Comparative Analysis of ResNet50V2, ResNet152V2, and MobileNetV2 Architectures in Monkeypox Classification

Habi Jiyan Mustaqim [1]*, Ozzi Suria [2]*
* Information System Department, Universitas Mercu Buana Yogyakarta
habijiyan111@gmail.com [1], ozzisuria@mercubuana-yogya.ac.id [2]

Article Info

Article history:

Received 2025-06-05
Revised 2025-07-03
Accepted 2025-07-04

Keyword:

Monkeypox,
ResNet,
MobileNet,
Image Classification,
CNN.

ABSTRACT

Convolutional Neural Networks (CNN) are recognized for their high accuracy in image classification, but large-scale datasets and significant computer resources are needed to train them from scratch, though. Transfer learning offers a practical solution by leveraging pre-trained models to accelerate training even when data is limited. Although CNNs have been widely applied to skin disease classification, specific evaluations of architectures such as ResNet50V2, ResNet152V2, and MobileNetV2 for monkeypox image classification remain scarce. Therefore, this study aims to comprehensively compare the effectiveness and trade-offs of these architectures in detecting monkeypox through transfer learning. The evaluation focuses on balancing accuracy and computational efficiency across stages, including data collection, preprocessing, model design, training, and testing. The dataset, obtained from Kaggle, consists of 2,310 images across four classes: monkeypox, chickenpox, measles, and normal. Transfer learning was implemented using fine-tuned weights from ImageNet. According to the results, ResNet152V2 needed the most training time but had the lowest loss and the greatest validation accuracy (98.28%). ResNet50V2 maintained a good compromise between accuracy (97.84%) and training efficiency, while MobileNetV2 yielded the best overall classification metrics (97.86% for accuracy, precision, recall, and F1-score), indicating strong generalization. These findings highlight the distinct strengths of each model, offering insights into architecture selection based on specific operational constraints and goals.

I. Introduction

Monkeypox cases in 2022 emerged as a new global health concern while the world was still recovering from the impact of the COVID-19 pandemic that began in 2020 [1]. Monkeypox is an infectious illness caused by a virus belonging to the Orthopoxvirus category. This virus was initially discovered in 1958 from blister-like lesions observed in monkeys that were isolated in Copenhagen [2]. The only way for the virus to spread is through physical contact with infected individuals, animals, or other things. The routes of transmission are respiratory droplets, nasal secretions, or saliva [3]. The monkeypox virus, which causes monkeypox, was first discovered to infect people in the Congo in 1970. On August 19, 2022, a 27-year-old man in Jakarta was found to be the first instance in Indonesia. There were two suspected cases and 63 cases that tested negative as of September 15, 2022. Ten Indonesian provinces made up the distribution of the 63 negative instances [4]. This fact indicates the potential for the disease to spread more widely, making early detection an important step in supporting the overall effectiveness of prevention and infection control efforts.

Convolutional Neural Networks (CNNs) are among the most promising techniques for image processing, demonstrating high accuracy and practical performance across various classification and recognition tasks [5]. In 1998, the first CNN architecture, LeNet, was introduced. However, due to technological limitations, limited data, and errors in the use of non-linear functions, this method did not receive much attention at the time [7]. Convolutional Neural

Networks (CNN) are increasingly used in image classification due to their high accuracy. However, training them from scratch demands large amounts of data and substantial computational resources. This challenge can be mitigated through transfer learning, which leverages pre-trained networks and enables faster, more efficient training by fine-tuning on smaller datasets [8].

Several previous studies have explored the use of deep learning for monkeypox classification, producing promising results while indicating opportunities for further improvement and investigation. For example, Islam at al uses a variety of CNN training models, including VGG19, VGG16, ResNet50, MobileNetV2, AlexNet, InceptionV3, and DenseNet121, and achieves a high accuracy of 99.52% through ensemble modeling and comparison with machine learning algorithms, such as Random Forest, Decision Tree, and K-Nearest Neighbors. However, even if the results are encouraging, this study focuses more on accuracy without providing any explanations on computer efficiency or the ability to change models in environments with changing daylight levels [9]. Other research by Fransisca compares the performance of Adam and RMSprop and uses MobileNetV2 with transfer learning to detect monkeypox. Although the Adam optimizer resulted in higher loss values, this study does not examine the generalizability of the model to various skin conditions [10].

