2024 4th International Conference of Science and Information Technology in Smart Administration (ICSINTESA) | 979-8-3503-7611-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICSINTESA62455.2024.10747897

Enhanced Flood Detection on Highways: A Comparative Study of MobileNet and VGG16 CNN Models Based on CCTV Images

Muh. Taufik Hidayat
Department of Computer Engineering
Universitas Negeri Makassar
Makassar, Indonesia
muhtaufikunm17@gmail.com

Fhatiah Adiba
Department of Computer Engineering
Universitas Negeri Makassar
Makassar, Indonesia
adibafhatiah@unm.ac.id

Muhammad Yahya
Department of Automotive Engineering Education
Universitas Negeri Makassar
Makassar, Indonesia
m.yahya@unm.ac.id

Dyah Darma Andayani
Department of Computer Engineering
Universitas Negeri Makassar
Makassar, Indonesia
dyahdarma@unm.ac.id

Andi Baso Kaswar
Department of Computer Engineering
Universitas Negeri Makassar
Makassar, Indonesia
a.baso.kaswar@unm.ac.id

***Abstract*— Flooding is one of the most common natural disasters, especially during the rainy season. Flooding on highways can cause material losses and casualties, as well as disrupt community mobility and adversely affect the economic sector. Early flood detection can help reduce the impact of these natural disasters. This research was designed to detect the occurrence of flooding on highways by comparing the performance of two Convolutional Neural Network (CNN) architectures, namely MobileNet and VGG16, for the detection and classification of flood images on highways. This research used a dataset consisting of 2,000 images, with 1,000 images for the flood class and 1,000 images for the non-flood class. The results showed that the MobileNet model performed better than the VGG16 model. The MobileNet model has an accuracy of 99% and a shorter computation time, which is only 9 seconds, while the VGG16 model has an accuracy of 96% and takes 69 seconds for computation time. Based on the results of this study, it can be concluded that the MobileNet model can be used for early detection of flooding on highways. It is proven to have high accuracy and a shorter computation time; therefore, it can be used in real-time flood detection systems.**

Keywords—Flood, CNN, Flood Detection, MobileNet, VGG16

I. INTRODUCTION

Flooding is a natural disaster that often occurs especially during the rainy season where the intensity of rainfall is high causing inundation [1][2]. The impact of flooding is very detrimental, especially for highways which are an important means of community mobility [3][4]. Water puddles on the highway not only disrupt vehicle activities, but also have the potential to cause serious damage to road infrastructure [5][6]. In addition, flooding can create significant financial losses and threaten the safety of people who use roads as the main means of transportation [7][8]. Based on data recorded from dibi.bnpb.go.id for the last 5 years, it shows that Indonesia has experienced as many as 4,407 flood events starting from 2019 to 2023, and the region that has highest numbers of flooding is the Central Java with 874 events [9].

The primary reason for flooding on highways is typically prolonged heavy rainfall. This excess rainwater does not absorb into the ground quickly, leading to surface water accumulation that disrupts waterflow [6]{10]. Moreover inadequate drainage systems can exacerbate the issue by impeding waterflow [11]. Flooding not only disrupts mobility but also damages vehicles, road infrastructure, and causes traffic congestion. These delays in goods delivery can have adverse effects on the economy [6][12].

To mitigate the negative impact of flooding on highways, efficient preventive measures are essential. One possible solution is the application of modern technology in the field of artificial intelligence using Convolutional Neural Network (CNN). It can be used to detect floods in CCTV images captured on highways, thus aiding in flood monitoring and mitigation efforts.

Multiple studies have been undertaken on flood detection and classification. One study, for instance, proposed an urban road flood monitoring system using CNN Mobilenet V2 architecture. This system can accurately detect floods and classify their severity achieving a detection accuracy of 95.33% and a classification accuracy of 89.58% [13]. Another study utilized CNN models, specifically Inception v3 and DenseNet, to identify flood-affected area. The findings indicated that the Inception v3 architecture outperformed, achieving an accuracy of 83%. [14].

In prior research, the U-net transfer learning CNN architecture was combined with MobilenetV2. This particular study utilized a dataset of satellite image flood and achieved a Mean Intersection over Union (MioU) segmentation rate of 76.62%, significantly outperforming the standard U-net. This result demonstrates the potential effectiveness of this approach for automatic flood detection [15]. In a subsequent study, the MobileNet V3 architecture was employed achieving an accuracy rate of 83.3% [16].

