Classification of Cumulonimbus Cloud Formation based on Himawari Images using Googlenet

Mohammad Rizal Abidin[1], Dian C. R. Novitasari[2], Hani Khaulasari[3], Fajar Setiawan [4]

Department of Mathematics, Faculty of Science & Technology, UIN Sunan Ampel, Surabaya,
Jl. Ahmad Yani No.117, Jemur Wonosari, Kec. Wonocolo, Surabaya, 60237, Jawa Timur,
Indonesia
Email: [1]mrizalabidinn@gmail.com, [2]diancrini@uinsby.ac.id, [3]hanikhaulasari@gmail.com,
[4]fajar.setiawan@bmkg.go.id

Abstract. *Awan cumulonimbus (Cb) merupakan awan yang berbahaya bagi banyak aktivitas manusia. Untuk mengurangi efek tersebut, diperlukan sistem untuk mengklasifikasikan pembentukannya. Pembentukan awan Cb dapat dilihat pada citra Himawari-8 IR. Tujuan penelitian ini adalah membuat sistem klasifikasi formasi awan Cb dengan citra Himawari-8 IR Enhanced menggunakan metode CNN model GoogleNet. Total data yang akan digunakan sebanyak 2026 data citra. Pengujian parameter dilakukan pada model CNN Googlenet pada penelitian ini yaitu rasio sebaran data 90:10 dan 80:20. Probabilitas drop out 0,6, 0,7, dan 0,8. dan batch size 8, 16, 32, dan 64. Uji coba yang dilakukan pada penelitian ini menghasilkan nilai sensitivitas 100,00%, akurasi 99,00%, dan spesifisitas 99,60% yang diperoleh dari distribusi data eksperimen sebesar 90: 10, probabilitas 0,8, dan ukuran batch 8*
Kata Kunci: *Batch Size, Cumulonimbus, CNN,GoogleNet, Himawari-8 IR Enhanced*

Abstrak. *Cumulonimbus clouds (Cb) are clouds that are dangerous for many human activities. To reduce this effect, we need a system to classify formations. The formation of Cb clouds can be seen in the Himawari-8 IR image. This research aims to create a Cb cloud classification system with Himawari-8 IR Enhanced imagery using the GoogleNet model CNN method. The total data to be used is 2026 image data. Parameter testing was carried out on the CNN Googlenet model in this study, namely the data distribution ratio of 90:10 and 80:20. The probability of dropout is 0.6, 0.7, and 0.8. and batch sizes of 8, 16, 32, and 64. The results of the trials conducted in this study yielded a sensitivity value of 100.00%, an accuracy of 99.00%, and a specificity of 99.60% obtained from the experimental data distribution of 90: 10, probability 0.8, and batch size 8*
Keywords: *Batch Size, Cumulonimbus, CNN, GoogleNet, Himawari-8 IR Enhanced*

1. Introduction

Cumulonimbus clouds (Cb) are a type of dangerous cloud that can cause extreme weather such as heavy rain, lightning, and thunder. Cb clouds can form if there is very intensive heating by the sun in moist air in areas where the winds meet (convergent) and in areas where strong winds blow (troughs) [1]. To anticipate the presence of Cb clouds, it is necessary to classify the formation of Cb clouds. The process of forming clouds into Cb clouds can be seen from the captured Himawari-8 IR Enhanced satellite imagery. Himawari-8 IR Enhanced satellite imagery can also depict the presence of Cb clouds. Where the cloud top temperature is less than -60 ◦C is a Cb cloud while the cloud top temperature is more than -60 is the temperature owned by other clouds. The process of Cb cloud formation can be seen by decreasing the cloud temperature to less than -60◦C [2]. So that it can be classified as Cb cloud formation.

The classification process is carried out by recognizing the pattern of Cb cloud formation from Himawari-8 IR Enhanced imagery. Pattern recognition from satellite imagery is done by looking at the pixel values in the satellite imagery. The pixel value indicates the temperature that the cloud has. The higher the pixel value for red intensity, the cooler the cloud temperature.