The study conducted by Saputra using ResNet50 shows a range of accuracy across scenarios, with the highest accuracy in the uji data being 76.10 percent, indicating a lack of control in performance [11]. Conversely, Prasetyo and Ichwan evaluated the ekstraksi feature and classification using ResNet50 and ResNet152 on X-ray lung images, and they concluded that ResNet152 had better performance. However, they did not focus their research on monkeypox, and they also noted that the quantity and composition of the dataset significantly impacted the results [12]. All things considered, these investigations demonstrate that CNN architecture varies for monkeypox classification, especially with regard to accuracy, efficiency in computation, and practical use in the afflicted area.

Research that precisely examines the performance of CNN designs like ResNet50V2, ResNet152V2, and MobileNetV2 in monkeypox picture classification is still scarce, even though several studies have successfully used CNN for skin disease image classification. Thus, the purpose of this study is to assess the efficacy of these three designs by taking into account the trade-off between computing efficiency and accuracy. The findings of this study are expected to support the development of more accurate and computationally efficient image-based monkeypox detection systems.

II. Research Methods

The study technique encompasses the following phases: data collection, data preprocessing, model architecture design, model training, and model evaluation. Figure 1 illustrates the research methodology's flow.

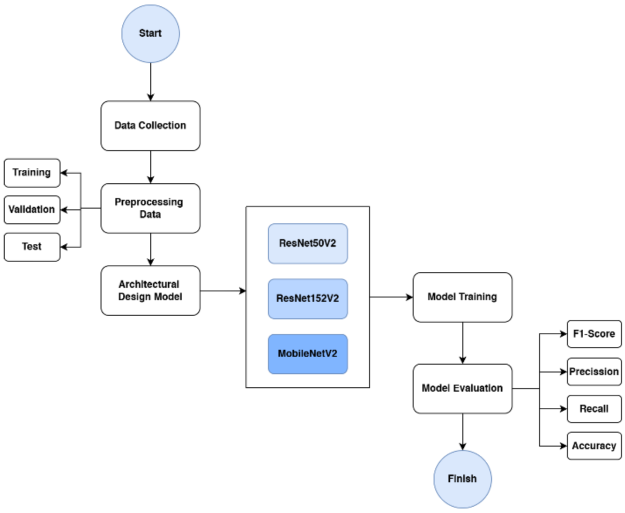

Figure 1. Research methods

A. Data Collection

The dataset used in this study was acquired via Kaggle [13]. The dataset contains 2,310 images that have been augmented and is divided into four classes: monkeypox (837 images), chickenpox (321 images), measles (273 images), and normal (879 images). All images are in PNG format with a resolution of 224x224 pixels. A comprehensive breakdown of data for each grade may be seen in TABLE I. Figure 2 displays sample data from each class in this dataset.

Figure 2. Image Dataset

B. Preprocessing Data

Data preparation encompasses all the steps done before the start of the real data analysis process. To make learning easier and more applicable in real-world situations, data preparation enables users to choose how to convey the analytic findings, what concepts will be learnt, and how to format the data [14]. With an 80:10:10 ratio, three categories are created from the data: test, validation, and training. The amount of data used after preprocessing is shown in TABLE I.

TABLE I
DATA SPLITTING

Class	Training	Validation	Test
Monkeypox	669	84	84
Chickenpox	256	32	33
Measles	218	27	28
Normal	701	89	89
Total	**1844**	**232**	**234**
		2.310	

C. Architectural Design Model

At this point, a transfer learning methodology was utilized to construct the model architecture. Using initial weights pre-trained on the ImageNet dataset, three convolutional neural network (CNN) architectures, ResNet50V2, ResNet152V2, and MobileNetV2, were employed as base models.

