JPPIPA 11(7) (2025)

Jurnal Penelitian Pendidikan IPA

http://jppipa.unram.ac.id/index.php/jppipa/index

Advanced Chicken Breed Identification Using Transfer Learning Techniques with the VGG16 Convolutional Neural Network Architecture

Nagala Wangsa Kencana[1*], Rusydi Umar[1], Murinto[2]

[1]Master's Program in Informatics, Universitas Ahmad Dahlan, Indonesia.
[2]Department of Informatics, Universitas Ahmad Dahlan, Indonesia.

Received: May 19, 2025
Revised: June 24, 2025
Accepted: July 25, 2025
Published: July 31, 2025

Corresponding Author:
Nagala Wangsa Kencana
wangsanagala@gmail.com

DOI: 10.29303/jppipa.v11i7.11870

Abstract: This study proposes a deep learning-based classification system to identify chicken breeds from image data. A dataset of 2,400 labeled images representing twelve distinct chicken breeds was collected and divided into training, validation, and testing sets. The system employs transfer learning by integrating the Mobile VGG16 convolutional neural network as the feature extraction backbone. The extracted features were then passed through custom classification layers to differentiate among the breeds. The model was trained using 1,800 images, validated with 300 images, and evaluated on a separate test set of 300 images. During testing, the model achieved an accuracy of 81% and a categorical cross-entropy loss of 0.378. These results indicate that the model can effectively recognize subtle visual distinctions between similar-looking chicken breeds. The system demonstrates practical potential for applications in poultry farming, biodiversity documentation, and automated livestock management. The findings confirm that deep convolutional neural networks, particularly VGG16 in a transfer learning setup, are capable of performing fine-grained classification tasks in real-world scenarios. The proposed method provides a reliable and scalable solution for automatic chicken breed identification based on image input.

Keywords: Chicken; Convolutional neural network; Deep learning; Fine-tuning; VGG16

Introduction

Identifying chicken breeds plays a critical role in poultry farming, conservation, and trade. However, most laypeople struggle to accurately distinguish between breeds due to the physical similarities in body shape, feather color, and overall appearance. This lack of knowledge is often exploited in local markets, where buyers are misinformed about the breed and quality of chickens, leading to inappropriate care and potential economic losses (Menaga et al., 2025).

Indonesia is home to a wide variety of local and crossbred chicken breeds, each with unique characteristics and specific management requirements. Understanding the breed is essential, as each type of chicken requires tailored treatment in terms of feeding, housing, and healthcare. Unfortunately, such information is not widely accessible to the general public (Chuquimarca et al., 2024).

Traditional methods for identifying chicken breeds rely heavily on manual observation, which is subjective and often inaccurate. Therefore, there is a pressing need for a technological solution that can automate the identification process with high accuracy (Srinivasu et al., 2021).

This study proposes the use of Convolutional Neural Networks (CNN), specifically applying transfer learning with the Mobile VGG16 architecture, to

How to Cite:
kencana, N. W., Umar, R., & Murinto. Advanced Chicken Breed Identification Using Transfer Learning Techniques with the VGG16 Convolutional Neural Network Architecture. *Jurnal Penelitian Pendidikan IPA*, *11*(7), 152-159. https://doi.org/10.29303/jppipa.v11i7.11870

develop an automated chicken breed classification system based on full-body images (Brianorman & Utami, 2024). CNNs are chosen due to their proven effectiveness in pattern recognition and visual data processing, especially for complex image classification tasks (Imawati et al., 2025).

The novelty of this research lies in the application of transfer learning using Mobile VGG16 to identify a wide range of local and imported chicken breeds using complete body images, an area that has been minimally explored. Unlike previous studies that focused on classifying only a few breeds or specific body parts, this research involves a broader range of breeds and utilizes full-body image data for more accurate classification (Zaid et al., 2025).

With this approach, the study aims to contribute a practical, accurate, and accessible solution for automatic chicken breed identification, empowering the general public and improving efficiency in the poultry sector (Beltran et al., 2025).

Method

A general overview of the proposed chicken breed classification system using transfer learning is illustrated in Figure 1. The approach is designed to address the challenge of accurately identifying multiple chicken breeds from image data by leveraging the power of deep learning, particularly Convolutional Neural Networks (CNN). The methodology consists of a structured sequence of stages, including dataset preparation, model adaptation using pre-trained CNN architectures, fine-tuning, and performance evaluation. This systematic process aims to improve classification accuracy and ensure the model's robustness across varying visual conditions (Ramadhani et al., 2023). A detailed explanation of each stage is presented in the following sections (Al-Gaashani et al., 2025).

