Institute of Research and Publication Indonesia (IRPI)
Public Research Journal of Engineering, Data Technology and Computer Science
Journal Homepage: https://journal.irpi.or.id/index.php/predatecs
Vol. 2 Iss. 2 January 2025, pp: 127-137
ISSN(P): 3024-921X | ISSN(E): 3024-8043

Convolutional Neural Networks Using EfficientNetB0 Architecture and Hyperparameters on Skin Disease Classification

Putri Khairunnisa[1*], Wahyu Eka Putra[2], Wu Yitong[3], Abni Jufrizal[4], Muhammad Nur Aflah Makmum[5]

[1,2] Department of Information System, Faculty of Science and Technology,
Universitas Islam Negeri Sultan Syarif Kasim Riau, Indonesia
[3]Faculty of Law, Shantou University, China
[4]Department of Geography, Faculty of Humanities and Social Sciences,
Süleyman Demirel University, Turkey
[5]Department of International Relations, Faculty of Economics and Administrative Sciences,
Süleyman Demirel University, Turkey

E-Mail: [1]12150323489@students.uin-suska.ac.id, [2]wahyueka033@gmail.com,
[3]wuyitongus@gmail.com, [4]jufrizalabni@gmail.com, [5]aflahmakmur8@gmail.com

Received Jul 20th 2024; Revised Jan 5th 2025; Accepted Jan 9th 2025; Available Online Jan 19th 2025, Published Jan 24th 2025
Corresponding Author: Putri Khairunnisa

Abstract

Skin diseases are often caused by air temperature, environmental hygiene and personal hygiene, with symptoms such as itching, pain and redness. Contributing factors include exposure to chemicals, sunlight, infections, a weak immune system, microorganisms, and poor personal hygiene. This study uses Convolutional Neural Networks (CNN) with EfficientNetB0 model and hyperparameter optimization for skin disease classification. The dataset consists of 1158 images that have been divided into eight categories, with 80% for training data and 20% for test data. Data augmentation is applied to increase the variety of training data. Various combinations of hyperparameters such as learning rate, optimizer (Adamax and AdamW), and activation function (ReLU and LeakyReLU) were tested in 16 training scenarios. The best results was obtained from the third scenario with the original dataset, Adamax optimizer, ReLU activation function, and 0.01 learning rate, which gave a testing accuracy of 95.70%. The model also showed good generalization and low loss values. Confusion matrix analysis and classification report showed high accuracy in skin disease classification. This study concludes that EfficientNetB0 with proper hyperparameter optimization can improve the accuracy and effectiveness of skin disease diagnosis.

Keyword: EfficientNetB0, Hyperparameter, Optimization, Skin Disease

1. INTRODUCTION

Health is an important aspect of human life. Maintaining a healthy lifestyle is key to preventing disease. Skin diseases, which affect about a third of the world's population, are often a significant obstacle for individuals. These conditions can reduce quality of life, affecting psychological health, social interaction, and daily activities [1]. Skin is an organ of the human body that is located on the surface and plays a role in receiving external stimuli, such as touch, pain, and various other influences [2]. Neglecting skin health can lead to various skin diseases, so it is important to take care of the skin early on to keep it healthy and avoid disease. The skin also plays a role in protecting the body from infection, regulating body temperature, and preventing fluid loss. Therefore, maintaining healthy skin is not only important to avoid skin diseases, but also to ensure the body's overall protective function and balance [3].

Skin diseases are conditions that affect the outside of the body with symptoms such as itching, pain, numbness and redness. Causes can include exposure to chemicals, sunlight, viral infections, air temperature, a weak immune system, microorganisms, microbes, fungi, and personal hygiene factors. Skin diseases can cause changes in the texture or color of the skin. In general, these conditions can be chronic, contagious, and in some cases can develop into skin cancer [4]. Some skin diseases display symptoms that require great effort for treatment. This is because the disease can progress for months before it is finally diagnosed [5]. The prevalence of infectious skin diseases according to the World Health Organization (WHO) worldwide is reported to be around 300 million cases per year. The prevalence of skin diseases in Indonesia is 4.60%-12.95%, ranking third out of the top 10 diseases [6].

There are various factors that contribute to the development of skin diseases, both long-term and short-term. Direct causes that the skin is exposed to include chemicals used in daily activities such as detergents, soaps, shampoos, toothpastes, as well as frequently used metals such as in watches. These materials can damage the skin or trigger allergic reactions [7]. In addition, there are several factors that cannot be ignored for a long time, such as age, gender, level of knowledge, history of skin and allergies, personal hygiene, community behavior, and use of river water [8]. Skin diseases do not recognize certain age restrictions that are more vulnerable. Everyone, from infants to the elderly, is at risk of developing skin diseases. Some types of skin diseases, if left untreated, can lead to complications of other diseases [9]. Some types of skin diseases can cause various complications. For example, chickenpox can cause complications such as diarrhea, pneumonia, malnutrition, middle ear infections, mouth ulcers, and eye problems. Shingles can cause complications such as neuralgia, skin infections, eye problems and muscle weakness. Leprosy can cause damage to the skin, nerves, limbs and eyes. Meanwhile, eczema or dermatitis can cause ulcers that can spread to uninfected skin areas [10].

Digital image processing technology is very useful for overcoming various image analysis problems. With the application of image processing, identification and classification of image-based data can be done more efficiently. One method that is very effective in this case is the Convolutional Neural Network (CNN) [11]. Technological advances in the field of artificial intelligence and deep learning, especially Convolutional Neural Network (CNN), open up new opportunities to improve accuracy and efficiency in the diagnosis of skin diseases [12]. CNN, which is a type of artificial neural network specifically designed to process and analyze image data, is becoming a very effective tool for image classification tasks, including skin disease detection and classification [13]. The research results provide clear evidence that CNNs are capable of detecting skin diseases with high accuracy from various angles.

