Identification of ground monitoring cloud images using the CNN Learning Transfer Model

Silfia Andini[1], Erni Rouza[2], Luth Fimawahib[2], Satria Riki Mustafa[2], Ahmad Fathoni[2], Teguh Iman Hermanto[3], Agus Perdana Windarto[4*]

[1]Universitas Putra Indonesia YPTK Padang, West Sumatra, Indonesia
[2]Universitas Pasir Pengaraian, Riau, Indonesia
[3]Sekolah Tinggi Teknologi Wastukancana, Indonesia
[4]STIKOM Tunas Bangsa, Pematangsiantar, North Sumatra, Indonesia

Email: [1]silfiaandini68@upiyptk.ac.id,[2]ernirouzait@gmail.com, [2]luthfimawahib@gmail.com, [2]satriarikimustafa@gmail.com, [2]ahmadfathoniupp@gmail.com, [3]teguh@stt-wastukancana.ac.id, [4]agus.perdana@amiktunasbangsa.ac.id*

Abstract. This research aims to produce the best architectural model using a Convolutionary Neural Network from the results of the detection of a cloud image classification. The source dataset is composed of 11 cloud categories, covering 2545 cloud images, and is used by Cirrus Cumulus stratus Nimbus (CCSN). This study makes use of the basic google architecture to retrain the optimal Convolutional Neural Network almost instantly by transferring education. The classification process uses two phases, namely training and testing, based on the modified Googlenet architecture. During the training phase, the dataset is divided into three parts: training data of 70%, validation data of 15% and test data of 15%. During the test phase, however, two experiments are performed to classify the cloud images, one of which is composed by 10 cloud types which can be randomly selected. The results showed that the precision produced during training was 44.5%. In the two tests the results are 75%, with an average error value of 0.25. The result is 75% in the test phase.

Keywords: Cloud image, classification, googlenet, CNN, CCSN dataset.

1. Introduction

Today's digital era is in the world. An era in which virtually all aspects of human life are closely related to computer technology. As time goes by, people continue to develop knowledge and technology in order to support and ease their work. Artificial intelligence or more known as Artificial Intelligence is still developing (AI) [1]–[3]. Cloud detection is critical for a large number of tasks for remote optical data sensing. For example, clouds mask the Earth's surface and provide incorrect reflectance values for ground-based targets [4]. Remote sensing clouds have certain specific features including luminosity, color, texture, shape, etc [5]. Cloud detection techniques are employed by cloud investigators with physical cloud parameters such as (a) shape attributes; (b) fusion of cloud net multi-stage convolutionary features; (c) color transition; (d) cloud densities; (e) cloud shadows [6]. It's the main step of many object recognition and computer vision tasks to extract effective features. Several researchers therefore focused on robust features for a range of tasks of image classification [7]. Currently, much attention is paid to learning algorithms and revolutionary networks (CCN). The algorithm provides the image directly to the convolutionary neural networks, and the algorithm removes the most important features of the image [8]. In the findings indicate that CNN functionality extracted from profound learning must be taken into account in the most visual recognition tasks [9]. To identify cloud image classifications, priority knowledge is needed, which is learned through identified cloud image types with a similar composition. The data sets of the CCSN (Cirrus Cumulus Stratus Nimbus) divides into 11 different cloud genus (main group): Ci = Cirrus; Cs = Cirrostratus; Cc = Cirrocumulus; Ac = Altocumulus; As = Altostratus; Cu = Cumulus; Cb = Cumulonimbus; Ns = Nimbostratus; Sc = Stratocumulus; St = Stratus; Ct = Contrail. In this experiment, the CNN is used to classify image types of cloud-based objects. The focus is on modeling for cloud-type object classification. Highlights of the paper are:

a) To address the cloud classification problem, we propose a CNN Learning Transference Model that incorporates state of the art Transfer learning technology.
b) We conducted cloud experiments collected by the World Metroliferative Organization and the results showed that the Model for Learning Transfer has been effective and potential.