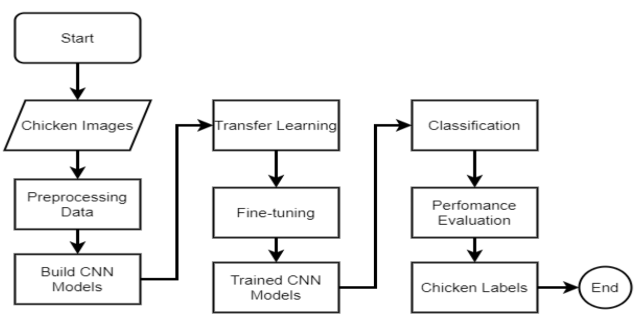

Figure 1. Research methodology

Dataset Description

The dataset used in this study comprises 2,400 images across 12 chicken breed classes, including Cemani, Kate, Pelung, Serama, Ketawa, Hutan, Mutiara, Turkey, Bangkok, Poland, Brahma, and Dong Tao (Hussain et al., 2024). The images were obtained from publicly available datasets, online sources, and local poultry farms, selected based on clarity, full-body visibility, and accurate labeling (Dihan et al., 2024). The data collection process involved a combination of manual photography under varied natural lighting conditions and web-sourced images under open licenses, ensuring diversity in background and appearance to enhance model generalization. An overview of sample images representing each breed is illustrated in Figure 2, highlighting the visual variety within the dataset. The dataset was split into 1,800 training images, 300 validation images, and 300 testing images to support consistent and balanced (Kurniawan et al., 2023).

Figure 2. Chicken dataset

Data Preprocessing

Data preprocessing is a crucial step in preparing datasets for training machine learning models, particularly in image classification tasks. It involves a series of techniques aimed at optimizing the dataset for model performance while preserving the integrity of the input data. Key procedures include resizing images to standard dimensions of 224 × 224 × 3 to ensure compatibility with model architectures, with padding applied as needed to maintain aspect ratios and avoid distortion (Adhinata et al., 2021). Data augmentation techniques were implemented to increase dataset variability and robustness, including random rotations within a range of ±25 degrees, horizontal flipping with a probability of 0.5, and scaling transformations within a range of 80 to 120% of the original image size (Garg et al., 2020). These augmentation strategies simulate diverse visual conditions and improve the model's ability to generalize to unseen data (Muttaqin & Sudiana, 2025).

CNN Models

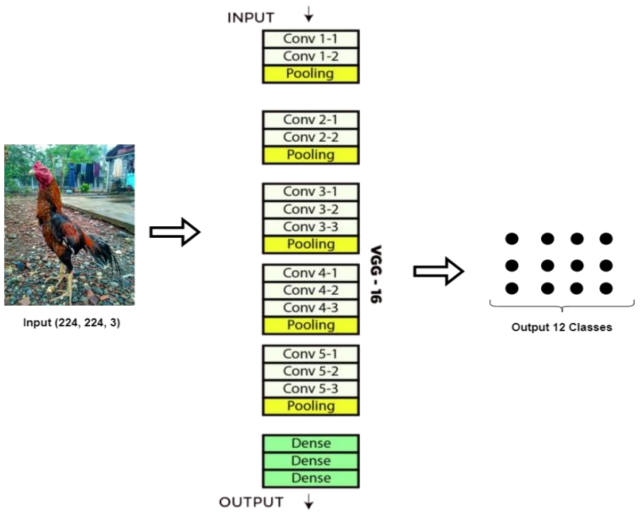

Figure 3. VGG16 models

Convolutional Neural Networks (CNNs) are widely used for processing image data due to their ability to extract meaningful spatial features. A typical CNN model consists of convolutional layers that apply filters to detect patterns from various regions of the input image, followed by pooling layers—such as max pooling or average pooling—that reduce dimensionality while preserving important features (Wibowo et al., 2020). These operations lower computational cost and help prevent overfitting. In this study, the VGG16 architecture was adopted through a transfer learning approach. The original fully connected layers of VGG16 were removed and replaced with a custom classifier tailored for chicken breed classification. Specifically, the modified architecture includes a flattening layer after the convolutional base, followed by two dense layers with 256 and 128 neurons respectively, both using ReLU activation (Chihaoui et al., 2024). The final output layer consists of 12 neurons, corresponding to the 12 chicken breed classes, and uses a softmax activation function to perform multi-class classification. This configuration allows the model to output a probability distribution across the target classes, facilitating accurate breed identification which can be seen in Figure 3.