Previous research has explored object detection in disaster images, specifically focusing on floods and earthquakes. Three pre-trained CNN models, namely ResNet50, VGG-16, and VGG-19, were employed in the study. The findings revealed that the VGG-19 model achieved the highest accuracy of 94.22% for object detection in flood and earthquake images [17].

While previous studies have demonstrated relatively good accuracy in flood detection and classification, there remains room for improvement to achieve optimal results in real-time monitoring and detection of highway flooding. The goal is to enhance model's ability to swiftly and effectively detect and classify floods, enabling efficient flood mitigation measures on highways.

Thus, addressing the aforementioned challenges, this study proposes the application of Convolutional Neural Network (CNN) architecture with transfer learning. This approach aims to detect and classify highway flooding using image data obtained from CCTV cameras.

II. METHOD

Figure 1 illustrates the flow of research stages commencing with data collection, followed by data preprocessing. Subsequently, the CNN architecture is constructed and trained using the pre-processed data. Finally, the model training results are evaluated.

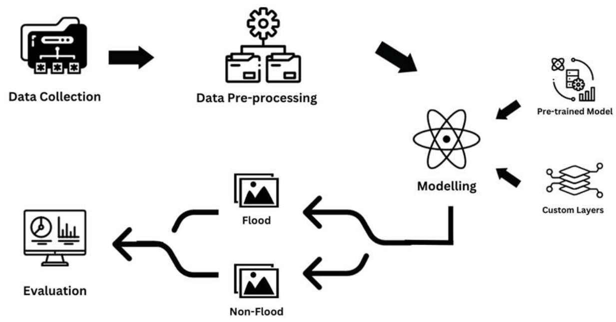


Fig. 1. Stages of Method

A. Data Collection

This study employed two methods of data collection : primary and secondary. Primary data was obtained directly from the collection of CCTV images at the Makassar City Communication and Informatics Office (Diskominfo), which covers various strategic locations in Makassar City, with a total of 1,120 images. Secondary data consisted of 880 images from two additional datasets, namely Roadway Flooding Image [18] and FloodIMG Dataset [19]. Thus, the dataset comprised of a total of 2,000 images categorized into two classes, namely Flood and Non-Flood classes. The Flood class included images depicting flooding, while the Non-Flood class comprised images without flooding. Figure 2 provides a visual presentation of the dataset used in this study.

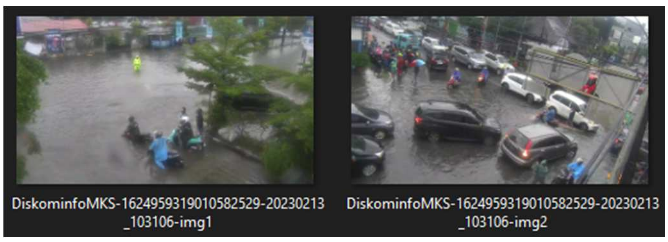


Fig. 2. Datasets

B. Pre-processing

After data collection, the next stage is the data preprocessing stage. This stage applies various techniques to enhance the quality of image data and ensure its compatibility with the deep learning model's requirements [20][21]. Techniques used in this stage include image data normalization and transformation, as well as image augmentation. Image data normalization and transformation are performed using the preprocess_input function to adjust the scale of image pixels to align with the needs of the deep learning model.

Image augmentation on MobileNet and VGG16 architectures apply identical hyperparameters to ensure both architectural models are trained with data that has similar characteristics to allow for a fair comparison of results. The hyperparameters used include an epoch of 20, a learning rate of 0.001, a maximum image rotation of 20 degrees, a maximum horizontal image shift of 10% of the image width, a maximum vertical image shift of 10% of the image height, a shear shape change of 0.2, an image zoom of 0.2 (signifying enlargement or reduction of the image to 20% of the original size), and a horizontal image flip to increase dataset variety.