2. Methods

2.1. Dataset

For cloud detection purposes, we use the Cirrus Cumulus Stratus Nimbus (CCSN). This dataset contains 11 categories. The data set of CCSN includes 2545 images of the Cloud. The representatives of each category are Ci = cirrus; Cs = cirrostratus; Cc = cirrocumulus; Ac = altocumulus; As = altostratus; Cu = cumulus; Cb = cumulonimbus; Ns = nimbostratus; Sc = stratocumulus; St = stratus; Ct = contrail. All pictures are 256 / 256 pixels of fixed resolution with JPEG format.

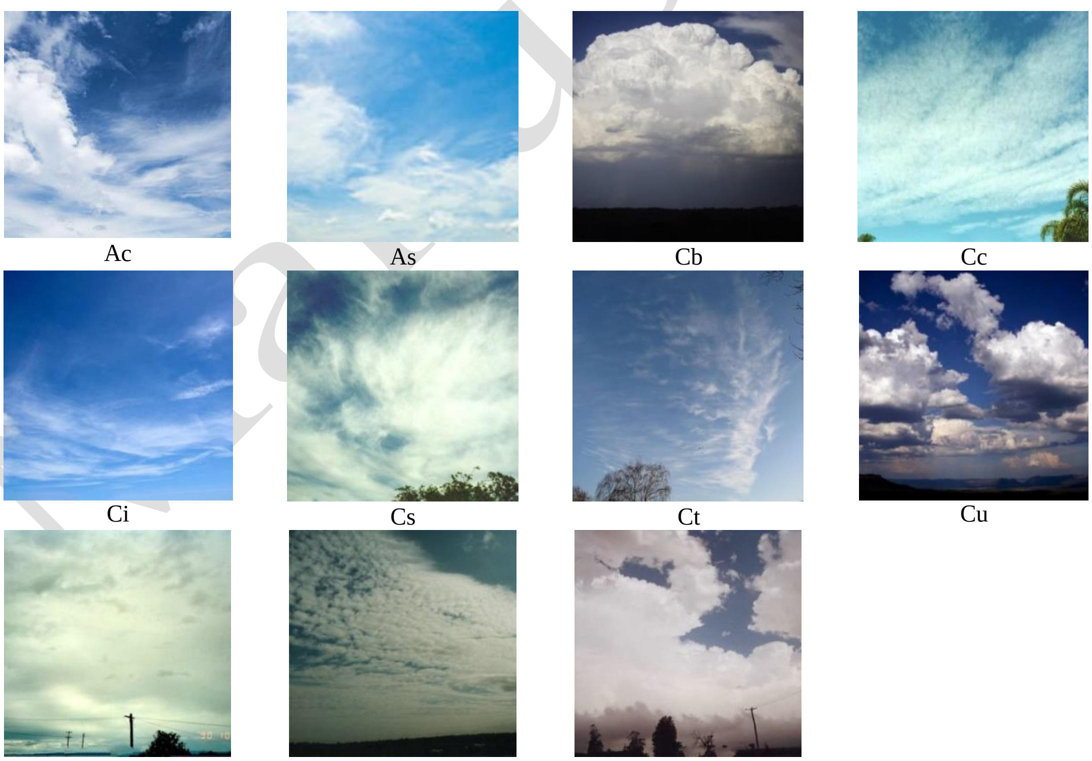


Figure 1. Example of a Cirrus Cumulus Stratus Nimbus (CCSN) cloud type

2.2. CNN Architecture

Convolutional Neural Networks (CNN) is a deep learning network architecture that can automatically learn how to represent picture features. CNN is part of a particular category of methods of the neural network. The CNN consists of three layer types. The layers can be configured, combined or completely connected. Usually, CNN is two-part structured. The first part of the system, called extraction of functions, uses coalescing and grouping layers. The second part is a classification that uses layers fully connected. The layer description is shown in table 1 for further details [7].

Table 1. The description of CNN layers

Layer Name	Layer Description
Convolutional layer	The convolution process was performed on the input image using a set of filters called the kernel. The characteristic map is the operation output.
Pooling layer Fully	In this layer, the convolutionary layer output was reduced while the main information contained in the input layer was saved. The bundling process may be performed (max or average), the most common kind of bundling is the choice of maximum value.
Fully Connected Layers	The extracted features of these layers were used for the classification task in the previous layers.