Previous research conducted by Jenan A. Alhijaj and Raidah S. Khudeyer (2023) with the title "Integration of EfficientNetB0 and Machine Learning for Fingerprint Classification". This research uses the CNN method with the EfficientNet-B0 architecture. This research applies two testing methods. The first method involves feature extraction and fingerprint gender classification using EfficientNet B0, while the second method uses PCA and RF after feature extraction. The test results show that the second proposed method is superior to the first method in terms of precision and accuracy, with scores reaching 99% [14]. Another study conducted by R. Indumathi and friends (2024) with the title "Predicting Heart Attack from Retinal Fundus Image Classification Using CNN with EfficientNet B0". The results of this study show that the model is able to provide fairly accurate predictions related to the risk of heart attack based on eye fundus images, with an accuracy of 96.12%. This shows the potential of the model in clinical applications for more effective cardiovascular risk assessment [15]. In addition, other research conducted by R. Angeline and A. Alice Nithya (2024) with the title "Deep Human Facial Emotion Recognition: A Transfer Learning Approach Using EfficientNet B0 Model". This research proposes the use of the Convolutional Neural Network method by applying the EfficientNetB0 model. The model is adjusted and validated after performing Test Time Augmentation (TTA). The test results showed that the EfficientNetB0 model achieved a training accuracy of 87%. [16].

Previous research has shown various approaches in optimizing hyperparameters to improve model performance. One study implemented a genetic algorithm to optimize hyperparameters in a deep learning-based melanoma automated diagnosis system. Hyperparameters tested include random seed, solver type, base learning rate, number of epochs, and batch size. The highest validation accuracy result was 89.75%, superior to other methods such as brute force and Bayesian optimization [17]. Another study combined hyperparameter optimization (HPO) with SMOTE and Extra Trees techniques in heart disease detection, which achieved 99.2% and 98.52% accuracy, and 95.73% in severity classification [18]. In addition, another study proposed a deep learning-based network intrusion detection system integrating pre-training using a Deep Autoencoder (DAE), which provided the best results in intrusion detection and defeated previous approaches, with hyperparameter optimization via grid search and random search [19].

Based on previous research, the EfficientNetB0 architecture was shown to be more efficient in producing high accuracy with a smaller number of parameters compared to other CNN architectures, so EfficientNetB0 was chosen as the model in this study. The computational efficiency offered by this model allows its application to image classification tasks that require large data processing without sacrificing performance. In addition, hyperparameter optimization plays a very important role in improving the performance of deep learning models. The use of proper hyperparameter optimization techniques, as has been proven in previous studies, can significantly improve model accuracy, reduce overfitting, and ensure the model can adapt well to complex data. Therefore, hyperparameter optimization is also applied in this study to obtain a model with optimal performance in skin disease image classification.

This study aims to measure the performance of skin disease classification models using EfficientNetB0 architecture and optimize hyperparameters to get optimal results. Unlike previous studies that discuss similar topics, this research specifically tests hyperparameters to find the optimal model combination. Previous studies have not tested the effect of hyperparameters on model performance, so the best accuracy on various hyperparameter combinations is unknown. Through this approach, this research is expected to improve the accuracy of skin disease detection by considering computational efficiency and the ability of the model to handle complex data, as well as improving performance through the right hyperparameter combination.

2. MATERIAL AND METHOD

The research stages are an integral part of the methodology that is systematically designed to achieve the set objectives. Each stage plays an important role in developing a research framework, designing appropriate methods, collecting and analyzing data, and drawing relevant conclusions. The research stages that will be applied in this study include the following steps.

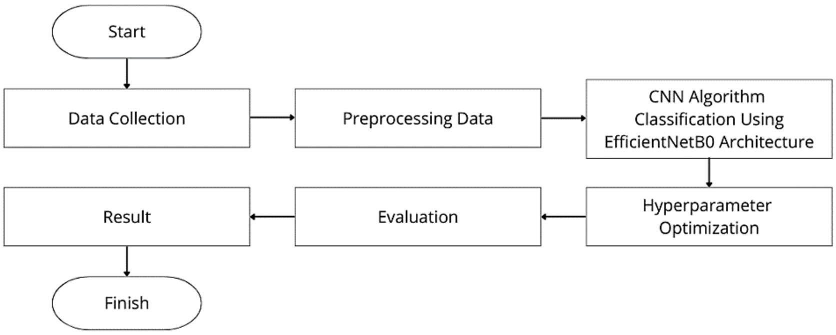

Figure 1. Research Methodology

2.1. Data Collection

Data Collection The dataset used in this study consists of 1158 skin disease images from the Kaggle database. This image data has been divided into 80% training data and 20% testing data with 8 (eight) categories, namely Bacterial Infection (cellulitis), Bacterial Infection (impetigo), Fungal Infection (athlete's foot), Fungal Infection (nail fungus), Fungal Infection (ringworm), Parasitic Infection (skin larva migrans), Viral Skin Infection (chickenpox), Viral Skin Infection (shingles). Figure 2 in this study provides a visual depiction of the variations and differences in characteristics between diseases by displaying several photo examples of each of the different disease categories.

Figure 2. Data Visualization. (a) Bacterial Infection (cellulitis), (b) Bacterial Infection (impetigo), (c) Fungal Infection (athlete's foot), (d) Fungal Infection (nail fungus), (e) Fungal Infection (ringworm), (f) Parasitic Infection (skin larva migrans), (g) Viral skin infection (chickenpox), (h) Viral skin infection (shingles).