The image size was set to 224x224 pixels to match the input format accepted by the deep learning model. The data was also divided into batches with each batch containing 20 images. The data generator's class mode was set as categorical, interpreting the labels on the images as class categories. Figure 3 shows a label consisting of two numbers, 0 and 1, which are used to indicate whether the image shows a flood condition or not. A label with 1 on the left and 0 on the right indicates classification of the image as flooded, while a label with 0 on the left and 1 on the right indicates classification of the image as not flooded. The preprocessed dataset can be seen in Figure 3.

Fig. 3. Image Pre-processing Result

C. CNN Model Architecture Design

The Convolutional Neural Network (CNN) architecture consists of several main parts namely the input layer, convolution layer, pooling layer, output layer, and additional layers for prediction or classification generation. The input layer receives input data, while the convolution layer detects patterns and important features in the data through convolution operations. The pooling layer reduces the spatial dimension of the features that have been generated by the convolution layer. The output layer generates a prediction or classification based on the information from the previous layer. Additional layers in the CNN architecture can improve prediction or classification accuracy. Effectively, the CNN architecture extracts important features from the input image data and produces accurate predictions or classifications [22].

This research uses VGG16 and MobileNet as the architecture for image classification. This selection is based on two main reasons. First, VGG16 is known for its high classification accuracy, which is in line with our research objective of evaluating image classification performance on flood datasets. Second, MobileNet has better computational efficiency and performance on mobile devices, which is relevant to the potential future application of the model for real-time flood monitoring and detection on highways [23].

D. MobileNet

MobileNet is a CNN architecture specifically designed for mobile devices and devices with limited resources [24]. It uses a depthwise separable convolution technique to reduce the number of parameters needed in the convolution process, resulting in a lighter and more efficient model [25].

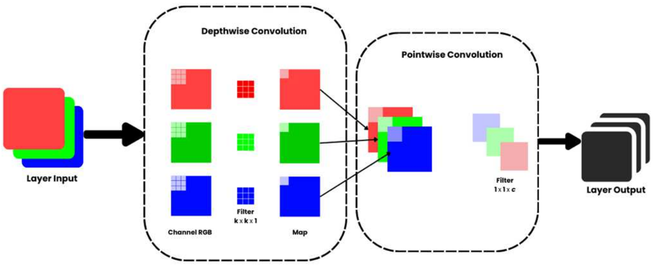


Fig. 4. MobileNet Architecture [26]

As can be seen in Figure 4, the depthwise separable convolution in MobileNet consists of two stages, namely depthwise and pointwise convolution. Depthwise convolution processes each channel in the input separately, while the pointwise combines the results of depthwise convolution using a 1x1 filter. The pointwise convolution layer is crucial for reducing model size and enhancing efficiency. Through this approach, MobileNet effectively reduces the number of parameters necessary for convolution process, leading to a more light and efficient model [27]. MobileNet's ability to achieve balance between accuracy and speed makes it well-suited for deployment in mobile applications with computational constraints.

E. VGG16

The VGG16 architecture is a CNN architecture with 16 layers, consisting of 13 convolutional layers and 3 fully connected layers [28]. Renowned for its accuracy in image classification, particularly in object recognition. It can be seen in Figure 5 that VGG16 uses a convolution filter of 3x3 and stride 1 pixel, and max pooling of 2x2 and stride 2 pixels. ReLU activation function is used in each convolutional layer and is fully connected.

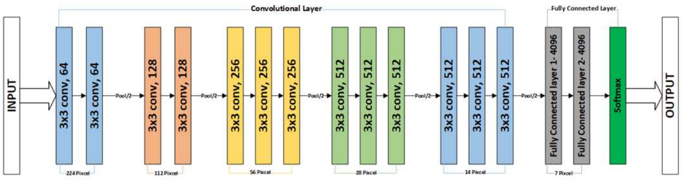


Fig. 5. VGG16 Architecture [29]

The way VGG16 works is by extracting features from images through convolution and max pooling layers. The results of the feature extraction are then used as input to the fully connected layer for classification. VGG16 also utilizes transfer learning techniques, adopting models that have been trained on large datasets such as ImageNet, enabling the achievement of high accuracy with smaller datasets. This architecture can also be used for other tasks such as object detection and image segmentation.

F. Evaluation

The final stage of this research involves evaluating model performance. The confusion matrix is a valuable tool for assessing the performance of classification models in statistical modeling or machine learning [30]. It presents model performance in a matrix format, illustrating the number of correct and incorrect predictions made by the classification model. The matrix comprises four primary cells: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). TP and TN signify accurate predictions, while FP and FN denote incorrect predictions.