The three CNN architectures evaluated in this study, ResNet50V2, ResNet152V2, and MobileNetV2, exemplify different design principles to optimize performance and computational efficiency. An improvement on the 50-layer Residual Network (ResNet) model, the ResNet50V2 architecture uses residual learning to solve the vanishing gradient issue in deep networks. This model uses pre-activation residual blocks, where batch normalization and the ReLU activation function are placed before the convolutional layers, thereby improving gradient flow during training [15], [16]. With 152 layers, ResNet152V2 is a deeper version of ResNet that makes it possible to extract more intricate and representative characteristics. It also applies pre-activation residual blocks, similar to ResNet50V2, to maintain training stability in very deep networks [17]. MobileNetV2 is designed as a lightweight and efficient CNN architecture that minimizes computational complexity and memory usage, making it suitable for environments with constrained resources. Inverted residual blocks with bottlenecks and depthwise separable convolutions are used in this model to considerably cut down on parameters and processing demands without compromising accuracy [15], [18].

The top levels of each model were swapped out with custom layers that were determined by the number of classes in the target dataset. These additional layers included a GlobalAveragePooling2D layer to flatten the spatial features, a BatchNormalization layer to accelerate and stabilize the training process, a Dense layer with 128 neurons using the ReLU activation function, a Dropout layer with a rate of 0.5 to reduce the risk of overfitting, and an output layer with a softmax activation function to enable multi-class classification. During the initial training phase, all weights of the base models were frozen to retain the pre-trained weights, allowing fine-tuning to be applied solely to the newly added layers.

D. Model Training

The experimental process in this study was conducted using Google Colab with a T4 GPU runtime to facilitate efficient training and execution of the models. At this stage, each architecture, ResNet50V2, ResNet152V2, and MobileNetV2, was compiled using the Adam optimizer. This optimizer was chosen based on previous studies indicating its superior performance in monkeypox image classification tasks [10]. Using categorical cross-entropy as the loss function, accuracy served as the assessment metric. A learning rate of 0.0001 was applied, and the training process was carried out for a maximum of 50 epochs. Two callbacks were used to improve the training procedure and avoid overfitting.

The first was called EarlyStopping, which immediately stops training after five consecutive epochs if the validation loss does not improve. This helps avoid overfitting and ensures efficient use of computational resources. Additionally, the restore_best_weights=True option was enabled to automatically revert the model to the best-performing weights based on validation loss. The second callback was ReduceLROnPlateau, which lowers the learning rate if validation loss does not improve over three consecutive epochs, enabling more stable learning during the later phases of training. Each of the three architectures was trained separately using the same dataset and hyperparameters, ensuring a fair and consistent comparison of model performance.

E. Model Evaluation

The confusion matrix was used for model assessment in order to evaluate multi-class classification performance in a comprehensive manner. A binary matrix known as the confusion matrix illustrates the classification model's performance and how well it predicts outcomes [19]. By utilizing the Confusion Matrix, we can analyze and calculate evaluation metrics such as Accuracy, Precision, Recall, and F1-score to comprehensively assess the model’s performance [20], [21]. These evaluation results serve as the basis for determining the most optimal model architecture for accurately and efficiently classifying monkeypox images.

Weighted averaging is used to determine class imbalance in the dataset, and metrics such as accuracy, precision, recall, and F1-score can be used to evaluate a classification model's performance. Based on the Confusion Matrix, the four metrics were computed using the following equations:

1) Accuracy

Accuracy is defined as the proportion of correct model predictions over all tested data. To put it another way, accuracy quantifies the proportion of accurate predictions the model makes, whether for the positive or negative class. Equation (1) presents the formula used to calculate accuracy.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (1)$$

2) Precision

Precision indicates how accurately the model identifies positive cases. Equation (2) illustrates the percentage of accurately predicted positives among all positive forecasts.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

3) Recall

Recall, sometimes referred to as sensitivity, gauges how well the model can recognize all positive data. This measure shows the percentage of real positive cases that the model accurately detects. The equation demonstrates the recall formula (3).