Fine-Tuning and Transfer Learning

Transfer learning is a technique that enables a model trained on a large-scale source task to be repurposed for a related target task, thereby reducing training time and improving performance, particularly in domains with limited data availability. In this study, transfer learning was applied using the pre-trained VGG16 model, originally trained on the ImageNet dataset (Kurniawan & Utami, 2025). The convolutional base of the model was initially frozen to retain the general image features learned from the source task, while the newly added fully connected layers were trained on the target dataset. Subsequently, fine-tuning was conducted by unfreezing all layers and continuing training to adapt the feature representations to the specific characteristics of chicken breed images (Noor et al., 2023). The model was fine-tuned using the Adam optimizer, with an initial learning rate of 0.0001, which was reduced by a factor of 0.1 upon plateau in validation accuracy. A total of 150 epochs were used during training, incorporating early stopping to prevent overfitting. The categorical cross-entropy loss function was employed, as it is suitable for multi-class classification problems (Mirza et al., 2025). This two-stage training strategy enhances the model's ability to generalize while leveraging the robustness of the pre-trained network (Islami et al., 2024).

Tools and Materials

The implementation of this research required specific hardware and software, with the necessary specifications detailed in Table 1. These tools and software were essential for the study, providing the resources needed to carry out the research effectively. The specifications outlined in Table 1. ensured that all requirements for the research process were met, supporting the successful execution of the study (Putri et al., 2025).

Table 1. Tools

Device/Software	Operating System	Purpose
Lenovo Ideapad	Windows 10	To run the system
TensorFlow	Linux	Program development
Python 3.10.12	Windows 10	To run the program
Google Colab	Browser/Cloud	To run the program
Draw.io	Browser/Cloud	For design purposes

Evaluation Metrics

The performance of a multiclass classification model can be evaluated using several metrics, including accuracy, recall, precision, and F1-score (Zhong et al., 2023). To compute these metrics, a matrix known as the confusion matrix is required. The equations for accuracy, precision, recall, and F1-score are presented in Equations (1), (2), (3), and (4), respectively, where TP represents true positives, TN represents true negatives, FP represents false positives, and FN represents false negatives (Diharja et al., 2022).

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

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

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

$$\text{F1 Score} = \frac{\text{precission.recall}}{\text{precision} + \text{recall}} \quad (4)$$

Result and Discussion

This method was implemented using Python, supported by libraries such as TensorFlow, OpenCV, and Scikit-learn. The program was executed on Google Colab, a cloud-based platform that leverages a T4 GPU to accelerate training and processing (Ramadhani et al., 2023). By integrating these advanced tools and infrastructure, this approach ensures efficient performance and rapid development of the machine learning model (Zulirfan & Yennita, 2022).

Chicken Dataset

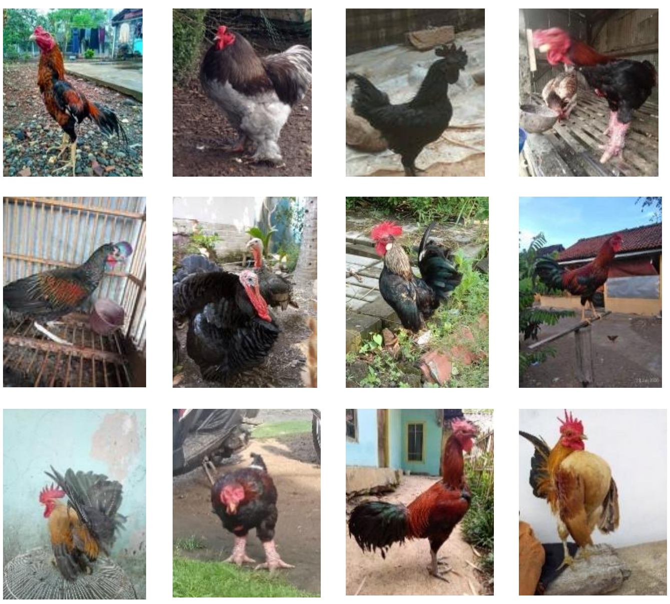

Figure 4. Chicken dataset

The dataset used in this study consists of 12 chicken breeds, including Cemani, Kate, Pelung, Serama, Ketawa, Hutan, Mutiara, Turkey, Bangkok, Poland, Brahma, and Dong Tao. The dataset contains a total of 1,800 images of these chicken breeds (Brianorman & Utami, 2024). These images were then divided into two main subsets: a training set containing 900 images and a validation set containing 300 images for each breed. This division ensures a balanced and robust dataset for training and evaluating the model, facilitating an accurate analysis of breed-specific features (Sari et al., 2025).