The evaluation of model performance using the confusion matrix offers a more comprehensive insight into the model's capability to classify flood data [31]. The matrix values can be utilized to compute evaluation metrics such as accuracy, precision, recall, and F1-score. These metrics help assess the model's performance in the flood detection task, which is the primary focus of this research.

III. Result and Discussion

In this research, the dataset comprises 2000 images divided into two classes, namely flood images and non-flood images. The dataset is subsequently divided into three parts: training data, validation data, and test data. The training data constitutes 70% of the entire dataset, equating to 1400 images. The validation data consists of 20% of the entire dataset, totaling 400 images. Moreover, the test data consists of 10% of the entire dataset, amounting to 200 images. An example of each class in the dataset is illustrated in Figure 6.

(a)

(b)

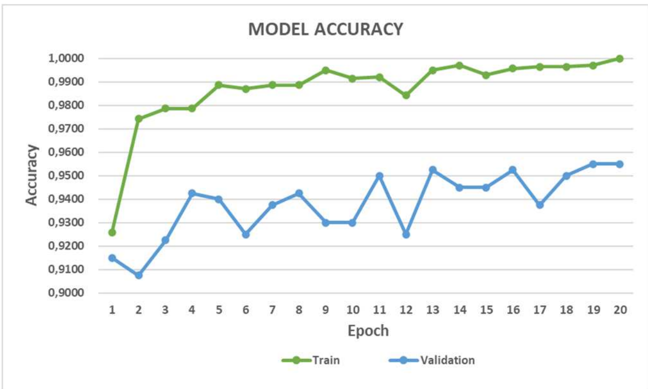
Fig. 6. (a)Non-Flood Image and (b)Flood Image

Figure 6 shows the notable distinction between flood and non-flood images. The non-flood image in Figure 6.a is distinguished by the absence of standing water. In contrast, the flood image in Figure 6.b is characterized by the presence of standing water, evident from the water waves on the road.

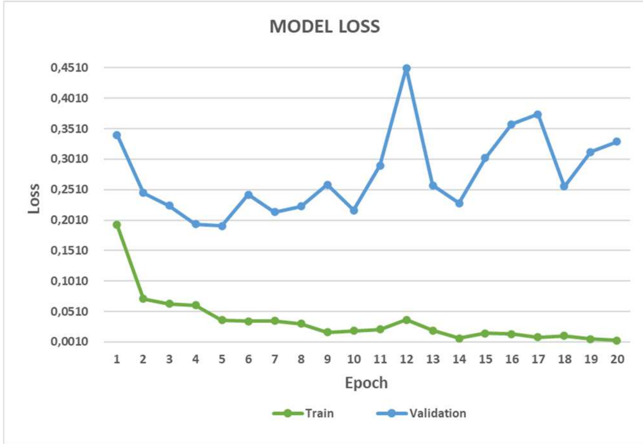
In this research, the image used as input to the model undergoes data processing as a preliminary step. The data processing is executed using the ImageDataGenerator function of Keras TensorFlow. This function performs transformation and normalization of image data, such as rotation, shifting, flipping, and zooming. Since images vary in sizes, a target size parameter of 224x224 pixels is applied during the data processing stage to standardize the image size across training data, validation data, and test data. Additionally, a batch size parameter of 20 is applied during data processing to partition the data into several batches for model training. Each batch contains a number of data samples processed by the model.

After preprocessing the image data, the subsequent step in this research is to build a model utilizing Convolutional Neural Network (CNN) architecture. Two types of CNN architectures are MobileNet and VGG16 are trained for comparison and analysis of their performance in road flood classification. Model training is conducted iteratively for 20 epochs, with each iteration involving a validation phase to assess model performance.

A. MobileNet



(a)



(b)

Fig. 7. (a)Model Accuracy MobileNet and (b)Model Loss MobileNet

Figure 7 displays two graphs : the model accuracy graph and the model loss graph. The highest model accuracy is 1,000, while the lowest model loss is 0. Both graphs feature two curves, namely the training curve and the validation curve. The training curve shows the accuracy and loss of the model during the training process, whereas the validation curve illustrates the accuracy or loss of the model after training completion.