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

4) F1-Score

F1-score is the mean of precision and recall. This metric is used to achieve a balance between the two, especially when there is a trade-off between precision and recall. The formula for the F1-score can be seen in Equation (4).

$$F1\ Score = 2 \text{ x} \frac{Precision * Recall}{Precission + Recall} \quad (4)$$

Information:
True Positive (TP): Accurately predicting a positive outcome
True Negative (TN): Accurately predicting a negative outcome
False Positive (FP): An inaccurate, optimistic forecast
False Negative (FN): An inaccurate, negative prediction

III. RESULTS AND DISCUSSION

A. Model Training Results

Here, the discussion focuses on the training results of the ResNet50V2, ResNet152V2, and MobileNetV2 models in detail according to evaluation measures, including train accuracy, loss accuracy, validation accuracy, validation loss, and epoch time. Per-epoch time is useful for measuring the efficiency of each training iteration and helps identify whether the training process is running smoothly or facing computational bottlenecks. For example, the VGG16 CNN model requires approximately 29 seconds per epoch, while ResNet50 takes around 16 seconds per epoch [22], [23]. Epoch time also helps determine the ideal number of epochs to prevent overfitting and ensuring that training doesn't take too long. This comparison uses the training and validation datasets. Such a discussion is important to justify which model is the most optimal to be used for image classification in this study. The results of comparing the three CNN architectures can be observed in TABLE II.

TABLE II
TRAINING AND VALIDATION RESULTS

Model	Train Acc	Train Loss	Val Acc	Val Loss	Epoch Time
ResNet50V2	0.9931	0.0352	0.9784	0.0633	12-21s
ResNet152V2	0.9960	0.0364	0.9828	0.0456	14-21s
MobileNetV2	0.9917	0.0445	0.9741	0.0881	11-21s

B. Confusion Matrix

This part thoroughly assesses the model's performance by examining and calculating evaluation metrics like accuracy, precision, recall, and F1-score. These metrics are crucial for understanding the model's overall accuracy as well as its capacity to accurately identify each class, particularly in situations involving multi-class classification or unbalanced data. All evaluation results are calculated based on the formulas presented in the research methodology section.

This matrix displays the number of true positives, false positives, true negatives, and false negatives for each class, providing insight into the model's performance and any issues. Weighted averaging was used to compute evaluation measures, including precision, recall, and F1-score, to correct for class imbalance in the dataset and avoid the metrics being disproportionately affected by the majority class alone. The confusion matrix results for the ResNet50V2, ResNet152V2, and MobileNetV2 architectures can be seen in Figure 3, Figure 4, and Figure 5.