Clouds with higher temperatures have a lower red intensity. The cloud temperature at a certain point is very influential on the temperature of the clouds around it. As a result, a pixel will affect the pixels around it [3]. One of the algorithms that can learn patterns is deep learning. The deep learning algorithm has various methods for learning patterns, namely DeepESN, RNN, and CNN. Convolutional Neural Network (CNN) is one of the deep learning methods that currently achieves the best results in trying image pattern recognition because the CNN method mimics the image recognition system in the human visual cortex [4]. The CNN method has been used in various computer calculations such as segmentation, expression, pattern recognition, and image data classification [5]. Compared to other deep learning methods, CNN is known as the best method used in image classification. CNN has the advantage that it does not require a complicated feature extraction process like traditional image processing because CNN has a convolution layer that is easy to detect and extract features from the input image [6]. There are several architectural models available at CNN, namely GoogLeNet, Resnet, Alexnet, and VGGNet. GoogleNet has the advantage of Inception Modules which consist of several small convolutions designed to reduce the number of parameters without sacrificing network performance [7]. The research used several CNN architectural models carried out by Chengcheng Ma, namely the classification of high-resolution remote sensing image displays using 3 CNN models AlexNet, GoogleNet, and VGGNet. The accuracy of the 3 architectural models, namely the Alexnet model has an accuracy of 97.17% then GoogleNet is 98.33% and VGGNet is 98.10% where the GoogleNet architectural model has the best accuracy [8].

Based on the background explanation and previous research, if the presence of Cb clouds can have a serious impact and Cb clouds can be detected from the formation process, then the purpose of this study is to classify the formation of Cb clouds using IR Himawari -8 data with a deep learning algorithm, namely the CNN GoogleNet model. to distinguish the formation of clouds that have the potential to become Cb clouds and those that do not

2. Theoretical Framework

2.1 Cumulonimbus Clouds

Cumulonimbus clouds (Cb) are clouds that grow vertically. The water vapor content of Cb clouds is very high. Cb cloud activity often occurs during thunderstorms [2]. There are three stages in the formation of Awab CB, namely the growing stage, the mature stage, and the releasing stage. The growth stage describes the cloud growth continuing until its thermal buoyancy equals zero. These clouds are dominated by updrafts. Clouds like these don't cause rain. In the mature stage, clouds that have gathered and formed clumps are dominated by water droplets by air currents that start moving downwards causing light to heavy rain. In the dissipation stage, When the air current moves downward by more than 50%, the cumulonimbus cloud enters the dissipation stage. The dissipation stage will change the production of rain which weakens so that drizzle occurs and finally the clouds disappear [9].

2.2. Satellite Image

Satellite imagery was developed to make visual observations at distances beyond the range of human vision [10]. The Himawari-8 IR Enhanced image is one of the satellite images showing the cloud's top temperature. The cloud top temperature is obtained by sending waves of 10.4 microns. The resulting emitted waves are then converted to a temperature scale between -100°C and 100°C. In addition, cloud top temperature values are converted to pixel values between 0 and 255 in the image. The displayed image is an RGB image. Images that are blue or black mean that not many clouds are forming. When the point cloud top temperature is higher, the resulting color is closer to orange or red. An orange or red image indicates significant cloud growth, and Cb clouds are very likely to form. [2]. An example of the Himawari-8 IR Enhanced image can be seen in Figure 1.

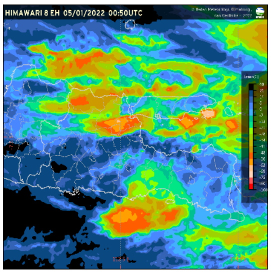


Figure 1. Image Himawari-8 IR Enhanced [11]

2.3. Convolutional Neural Network (CNN)

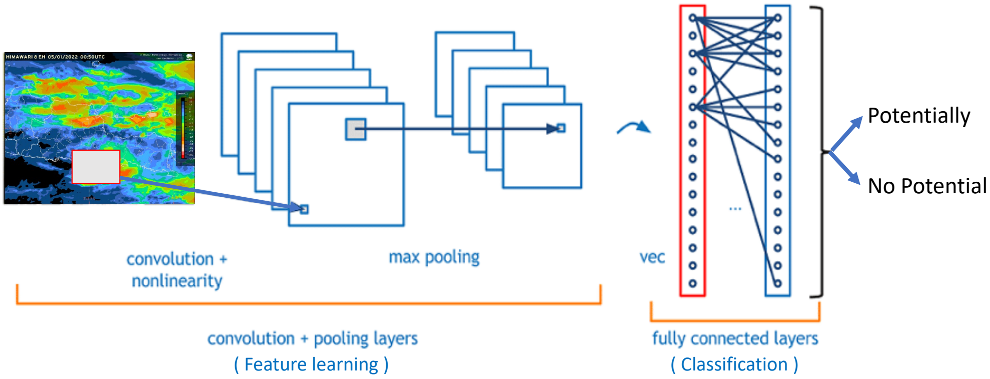


Figure 2. Illustration of Convolutional Neural Network (CNN) Architecture [12]

