mathrm{IDM} = \sum_{i=1}^{M_g}\sum_{j=1}^{N_g}\frac{g(i, j)}{1+(i-j)^2}. \tag{3}$$

3. Energy.

$$\mathrm{ENE} = \sum_{i=1}^{M_g}\sum_{j=1}^{N_g} g(i, j)^2. \tag{4}$$

4. Correlation.

$$\mathrm{COR} = \sum_{i=1}^{M_g}\sum_{j=1}^{N_g}\frac{(i-\mu)(j-\mu)g(i, j)^2}{\sigma^2}. \tag{5}$$

5. Entropy.

$$\mathrm{ENT} = -\sum_{i=1}^{M_g}\sum_{j=1}^{N_g} g(i, j)log(g(i, j)). \tag{6}$$

6.G. The gray-value inconsistency of an image is an important index to represent the image's background gray value. The image's gray-value inconsistency is defined as (Wang et al. 2019)

$$G = 10lg(\frac{P_s}{P_n}), \tag{7}$$

in which P_s and P_n are respectively the maximum and minimum standard deviation of local images. This study selects the template of 9 × 9 pixels that can be spread all over the image.

After normalizing the computed feature values to a range between 0 and 1, it was observed that these six features can effectively distinguish between the two categories, as shown in the Figure 6 below. Therefore, this study will utilize these six features for training the SVM model. The SVM model training and testing results are displayed in Section 4.4.

4. Results

In this section, the results for the four mentioned models are presented. Table 2 below shows the accuracy of these models on the test set and the training set.

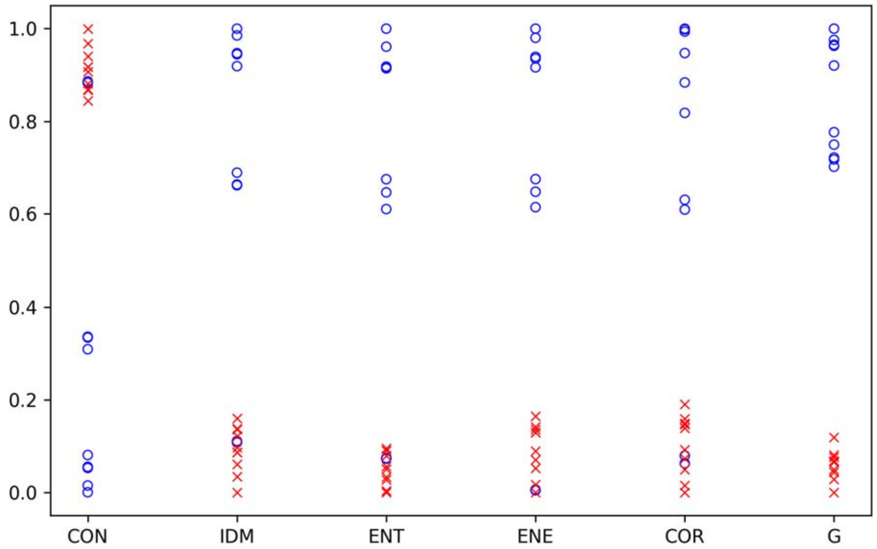

Figure 6. The distribution of the six features for the 20 images. Blue: normal images, while red: cloud-contaminated images.

Table 2
Accuracies on the Training Set and Test Set of All Four Models

Contamination	Model	Training Set	Test Set
stellar	CNN	0.9389	0.9221
	CNN with stamp images	1.0000	1.0000
cloudy	ResNet	0.8000	0.8000
	lightGBM	1.0000	0.9995
	SVM	0.9753	0.9712

Note. The accuracies of the lightGBM model shown above are very high, but this model has weak generalization ability.

4.1. Identify Stellar Contamination

4.1.1. Training and Testing CNN Model

From the growth curve in the above Figures 7 and 8, it can be observed that the CNN model trained with stamp images converges faster and exhibits better performance on the test set. We import module `f1_score` from `sklearn.metrics` and use it to calculate F1 scores. The F1 scores for these two different cases are 0.90 and 1.00, respectively. Their confusion matrices on the test set are illustrated in Figures 9 and 10 below. The confusion matrix is a visual tool designed for supervised learning. In a confusion matrix, each column represents the predicted results, while each row corresponds to the truth. "Positive" represents contaminated images, while "Negative" represents normal images. In Figure 9, the CNN model incorrectly classified 51 contaminated images as normal images and 12 normal images as contaminated images. In Figure 10, the CNN model trained with stamp images made all correct predictions.