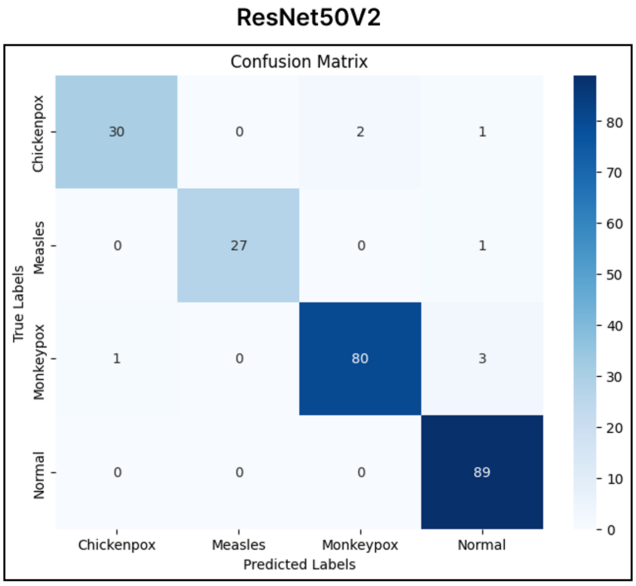

Figure 3 Confusion Matrix ResNet50V2

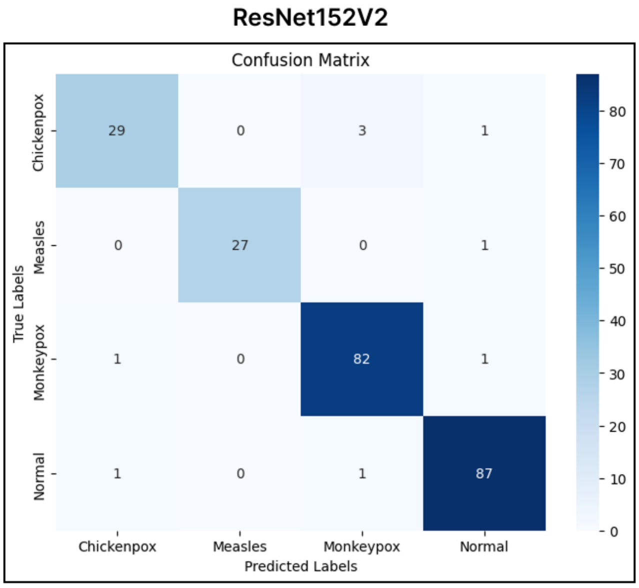

Figure 4 Confusion Matrix ResNet152V2

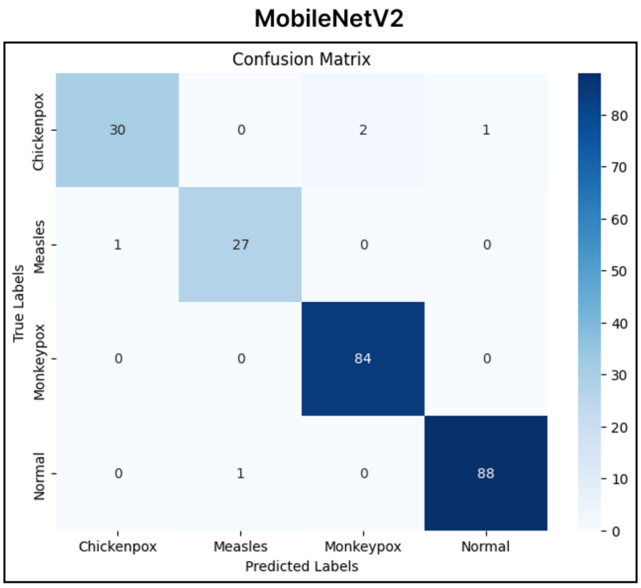

Figure 5 Confusion Matrix MobileNetV2

In TABLE III, the model demonstrates strong categorization capabilities, as reflected by the Confusion Matrix. This table shows the number of accurate and inaccurate predictions for each class and was produced using the confusion matrix computation. It serves as the foundation for computing crucial evaluation metrics, such as accuracy, precision, recall, and F1-score, to fully analyze the model's overall performance.

TABLE III
CONFUSION MATRIX RESNET50V2, RESNET152V2 DAN MOBILENETV2

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**
ResNet50V2	96.58	96.65	96.58	96.56
ResNet152V2	96.15	96.15	96.15	96.13
MobileNetV2	97.86	97.86	97.86	97.84

C. Discussion

The ResNet152V2 model, which had more parameters, had the best training accuracy (99.60%) and validation accuracy (98.28%). The training loss and validation loss were also lower compared to the other models (0.0364 and 0.0456, respectively). However, the training time per epoch was longer, ranging from 14 to 21 seconds. Based on the classification evaluation, this model achieved an accuracy of 96.15%, precision of 96.15%, recall of 96.15%, and F1-score of 96.13%. These results support ResNet152V2 superiority and are in line with those of Merlina et al., who demonstrated that the residual architecture of this model enables it to perform very well in challenging classification tasks [24].

The ResNet50V2 model demonstrated solid performance with a training accuracy of 99.31% and a validation accuracy of 97.84%. The training and validation losses were relatively low, at 0.0352 and 0.0633, respectively. The training time per epoch ranged from 12 to 21 seconds. From the confusion matrix evaluation, this model achieved an accuracy of 96.58%, a precision of 96.65%, a recall of 96.58%, and an F1-score of 96.56%. These findings show consistent performance on both training and validation data, with a good trade-off between computational efficiency and accuracy [25].

The MobileNetV2 model, designed for efficiency, showed a training accuracy of 99.17% and a validation accuracy of 97.41%. It recorded a training loss of 0.0445 and a validation loss of 0.0881, with the shortest epoch training time (11–21 seconds). With the greatest evaluation metrics accuracy, precision, recall, and F1-score all at 97.86%, this model beat the others based on the confusion matrix, demonstrating its high efficiency and dependability, and making it particularly appropriate for deployment on devices with limited resources [26].

Overall, each model exhibits different performance characteristics: ResNet152V2 excels in achieving the highest validation accuracy, ResNet50V2 offers shorter training time with solid performance, while MobileNetV2 stands out for its efficiency and superior classification results. The best model selection depends on the application context, whether prioritizing accuracy, computational efficiency, or a balanced trade-off between both.

