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 60237, 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 2.026 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 dangerous for many human activities. To reduce this effect, a system to classify formations is needed. The formation of Cb clouds can be seen in the Himawari-8 IR image. This research aimed to create a Cb cloud classification system with Himawari-8 IR Enhanced imagery using the GoogleNet model CNN method. The total data used was 2026 image data. Parameter testing was carried out on the CNN GoogleNet model in this study, namely a 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 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°C is the temperature owned by other clouds. Cb cloud formation can be seen by decreasing the cloud temperature to less than -60°C [2], which 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 influences 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: 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]. CNN is the best method used in image classification compared to other deep learning methods.

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]. Several architectural models are available at CNN: GoogLeNet, Resnet, Alexnet, and VGGNet. GoogleNet's Inception Modules consist of multiple small convolutions which reduce parameters without compromising performance [7]. The research used several CNN architectural models 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 three architectural models, namely the Alexnet model is 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 severe 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. Literature Review

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: the growing, maturity, and releasing stages. During the growth stage, the cloud continues to expand until it reaches a point where its thermal buoyancy is balanced and equals zero. Updrafts dominate these clouds. These clouds do not 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 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 increases, the resulting colour is closer to orange or red. An orange or red image indicates significant cloud growth, and Cb clouds are likely to form [2]. An example of the Himawari-8 IR Enhanced image can be seen in Figure 1 [11].

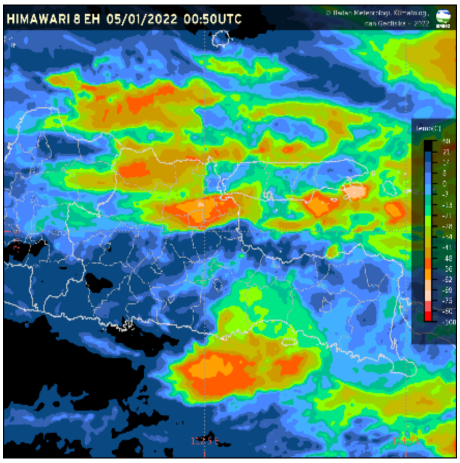


Figure 1. Image Himawari-8 IR Enhanced

2.3. Convolutional Neural Network (CNN)

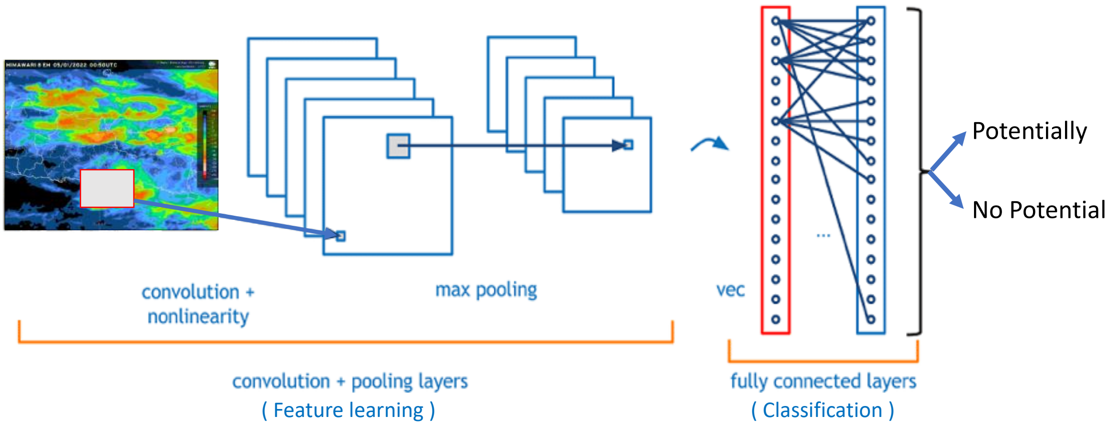


Figure 2. Illustration of Convolutional Neural Network (CNN) Architecture

Convolution works by finding pixels based on the values of the pixels themselves and their neighbours 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, as shown in Figure 2. The CNN hidden layer generally 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, the main layer that underlies a CNN process that aims to perform convolution operations. The convolution process is the primary process in CNN, extracting features from the input image [15]. The output of the convolution process is a pile of feature maps from all filter layers [16]. 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.