Convolution works by finding pixels based on the values of the pixels themselves and their neighbors using matrices called kernels, which represent weights. The pooling layer is a layer that takes features as input from the feature map and processes them using various operation statistics based on the closest pixel value. The fully connected layer is a layer that aims to change data dimensions in such a way that data can be classified linearly [12]. The CNN algorithm has been used in various computer calculations, such as segmentation, recognition, pattern recognition, and image data classification [13]. The advantage of CNN is that it does not require other complicated extraction methods because CNN has a convolutional layer that can be used to detect and extract features from the input image easily [6]. CNN consists of input, output and several hidden layers. In general, the CNN hidden layer consists of a convolutional layer, rectified linear unit (ReLU), join layer, dropout layer, and fully connected layer. The CNN process consists of two phases, namely the learning feature phase and the classification phase [14].

In the first feature learning stage, there is a convolution layer which is the main layer that underlies a CNN process that aims to perform convolution operations where the convolution process is the main process in CNN which has the function of extracting features from the input image [15]. The output of the convolution process is a pile of feature maps from all filter layers [16]. Then the second is the Rectified Linear Unit (ReLU) which changes the input value of the neuron feature map resulting from the convolution layer in the range 0 to infinity [17]. The form of the ReLU function can be expressed by Equation (1).

$$f(x) = max(0,x) \tag{1}$$

Information:

x = input value, $\quad$ $x \in R$

Which means it will cut off the input signal that has a value less than 0 [6]. So, ReLU assigns a value of 0 instead of a negative value in the feature map and the value of the neuron input feature map remains the same if the value is greater than or equal to 0 [18]. Then the third is the pooling layer, which speeds up computing by reducing the volume per stack of feature maps without losing important information [18].

At the classification stage, there is a dropout layer which a layer for carrying out the dropout process where dropout is a process to prevent or reduce overfitting where overfitting is unwanted machine learning behavior that occurs when machine learning models provide accurate predictions for training data but not for new data [19]. Then there is a Fully connected layer which is the last layer or perceptron layer which collects all the signals from the previous layer and then processes the sum of these signals using the softmax function [6]. The softmax activation function converts values into probabilities using Equation (2).

$$Softmax(y_i) = \frac{e^{(y_i)}}{\sum_{j=1}^{k} e^{(y_i)}}, i = 1,2, \dots, k \tag{2}$$

Information:

y = input value

e = 2,7182

k = number of classes

Softmax values range from 0 to 1. Data will be classified with softmax values close to 1 or the highest [20].

2.4. Googlenet

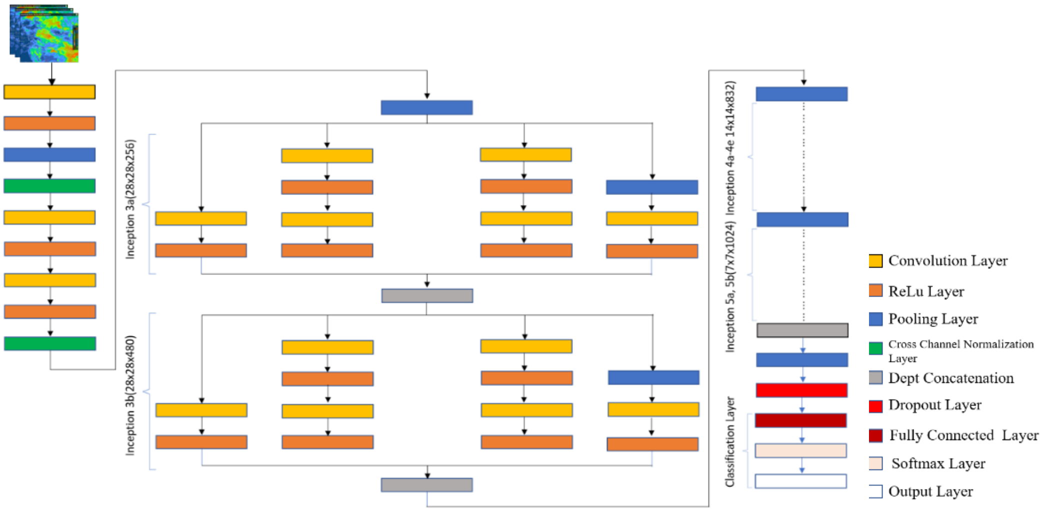


Figure 3. Illustration of Googlenet Architecture [7]