3. Results And Discussion

A well-trained image classification network can accurately classify images into categories. The trained network's limitation is that it can only classify the trained object to be classified. Assume GoogleNet has been trained to classify over 1000 different objects. If there are objects that are not among those 1000, the network will fail to classify them. As a result, we require a network that can train and classify new data sets. This is referred to as transfer learning. Transfer learning is a deep learning technique that employs existing networks as a starting point for learning new tasks [7]. Transfer learning involves removing a specific task layer from an existing network and adding a new layer so that it can be trained to learn new features for some of the new tasks. Then, using the new data set, this new layer is trained, validated, and tested [6], [8]–[11]. If the network is properly trained with appropriate data sets, it will be capable of classifying recently learned objects.

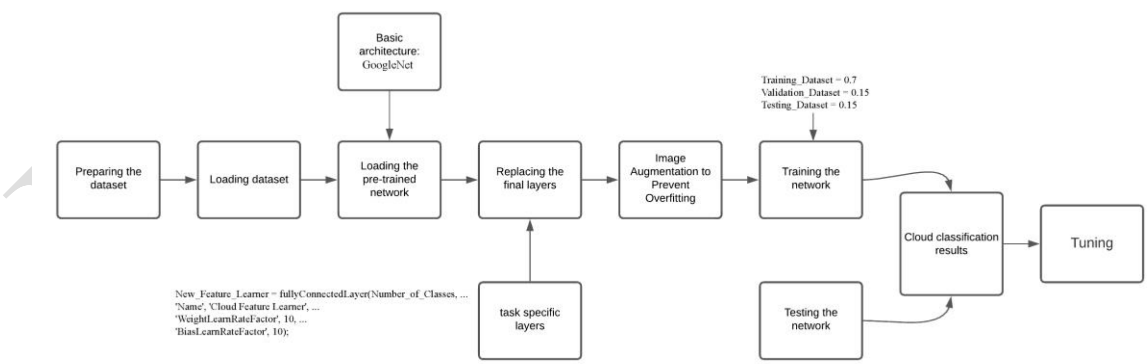


Figure 2. Architecture Identification of ground monitoring cloud imagery using the CNN Learning Transfer Model

Figure 2 shows that a neural network is trained to classify images by making certain changes to the network architecture. GoogleNet is one of the most popular neural convolution Networks in this study.

This article shows the classification of the cloud type by the Google Net neural network (CNN). The layers are modified. The modified architecture layers are the FullyConnectredLayer and ClassificationOutputLayer layers. The FullyConnectredLayer layer renames the layer to 'Cloud Feature Learner', WeightLearnRateFactor = 10, and BiasLearnRateFactor = 10. Whereas the ClassificationOutputLayer layer changes the layer name to Cloud Classifier. Following are the modifications that have been made are shown in Figure 3 by using the Matlab 2021a programming language.

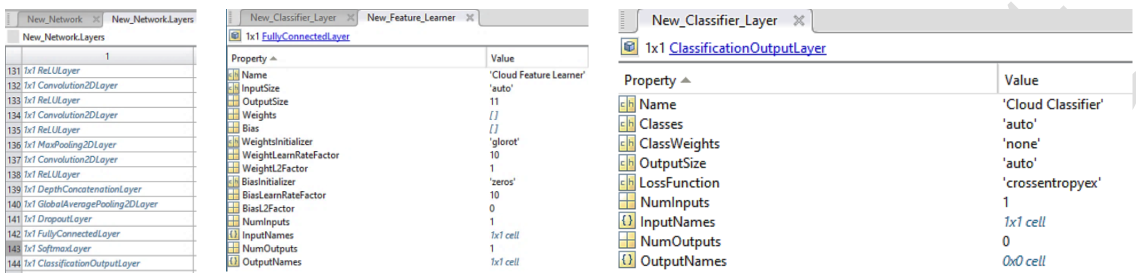


Figure 3. Modify the GoogleNet architecture layer

Following the modification of the GoogleNet architecture screen, the training process is carried out by dividing the data into three parts, namely Training Dataset of 70%, Validation Dataset of 15%, and Testing Dataset of 15%, which is carried out at random on 2545 cloud images divided into 11 cloud types (Figure 1). The dataset sharing program code is shown below, along with the results of dividing the dataset into three parts (Training Dataset, Validation Dataset, and Testing Dataset), as shown in Figures 4 and 5.

```
Dataset = imageDatastore('Dataset', 'IncludeSubfolders', true, 'LabelSource', 'foldernames');
[Training_Dataset, Validation_Dataset, Testing_Dataset] = splitEachLabel(Dataset, 0.7, 0.15, 0.15);
```