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

where, x is input value, $x \in R$

This means it will cut off any input signals with a value less than 0 [6]. ReLU replaces negative values in the feature map with 0 and keeps the same value for non-negative values in the neuron input feature map [18]. The third is the pooling layer, which speeds up computing by reducing the volume per stack of feature maps without losing essential information [18]. A dropout layer is included during the classification stage to facilitate the dropout process. Dropout is a technique used to prevent overfitting in machine learning models. Overfitting occurs when a model provides accurate predictions for training data but needs to generalize better to new data [19]. The final layer in a neural network is a fully connected layer, also known as the perceptron layer, which receives input from the previous layer and computes the sum of those inputs using the softmax function [6]. The softmax activation function transforms values into probabilities using Equation 2. Softmax values range from 0 to 1, and data is classified based on the softmax values, which are close to 1 or the highest.[20].

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

where, y is the input value, e is 2.7182, and k is the number of classes.

2.4. GoogleNet

Figure 3 [7] presents the GoogleNet architecture which consists of three inception modules. The inception module is a layer that allows multiple filters, such as 1×1, 3×3, and 5×5, to operate on the same part of an image. It then consolidates the results of these filters into a single output, along with the output from the pooling layer. The GoogleNet model is trained using the sampling method, and six models are employed for different images. All convolution processes, including the initial ones, employ linear activation. The GoogleNet architecture is composed of nine initial modules, containing a total of 22 layers (or 27, including the binder layer). One of the highlights of GoogleNet is the inception module, which is made up of multiple small coils, designed to reduce the number of parameters while maintaining network performance. This feature is particularly advantageous [21]. On GoogleNet, there is a particular module, namely the inception module, where this module has a multiscale kernel convolution [22]. This network can recognize thousands of objects using extensive training data while reducing 60 million parameters [21]. The classification of image data by GoogleNet resulted in an error rate of 6.7% [20].

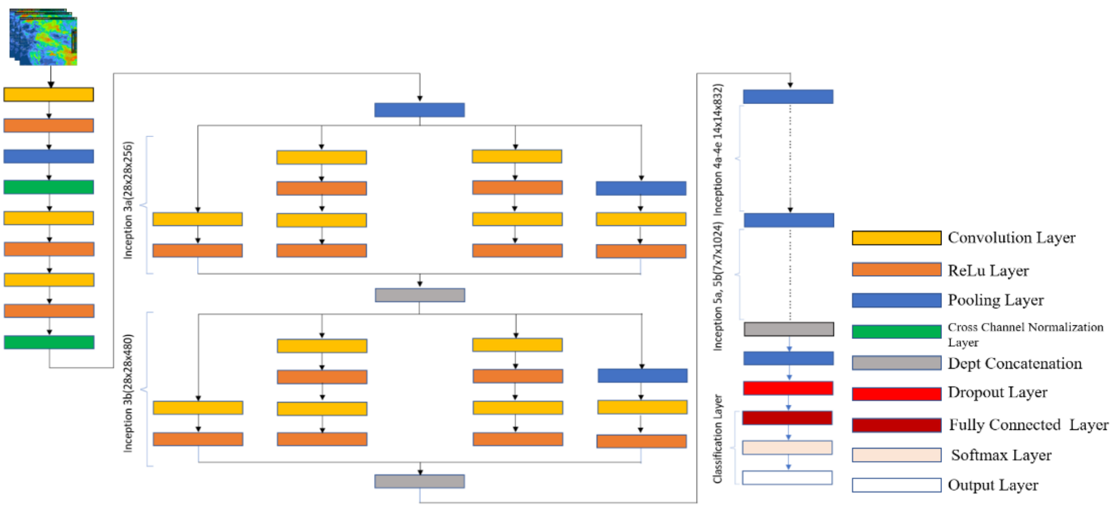


Figure 3. Illustration of GoogleNet Architecture

2.5. Evaluation System

This study employs the confusion matrix to evaluate classification accuracy [22]. When evaluating classification results, several indicators are analyzed, such as accuracy, sensitivity, and specificity. Accuracy measures the number of successful classifications. Sensitivity is used to identify positive results from accurate positive data, while specificity determines negative results from actual negative data [23]. In data analysis, TP (true positive) refers to the data that is correctly classified as positive, while TN (true negative) refers to the data that is correctly classified as negative. FP (false positive) is the data that is mistakenly classified as positive, and FN (false negative) is the data that is mistakenly classified as negative. These terms are represented in Table 1.

Table 1. Confusion Matrix

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

The evaluation of a model is based on the True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) values. Several indicators are used to analyze the performance, including accuracy, specificity, and sensitivity. Accuracy measures the ratio of correctly predicted data in the entire dataset. Specificity measures how many negative samples are correctly classified into the negative class. Sensitivity measures how many valuable positive samples are correctly classified into the positive class to determine the model's performance. These indicators are calculated using 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.