The GoogleNet architecture in figure 3, which is the 3 Inception modules. The Inception module is a layer that allows all filters (1×1, 3×3, and 5×5) to operate on the same portion of the image and then strings the loop results into a single output with the resulting output from the pooling layer. Active network GoogleNet is trained with the sampling method Different pictures use 6 models. All convolution processes, including those at the start, use linear activation. The GoogleNet architecture consists of 22 layers (27 layers including the binder layer) in total of nine initial modules. GoogleNet has the advantage of the Inception Module which consists of several small coils designed to reduce the number of parameters without decreasing network performance [21]. On GoogleNet there is a special module, namely the inception module, where this module has a multiscale kernel convolution [22]. This architecture can be used for very large training data and can recognize thousands of different objects. This network can reduce 60 million parameters that are lacking [21]. GoogleNet returns an error value in the classification of image data reaching 6.7% [20].

2.5. System Evaluation

This study uses the evaluation of the confusion matrix system. The confusion matrix is an evaluation method used to measure the accuracy of classification [22]. The evaluation of classification results is analyzed from several indicators such as accuracy, sensitivity, and specificity. Accuracy is a value to measure the number of success rates in the classification carried out, then sensitivity is a measurement to find out positive results from true positive data, and specificity is a test to determine negative results from real negative data. [23]. TP (true positive) is data that is classified as potential, TN (true negative) is data that is not classified as potential, FP (false positive) is data that is not classified as potential, and FN (false negative) is data that is classified as potential. to be potential. These terms are represented in Table 1.

Table 1. Confusion Matri

Classification Result Data	**Original Data**	
	Potentially	No Potential
Potentially	TP	FP
No Potential	FN	TN

Based on the True Positive (TP) value, False Positives (FP), True Negatives (TN), and False Negative (FN), the evaluation is analyzed from several indicators, one of which is indicator accuracy, specificity, and sensitivity. Accuracy is the ratio of numbers predicted correctly in all data. Specificity is a value that shows a lot of negative data can be classified correctly into the negative class. Sensitivity is a value indicating a lot of valuable data positive that can be correctly classified into the positive class with the aim of knowing how good the model. Indicators are calculated by Equations (3), (4), and (5) [23].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \tag{3}$$

$$Sensitivity = \frac{TP}{TP + FN} \times 100\% \tag{4}$$

$$Specificity = \frac{TN}{TN + FP} \times 100\% \tag{5}$$

3. Methodology

In this section, the stages of data processing will be explained for the classification process for the formation of Cb clouds, which can be seen in Figure 4.