Preprocessing Data

Preprocessing was applied to prepare the dataset before the training process began. This step ensured that the data was in the optimal format and quality for the model to learn effectively (Murinto et al., 2023). Each stage of the preprocessing pipeline contributed to refining and organizing the dataset. The results of these preprocessing steps are outlined as follows, illustrating how raw data was transformed into clean and standardized input for the training phase.

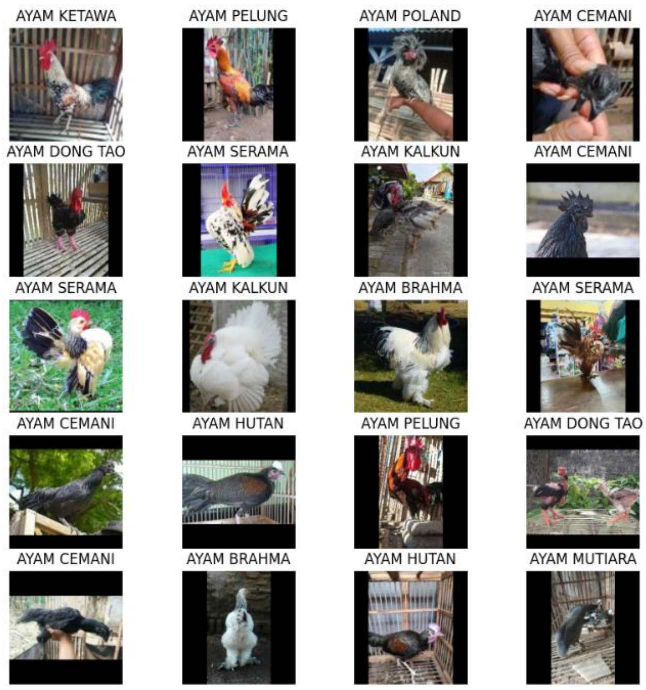

Figure 5. Data preprocessing

Padding was used to maintain the aspect ratio of the image by adding extra pixels around the edges of the original image. This ensures that the proportions of the image remain consistent even after resizing. The image was resized to the standard input dimensions of 224 x 224 x 3, with zero padding added to fill the borders while preserving the original aspect ratio

(Supriyanto et al., 2024). This method prevents distortion or stretching of the image and ensures that the input data aligns with the model's requirements for training and inference. The padded images provide a uniform format, allowing the model to process them more effectively while preserving important visual features from the original data.

Data augmentation is the process of applying various transformations to images in a dataset to increase its size and diversity, thereby improving model generalization during training. In this study, the augmentation techniques included 20-degree rotations and horizontal flips. Figure 5 illustrates the results, with the original image displayed on the left and the augmented images shown alongside it. Following augmentation, the total number of chicken images increased to 900, significantly enhancing the dataset's richness and effectiveness for training (Azhmy et al., 2024).

Figure 6. Data augmentation

To align with the pre-trained configuration of the VGG16 model, all pixel values in the input images were normalized to a range between [0, 1]. This standardization ensures that pixel intensities are appropriately scaled for optimal compatibility with the model's input requirements. By converting pixel values from their original range (typically [0, 255]) to this normalized range, the data becomes consistent with the training conditions under which the VGG16 model was initially developed. This process not only improves computational efficiency but also enhances the model's ability to interpret and process images effectively, contributing to better performance during both training and evaluation (Elsayed et al., 2023).

Transfer Learning

The training process employed the Adam optimizer with an initial learning rate of 0.001 and was conducted over 150 epochs (Ekmekyapar & Taşcı, 2023). The final model was selected based on the lowest validation loss observed during training. Using transfer learning with the VGG16 architecture, the model achieved a training accuracy of 36%, while validation and test accuracies both reached 77%. The training and validation performance curves are illustrated in Figure 7. Although the graph indicates signs of overfitting—evident from the performance gap between training and validation accuracy—kernel regularization techniques were applied within the fully connected layers to mitigate this issue. Despite these challenges, the VGG16-based model demonstrated consistent and reliable performance across all evaluation sets (Jee et al., 2023).