IV. CONCLUSION

This study compared the performance of three CNN models, ResNet152V2, ResNet50V2, and MobileNetV2, for classifying monkeypox images. Each model demonstrated specific strengths. The best accuracy was attained by ResNet152V2, which makes it appropriate for applications that need the highest level of precision, even with prolonged training periods. For systems with limited resources, ResNet50V2 provided a fair compromise between computational efficiency and accuracy. MobileNetV2 excelled in speed and efficiency, making it appropriate for devices with limited processing power. With proper dataset optimization, all three models show strong potential for use in image-based disease detection systems, depending on the application's specific needs and constraints.

REFERENCES

[1] M. E. Wilson, J. M. Hughes, A. M. McCollum, and I. K. Damon, "Human Monkeypox," *Clinical Infectious Diseases*, vol. 58, no. 2, pp. 260–267, Jan. 2014, doi: 10.1093/CID/CIT703.

[2] M. Pal, F. Mengstie, and V. Kandi, "Epidemiology, Diagnosis, and Control of Monkeypox Disease: A comprehensive Review," *American Journal of Infectious Diseases and Microbiology*, vol. 5, no. 2, pp. 94–99, 2017, doi: 10.12691/ajidm-5-2-4.

[3] P. Y. Nguyen, W. S. Ajisegiri, V. Costantino, A. A. Chughtai, and C. R. MacIntyre, "Reemergence of Human Monkeypox and Declining Population Immunity in the Context of Urbanization, Nigeria, 2017–2020," *Emerg Infect Dis*, vol. 27, no. 4, p. 1007, Apr. 2021, doi: 10.3201/EID2704.203569.

[4] Kementerian Kesehatan Republik Indonesia, "Kasus Monkeypox Pertama di Indonesia Terkonfirmasi," 2022.

[5] J. Liu, H. Sun, and J. Katto, "Learned Image Compression With Mixed Transformer-CNN Architectures," 2023. doi: https://doi.org/10.1109/CVPR52729.2023.01383.

[6] A. W. Salehi *et al.*, "A Study of CNN and Transfer Learning in Medical Imaging: Advantages, Challenges, Future Scope," *Sustainability 2023, Vol. 15, Page 5930*, vol. 15, no. 7, p. 5930, Mar. 2023, doi: 10.3390/SU15075930.

[7] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2323, 1998, doi: 10.1109/5.726791.

[8] D. M. Wonohadidjojo, "Perbandingan Convolutional Neural Network pada Transfer Learning Method untuk Mengklasifikasikan Sel Darah Putih," *Ultimatics : Jurnal Teknik Informatika*, vol. 13, no. 1, pp. 51–57, Jun. 2021, doi: 10.31937/TI.V13I1.2040.

[9] S. Islam, F. A. Nishi, T. Akter, and M. A. Azim, "Monkeypox Skin Lesion Detection with Deep Learning and Machine Learning," *Int J Comput Appl*, vol. 185, no. 23, pp. 39–45, Jul. 2023, doi: 10.5120/IJCA2023922984.

[10] P. Sarah Fransisca and N. Matondang, "Deteksi Citra Digital Penyakit Cacar Monyet menggunakan Algoritma Convolutional Neural Network dengan Arsitektur MobileNetV2 Digital Image Detection of Monkeypox Disease using Convolutional Neural Network Algorithm with MobilenetV2 Architecture", [Online]. Available: http://journal.ipb.ac.id/index.php/jika

[11] T. O. Saputra, "Klasifikasi Penyakit Monkeypox Berdasarkan Lesi Kulit Menggunakan Metode CNN Arsitektur ResNet-50," 2023.

[12] R. R. E. Prasetyo and M. Ichwan, "Perbandingan Metode Deep Residual Network 50 dan Deep Residual Network 152 untuk Deteksi Penyakit Pneumonia pada Manusia," *MIND Journal*, vol. 6, no. 2, pp. 168–182, Dec. 2021, doi: 10.26760/mindjournal.v6i2.168-182.