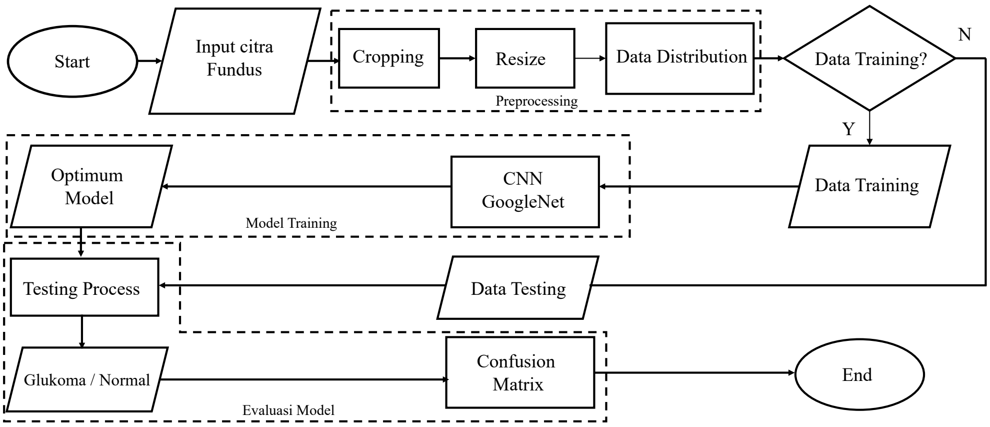


Figure 4. Research Flowchart

3.1. Data Collection

The data used is Himawari-8 IR Enhanced satellite image data taken from the BMKG Maritime Perak 2 in Surabaya which represents the province of East Java. Satellite imagery is taken once every ten minutes. The data used is in the form of RGB images. The pixel value in the image shows the temperature that a cloud has. Clouds with a low temperature are marked with a redder color, and white clouds with a high temperature are marked with a blacker color. The image data obtained totaled 2026 satellite images with a division of 1023 satellite images having the potential to become Cb clouds and 1003 satellite images not having the potential to become clouds. Examples of these satellite images are presented in Figures 5 and 6.

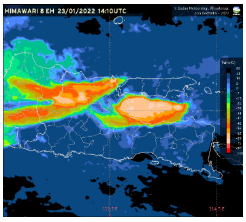


Figure 5. Potential Data [11]

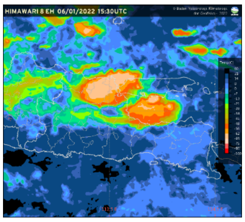


Figure 6. Non-Potential Data [11]

3.2. Preprocessing

Data preprocessing is carried out with the aim of optimizing the data for the learning model. All data will go through a preprocessing process by cropping and resizing to a size of 224×224×3 pixels. Cropping is done with a size of 1220x1220 to remove the color index and date information in the image, then resize it to 224x224x3. Because if the size of the image data is not 224 × 224 × 3 pixels, then the image data cannot be used as input to the CNN method, the Google Net model.

3.3. Model Training

After the image data is 224 × 224 × 3 pixels in size, the data is divided into training data and test data, then the training process will be carried out using the CNN GoogleNet model. This study uses trial parameters, namely data distribution, probability or dropout values, and the number of batch sizes. The process of dividing the training data and test data is done to get the best percentage of data. The probability or dropout value serves to randomly remove several neurons with a predetermined probability at each iteration of the training process, resulting in a reduction or depletion of the network. This can reduce overfitting because the thicker the network, the more likely it is that the model is too well suited for a given data set [23]. Batch size is a parameter to control the learning process by selecting training data samples with the same number of repetitions as the epoch in each sample [23].

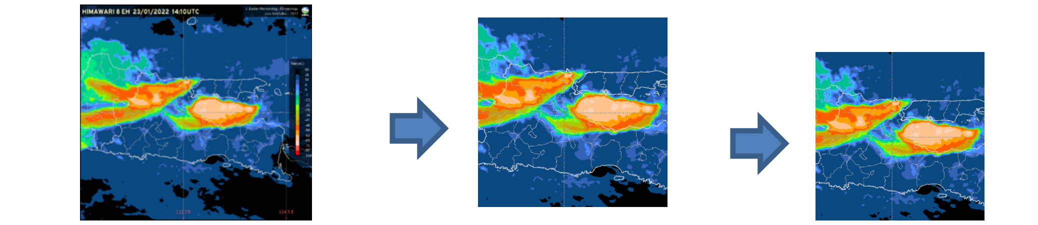
3.4. Evaluasi Model

after getting the model in the training process, the next model will be tested using test data which aims to find out how good the model is. in the evaluation of the system will use the confusion matrix which will be continued to look for the best accuracy, sensitivity, and specificity values of this study.

4. Results

4.1. Preprocessing

All data will go through a preprocessing process by cropping it to a size of 1220x1220 and resizing to a size of 224x224x3 pixels. The difference in image size before and after preprocessing is shown in Figure 7.



(Data before cropping and resizing) (Data after cropping) (Data after being resized)

Figure 7. Preprocessing results (crop and resize) [11]

4.2. Model Training

In the data training process learning was carried out using the GoogleNet model with the selected data distribution, namely 60:40, 70:30, 80:20, and 90:10 to get the best percentage of data, with probability and batch size parameters. In this study, the probability values were randomly selected with values 0.6, 0.7, and 0.8, while the number of batch sizes to be tested was also randomly selected with values 8, 16, 32, and 64. The training model image can be seen in Figure 8.