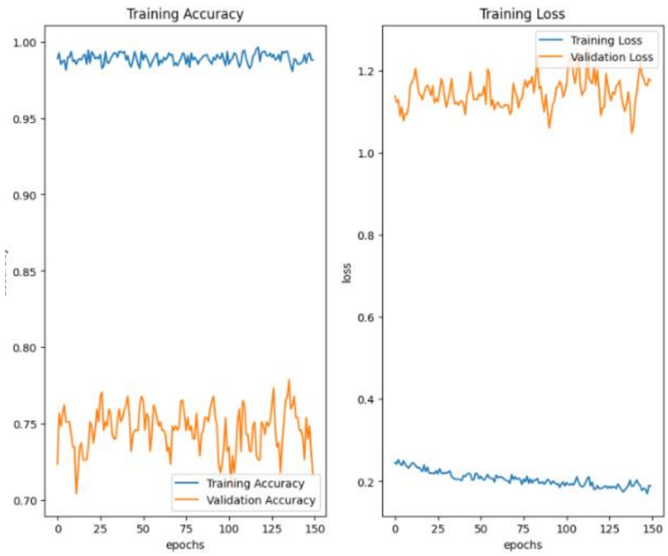

Figure 7. Transfer learning

Fine-Tuning

The training process employed the Adam optimizer with an initial learning rate of 0.001 and was conducted over 150 epochs. The final model was selected based on the lowest validation loss observed during training. In this study, transfer learning was applied using the pre-trained VGG16 architecture, which was originally trained on the ImageNet dataset (Kaya & Gürsoy, 2023). During the initial phase, the convolutional base of VGG16 was frozen, and only the newly added dense layers were trained on the chicken breed dataset. At this stage, the model achieved a validation and test accuracy of 77%, indicating that the generic features extracted from the frozen layers were relevant but not yet optimally adapted to the specific domain.

To further enhance performance, a fine-tuning stage was conducted by unfreezing the entire convolutional base and continuing training with a lower learning rate. This allowed the model to adjust the pre-trained weights more precisely to the visual characteristics of the chicken breed images. As a result of this refinement, the model's test accuracy increased to 81%, demonstrating a significant improvement in classification capability (Gao et al., 2023). The training and validation curves, shown in Figure 7, reflect this gain. Although slight overfitting was observed, it was mitigated using kernel regularization in the fully connected layers. Overall, the integration of transfer learning followed by fine-tuning proved to be a highly effective strategy, enabling the model to generalize better and achieve higher accuracy on unseen data (Hamim et al., 2023).

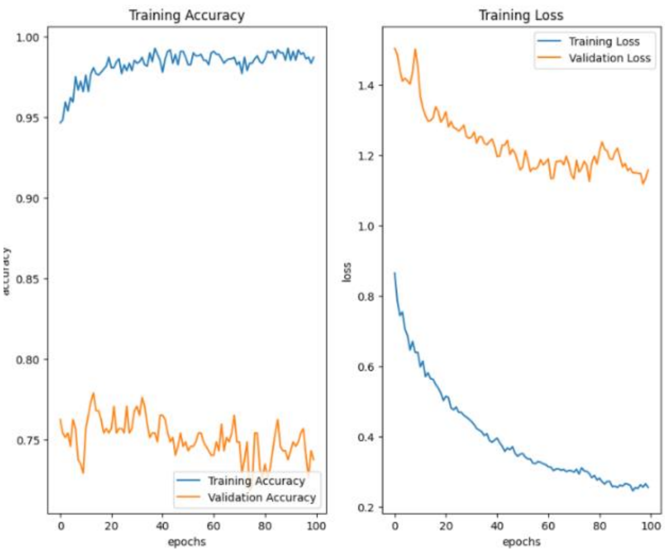

Figure 8. Fine-tuning graph

Performance Evaluation

The model was evaluated using a test dataset consisting of 1,800 images. A confusion matrix was used to assess the model's performance in classifying chicken breeds. As shown in Table 2, the VGG16 model performed the lowest in the Dong Tao chicken breed category, with an accuracy of 62%. This may be attributed to certain images having lower quality, which could negatively affect the test results (Ecemiş & O.Ilhan, 2022). Out of the 1,800 test images, the model accurately predicted 900 occurrences. The overall accuracy achieved by the VGG16 model was 81.55%.