In the accuracy graph, the training curve begins at a value of 0.9257 in the initial epoch, and reaches 1.000 in the final epoch. The validation curve starts at 0.9150 in the initial epoch, and at the last epoch is at a value of 0.9550. This indicates that the model performs well, as the gap between the training curve and the validation curve is not overly narrow, despite some fluctuation in the distance between them in certain epochs.

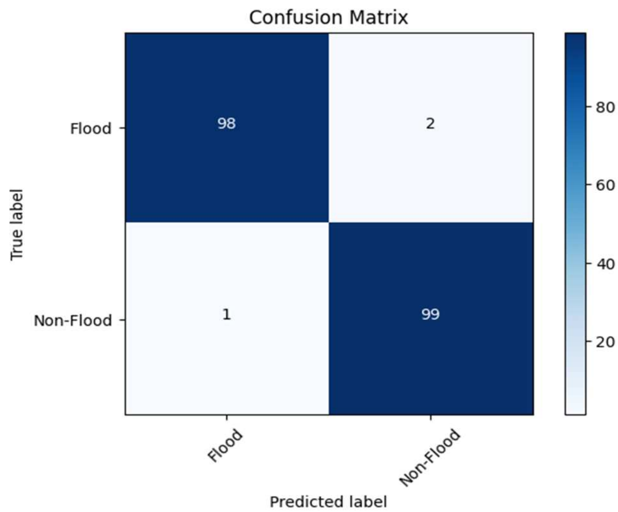
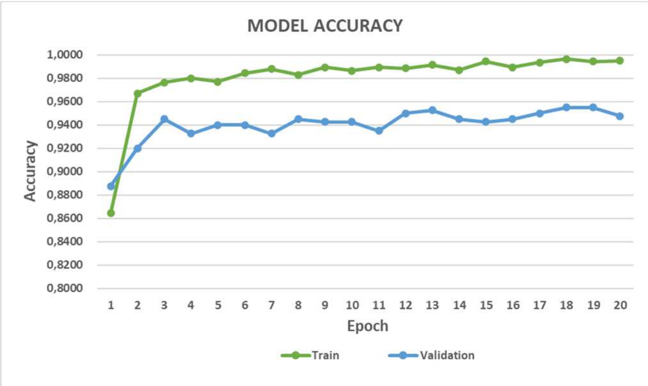


Fig. 8. MobileNet Confusion Matrix

Based on the confusion matrix in Figure 8, there are 98 images that are correctly predicted as flood images. This shows that the model has a good performance in predicting flood images. However, two images are incorrectly predicted as non-flood by the model, when they are actually flood images. This is known as a false negative (FN) and is important indicator of model performance. The smaller the FN value, the better the model performance.

In addition, there was one image that was predicted as a flood image by the model when it was actually a non-flood image. This is referred to as a false positive (FP) and is another indicator of model performance. The smaller the FP value, the better the model performance.

B. VGG16



(a)

(b)

Fig. 9. (a)Model Accuracy VGG16 and (b)Model Loss VGG16

Figure 9 shows the accuracy and loss graphs of the model. In the accuracy graph, the training curve at the initial epoch is at a value of 0.8643, and at the last epoch is at a value of 0.9950. The validation curve at the initial epoch is at a value of 0.8875, and at the last epoch is at a value of 0.9475. This shows that the performance of the model is quite good, because the distance between the training curve and the validation curve is not too tenuous. However, the model loss graph exhibits fluctuation in the distance between the curves at several epochs.