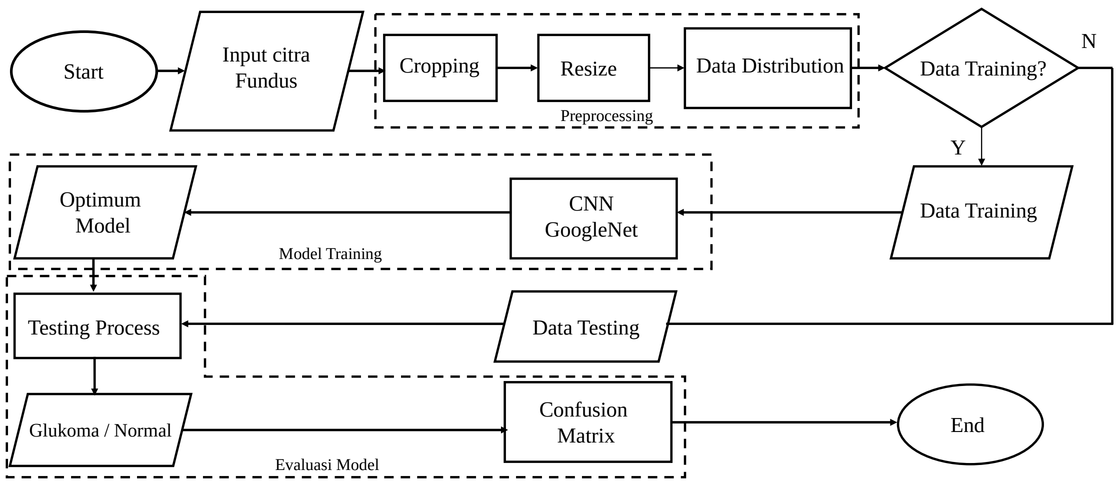


Figure 4. Research Flowchart

3.1. Data Collection

The data used in this study is the Himawari-8 IR Enhanced satellite image data obtained from the BMKG Maritime Perak 2 in Surabaya, which represents the province of East Java. The satellite imagery is captured every ten minutes, providing 2026 images in RGB format. The image pixel values represent the temperature of clouds, with lower temperatures shown in redder colors and higher temperatures in blacker colors. Of the 2026 images, 1023 have the potential to become Cb clouds, while the remaining 1003 do not. Examples of these satellite images can be seen in Figures 5 and 6 [11].

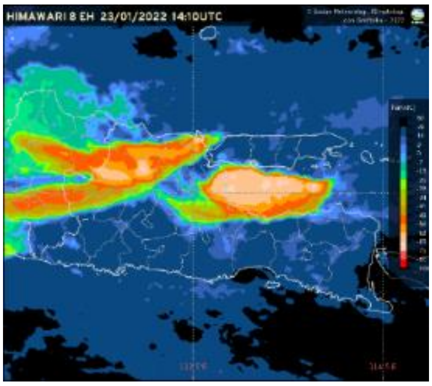


Figure 5. Potential Data

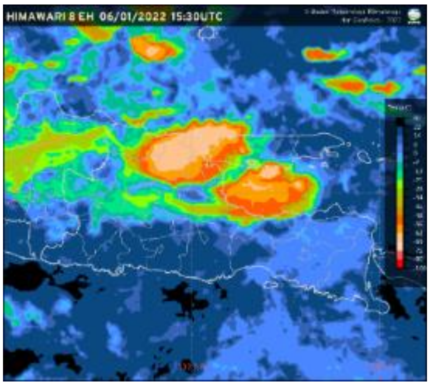


Figure 6. Non-Potential Data

3.2. Preprocessing

Data is preprocessed to optimize it for the learning model. All data is cropped and resized to 224x224x3 pixels. Images are cropped to 1220x1220, removing the color index and date information before resizing. The GoogleNet model requires image data in 224x224x3 pixels, making this preprocessing necessary.

3.3. Model Training

After the image data is 224 × 224 × 3 pixels, the data is divided into training 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 randomly removes 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 the model will be 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].