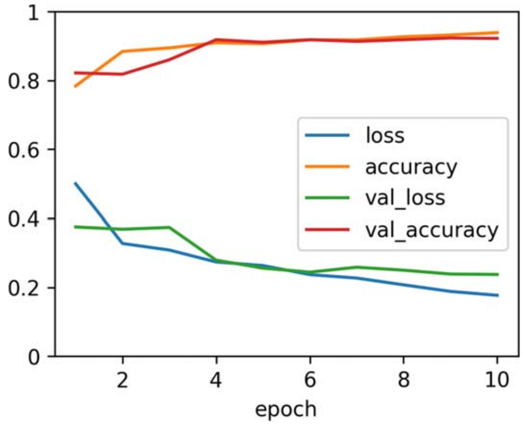

Figure 7. Growth curves of CNN model.

4.2. Identify Cloudy Contamination

4.2.1. Training and Testing ResNet-18

The model's accuracy on the test set is below 0.8 with an epoch being set to 10, as shown in Figure 11, which is lower than Mommert's accuracy of 0.85. Our analysis suggests that this could be due to the smaller field of view in the images and the relatively faster motion of clouds within the field of view, resulting in less distinct cloudy features.

4.2.2. Training and Testing LightGBM Model

The lightGBM model, when trained with data from 2022 October 4, achieved an impressive classification accuracy of 99.95%, as shown in the confusion matrix in Figure 12. However, it displayed poor generalization with nearly zero accuracy when this model was used to classify data from

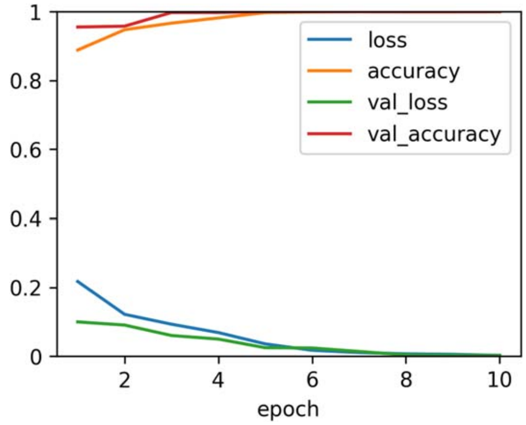

Figure 8. Growth curves of CNN model with stamp images.

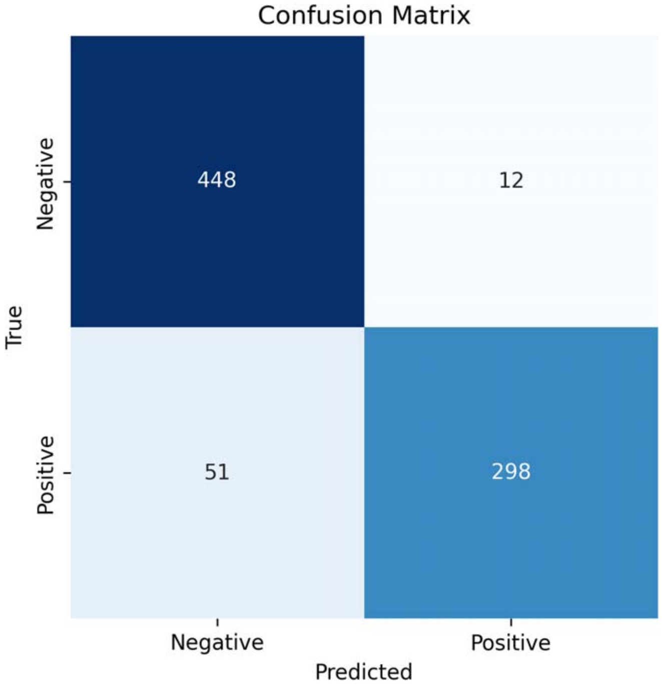

Figure 9. Confusion matrix of CNN model.