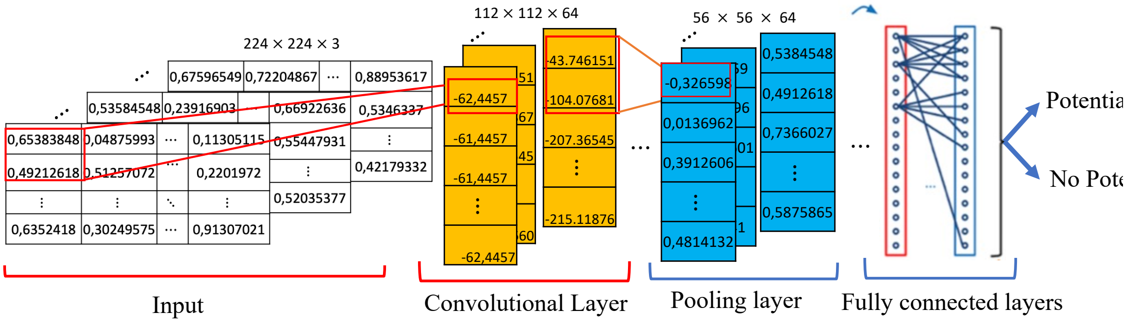


Figure 8. Illustration of Training Model

4.3. Evaluasi Model

After doing several experiments by changing the number of batch sizes and the number of epochs in each test. The optimum test model for this study is the data division of 90:10 with a probability value or dropout of 0.6 with a batch size of 8 with an accuracy of 99.00%, a sensitivity value of 100.00% and a specificity value of 99.60%. This value indicates that the classification system created using the CNN GoogleNet model can detect data correctly like the original data. The test results with the parameters of the probability value data distribution or dropout and bath size as well as the optimum model results can be seen in Table 2

Table 2. Test Results with The Parameters of The Number of Data Distributions, Probability or Dropout, and Batch Size

Part Data	Probability	Batch Size	Sensitivity (%)	Accuracy (%)	Specificity (%)
60	0.6	8	88.00	93.66	96.20
		16	89.60	94.41	96.60
		32	87.60	93.42	96.00
		64	84.30	92.42	96.00
	0.7	8	89.20	94.91	97.50
		16	83.90	93.66	98.00
		32	86.30	93.17	96.20
		64	82.70	92.17	96.40
	0.8	8	89.60	94.91	97.30
		16	88.80	94.16	96.60
		32	89.60	94.04	96.00

		64	88.80	92.05	93.50
70	0.6	8	93.00	96.53	97.80
		16	95.20	96.36	97.80
		32	92.00	95.36	96.90
		64	93.00	93.37	96.80
	0.7	8	95.70	95.86	98.00
		16	95.20	95.36	97.80
		32	89.80	94.20	95.50
		64	92.00	93.87	96.30
	0.8	8	94.10	95.36	97.30
		16	94.10	95.53	97.30
		32	94.10	95.03	97.30
		64	92.50	94.20	96.60
80	0.6	8	93.40	96.31	96.50
		16	94.60	96.11	97.50
		32	95.40	95.52	96.10
		64	93.40	96.79	97.10
	0.7	8	96.80	96.60	96.80
		16	95.40	95.77	97.50
		32	96.80	96.21	97.90
		64	95.70	95.53	96.80
	0.8	8	97.60	97.77	98.00
		16	96.20	96.93	97.80
		32	93.00	95.49	96.10
		64	93.40	94.64	96.30
90	0.6	8	100.00	99.00	99.60
		16	95.50	97.30	98.00
		32	94.50	97.50	97.80
		64	96.50	98.51	97.20
	0.7	8	95.50	97.57	98.30
		16	96.30	97.00	98.60
		32	97.20	98.20	98.30
		64	97.20	98.65	99.10
	0.8	8	96.00	98.53	99.00
		16	95.80	97.50	98.00
		32	96.50	97.40	98.60
		64	95.20	97.62	97.80