2.2. Preprocessing

Preprocessing aims to homogenize skin disease images so that the information extraction process becomes more accurate. At this stage, data augmentation is performed. Augmentation is the process of modifying an image in such a way that the computer still detects the modified image as the same image [20]. Data augmentation techniques use data alteration and oversampling to increase the number of images in the dataset. This technique can minimize the problem of overfitting in the resulting image [21]. There are several ways to apply various augmentation methods such as rotation, rescaling, shear, zooming, horizontal flipping, and fill mode using principal component analysis [22]. By applying these operations, data augmentation can effectively enrich the training dataset, allowing the model to learn from a wider variety and reducing the possibility of overfitting.

2.3. Convolutional Neural Network Model

Convolutional Neural Networks (CNN) is a type of artificial neural network that works similar to the working process of the human brain [23]. The main principle behind CNN is its ability to extract local features from inputs at higher layers and send them to lower layers to form more complex features [24]. CNN is a form of Multilayer Perceptron (MLP) specifically designed to process two-dimensional information. This type of neural network belongs to the complex category because it has many layers and is often used for image data processing. CNNs usually consist of input layers, convolution layers, pooling, and fully connected layers [25]. CNN learns the input image by first calculating the weights and then performing clustering [26]. Therefore, CNNs have been used in various applications such as segmentation, pattern recognition, detection, and classification [27]. The basic structure of CNN includes convolution, pooling, activation function, and fully connected layers [28].

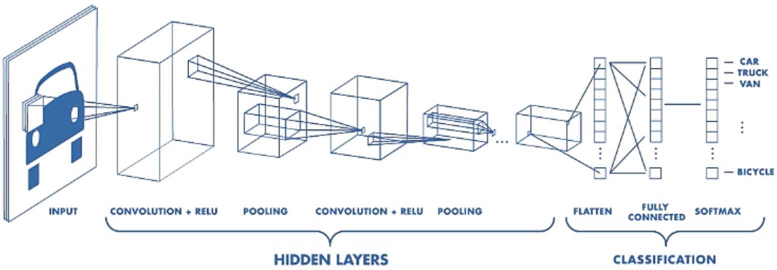

Figure 3. CNN Architecture

Figure 3 shows the elements that make up a CNN. To understand the development of CNN architecture, it is important to recognize the various parts of a CNN and how they are implemented [29]. The output of the convolution layer is known as the feature map, which describes the unique characteristics of the image. The convolution layer consists of filters that perform convolution on the input image matrix. The main purpose of using pooling layers is to reduce the complexity of the model by reducing the number of parameters required for computation [30]. Highly complex models are prone to overfitting conditions. This parameter reduction is done by combining several adjacent pixels into a single value. The most commonly used pooling types are max pooling and average pooling.

2.4. EfficientNetB0

EfficientNet was first introduced by Tan and Le in their research. They mentioned that EfficientNet is one of the most efficient models and is able to achieve the highest accuracy in ImageNet and transfer learning for image classification. EfficientNet is a CNN architecture designed with regular scaling on three main components: depth, area, and resolution. The addition of the three components is done systematically to reduce the number of parameters, so the process is faster and the accuracy is higher than previous models [31]. EfficientNet-B0 is a basic model in the EfficientNet series that has a small number of parameters and moderate computational cost. Despite its simplicity, it remains highly reliable and is a good initial choice for various image classification tasks [32]. EfficientNetB0 is designed with an architecture consisting of iterative blocks that implement Depthwise Separable Convolutions, batch normalization, as well as a non-linear Swish activation function to improve efficiency and performance [33].

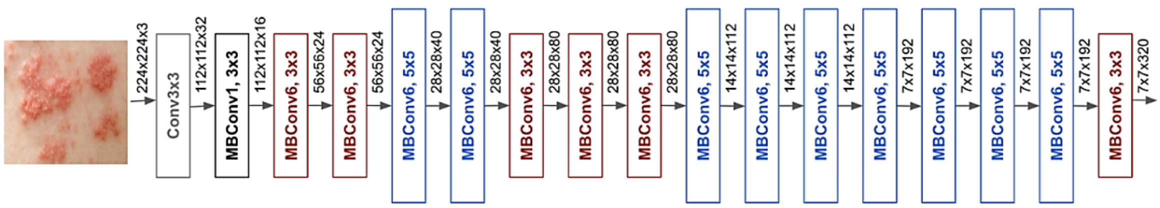

Figure 4. EfficientNetB0 Architecture

2.5. Hyperparameter Optimization

Hyperparameter optimization is the process of determining how many function evaluations need to be performed for each optimization to select the best hyperparameters in a model. Although complex and time-consuming, this optimization has the potential to significantly improve the model [34]. Hyperparameter optimization consists of four main elements that complement each other. First, there is an estimator, either a regressor or a classifier, designed to achieve one or more objective functions. Second, there is a search space that contains various possible hyperparameter values as an exploration arena. Thirdly, there is an optimization method that acts as a strategy to find the most optimal combination of hyperparameters. Finally, there is the evaluation function that compares the effectiveness of each configuration to ascertain the best performance [35]. Some of the hyperparameters that can be used include learning rate, optimizer, and activation function [36].