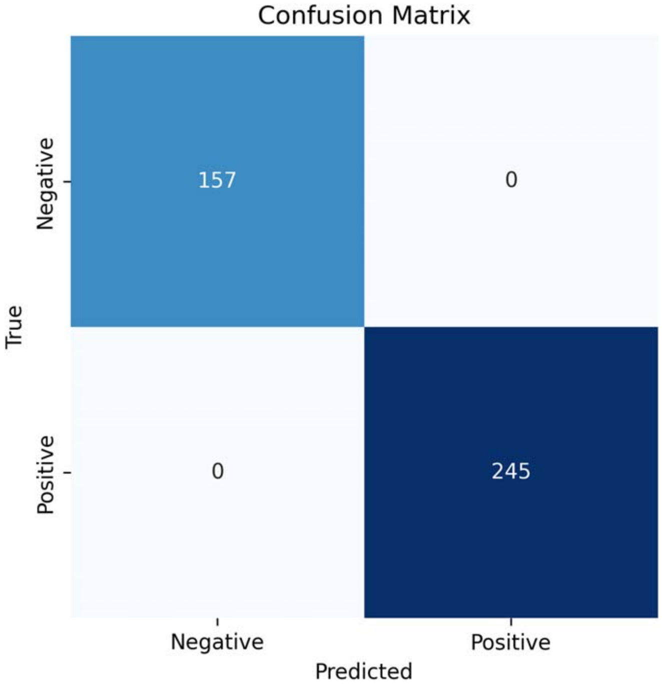

Figure 10. Confusion matrix of CNN model with stamp images.

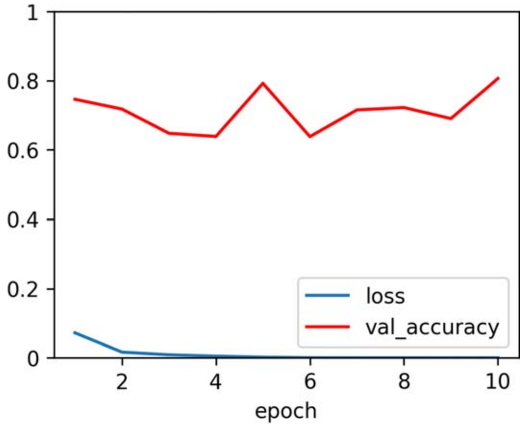

Figure 11. Growth curve of ResNet-18 model.

October 12. But when data from these 2 days were combined for model training, the consequent model performed well when applied to classify images from October 12 again, as indicated in the confusion matrix in Figure 13. It mistakenly recognized only two contaminated images as normal.

The analysis suggests that learning cloudy imaging features from data on a single day or a few days may not be able to capture the general characteristics of cloudy layers. Therefore, long-term data accumulation is needed to improve the generalization ability of the lightGBM model.

4.2.3. Training and Testing SVM Model

The SVM model, trained with 15,000 samples of normal data and 3,199 samples (October 4) of cloudy contamination data, achieved an accuracy of 97.12% on the test set. This model's accuracy for classifying data from October 12 remains high at 98.34%, indicating that the SVM classification model exhibits good generalization ability. The confusion matrices for the SVM model on the test set and the data from October 12 can be found in Figures 14 and 15, respectively. The respective recall rates are 89.44% and 98.34%.