Table 2. Performance evaluation

Chicken Breed	Data Count	Accuracy Result (%)
Dong Tao	150	62
Ketawa Chicken	150	73
Pelung	150	93
Brahma	150	76
Bangkok	150	83
Kate	150	83
Serama	150	83
Polish	150	83
Cemani	150	76
Jungle Fowl	150	63
Pelung	150	93
Turkey	150	80

Conclusion

The results demonstrate that the proposed approach effectively classifies chicken breeds from images by utilizing the VGG16 model with transfer learning and fine-tuning techniques. The dataset consists of 2,400 images across 12 classes, divided into 1,800 for training, 300 for validation, and 300 for testing. After applying fine-tuning, the VGG16 model achieved strong performance, with a precision of 77%, recall of 78%, an F1 score of 81%, and an overall accuracy of 81%. These findings highlight the robustness and effectiveness of the VGG16 architecture in handling fine-grained image classification tasks such as chicken breed identification. However, several limitations remain. The relatively small dataset may limit the model's generalizability to broader, more diverse real-world scenarios. Furthermore, model performance under varying lighting conditions and image quality was not thoroughly evaluated, which may affect reliability in uncontrolled environments. Despite these challenges, the approach shows strong potential for practical deployment, particularly in mobile applications that allow users to identify chicken breeds directly from captured images in real-time.

Acknowledgments

The authors would like to thank to all parties who helped in this research.

Author Contributions

Conceptualization, resources, data curation, and writing—preparation of the original draft, N.W.K. and R.U.; methodology, formal analysis, investigation, writing—review and editing and visualization, M. All authors have read and approved the published version of the manuscript.

Funding

This research received no external funding.

Conflicts of Interest

All authors declare that there is no conflict of interest.

References

Adhinata, F. D., Tanjung, N. A. F., Widayat, W., Pasfica, G. R., & Satura, F. R. (2021). Comparative Study of VGG16 and MobileNetV2 for Masked Face Recognition. *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, *7*(2), 230. https://doi.org/10.26555/jiteki.v7i2.20758

Al-Gaashani, M. S. A. M., Xu, W., & Obsie, E. Y. (2025). MobileNetV2-Based Deep Learning Architecture with Progressive Transfer Learning for Accurate Monkeypox Detection. *Applied Soft Computing*, *169*, 112553.

https://doi.org/10.1016/j.asoc.2024.112553

Azhmy, M. F., Suyar, A. S., & Setiamy, A. A. (2024). Digital HRM with Machine Learning Approach. *Jurnal Penelitian Pendidikan IPA*, *10*(SpecialIssue), 18–23. https://doi.org/10.29303/jppipa.v10iSpecialIssue.7213

Beltran, T. D., Villao, R. J., Chuquimarca, L. E., Vintimilla, B. X., & Velastin, S. A. (2025). Fruit Deformity Classification Through Single-Input and Multi-input Architectures Based on CNN Models Using Real and Synthetic Images. In Hernández-García, R., Barrientos, R. J., & Velastin, S.A. (eds) *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. CIARP 2024*. Lecture Notes in Computer Science, vol 15368. Springer, Cham. https://doi.org/10.1007/978-3-031-76607-7_4

Brianorman, Y., & Utami, D. (2024). Comparative Analysis of CNN Architectures for SIBI Image Classification. *JUITA: Jurnal Informatika*, *12*(1), 61. https://doi.org/10.30595/juita.v12i1.20608

Chihaoui, M., Dhibi, N., & Ferchichi, A. (2024). Optimization of Convolutional Neural Network and Visual Geometry Group-16 Using Genetic Algorithms for Pneumonia Detection. *Frontiers in Medicine*, *11*, 1498403. https://doi.org/10.3389/fmed.2024.1498403

Chuquimarca, L., Vintimilla, B., & Velastin, S. (2024). Classifying Healthy and Defective Fruits with a Multi-Input Architecture and CNN Models. *2024 14th International Conference on Pattern Recognition Systems (ICPRS)*, 1–7. https://doi.org/10.1109/ICPRS62101.2024.10677833

Dihan, F. J., Murad, S. A., Muzahid, A. J. M., Uddin, K. M. A., Alenazi, M. J. F., Bairagi, A. K., & Biswas, S. (2024). *MpoxSLDNet: A Novel CNN Model for Detecting Monkeypox Lesions and Performance Comparison with Pre-trained Models*. https://doi.org/10.48550/arXiv.1409.1556