Figure 4. Kode script dari pembagian dataset

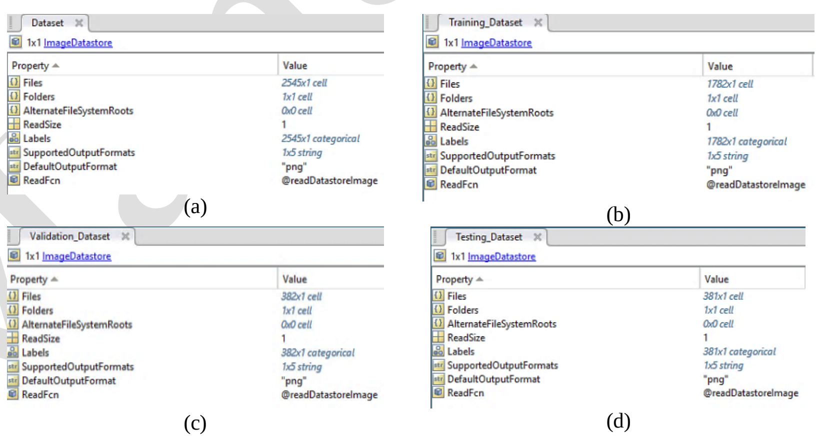


Figure 5. The results of dividing the dataset into three parts (Training Dataset, Validation Dataset, Testing Dataset).

Figure 5 shows that the number of datasets for Training Dataset (b) is 1782 (70%), the number of datasets for Validation Dataset (c) is 382 (15%), and the number of datasets for Testing Dataset (d) is 381 (15%). (a) The total number of datasets is 2545. The following are the outcomes of the training process, which was carried out with the following parameters:

```
Minibatch_Size = 11;
trainingOptions='sgdm';
MaxEpochs= 10;
InitialLearnRate= 0.0001;
Shuffle= every-epoch;
ValidationData= Resized_Validation_Dataset;
ValidationFrequency= Validation_Frequency;
Verbose= false.
```