The results of the Confusion matrix in this study are 98 data that are targeted to potentially occur Cb clouds classified as potential Cb clouds, 2 data targeted to potentially occur Cb clouds classified as not potentially occurring Cb clouds, 0 data targeted not to occur Cb clouds and classified as Cb clouds, and 100 targets data that does not occur Cb clouds are classified as Cb clouds do not occur so that the confusion matrix image above has sensitivity, specificity and accuracy values of 100%, 99% and 98%, Each batch of size experiments shows that the average accuracy value decreases with increasing size values batch. Classification system performance is also evaluated with sensitivity and specificity. A high sensitivity value indicates that the resulting model can recognize images that do not have the potential to become Cb clouds well. Meanwhile, a high specificity value indicates that the resulting model can recognize images that have the potential to become Cb clouds well. so that the resulting model can classify Cb cloud formation properly and the model does not experience overfitting. for the complete Confusion Matrix results can be seen in Figure 9.

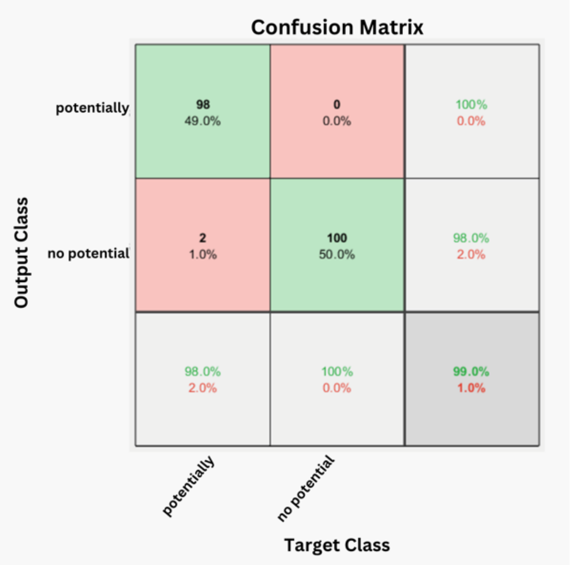


Figure 9. Results Confusion Matrix Optimum Model

5. Conclusion

From the results of this study on the classification of Cb cloud formation using the CNN GoogleNet model, it can be concluded that the recommended image for Cb cloud classification is Himawari-8 IR Enhanced imagery. The probability of dropout is 0.6 and the batch size is 8. The sensitivity, accuracy, and specificity values are 100.00%, 99.00%, and 99.60%. from the trial results obtained is relatively high. However, it requires a relatively long time and the classification system created using the CNN GoogleNet model obtains effective results. This research can create an accurate Cb cloud classification system based on the results obtained.

REFERENCES

[1] K. B. Damanik, Y. Fitri, S. Gautami Mahasiswa, J. Fisika, D. J. Fisika, and U. M. Riau, “Pertumbuhan Awan Cumulonimbus (Cb) di Bandar Udara (BANDARA) Sultan Syarif Kasim II Pekanbaru,” 2016.

[2] M Fahrur R, “Prediksi Pertumbuhan Awan Cumulonimbus pada Citra Himawari IR Enhanced Menggunakan Deep Echo State Network (DEEPESN).”

[3] B. S. Pandjaitan and M. I. Damayanti, “Kajian Mikrofisis Awan Menggunakan Satelit Himawari 8 pada Kejadian Hail (Studi Kasus: Kejadian Hail di Jakarta Tanggal 28 Maret 2017) Study of Cloud Microphysic Using Himawari 8 Satellite on the Hail Stone Event (Case Study: Hail Stone Event in Jakarta on 28 March 2017).” Go to reference in article.

[4] N. Hanum Harani, C. Prianto, and M. Hasanah, “Deteksi Objek dan Pengenalan Karakter Plat Nomor Kendaraan Indonesia Menggunakan Metode Convolutional Neural Network (CNN) Berbasis Python,” *Jurnal Teknik Informatika*, vol. 11, no. 3, pp. 47–53, 2019.

[5] J. Ker, L. Wang, J. Rao, and T. Lim, “Deep learning applications in medical image analysis,” *Ieee Access*, vol. 6, pp. 9375–9389, 2017.

[6] H. Akbar and Sandfreni, “Klasifikasi Kanker Serviks Menggunakan Model Convolutional Neural Network (Alexnet),” *JIKO (Jurnal Informatika dan Komputer)*, vol. 4, no. 1, pp. 44–51, 2021, doi: 10.33387/jiko.v4i1.2606.