Diharja, R., Fahlevi, M. R., Rahayu, E. S., & Handini, W. (2022). Prototype-Design of Soil Movement Detector Using IoT Hands-on Application. *Jurnal Penelitian Pendidikan IPA*, *8*(4), 2245–2254. https://doi.org/10.29303/jppipa.v8i4.1709

Ecemiş, İ. N., & O.Ilhan, H. (2022). Domates Yapraklarında Hastalık Tespiti Için Önerilen Hafif Evrişimli Sinir Ağı Ile Önceden Eğitilmiş Ağların Performans Karşılaştırması. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, *38*(2), 693–706. https://doi.org/10.17341/gazimmfd.1003730

Ekmekyapar, T., & Taşcı, B. (2023). Exemplar MobileNetV2-Based Artificial Intelligence for Robust and Accurate Diagnosis of Multiple Sclerosis. *Diagnostics (Basel, Switzerland)*, *13*(19). https://doi.org/10.3390/diagnostics13193030

Elsayed, O. R., ElKot, Y. G., ElRefaai, D. A., Abdelfattah, H. M., ElSayed, M., & Hamdy, A. (2023). Automated Identification and Classification of Teeth Fossils. *2023 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, 179–186. https://doi.org/10.1109/MIUCC58832.2023.10278368

Gao, W., Wang, D., & Huang, Y. (2023). Designing a Deep Learning-Driven Resource-Efficient Diagnostic System for Metastatic Breast Cancer: Reducing Long Delays of Clinical Diagnosis and Improving Patient Survival in Developing Countries. *Cancer Informatics*, *22*. https://doi.org/10.1177/11769351231214446

Garg, T., Garg, M., Mahela, O. P., & Garg, A. R. (2020). Convolutional Neural Networks with Transfer Learning for Recognition of COVID-19: A Comparative Study of Different Approaches. *AI*, *1*(4), 586–606. https://doi.org/10.3390/ai1040034

Hamim, M. A., Haque, A. U., Ray, K., & Hasan, M. S. (2023). Multiple Nail-Disease Classification Based on Machine Vision Using Transfer Learning Approach. *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 1–8. https://doi.org/10.1109/ICCCNT56998.2023.10306501

Hussain, M., Thaher, T., Almourad, M. B., & Mafarja, M. (2024). Optimizing VGG16 Deep Learning Model with Enhanced Hunger Games Search for Logo Classification. *Scientific Reports*, *14*(1), 31759. https://doi.org/10.1038/s41598-024-82022-5

Imawati, I. A. P. F., Sudarma, M., Putra, I. K. G. D., Bayupati, I. P. A., & Jo, M. (2025). Training VGG16, MobileNetV1 and Simple CNN Models from Scratch for Balinese Inscription Recognition. *Lontar Komputer: Jurnal Ilmiah Teknologi Informasi*, *15*(03), 149. https://doi.org/10.24843/LKJITI.2024.v15.i03.p01

Islami, F., Sumijan, S., & Defit, S. (2024). Customized Convolutional Neural Network for Glaucoma Detection in Retinal Fundus Images. *Jurnal Penelitian Pendidikan IPA*, *10*(8), 4606–4613. https://doi.org/10.29303/jppipa.v10i8.7614

Jee, G., Gm, H., Gourisaria, M. K., Singh, V., Rautaray, S. S., & Pandey, M. (2023). Efficacy Determination of Various Base Networks in Single Shot Detector for Automatic Mask Localisation in a Post COVID Setup. *Journal of Experimental & Theoretical Artificial Intelligence*, *35*(3), 345–364. https://doi.org/10.1080/0952813X.2021.1960638

Kaya, Y., & Gürsoy, E. (2023). Retracted Article: A MobileNet-Based CNN Model with a Novel Fine-Tuning Mechanism for COVID-19 Infection Detection. *Soft Computing*, *27*(9), 5521–5535. https://doi.org/10.1007/s00500-022-07798-y

Kurniawan, M. A., Wijaya, S. K., & Hanifa, N. R. (2023). Convolutional Neural Network for Earthquake Ground Motion Prediction Model in Earthquake Early Warning System in West Java. *Jurnal Penelitian Pendidikan IPA*, *9*(11), 1004–1010. https://doi.org/10.29303/jppipa.v9i11.3514

Kurniawan, M. B., & Utami, E. (2025). Performance Comparison of ResNet50, VGG16, and MobileNetV2 for Brain Tumor Classification on MRI Images. *SISTEMASI*, *14*(2), 767. https://doi.org/10.32520/stmsi.v14i2.5054