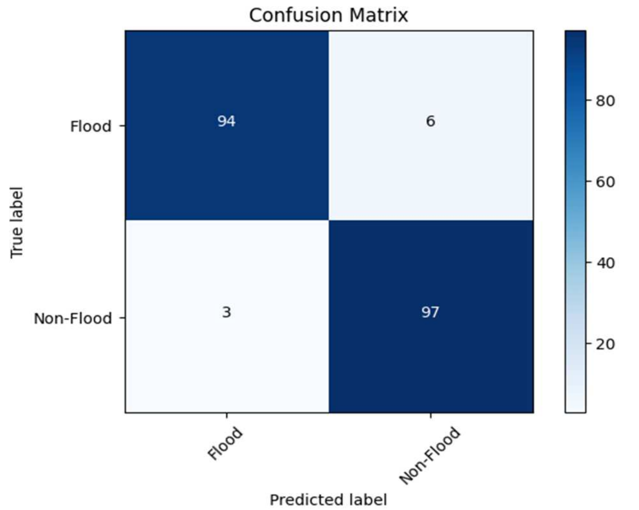


Fig. 10. VGG16 Confusion Matrix

Based on the confusion matrix in Figure 10, there are 94 true positives (TP), which are images predicted as flood by the model and are actually flood images. There are three false negatives (FN), which are images predicted as flood by the model, but are actually non-flood images. There are 97 true negatives (TN), which are images predicted as non-flood by the model and are indeed non-flood images. Finally, there are six false positives (FP), which are images predicted as non-flood by the model, but are actually flood images.

C. Comparison Results of MobileNet and VGG16 Architecture

Based on table 1, it is evident that the MobileNet architecture model has better performance than the VGG16 architecture model in terms of accuracy and computation time. The MobileNet architecture model achieves 99% accuracy with a computation time of 9 seconds, whereas the VGG16 architecture model reaches 96% accuracy with a computation time of 69 seconds.

TABLE I. COMPARISON OF MOBILENET AND VGG16 ARCHITECTURE RESULTS

Evaluation Metrics	Architecture Comparison Results	
	MobileNet	***VGG16***
Precision	98	94
F1-Score	99	96
Recall	99	97
Accuracy (%)	99	96
Computation Time (s)	9	69

The superior performance of the MobileNet model can be attributed to its simpler architecture compared to VGG16. MobileNet uses fewer layers and parameters, resulting in shorter computation time. In contrast, VGG16 has more complex architecture with additional layers and parameters, requires a greater number of computational operations leading to longer computation time.

The high accuracy of both models indicates their ability to predict the object class with a high degree of success. However, computation time is also a crucial factor in model assessment. Models that achieve high accuracy while requiring less computational time are preferred, as the speed of data processing is also an important consideration.

IV. Conclusion

Based on the results of the conducted research, it can be concluded that the MobileNet architecture shows better performance than the VGG16 architecture for flood image classification in detecting floods on highways. This superior performance is shown with an accuracy rate of 99% and a computation time of 9 seconds. The determination of the best performance is based on the comparison of accuracy and computation time from trials conducted on both architectures. However, the choice between MobileNet and VGG16 depend largely on the objectives and characteristics of the dataset used in the study. Therefore, the conclusion is that both architectures can work optimally if they are suitable for the characteristics and intended use of the dataset used.

References

[1] W. Kusumawati And F. Arofiati, "Biopori: Alternatif Pencegahan Banjir Melalui Pemberdayaan Masyarakat Pondok Pesantren," Pros. Semin. Nas. Program Pengabdi. Masy., Mar. 2021, Doi: 10.18196/Ppm.32.222.

[2] Indra Khaidir, "Mitigasi Bencana Banjir Untuk Mengurangi Dampak Terhadap Lingkungan Dan Kehidupan Sosial Masyarakat," J. Rekayasa, Vol. 8, No. 2, Pp. 154–160, Nov. 2019, Doi: 10.37037/Jrftsp.V8i2.29.

[3] S. Wiśniewski, M. Kowalski, And M. Borowska-Stefańska, "Flooding And Mobility: A Polish Analysis," Environ. Hazards, Vol. 20, No. 3, Pp. 300–322, May 2021, Doi: 10.1080/17477891.2020.1810608.

[4] M. H. Andreasen, J. Agergaard, L. Møller-Jensen, M. Oteng-Ababio, And G. A. B. Yiran, "Mobility Disruptions In Accra: Recurrent Flooding, Fragile Infrastructure And Climate Change," Sustainability, Vol. 14, No. 21, P. 13790, Oct. 2022, Doi: 10.3390/Su142113790.

[5] S. I. Islam, S. Shams-E-Rabbi, And M. S. Anita, "Causes And Effects Of Water Logging In Mirpur Area, Dhaka City," 2020.

[6] C. Ntakiyimana, Y. Zhang, And G. Twagirayezu, "Road Flooding In Kigali City, Rwanda:Causes, Effects On Road Transportation Mitigation Measures," Pol. J. Environ. Stud., Vol. 31, No. 4, Pp. 3735–3744, Jul. 2022, Doi: 10.15244/Pjoes/146215.