2.6. Evaluation Model

Model Evaluation Evaluation is the stage of checking the accuracy of experimental results, including scenario testing and hyperparameter optimization. The experimental results will be analyzed to draw conclusions as a basis for determining the hyperparameter combination in the next experiment to find the combination with the best performance. In this research, the evaluation is done using confusion matrix. Confusion matrix is an evaluation matrix that compares the classification results with the true value [37]. The model is evaluated using accuracy, precision, recall, and F1-score metrics. The following is the equation:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FN + FP} \tag{1}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{2}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{3}$$

$$\text{F1-Score} = \frac{2 \times P \times R}{P + R} \tag{4}$$

3. RESULTS AND DISCUSSION

3.1. Preprocessing

Data preprocessing is done by performing data augmentation by randomly rotating up to 45 degrees, randomly distorting the image with a sliding angle value of 15%, randomly zooming the image up to 15%, flipping the image horizontally and vertically. Data augmentation is performed on training data, while validation data and test data are not augmented. Figure 5 is the result of the augmentation of skin disease image data.

Figure 5. Augmentation Result

3.2. Hyperparameter Optimization

This research will implement various combinations of hyperparameters to evaluate the performance of the model. Some hyperparameters that will be applied include learning rate, optimizer, and activation function. The learning rates used are 0.01 and 0.1. The optimizer to be used consists of two types, namely Adamax and AdamW, and the activation function uses ReLU and LeakyReLU. The combination of various learning rate values, optimizers, and activation functions in this study aims to get the best combination that produces optimal performance of the model used.

3.3. CNN Model with EfficientNetB0 Architecture

Data training is performed using a CNN model with the EfficientNetB0 architecture. This model adds an AveragePooling2D layer with a kernel size of 7x7, followed by a BatchNormalization layer to stabilize and speed up the training process. A Dropout layer (50%) is used to prevent overfitting. The data is then flattened using Flatten to produce a one-dimensional vector. A Dense layer with 256 units and he_uniform initialization is added after the second BatchNormalization. An additional BatchNormalization is reapplied to maintain training stability, followed by an activation layer (ReLU or LeakyReLU). A second dropout with a 50% dropout rate is added, followed by a Dense layer with 8 units and a softmax activation function. This combination is designed to effectively process and classify images, as well as prevent overfitting and improve training stability and speed.

The training process was conducted for 16 scenarios to obtain comprehensive results. The types of datasets used are the original dataset and the dataset that has been augmented with images. The training scenarios are based on the type of dataset and hyperparameters such as optimizer, activity function, learning rate, and 25 epochs. The results of the scenarios that have been carried out can be seen in Table 1.

Table 1. Training Scenario Results

No.	Dataset	Optimizer	Activation Functions	Learning Rate	Accuracy
1	Original	Adamax	LeakyReLU	0,01	0,957
2				0,10	0,944
3			ReLU	0,01	0,957
4				0,10	0,948
5		AdamW	LeakyReLU	0,01	0,961
6				0,10	0,922
7			ReLU	0,01	0,957
8				0,10	0,871
9	Augmentasi	Adamax	LeakyReLU	0,01	0,957
10				0,10	0,931
11			ReLU	0,01	0,957
12				0,10	0,935
13		AdamW	LeakyReLU	0,01	0,944
14				0,10	0,824
15			ReLU	0,01	0,944
16				0,10	0,858

Based on Table 1, the test results with the highest accuracy are in the fifth scenario with the original dataset type, AdamW optimizer, LeakyReLU activity function with a learning rate of 0.01, which is 0.961. If analyzed more deeply, it is necessary to compare the model with other scenarios in terms of accuracy and loss values on the train, validation, and testing datasets. The accuracy and loss value models of some of the best scenarios with the highest accuracy can be seen in Table 2 and their visualizations in Figure 6, Figure 7, Figure 8, Figure 9, Figure 10, Figure 11.

Table 2. Modeled Accuracy & Loss Values of Some of the Best Scenarios

No.	Scenario	Model	Train	Validation	Testing
1	Scenario 1	Accuracy	100%	95,62%	95,70%
		Loss	0.004921	0.163553	0.119098
2	Scenario 3	Accuracy	100%	96,17%	95,70%
		Loss	0.003685	0.131901	0.127379
3	Scenario 5	Accuracy	99,59%	93,44%	96,13%
		Loss	0.013562	0.200492	0.109216
4	Scenario 7	Accuracy	100%	94,53%	95,70%
		Loss	0.002112	0.226405	0.171253
5	Scenario 9	Accuracy	98,65%	94,53%	95,70%
		Loss	0.053661	0.175659	0.151354
6	Scenario 11	Accuracy	99,73%	91,80%	95,70%
		Loss	0.042396	0.177617	0.140359