Menaga, D., Deborah, S. L., & Makizhna, M. (2025). *Deep Learning Models for Coral Reef Health Classification: MobileNetV2, VGG16 and ResNet* (pp. 1241–1251). https://doi.org/10.2991/978-94-6463-718-2_103

Mirza, A. H., Halim, R. M. N., & Hasbiallah, M. J. (2025). Development of an Efficient 1D-CNN Model for Myocardial Infarction Classification Using 12-Lead ECG Signals. *Jurnal Penelitian Pendidikan IPA*, *11*(3), 1167–1182. https://doi.org/10.29303/jppipa.v11i3.10238

Murinto, M., Rosyda, M., & Melany, M. (2023). Klasifikasi Jenis Biji Kopi Menggunakan Convolutional Neural Network dan Transfer Learning pada Model VGG16 dan MobileNetV2. *JRST (Jurnal Riset Sains dan Teknologi)*, *7*(2), 183. https://doi.org/10.30595/jrst.v7i2.16788

Muttaqin, R. Z., & Sudiana, D. (2025). Design of Realtime Web Application Firewall on Deep Learning-Based to Improve Web Application Security. *Jurnal Penelitian Pendidikan IPA*, *10*(12), 11121–11129. https://doi.org/10.29303/jppipa.v10i12.8346

Noor, A. Z. M., Gernowo, R., & Nurhayati, O. D. (2023). Data Augmentation for Hoax Detection Through the Method of Convolutional Neural Network in Indonesian News. *Jurnal Penelitian Pendidikan IPA*, *9*(7), 5078–5084. https://doi.org/10.29303/jppipa.v9i7.4214

Putri, S. A., Murinto, M., & Sunardi, S. (2025). Intelligent Monitoring of Smoking Prohibition in Public Spaces Using YOLOv8: Real-Time Detection and Telegram Notifications. *Jurnal Penelitian Pendidikan IPA*, *11*(4), 601–610. https://doi.org/10.29303/jppipa.v11i4.10519

Ramadhani, A. N. M., Saraswati, G. W., Agung, R. T., & Santoso, H. A. (2023). Performance Comparison of Convolutional Neural Network and MobileNetV2 for Chili Diseases Classification. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, *7*(4), 940–946. https://doi.org/10.29207/resti.v7i4.5028

Sari, M. W., Sitorus, S. P., Rohani, R., & Pane, R. (2025). Implementation of Convolutional Neural Network (CNN) Method in Determining the Level of Ripeness of Mango Fruit Based on Image. *Jurnal Penelitian Pendidikan IPA*, *11*(5), 419–428. https://doi.org/10.29303/jppipa.v11i5.11436

Srinivasu, P. N., SivaSai, J. G., Ijaz, M. F., Bhoi, A. K., Kim, W., & Kang, J. J. (2021). Classification of Skin Disease Using Deep Learning Neural Networks with MobileNet V2 and LSTM. *Sensors*, *21*(8), 2852. https://doi.org/10.3390/s21082852

Supriyanto, E., Isnanto, R. R., & Purnomo, S. H. (2024). Developing Chicken Health Classification Model Using a Convolutional Neural Network and Support Vector Machine (CNN-SVM) Approach. *Ingénierie Des Systèmes d Information*, *29*(6), 2525–2532. https://doi.org/10.18280/isi.290637

Wibowo, A., Hartanto, C. A., & Wirawan, P. W. (2020). Android Skin Cancer Detection and Classification Based on MobileNet v2 Model. *International Journal of Advances in Intelligent Informatics*, *6*(2), 135. https://doi.org/10.26555/ijain.v6i2.492

Zaid, M. M. A., Mohammed, A. A., & Sumari, P. (2025). Remote Sensing Image Classification Using Convolutional Neural Network (CNN) and Transfer Learning Techniques. *Journal of Computer Science*, *21*(3), 635–645. https://doi.org/10.3844/jcssp.2025.635.645

Zhong, J., Chen, J., & Mian, A. (2023). DualConv: Dual Convolutional Kernels for Lightweight Deep Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, *34*(11), 9528–9535. https://doi.org/10.1109/TNNLS.2022.3151138

Zulirfan, Z., & Yennita, Y. (2022). Feasibility Test of STEM at Home Prototype Kit as Science Project-Based Learning Media for Junior High School Students. *Jurnal Penelitian Pendidikan IPA*, *8*(1), 57–66. https://doi.org/10.29303/jppipa.v8i1.1122