[7] M. Dietze Et Al., "More Than Heavy Rain Turning Into Fast-Flowing Water – A Landscape Perspective On The 2021 Eifel Floods," Nat. Hazards Earth Syst. Sci., Vol. 22, No. 6, Pp. 1845–1856, Jun. 2022, Doi: 10.5194/Nhess-22-1845-2022.

[8] Q. Sholihah, W. Kuncoro, S. Wahyuni, S. Puni Suwandi, And E. Dwi Feditasari, "The Analysis Of The Causes Of Flood Disasters And Their Impacts In The Perspective Of Environmental Law," Iop Conf. Ser. Earth Environ. Sci., Vol. 437, No. 1, P. 012056, Feb. 2020, Doi: 10.1088/1755-1315/437/1/012056.

[9] "Data Informasi Bencana Indonesia (Dibi)." Accessed: Sep. 10, 2023. [Online]. Available: Https://Dibi.Bnpb.Go.Id/

[10] N. B. Dung, N. Q. Long, R. Goyal, D. T. An, And D. T. Minh, "The Role Of Factors Affecting Flood Hazard Zoning Using Analytical Hierarchy Process: A Review," Earth Syst. Environ., Vol. 6, No. 3, Pp. 697–713, Sep. 2022, Doi: 10.1007/S41748-021-00235-4.

[11] W. Xu, Y. Zhang, David. Proverbs, And Z. Zhong, "Enhancing The Resilience Of Road Networks To Flooding," Int. J. Build. Pathol. Adapt., Jul. 2022, Doi: 10.1108/Ijbpa-09-2021-0120.

[12] S. Praharaj, T. D. Chen, F. T. Zahura, M. Behl, And J. L. Goodall, "Estimating Impacts Of Recurring Flooding On Roadway Networks: A Norfolk, Virginia Case Study," Nat. Hazards, Vol. 107, No. 3, Pp. 2363–2387, Jul. 2021, Doi: 10.1007/S11069-020-04427-5.

[13] T. Q. Dam, D. K. Ninh, A. N. Le, V. D. Pham, And T. D. Le, "Danaflood: A Solution For Scalable Urban Street Flood Sensing," In Intelligent Systems And Networks, Vol. 752, T. D. L. Nguyen, E. Verdú, A. N. Le, And M. Ganzha, Eds., In Lecture Notes In Networks And Systems, Vol. 752. , Singapore: Springer Nature Singapore, 2023, Pp. 546–555. Doi: 10.1007/978-981-99-4725-6_65.

[14] M. A. Islam, S. I. Rashid, N. U. I. Hossain, R. Fleming, And A. Sokolov, "An Integrated Convolutional Neural Network And Sorting Algorithm For Image Classification For Efficient Flood Disaster Management," Decis. Anal. J., Vol. 7, P. 100225, Jun. 2023, Doi: 10.1016/J.Dajour.2023.100225.

[15] A. Rambhad, D. P. Singh, And J. Choudhary, "Detection Of Flood Events From Satellite Images Using Deep Learning," In Intelligent Data Engineering And Analytics, Vol. 327, V. Bhateja, X.-S. Yang, J. Chun-Wei Lin, And R. Das, Eds., In Smart Innovation, Systems And Technologies, Vol. 327. , Singapore: Springer Nature Singapore, 2023, Pp. 259–268. Doi: 10.1007/978-981-19-7524-0_23.

[16] Y. Liu, Z. Wang, R. Wang, J. Chen, And H. Gao, "Flooding-Based Mobilenet To Identify Cucumber Diseases From Leaf Images In Natural Scenes," Comput. Electron. Agric., Vol. 213, P. 108166, Oct. 2023, Doi: 10.1016/J.Compag.2023.108166.

[17] D. Salluri, K. Bade, And G. Madala, "Object Detection Using Convolutional Neural Networks For Natural Disaster Recovery," Int. J. Saf. Secur. Eng., Vol. 10, No. 2, Pp. 285–291, Apr. 2020, Doi: 10.18280/Ijsse.100217.

