JOIN (Jurnal Online Informatika)
p-ISSN: 2528-1682, e-ISSN: 2527-9165
Volume 9 Number 2 | December 2024: 219-227
DOI: 10.15575/join.v9i2.1195

Comparison of EfficientNetB0 and EfficientNetB7 Models in Classifying Malaria Based on Blood Cells

Muhammad Rizkiansyah Maulana[1], Zamah Sari[2], Didih Rizki[3]
[1,2,3]Department of Informatics, University of Muhammadiyah Malang, Malang, Indonesia

Article Info

Article history:

Received Sep 24, 2023
Revised April 28, 2024
Accepted August 24, 2024
Published December 27, 2024

Keywords:

Classification
Convolutional Neural Network
EfficientNet
Malaria

ABSTRACT

Malaria is a disease caused by the bite of malaria mosquitoes, which spreads through blood. Malaria mosquitoes will spread the Plasmodium parasite through their bites. Early malaria identification is essential so the disease can be prevented immediately. Through data science, which utilizes the CNN model, the classification of blood infected with parasites can be predicted accurately. This research uses data obtained from Kaggle website with 27,558 image samples. The data is divided into two classes, parasite-infected and uninfected, which are then divided again into two types. The first class is training data divided into 80% of the total data and the other 20% as validation data. This research used two test scenarios to obtain a more effective classification model. The first scenario uses Hyperparameter Tuning and the EfficientNetB0 model with classification results of 95%. Meanwhile, the classification achievement for scenario two was 99% by utilizing EfficientNetB7.

Corresponding Author:

Zamah Sari,
Informatics Department, Faculty of Engineering, University of Muhammadiyah Malang
Jl. Raya Tlogomas No.246, Babatan, Tegalgondo, Kec. Lowokwaru, Kota Malang, Jawa Timur 65144
Email: zamahsari@umm.ac.id

1. INTRODUCTION

Malaria, a deadly illness caused by the Plasmodium parasite, is spread through the bite of an infected Anopheles mosquito, posing a threat to life [2]. This disease poses a significant public health concern and has the potential to be fatal, particularly in regions with tropical and subtropical climates, like Africa, Asia, and South America [3]. While malaria is both preventable and treatable, it can result in severe sickness and even death, particularly among infants, children under five years old, pregnant women, travelers, and those with HIV or AIDS [1].

Typically, malaria signs manifest between 7 to 30 days post-infection, showing symptoms like fever, headaches, nausea, vomiting, muscle pains, and exhaustion [4]. Neglecting malaria can result in serious issues such as anemia, kidney malfunction, and respiratory difficulties [5]. Certain groups, including infants, youngsters, and pregnant women, are at elevated risk of contracting malaria [1].

Malaria represents a significant global health challenge, impacting millions of individuals across the world, with approximately 229 million cases and 409,000 fatalities reported each year. Most of the affected individuals are children below the age of five in Sub-Saharan Africa, where malaria is particularly lethal [6]. In Indonesia, there were 261,617 reported malaria cases [7]. Annually, an estimated 30 million malaria cases occur, with only 10% of individuals receiving treatment at healthcare facilities [8].

Malaria can also have a major economic impact, as it can affect productivity and reduce a person's ability to work and contribute to society [9]. Therefore, prevention and control of malaria is a global health priority, and WHO and other world health organizations are committed to reducing the burden of this disease worldwide through appropriate prevention and treatment [6].

Multiple techniques exist for malaria detection, such as Rapid Diagnostic Tests (RDT) and Polymerase Chain Reaction (PCR) [19]. Laboratory-based blood tests are available, but they may be time-consuming due to the minuscule size of the plasmodium parasite [10][11]. The established approach for malaria detection is the microscopic analysis of blood smears, but it exhibits low sensitivity and mandates skilled personnel [6][12]. Consequently, there is a necessity for a system to identify malaria in cell 2 of blood smear images, aiming to decrease inaccuracies and expedite the examination process conducted by medical professionals.

Several techniques, including machine learning and deep learning algorithms, are available for malaria detection. Machine learning enables autonomous learning, while deep learning, a subset of it, mimics the human brain's neural system. Commonly employed deep learning algorithms include CNN, LSTM, and RNN. Research has shown that they outperform traditional image processing and machine learning methods in identifying and detecting malaria parasites in microscopic blood smear images. Nonetheless, traditional approaches like microscopic scrutiny of blood smears continue to serve as the "benchmark" for lab-confirmed malaria diagnosis.