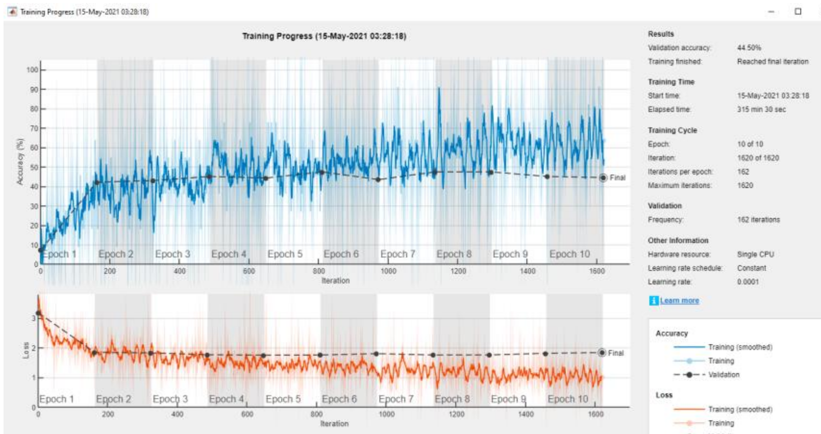
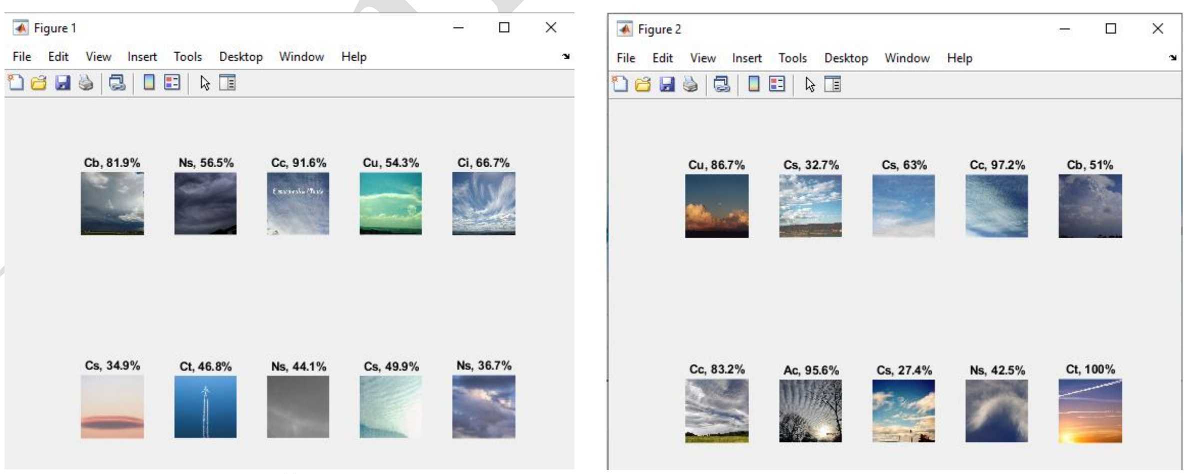


Figure 6. The training process of the GoogleNet architecture

Figure 6 explains that there are two types of plots. The upper plot represents accuracy or acc, while the bottom plot represents error or loss. The lower plot (error plot) shows that there is a decrease in the red line, but there is a flat epoch starting from epoch 2-10, whereas the upper plot (plot accuracy) shows an increase in epoch 1–10. This demonstrates that the formed fit model already has a fairly good accuracy value to use. The training process yielded a 44.5 percent accuracy after 315 minutes, 30 seconds, and 1620 iterations. Then, repeat the testing process with 2 trials based on the training process. The results of five experiments, as shown in Figure 7, are as follows.

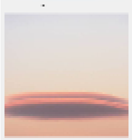


(a) (b)

Figure 7. Classification test results with 5 trials

Figure 7 shows a tabular explanation of each experimental result using ten different cloud images.

Table 1. Results of the First Experiment

No	Cloud images	Cloud type	Prediction Results	Probability	Accuracy	Conclusion
1		Cb	Cb	0.819	81.9%	True
2		Ns	Ns	0.565	56.5%	True
3		Cc	Cc	0.916	91.6%	True
4		Cu	Cu	0.543	54.3%	True
5		Ci	Ci	0.667	66.7%	True
6		Ac	Cs	0.349	34.9%	False
7		Ct	Ct	0.468	46.8%	True
8		Ns	Ns	0.441	44.1%	True
9		Cc	Cs	0.49.9	49.9%	False
10		Ns	Ns	0.367	36.7	True

Note: Ci = cirrus; Cs = cirrostratus; Cc = cirrocumulus; Ac = altocumulus; As = altostratus; Cu = cumulus; Cb = cumulonimbus; Ns = nimbostratus; Sc = stratocumulus; St = stratus; Ct = contrail

In the first experiment, there are two errors with an error value of 0.2 (80% accuracy) among the ten classified cloud types.

Table 2. Results of the Second Experiment

No	Cloud images	Cloud type	Prediction Results	Probability	Accuracy	Conclusion
1		Cu	Cu	0.887	88.7%	True
2		Sc	Cs	0.327	32.7%	False
3		Cs	Cs	0.630	63%	True
4		Cc	Cc	0.972	97.2%	True
5		Cb	Cb	0.510	51%	True
6		Cc	Cc	0.832	83.2%	True
7		Ac	Ac	0.956	95.6%	True
8		Sc	Cs	0.274	27.4%	False
9		Ci	Ns	0.425	42.5%	False
10		Ct	Ct	0.1	100%	True

Note: Ci = cirrus; Cs = cirrostratus; Cc = cirrocumulus; Ac = altocumulus; As = altostratus; Cu = cumulus; Cb = cumulonimbus; Ns = nimbostratus; Sc = stratocumulus; St = stratus; Ct = contrail

In the first experiment, there are three errors with an error value of 0.3 (70% accuracy) among the ten classified cloud types. The average truth accuracy from the two experiments is 75%, with an average error value of 0.25. As a result, even though the accuracy value in training is less than 50%, the model obtained in this experiment produces a high level of accuracy in test results.