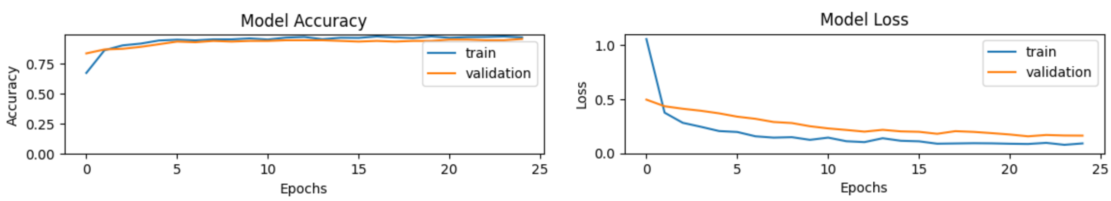

Figure 6. Model Visualization of the first Scenario

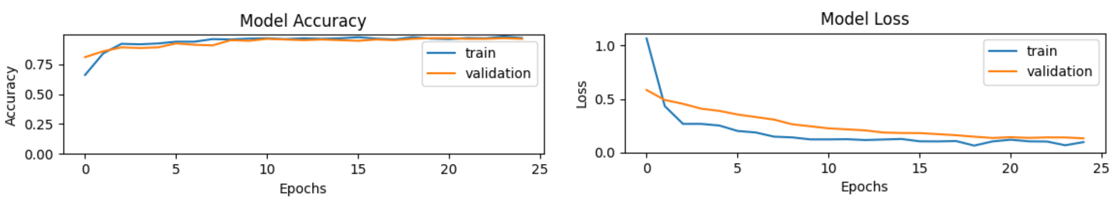

Figure 7. Model Visualization of the third Scenario

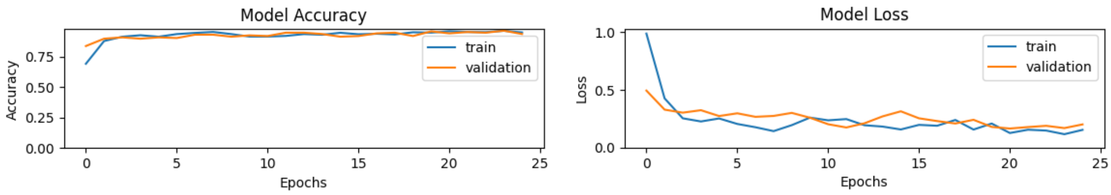

Figure 8. Model Visualization of the fifth Scenario

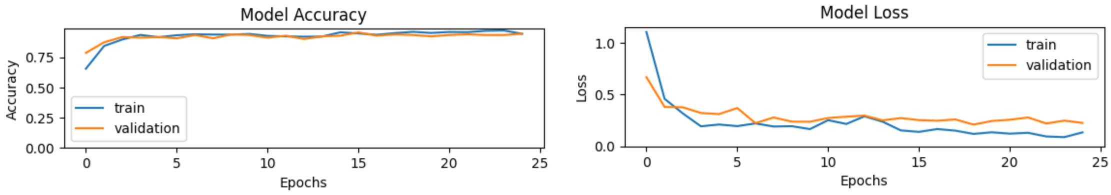

Figure 9. Model Visualization of the Seventh Scenario

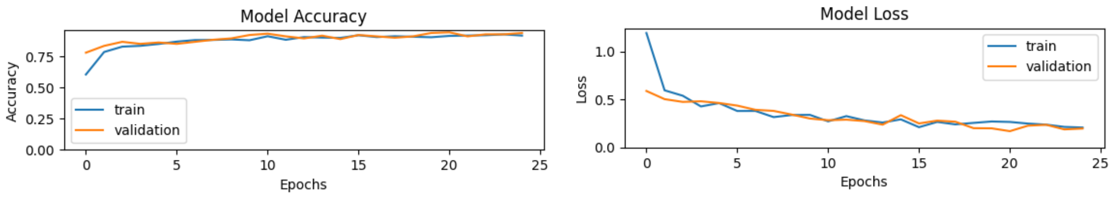

Figure 10. Model Visualization of the Ninth Scenario

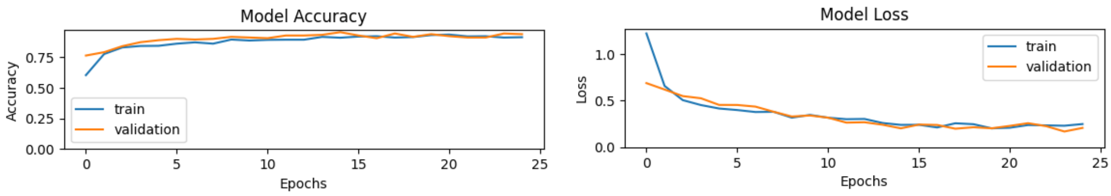

Figure 11. Model Visualization of the Eleventh Scenario

The model visualization graphs for the various scenarios show the changes in accuracy and loss over the 25 training epochs. Some scenarios show indications of overfitting, where the model performs very well on training data but less well on validation data. In the first scenario, the model achieved perfect training accuracy (100%), but lower validation accuracy (95.62%) with a significant difference in loss between the training and validation data. The seventh, ninth, and eleventh scenarios follow the same pattern with high training accuracy, but lower validation accuracy and large loss differences.

Although the fifth scenario has the highest testing accuracy of 96.13%, the difference between accuracy and loss in the training and validation data indicates potential overfitting. The model in the fifth scenario showed a training accuracy of 99.6% and a validation accuracy of 93.4%. The loss value on the training data is very low at 0.013562, while the loss on the validation data is higher at 0.200492. This 6.2% difference in accuracy and significant difference in loss suggests that the model may be too focused on the training data and not generalizing enough to new data.

Among all scenarios, the third scenario is the best model. The model in the third scenario showed excellent performance with a training accuracy value of 100%, validation accuracy of 96.17%, and testing accuracy of 95.70%. These values show that the model has an excellent ability to learn the training data and is able to maintain high performance on validation and testing data. The small difference in accuracy between the training data and validation data (about 3.83%) indicates that the model does not experience significant overfitting. Low loss values in training (0.003685), validation (0.131901), and testing (0.127379) data indicate that the model is able to minimize errors well on all datasets. Visualization of the model in the third scenario shows an increasingly convergent and stable curve between training and validation, as can be seen in Figure 7.

The best model generated from the third scenario training process has an accuracy of 95.70%. The classification report results from the third scenario in Table 3 show that the precision and recall values for most classes are very high, with some classes achieving a perfect score (1.00). The high f1-score values, consistent with the confusion matrix results, indicate the excellent performance of the model in skin disease classification.

The overall accuracy of the model was 96%, with the macro average and weighted average for precision, recall, and f1-score showing consistent and strong performance across all classes.