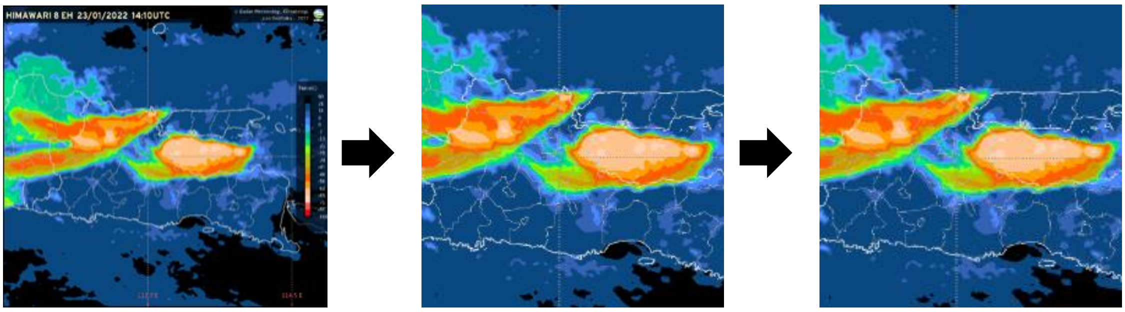
3.4. Model Evaluation

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

4. Results

4.1. Preprocessing

All data will be preprocessed by cropping it to 1220x1220 and resizing to 224x224x3 pixels. The difference in image size before and after preprocessing is shown in Figure 7 [11].



(a) (b) (c)

Figure 7. Preprocessing Results: (a) Data before cropping and resizing, (b) Data after cropping, (c) Data after being resized

4.2. Model Training

In the data training process, the selected data distribution of 60:40, 70:30, 80:20, and 90:10 was used with the GoogleNet model to determine the optimal percentage of data, utilizing 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 batch sizes to be tested were also randomly selected with values 8, 16, 32, and 64. The training model image can be seen in Figure 8.

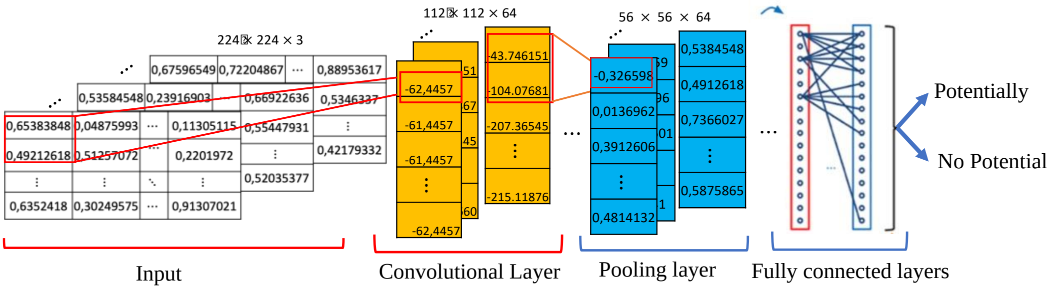


Figure 8. Illustration of Training Model

4.3. Model Evaluation

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. It 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

Part Data	Probability	Batch Size	Sensitivity (%)	Accuracy (%)	Specificity (%)
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

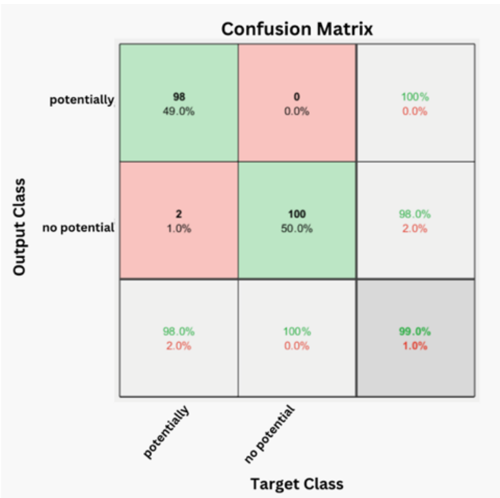


Figure 9. Results Confusion Matrix Optimum Model

The results of confusion matrix in this study are 98 data that are targeted to occur Cb clouds classified as potential Cb clouds potentially, 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%.

After conducting experiments of various batch sizes, we observed a decrease in average accuracy values as batch size increased. We evaluated the classification system performance using sensitivity and specificity. A high sensitivity value means that the model can accurately identify images that may not become Cb clouds. On the other hand, a high specificity value means that the model can accurately identify images that are likely to become Cb clouds. It is essential to have a balanced sensitivity and specificity values to ensure the model can classify Cb cloud formation correctly without overfitting. The complete Confusion Matrix results can be seen in Figure 9.