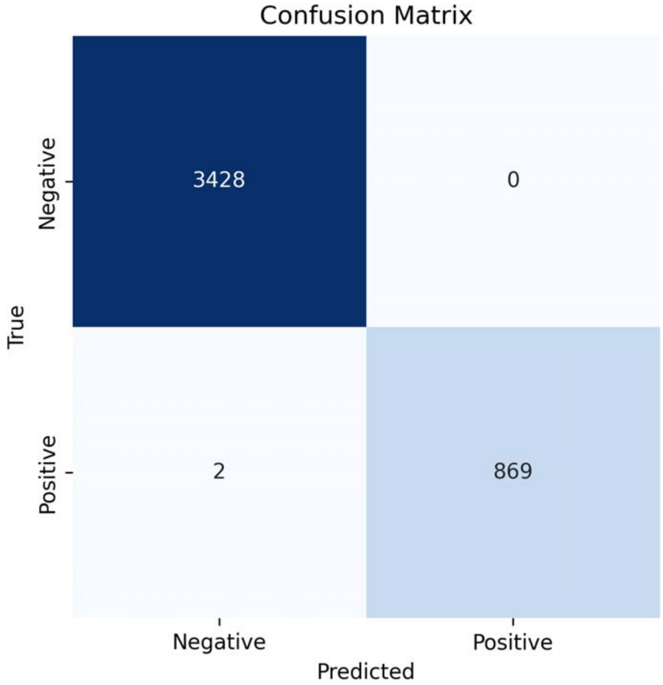

Figure 12. Confusion matrix.

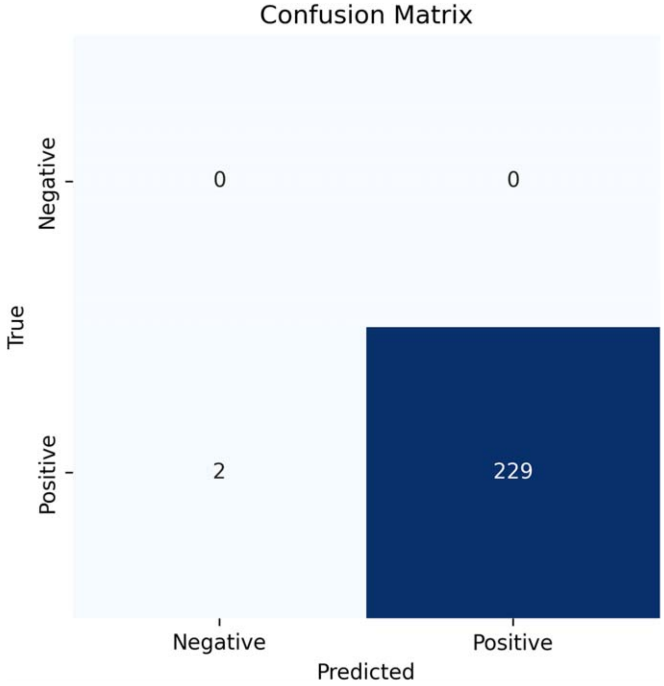

Figure 13. Confusion matrix on data from October 12.

5. Conclusions

A data set for machine learning training has been obtained through manual annotation of a portion of the YNAO’s 2022 observational data and additional data from 2023. The data set with a total 40,115 images is of high quality, well-classified, and easy to use.

From the results of the two CNN models, it is evident that the CNN model trained with stamp images achieves higher

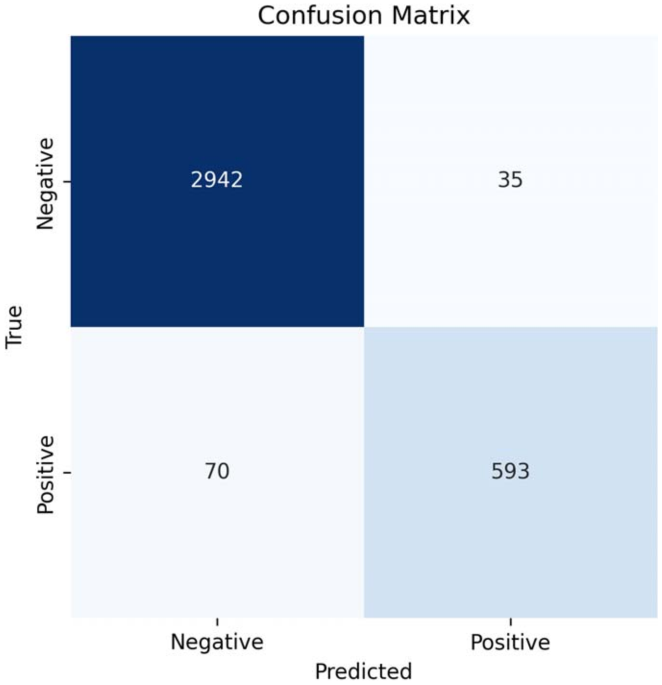

Figure 14. Confusion matrix.