Table 3. Classification Report of the Third Scenario

	Precision	Recall	F1-Score	Support
Cellulitis	0,97	0,91	0,94	33
Impetigo	1,00	1,00	1,00	20
Athlete Foot	0,94	1,00	0,97	32
Nail Fungus	1,00	1,00	1,00	33
Ringworm	1,00	0,83	0,90	23
Larva Migrans	1,00	0,88	0,94	25
Chickenpox	1,00	1,00	1,00	34
Shingles	0,82	1,00	0,90	33
Accuracy			0,96	233
Macro Avg	0,97	0,95	0,96	233
Weighted Avg	0,96	0,96	0,96	233

Based on the confusion matrix calculation results, the model generated from the third scenario shows excellent performance in classifying most classes with high true positive values and minimal misclassification. In the Cellulitis class, the model successfully classified 30 test data correctly, but there were 3 misclassified test data, namely 1 image considered as athlete's foot and 2 images considered as shingles. The Ringworm class showed 19 test data correctly classified, but there were 4 misclassified test data, namely 1 image considered ringworm and 3 images considered shingles. The larva migrans class shows 22 correctly classified test data and there are few errors, namely 1 image is considered water fleas and 2 images are considered shingles. As for the impetigo, athlete's foot, nail fungus, chickenpox, and shingles classes, the model did not misclassify at all, producing 20, 32, 33, 34, and 33 correctly classified test data respectively. The results show that the model has a high level of accuracy with most classes having a dominant number of positives and minimal errors.

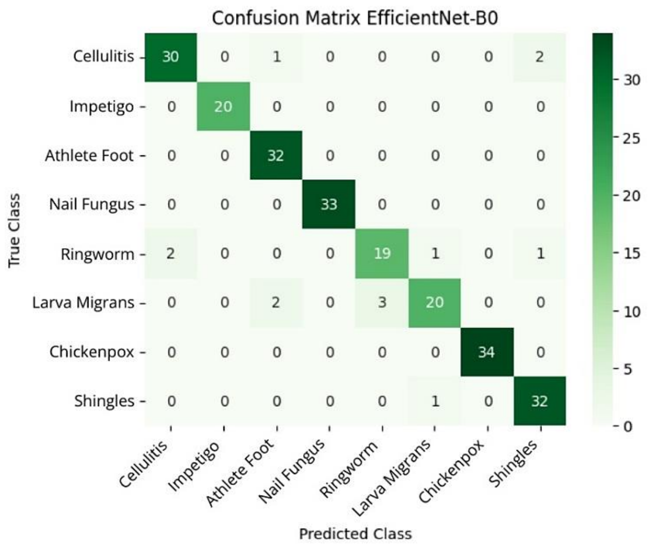

Figure 12. Confusion Matrix of the Third Scenario

The results showed that the model in the 3rd scenario gave the best performance in skin disease classification with a testing accuracy of 95.70%. This model outperformed models in other scenarios, including the 5th scenario which achieved the highest testing accuracy of 96.13% but experienced more significant overfitting. The performance of the 3rd scenario is supported by the low loss values in the training, validation, and testing data, as well as the small accuracy difference between the training and validation data, which indicates the good generalization ability of the model. These findings are in line with previous research using deep learning for skin disease classification, such as in a study on skin cancer classification using InceptionV3 as modeling, obtaining a diagnosis accuracy of 86.90% [38]. Another study using a similar model obtained an accuracy of 85.8% [39]. Compared to these studies, the EfficientNetB0 model used in this study shows a significant increase in accuracy, especially in terms of the ability to classify more complex datasets.

The superior performance of the model in the 3rd scenario can be attributed to the optimal combination of hyperparameters, such as the use of the Adamax optimizer, ReLU activation function, and a learning rate of 0.01. This supports the finding that proper hyperparameter selection can significantly improve model performance in medical image classification. Previous research on Hyperparameter Optimization for Automated Melanoma Diagnosis System also shows that proper selection of hyperparameters, such as learning rate and optimizer, can have a great impact on model accuracy [17]. In that study, experiments with various learning rate and optimizer values successfully improved the accuracy of the model in melanoma classification. The same is true in this study, where the right combination of hyperparameters can help achieve better performance in the model used.

Overall, this study shows that the EfficientNetB0 model with optimal hyperparameter combinations is able to provide superior performance in skin disease classification. Compared to previous studies, this model shows significant improvement in terms of accuracy and ability to handle complex datasets. Moreover, these findings confirm the importance of proper hyperparameter selection, as supported by related research [17]. Although challenges such as misclassification in certain classes still exist, these results provide a solid foundation for further development in medical classification, including the exploration of additional optimization techniques and improvement of data quality.

4. CONCLUSION

In this study, of the 16 training scenarios tested, the model in the third scenario with a combination of hyperparameters using the original dataset, Adamax optimizer, ReLU activation function, and learning rate 0.01 showed the best performance with a testing accuracy of 95.70%. Although the fifth scenario achieved the highest accuracy of 96.13%, the model experienced more significant overfitting than the third scenario. The model in the third scenario successfully minimized errors and showed good generalization with small accuracy differences between training and validation data and low loss values. Visualization and convergence curve analysis also reinforced that the model has high stability and performance. The results of the confusion matrix and classification report show that the model is able to classify most of the classes correctly and makes only a few errors. Overall, this study successfully demonstrated that the implementation of EfficientNetB0 with proper hyperparameter optimization can significantly improve accuracy and precision in skin disease classification, making a meaningful contribution to more effective and accurate skin disease diagnosis.