4. Conclusions

Based on the experimental results, it is possible to conclude that the architectural modification on GoogleNet for cloud type classification cases can be used with an accuracy value greater than 70%. There may be prediction errors between several types of clouds because they have nearly identical appearances, namely Stratocumulus, Cirrostratus, Cirrus, and Nimbostratus, resulting in a prediction error. Meanwhile, different types of clouds can be accurately predicted because they have a distinct appearance, both in terms of shape and color, making them easier to predict.

References

[1] Budiharjo, T. Soemartono, A. P. Windarto, and T. Herawan, "Predicting tuition fee payment problem using backpropagation neural network model," *Int. J. Adv. Sci. Technol.*, vol. 120, pp. 85–96, 2018, doi: 10.14257/ijast.2018.120.07.

[2] Budiharjo, T. Soemartono, A. P. Windarto, and T. Herawan, "Predicting School Participation in Indonesia using Back-Propagation Algorithm Model," *Int. J. Control Autom.*, vol. 11, no. 11, pp. 57–68, 2018.

[3] A. Waluyo, H. Jatnika, M. R. S. Permatasari, T. Tuslaela, I. Purnamasari, and A. P. Windarto, "Data Mining Optimization uses C4.5 Classification and Particle Swarm Optimization (PSO) in the location selection of Student Boardinghouses," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 874, no. 1, pp. 1–9, 2020, doi: 10.1088/1757-899X/874/1/012024.

[4] D. Tuia, B. Kellenberger, A. Pérez-Suay, and G. Camps-Valls, "A deep network approach to multitemporal cloud detection," *Int. Geosci. Remote Sens. Symp.*, vol. 2018-July, pp. 4351–4354, 2018, doi: 10.1109/IGARSS.2018.8517312.

[5] H. Liu, D. Zeng, and Q. Tian, "Super-pixel cloud detection using Hierarchical Fusion CNN," *arXiv*, 2018.

[6] S. Mahajan and B. Fataniya, "Cloud detection methodologies: variants and development—a review," *Complex Intell. Syst.*, vol. 6, no. 2, pp. 251–261, 2020, doi: 10.1007/s40747-019-00128-0.

[7] M. G Alaslni and L. A. Elrefaei, "Transfer Learning with Convolutional Neural Networks for IRIS Recognition," *Int. J. Artif. Intell. Appl.*, vol. 10, no. 5, pp. 49–66, 2019, doi: 10.5121/ijaia.2019.10505.

[8] E. Cromwell and D. Flynn, "Lidar cloud detection with fully convolutional networks," *Proc. - 2019 IEEE Winter Conf. Appl. Comput. Vision, WACV 2019*, pp. 619–627, 2019, doi: 10.1109/WACV.2019.00071.

[9] C. Gonzales and W. Sakla, "Semantic Segmentation of Clouds in Satellite Imagery Using Deep Pre-trained U-Nets," *Proc. - Appl. Imag. Pattern Recognit. Work.*, vol. 2019-Octob, 2019, doi: 10.1109/AIPR47015.2019.9174594.

[10] N. Li, H. Hao, Q. Gu, D. Wang, and X. Hu, "A transfer learning method for automatic identification of sandstone microscopic images," *Comput. Geosci.*, vol. 103, no. September 2016, pp. 111–121, 2017, doi: 10.1016/j.cageo.2017.03.007.

[11] I. Nurtanio, Z. Zainuddin, and B. H. Setiadi, "Cloud Classification Based on Images Texture Features," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 676, no. 1, 2019, doi: 10.1088/1757-899X/676/1/012015.