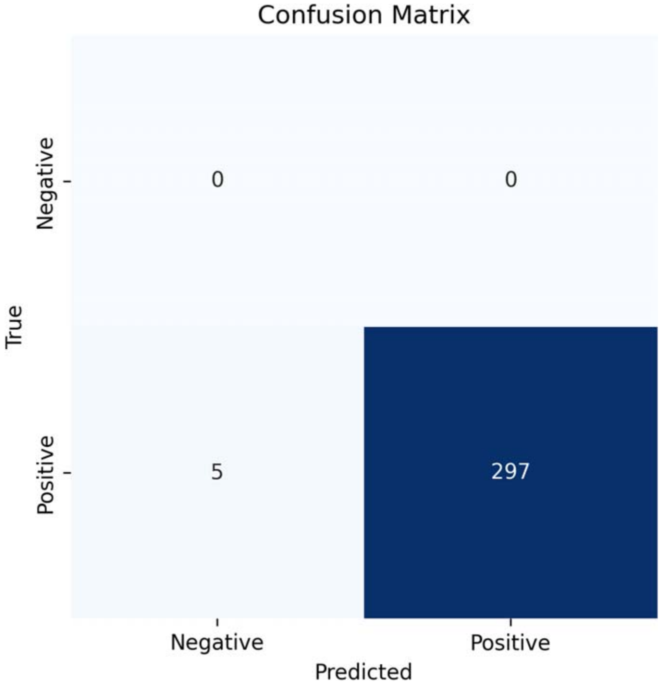

Figure 15. Confusion matrix on data from October 12.

accuracy, around 100%, meeting the practical requirements to identify stellar contamination from normal images.

Among the three models for identifying cloudy contamination, ResNet-18 has lower accuracy on both the training and testing data sets, but the other two models perform well. The lightGBM model exhibits poor generalization and requires long-term data accumulation to improve, while the SVM model has stronger generalization, maintaining high classification accuracy of 97.12% for new samples.

Acknowledgments

This work was funded by the National Natural Science Foundation of China (NSFC, Nos. 12373086 and 12303082), CAS "Light of West China" Program, Yunnan Revitalization Talent Support Program in Yunnan Province, National Key R&D Program of China, and Gravitational Wave Detection Project No.2022YFC2203800. Thanks go to He Zhao from Purple Mountain Observatory for the assistance. Thanks also go to Michael Mommert from Lowell Observatory for the open-source code.

ORCID iDs

Hui Li https://orcid.org/0009-0005-8405-3891

References

Bai, Y., Liu, J.-F., & Wang, S. 2018, RAA, 18, 118
Baraldi, A., & Panniggiani, F. 1995, ITGRS, 33, 293
Han, Y., Zou, Z., Li, N., & Chen, Y. 2022, RAA, 22, 76
He, K.-M., Zhang, X.-Y., Ren, S.-Q., & Sun, J. 2016, in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (Las Vegas, NV: IEEE), 770
Hinton, G., & Salakhutdinov, R. R. 2006, Science, 313, 504
Krizhevsky, A., Sutskever, I., & Hinton, G. E. 2012, in Proc. Adv. Neural Inf. Proc. Syst. (New York: Association for Computing Machinery), 1097
Lawrence, S., Giles, C. L., Tsoi, A. C., & Back, A. D. 1997, ITNN, 8, 98
Liu, L.-F., Chen, Y.-H., & Li, J. 2003, Remote Sensing Technology and Application, 18, 441
Mommert, M. 2020, AJ, 159, 178
Neubauer, C. 1998, ITNN, 9, 685
Ni, W.-J., Shen, Q.-L., Zeng, Q.-T., et al. 2022, RAA, 22, 155
Ulaby, F. T., Kouyate, F., Brisco, B., et al. 1986, ITGRS, 24, 235
Wang, L.-W., Jia, P., Cai, D.-M., & Liu, H.-G. 2019, ChJAA, 43, 128
Zhang, M.-B. 2018, Research on Cloud Detection Method of High Resolution Remote Sensing Image Master's thesis, Dalian Maritime Univ.
Zhou, F.-Y., Jin, L.-P., & Dong, J. 2017, Chin. J. Comput., 40, 1229
Zhu, K.-R., Kang, S.-J., & Zheng, Y.-G. 2021, RAA, 21, 15