[18] "Roadway Flooding Image Dataset." Accessed: Sep. 14, 2023. [Online]. Available: Https://Www.Kaggle.Com/Datasets/Saurabhshahane/Roadway-Flooding-Image-Dataset

[19] "Floodimg: Flood Image Database System." Accessed: Sep. 14, 2023. [Online]. Available: Https://Www.Kaggle.Com/Datasets/Hhrclemson/Flooding-Image-Dataset

[20] A. Isenko, R. Mayer, J. Jedele, And H.-A. Jacobsen, "Where Is My Training Bottleneck? Hidden Trade-Offs In Deep Learning Preprocessing Pipelines," 2022, Doi: 10.48550/Arxiv.2202.08679.

[21] Z. Xiao Et Al., "Impacts Of Data Preprocessing And Selection On Energy Consumption Prediction Model Of Hvac Systems Based On Deep Learning," Energy Build., Vol. 258, P. 111832, Mar. 2022, Doi: 10.1016/J.Enbuild.2022.111832.

[22] A. Asnidar Et Al., "Application Of Mobilenetv2 Architecture To Classification Of Knee Osteoarthritis Based On X-Ray Images," In 2023 International Conference On Advanced Mechatronics, Intelligent Manufacture And Industrial Automation (Icamimia), Surabaya, Indonesia: Ieee, Nov. 2023, Pp. 375–380. Doi: 10.1109/Icamimia60881.2023.10427581.

[23] D. Ramayanti, Sri Dianing Asri, And Lionie Lionie, "Implementasi Model Arsitektur Vgg16 Dan Mobilenetv2 Untuk Klasifikasi Citra Kupu-Kupu," Jsai J. Sci. Appl. Inform., Vol. 5, No. 3, Pp. 182–187, Nov. 2022, Doi: 10.36085/Jsai.V5i3.2864.

[24] E. Elfatimi, R. Eryigit, And L. Elfatimi, "Beans Leaf Diseases Classification Using Mobilenet Models," Ieee Access, Vol. 10, Pp. 9471–9482, 2022, Doi: 10.1109/Access.2022.3142817.

[25] M. Lopez-Montiel, U. Orozco-Rosas, M. Sanchez-Adame, K. Picos, And O. H. M. Ross, "Evaluation Method Of Deep Learning-Based Embedded Systems For Traffic Sign Detection," Ieee Access, Vol. 9, Pp. 101217–101238, 2021, Doi: 10.1109/Access.2021.3097969.

[26] C. Cheuque, M. Querales, R. León, R. Salas, And R. Torres, "An Efficient Multi-Level Convolutional Neural Network Approach For White Blood Cells Classification," Diagnostics, Vol. 12, No. 2, P. 248, Jan. 2022, Doi: 10.3390/Diagnostics12020248.

[27] A. S. Agung, S. Gunawan Zain, F. Adiba, D. D. Andayani, And A. Baso Kaswar, "Urban Flood Disaster Mitigation Through Image Classification Using Transfer Learning Method With Mobilenet Fine-Tuning," In 2023 International Conference On Advanced Mechatronics, Intelligent Manufacture And Industrial Automation (Icamimia), Surabaya, Indonesia: Ieee, Nov. 2023, Pp. 364–369. Doi: 10.1109/Icamimia60881.2023.10427560.

[28] M. Ye Et Al., "A Lightweight Model Of Vgg-16 For Remote Sensing Image Classification," Ieee J. Sel. Top. Appl. Earth Obs. Remote Sens., Vol. 14, Pp. 6916–6922, 2021, Doi: 10.1109/Jstars.2021.3090085.

[29] R. Rismiyati And A. Luthfiarta, "Vgg16 Transfer Learning Architecture For Salak Fruit Quality Classification," Telematika, Vol. 18, No. 1, P. 37, Mar. 2021, Doi: 10.31315/Telematika.V18i1.4025.

[30] P. R. Sihombing And A. M. Arsani, "Comparison Of Machine Learning Methods In Classifying Poverty In Indonesia In 2018," J. Tek. Inform. Jutif, Vol. 2, No. 1, Pp. 51–56, Jan. 2021, Doi: 10.20884/1.Jutif.2021.2.1.52.

[31] E. Kristiani, C.-T. Yang, And C.-Y. Huang, "Isec: An Optimized Deep Learning Model For Image Classification On Edge Computing," Ieee Access, Vol. 8, Pp. 27267–27276, 2020, Doi: 10.1109/Access.2020.2971566.