[13] "MonkeyPox_Aug_Munim." Accessed: Jun. 02, 2025. [Online]. Available: https://www.kaggle.com/datasets/ahmedmunim/monkeypox-aug-munim

[14] A. FAMILI, W. SHEN, R. WEBER, and E. SIMOUDIS, "Data preprocessing and intelligent data analysis," *Intelligent Data Analysis*, vol. 1, no. 1–4, pp. 3–23, Jan. 1997, doi: 10.1016/S1088-467X(98)00007-9.

[15] M. B. Kurniawan and E. Utami, "Performance Comparison of ResNet50, VGG16, and MobileNetV2 for Brain Tumor Classification on MRI Images," *SISTEMASI*, vol. 14, no. 2, pp. 767–777, Mar. 2025, doi: 10.32520/STMSI.V14I2.5054.

[16] "ResNet and ResNetV2." Accessed: Jun. 22, 2025. [Online]. Available: https://keras.io/api/applications/resnet/

[17] M. Harahap, E. M. Laia, L. S. Sitanggang, M. Sinaga, D. F. Sihombing, and A. M. Husein, "Deteksi Penyakit Covid-19 Pada Citra X-Ray Dengan Pendekatan Convolutional Neural Network (CNN)," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 1, pp. 70–77, Feb. 2022, doi: 10.29207/RESTI.V6I1.3373.

[18] M. N. Winnarto, "Penerapan Arsitektur Mobilenetv2 Pada Klasifikasi Penyakit Daun Teh," pp. 1–82, 2021, Accessed: Jun. 22, 2025. [Online]. Available: https://repository.bsi.ac.id/repo/35611/Penerapan-Arsitektur-Mobilenetv2-Pada-Klasifikasi-Penyakit-Daun-Teh

[19] L. Sharma and M. Carpenter, *Computer Vision and Internet of Things*. Boca Raton: Chapman and Hall/CRC, 2022. doi: 10.1201/9781003244165.

[20] R. B. Pereira, A. Plastino, B. Zadrozny, and L. H. C. Merschmann, "Correlation analysis of performance measures for multi-label classification," *Inf Process Manag*, vol. 54, no. 3, pp. 359–369, May 2018, doi: 10.1016/J.IPM.2018.01.002.

[21] A. F. Giraldo-Forero, J. A. Jaramillo-Garzón, and C. G. Castellanos-Domínguez, "Evaluation of Example-Based Measures for Multi-label Classification Performance," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9043, pp. 557–564, 2015, doi: 10.1007/978-3-319-16483-0_54.

[22] F. A. Nugroho, N. Wiliani, and D. Direvisi, "Perbandingan Kinerja ANN dan CNN dalam Tugas Klasifikasi Citra Berbasis Pembelajaran Mesin," *Teknomatika: Jurnal Informatika dan Komputer*, vol. 18, no. 1, pp. 22–27, Jun. 2025, doi: 10.30989/TEKNOMATIKA.V18I1.1561.

[23] A. L. Pramana, "Comparative Analysis of Deep Learning Method Evaluation on Vehicle Type Classification," *Journal of Computer Science and Visual Communication Design*, vol. 9, no. 1, pp. 451–465, Jul. 2024, doi: 10.55732/JIKDISKOMVIS.V9I1.1252.

[24] N. Merlina, A. Prasetyo, I. Zuniarti, N. A. Mayangky, D. N. Sulistyowati, and F. Aziz, "Improving Early Detection of Cervical Cancer Through Deep Learning-Based Pap Smear Image Classification," *Journal of Applied Data Sciences*, vol. 6, no. 2, pp. 952–968, Mar. 2025, doi: 10.47738/JADS.V6I2.576.

[25] Y. Azhar, W. P. Wicaksono, Z. Sari, and W. P. Wicaksono, "Pneumonia Diagnosis Through Deep Learning: ResNet50v2 Model Implementation," *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, vol. 13, no. 2, pp. 349–358, Jul. 2024, doi: 10.23887/JANAPATI.V13I2.72068.

[26] X. Liao *et al.*, "Automated detection of abnormal respiratory sound from electronic stethoscope and mobile phone using MobileNetV2," *Biocybern Biomed Eng*, vol. 43, no. 4, pp. 763–775, Oct. 2023, doi: 10.1016/J.BBE.2023.11.001.