REFERENCES

[1] M. A. M. Almeida and I. A. X. Santos, “Classification models for skin tumor detection using texture analysis in medical images,” J. Imaging, vol. 6, no. 6, pp. 1–15, 2020, doi: 10.3390/JIMAGING6060051.

[2] P. S. Mustafa, “Implikasi Pola Kerja Telensefalon dan Korteks Cerebral dalam Pendidikan Jasmani,” Media Ilmu Keolahragaan Indones., vol. 10, no. 2, pp. 53–62, 2020, doi: 10.15294/miki.v10i2.24901.

[3] N. Andrini, “Karakteristik Dan Perawatan Kulit Untuk Orang Asia,” J. Pandu Husada, vol. 4, no. 3, pp. 14–23, 2023, doi: https://doi.org/10.30596/jph.v4i3.16621.g10091.

[4] T. Srisantyorini and N. F. Cahyaningsih, “Analisis Kejadian Penyakit Kulit pada Pemulung di Tempat Pengolahan Sampah Terpadu (TPST) Kelurahan Sumur Batu Kecamatan Bantar Gebang Kota Bekasi,” J. Kedokt. dan Kesehat., vol. 15, no. 2, p. 135, 2019, doi: 10.24853/jkk.15.2.135-147.

[5] P. N. Srinivasu, J. G. Sivasai, M. F. Ijaz, A. K. Bhoi, W. Kim, and J. J. Kang, “Classification of Skin Disease Using Deep Learning Neural Networks with MobileNet V2 and LSTM,” Sensors, vol. 21, no. 8, pp. 1–27, 2021, doi: 10.3390/s21082852.

[6] N. S. Rahayu, A. D. Puteri, and L. M. A. Isnaeni, “Hubungan Perilaku Masyarakat Dan Penggunaan Air Sungai Dengan Gangguan Penyakit Kulit Di Desa Kampung Pinang Wilayah Kerja Puskesmas Pantai Raja,” J. Imliah Ilmu Kesehat., vol. 1, no. 3, p. 2023, 2023, doi: https://doi.org/10.31004/jiik.v2i1.23421.

[7] W. Liang, “Toxicity and Effect of Chemicals in Skin Care Products on Human Health,” IOP Conf. Ser. Earth Environ. Sci., vol. 512, no. 1, 2020, doi: 10.1088/1755-1315/512/1/012081.

[8] D. Novitasari, H. Akbar, A. Sutriyawan, Riswan, and H. Magdalena, “Analisis Jenis Kelamin, Riwayat Alergi, dan Personal Hygiene dengan Kejadian Dermatitis di Wilayah Kerja Puskesmas Passi Barat,” J. Keperawatan Cikini, vol. 4, no. 1, pp. 40–45, 2023, doi: https://doi.org/10.55644/jkc.v4i1.104.

[9] R. Lestari, “Hubungan Sanitasi Lingkungan Dengan Gejala Penyakit Kulit Di Wilayah Kerja Puskesmas Sukamenanti Kabupaten Pasaman Barat,” Nan Tongga Heal. Nurs., vol. 17, no. 1, pp. 14–23, 2022, doi: 10.59963/nthn.v17i1.98.

[10] O. A. Olusegun and B. S. Martincigh, “Allergic contact dermatitis: a significant environmental and occupational skin disease,” Int. J. Dermatol., vol. 60, no. 9, pp. 1082–1091, 2021, doi: 10.1111/ijd.15502.

[11] P. D. Rinanda, D. N. Aini, T. A. Pertiwi, S. Suryani, and A. J. Prakash, “Implementation of Convolutional Neural Network (CNN) for Image Classification of Leaf Disease In Mango Plants Using Deep Learning Approach,” Public Res. J. Eng. Data Technol. Comput. Sci., vol. 1, no. 2, pp. 56–61, 2024, doi: 10.57152/predatecs.v1i2.872.

[12] E. Goceri, “Diagnosis of skin diseases in the era of deep learning and mobile technology,” Comput.