[7] P. Jasitha, M. R. Dileep, and M. Divya, “Venation Based Plant Leaves Classification Using GoogLeNet and VGG,” *2019 4th IEEE International Conference on Recent Trends on Electronics, Information, Communication and Technology*, pp. 715–719, 2019, doi: 10.1109/RTEICT46194.2019.9016966.

[8] Institute of Electrical and Electronics Engineers and IEEE Geoscience and Remote Sensing Society, *2016 IEEE International Geoscience & Remote Sensing Symposium : proceedings : July 10-15, 2016, Beijing, China*.

[9] B. P. Dewa, R. E. Saputra, W. Harjupa, and I. Fathrio, “Perancangan Prediktor Awan Konvektif Menggunakan Logika Fuzzy Metode Sugeno (Design Of Convective Cloud Predictor Using Sugeno Fuzzy Logic Method).”

[10] N. Suwargana, “Resolusi spasial, temporal dan spektral pada citra satelit Landsat, SPOT dan IKONOS,” *Jurnal Ilmiah Widya*, vol. 1, no. 2, pp. 167–174, 2013.

[11] Media BMKG, “Himawari-9 IR Enhanced - Jawa Timur,” *BMKG*, May 17, 2022.

[12] S. Ilahiyah and A. Nilogiri, “Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network,” *JUSTINDO (Jurnal Sistem Dan Teknologi Informasi Indonesia)*, vol. 3, no. 2, pp. 49–56, 2018.

[13] J. Ker, L. Wang, J. Rao, and T. Lim, “Deep learning applications in medical image analysis,” *Ieee Access*, vol. 6, pp. 9375–9389, 2017.

[14] A. Alqumboz and M. Naser, “Classification of Avocado Using Deep Learning,” *International Journal of Academic Engineering Research (IJAER)*, vol. 3, no. 12, pp. 30–34, 2019.

[15] I. W. S. Suartika, A. Y. Wijaya, and R. Soelaiman, “Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) Pada Caltech 101,” *Jurnal Teknik ITS*, vol. 5, no. 1, pp. 65–69, 2016.

[16] A. A. Elsharif, I. M. Dheir, A. S. A. Mettleq, and S. S. A. Naser, “Potato Classification Using Deep Learning,” *International Journal of Academic Pedagogical Research (IJAPR)*, vol. 3, no. 12, pp. 1–8, 2019.

[17] S. H. Wang, Y. D. Lv, Y. Sui, S. Liu, S. J. Wang, and Y. D. Zhang, “Alcoholism Detection by Data Augmentation and Convolutional Neural Network with Stochastic Pooling,” *J Med Syst*, vol. 42, no. 2, pp. 1–11, 2018, doi: 10.1007/s10916-017-0845-x.

[18] M. Coskun, A. Ucar, O. Yildirim, and Y. Demir, “Face Recognition Based on Convolutional Neural Network,” *International Conference on Electrical Energy System*, pp. 376–379, 2017, doi: 10.13810/j.cnki.issn.1000-7210.2019.05.024.

[19] W. Wang *et al.*, “Detection of SARS-CoV-2 in Different Types of Clinical Specimens,” *JAMA - Journal of the American Medical Association*, vol. 323, no. 11, pp. 1843–1844, 2020, doi: 10.1001/jama.2020.1585.

[20] A. Rambaut, A. J. Drummond, D. Xie, G. Baele, and M. A. Suchard, “Posterior summarization in Bayesian phylogenetics using Tracer 1.7,” *Syst Biol*, vol. 67, no. 5, pp. 901–904, 2018.

[21] N. Fathanah Mustamin et al., “Klasifikasi Kualitas Kayu Kelapa Menggunakan Arsitektur CNN,” Kumpulan jurnaL Ilmu Komputer (KLIK), vol. 8, no. 1, 2021.

[22] S. Tabik, D. Peralta, A. H. Poyatos, and F. Herrera, “A Snapshot of Image Pre-Processing for Convolutional Neural Networks: Case Study of MNIST,” *International Journal of Computational Intelligence Systems*, vol. 10, no. 1, pp. 555–568, 2017, doi: 10.2991/ijcis.2017.10.1.38.

[23] L. Zmudzinski, “Deep learning Guinea Pig Image Classification Using Nvidia Digits and GoogleNet,” *CEUR Workshop Proc*, vol. 2240, 2018.