5. Conclusion and Recommendation

The study's results suggest that the Himawari-8 IR Enhanced imagery is the best option for identifying Cb cloud formation using the CNN GoogleNet model. The probability of dropout is 0.6 and the batch size is 8. The trial results show that the model is highly effective, with a sensitivity of 100%, accuracy of 99%, and specificity of 99.6%. Despite the lengthy processing time, the CNN GoogleNet model produces accurate classifications. As a result, this research may lead to the development of a precise Cb cloud classification system.

References

[1] K. B. Damanik, Y. Fitri, and S. Gautami, “Analisis Perubahan Suhu Dan Tekanan Udara Permukaan Terhadap Pertumbuhan Awan Cumulonimbus (Cb) Di Bandar Udara (Bandara) Sultan Syarif Kasim Ii Pekanbaru,” *Photon: Jurnal Sain dan Kesehatan*, vol. 6, no. 02, pp. 131–138, 2016.

[2] M. F. Rozi, “Prediksi pertumbuhan awan cumulonimbus pada citra himawari ir enhanced menggunakan deep echo state network (deepesn),” *Universitas Islam Negeri Sunan Ampel Surabaya*, 2019.

[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),” *Seminar Nasional Penginderaan Jauh ke-4 Tahun 2017*, 2017, pp. 413-420.

[4] N. H. 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," in *IEEE Access*, vol. 6, pp. 9375-9389, 2018, doi: 10.1109/ACCESS.2017.2788044.

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

[7] P. Jasitha, M. R. Dileep and M. Divya, "Venation Based Plant Leaves Classification Using GoogLeNet and VGG," *2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*, Bangalore, India, 2019, pp. 715-719, doi: 10.1109/RTEICT46194.2019.9016966.

[8] Y. Wang, C. Wang, and H. Zhang, “Combining a single shot multibox detector with transfer learning for ship detection using sentinel-1 SAR images,” *Remote Sensing Letters*, vol. 9, no. 8, pp. 780–788, 2018, doi: 10.1080/2150704X.2018.1475770.

[9] B. P. Dewa, R. E. Saputra, W. Harjupa, and I. Fathrio, “Perancangan Prediktor Awan Konvektif Menggunakan Logika Fuzzy Metode Sugeno,” *eProceedings of Engineering*, vol. 8, no. 5, 2021.

[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] BMKG, “Himawari-9 IR Enhanced - Indonesia,” May 22, 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," in *IEEE Access*, vol. 6, pp. 9375-9389, 2018, doi: 10.1109/ACCESS.2017.2788044.

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

[15] W. S. E. Putra, “Klasifikasi citra menggunakan convolutional neural network (CNN) pada caltech 101,” *Jurnal Teknik ITS*, vol. 5, no. 1, 2016.

[16] S. Mohammad, “Apple Fruits Classification using Deep Learning,” *International Journal of Academic Engineering Research (IJAER)*, vol. 3, no. 12, 2020.

[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,” *Journal of Medical Systems*, vol. 42, pp. 1–11, 2017, doi: 10.1007/s10916-017-0845-x.

[18] M. Coşkun, A. Uçar, Ö. Yildirim and Y. Demir, "Face recognition based on convolutional neural network," *2017 International Conference on Modern Electrical and Energy Systems (MEES)*, 2017, pp. 376-379, doi: 10.1109/MEES.2017.8248937.

[19] D. Wang *et al.*, “Clinical characteristics of 138 hospitalized patients with 2019 novel coronavirus–infected pneumonia in Wuhan, China,” *JAMA*, vol. 323, no. 11, pp. 1061–1069, 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,” *Systematic Biology*, vol. 67, no. 5, pp. 901–904, 2018, doi: 10.1093/sysbio/syy032.

[21] N. F. Mustamin, Y. Sari, and H. Khatimi, “Klasifikasi Kualitas Kayu Kelapa Menggunakan Arsitektur Cnn,” *KLIK-KUMPULAN JURNAL ILMU KOMPUTER*, vol. 8, no. 1, pp. 49–59, 2021, doi: 10.20527/klik.v8i1.370.

[22] S. Tabik, D. Peralta, A. Herrera-Poyatos, and F. Herrera Triguero, “A snapshot of image pre-processing for convolutional neural networks: case study of MNIST,” *International Journal of Computational Intelligence Systems*, vol. 10, 2017, pp. 555-568, doi: 10.2991/ijcis.2017.10.1.38.

[23] L. Zmudzinski, “Deep Learning Guinea Pig Image Classification Using Nvidia DIGITS and GoogLeNet,” *International Workshop on Concurrency, Specification and Programming*, 2018.