Biol. Med., vol. 134, p. 104458, 2021, doi: 10.1016/j.compbiomed.2021.104458.
[13] B. Haritha, N. Ramya, S. Afrin, and U. B. Hepsibha, "Skin Disease Detection System Using Convolutional Neural Network," Iarjset, vol. 11, no. 3, pp. 304–311, 2024, doi: 10.17148/iarjset.2024.11348.
[14] A. C. Ramdhana and N. Pratiwi, "Perbandingan Kinerja Model Convolutional Neural Network pada Klasifikasi Kanker Kulit," Edumatic J. Pendidik. Inform., vol. 7, no. 2, pp. 197–206, 2023, doi: https://doi.org/10.29408/edumatic.v7i2.19823.
[15] D. Anastasya, S. Fahri, and S. Situmorang, "Implementasi Metode Convolutional Neural Network (CNN) Dalam Klasifikasi Motif Batik.," Nuansa Inform., vol. 18, no. 1, pp. 1–5, 2024, doi: 10.25134/ilkom.v18i1.21.
[16] Nabila Asryani Sundari, Rita Magladena, and Sofia Saidah, "Klasifikasi Jenis Kulit Wajah Menggunakan Metode Covolutional Neural Network (CNN) Efficientnet-B0," e-Proceeding Eng., vol. 8, no. 6, pp. 3180–3187, 2022.
[17] T. Nagaoka, "Hyperparameter optimization for deep learning-based automatic melanoma diagnosis system," Adv. Biomed. Eng., vol. 9, pp. 225–232, 2020, doi: https://doi.org/10.14326/abe.9.225.
[18] A. Abdellatif et al., "An Effective Heart Disease Detection and Severity Level Classification Model Using Machine Learning and Hyperparameter Optimization Methods," IEEE Access, vol. 10, pp. 79974–79985, 2022, doi: 10.1109/ACCESS.2022.3191669.
[19] Y. N. Kunang, S. Nurmaini, D. Stiawan, and B. Y. Suprapto, "Attack classification of an intrusion detection system using deep learning and hyperparameter optimization," J. Inf. Secur. Appl., vol. 58, p. 102804, 2021, doi: 10.1016/j.jisa.2021.102804.
[20] M. Elgendi et al., "The Effectiveness of Image Augmentation in Deep Learning Networks for Detecting COVID-19: A Geometric Transformation Perspective," Front. Med., vol. 8, pp. 1–12, 2021, doi: 10.3389/fmed.2021.629134.
[21] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," J. Big Data, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0197-0.
[22] G. Shanqing, M. Pednekar, and R. Slater, "Improve Image Classification Using Data Augmentation and Neural Networks," SMU Data Sci. Rev., vol. 2, no. 2, pp. 1–43, 2019, [Online]. Available: https://scholar.smu.edu/datasciencereviewhttp://digitalrepository.smu.edu.Availableat:https://scholar.smu.edu/datasciencereview/vol2/iss2/1
[23] P. Kim, MATLAB Deep Learning: With Machine Learning, Neural Networks and Artificial Intelligence. 2017. doi: 10.1007/978-1-4842-2845-6.
[24] D. Hastari, S. Winanda, A. R. Pratama, N. Nurhaliza, and E. S. Ginting, "Application of Convolutional Neural Network ResNet-50 V2 on Image Classification of Rice Plant Disease," Public Res. J. Eng. Data Technol. Comput. Sci., vol. 1, no. 2, pp. 71–77, 2024, doi: 10.57152/predatecs.v1i2.865.
[25] H. Deng, W. X. Zhang, and Z. F. Liang, "Application of BP Neural Network and Convolutional Neural Network (CNN) in Bearing Fault Diagnosis," IOP Conf. Ser. Mater. Sci. Eng., vol. 1043, no. 4, pp. 1–7, 2021, doi: 10.1088/1757-899X/1043/4/042026.
[26] G. W. Lindsay, "Convolutional neural networks as a model of the visual system: Past, present, and future," J. Cogn. Neurosci., vol. 33, no. 10, pp. 2017–2031, 2021, doi: 10.1162/jocn_a_01544.
[27] T. Bezdan and N. Bačanin Džakula, "Convolutional Neural Network Layers and Architectures," pp. 445–451, 2019, doi: 10.15308/sinteza-2019-445-451.
[28] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach," Procedia Comput. Sci., vol. 132, pp. 679–688, 2018, doi: 10.1016/j.procs.2018.05.069.
[29] Y. S. HARIYANI, S. HADIYOSO, and T. S. SIADARI, "Deteksi Penyakit Covid-19 Berdasarkan Citra X-Ray Menggunakan Deep Residual Network," ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron., vol. 8, no. 2, p. 443, 2020, doi: 10.26760/elkomika.v8i2.443.
[30] A. Zafar et al., "A Comparison of Pooling Methods for Convolutional Neural Networks," Appl. Sci., vol. 12, no. 17, pp. 1–21, 2022, doi: 10.3390/app12178643.
[31] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," 36th Int. Conf. Mach. Learn. ICML 2019, vol. 2019-June, pp. 10691–10700, 2019.
[32] M. M. Islam, M. A. Talukder, M. A. Uddin, A. Akhter, and M. Khalid, "BrainNet: Precision Brain Tumor Classification with Optimized EfficientNet Architecture," Int. J. Intell. Syst., vol. 2024, 2024, doi: 10.1155/2024/3583612.
[33] Mohi Uddin Anando, Rabbi Hasan Himel, and Shifar Tanjam, "Enhancing the Identification of Brain Tumours Using the CNN Ensemble Model," Malaysian J. Sci. Adv. Technol., vol. 4, no. 3, pp. 338–353, 2024, doi: 10.56532/mjsat.v4i3.264.
[34] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," Neurocomputing, vol. 415, pp. 295–316, 2020, doi: 10.1016/j.neucom.2020.07.061.
[35] Md Riyad Hossain and Dr. Douglas Timmer, "Machine Learning Model Optimization with Hyper

Parameter Tuning Approach," Glob. J. Comput. Sci. Technol., vol. 21, no. 2, pp. 1–1, 2021.

[36] A. Gaspar, D. Oliva, E. Cuevas, D. Zaldívar, M. Pérez, and G. Pajares, "Hyperparameter Optimization in a Convolutional Neural Network Using Metaheuristic Algorithms," Stud. Comput. Intell., vol. 967, no. 6, pp. 37–59, 2021, doi: 10.1007/978-3-030-70542-8_2.

[37] M. Heydarian, T. E. Doyle, and R. Samavi, "MLCM: Multi-Label Confusion Matrix," IEEE Access, vol. 10, pp. 19083–19095, 2022, doi: 10.1109/ACCESS.2022.3151048.

[38] M. M. Mijwil, "Skin cancer disease images classification using deep learning solutions," Multimed. Tools Appl., vol. 80, no. 17, pp. 26255–26271, 2021, doi: 10.1007/s11042-021-10952-7.

[39] W. Gouda, N. U. Sama, G. Al-Waakid, M. Humayun, and N. Z. Jhanjhi, "Detection of Skin Cancer Based on Skin Lesion Images Using Deep Learning," Healthcare, vol. 10, pp. 1–18, 2022, doi: https://doi.org/10.3390/healthcare10071183.