Convolutional Neural Networks (CNN) excel in accurately classifying images by handling transformations like rotation and translation [13]. CNN also can conduct training before testing, eliminating the need for repeated training [14]. However, CNN may suffer from longer training times and overfitting when dealing with large datasets, resulting in inaccurate predictions [15]. Overfitting usually occurs due to the training data, the model works well, and the test data the model fails to work properly [16]. This can be overcome by increasing the effectiveness of learning on the CNN model, which can be done in several ways such as increasing the initialization of the model with transfer learning, data augmentation, using dropouts to reduce overfitting when training models, and using batch normalization to overcome long model training times [17][18].

Diyasa and colleagues conducted research on the classification of blood smear images for malaria detection, utilizing pre-trained deep convolutional neural networks. The dataset was split into 70% training data and 30% test data. Two pre-trained models, GoogleNet and ShuffleNetV2, were compared, and the models were trained for 15 epochs using the Adam optimizer. The evaluation of these models yielded an accuracy of 93.89% with GoogleNet and 95.20% with ShuffleNet V2 [20].

Qanbar and his team investigated malaria detection using the Residual Attention Network method. The dataset was divided into 20,658 training samples and 6,900 test samples. A comparison was made between the Residual Attention Network and Support Vector Machine (SVM), with a six-epoch model training. The results indicated an accuracy of 95.79% with the Residual Attention Network and 83.33% with SVM [21].

Shah and colleagues researched malaria detection using the CNN algorithm to classify blood smear images. The dataset consisted of two classes: infected with parasites and uninfected, totaling 27,558 images. The constructed model achieved an accuracy of 95.20% [22].

The study seeks to improve previous research on categorizing blood smear images for malaria identification. It introduces a new approach using the advanced EfficientNetB0 and EfficientNetB7 models, recognized for their exceptional precision and effectiveness compared to pre-existing models. The goal of the study is to achieve higher accuracy than previous research and improve the precision of blood smear image classification for malaria detection.

2. METHOD

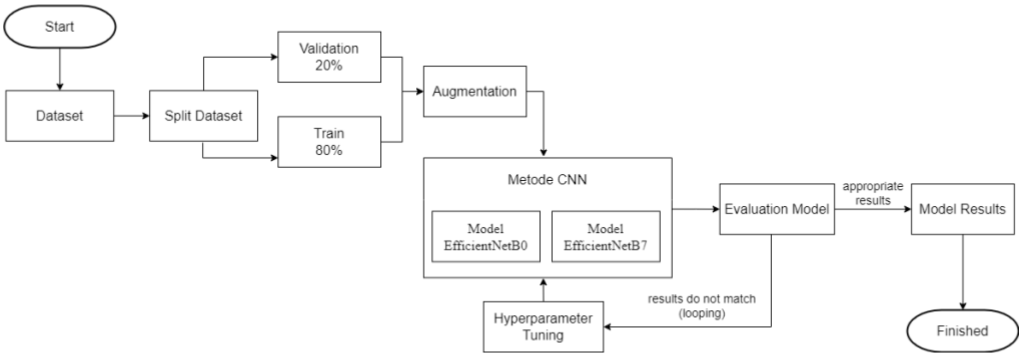

Figure 1. Research method flowchart

The research process for this study is illustrated in Figure 1. It commenced by obtaining the dataset from Kaggle and proceeded through various phases of data preprocessing, which involved dividing the data into training and validation sets. Subsequently, the results of data grouping were processed, leading to the modeling stage for data training. Data augmentation was applied, and the model was established and trained using the EfficientNetB0 and EfficientNetB7 models. To optimize the proposed model, hyperparameter techniques were integrated into the EfficientNetB7 model. The final phase of the research encompassed the evaluation of the model.

2.1. Dataset

Figure 2 illustrates a malaria-stained image from each category. The dataset utilized in this research was sourced from Kaggle under the name "Malaria Cell Images Dataset." The dataset contains two classes, parasitized and uninfected, with 27,558 images, 13,780 of which are parasitized and 13,780 are uninfected [23].

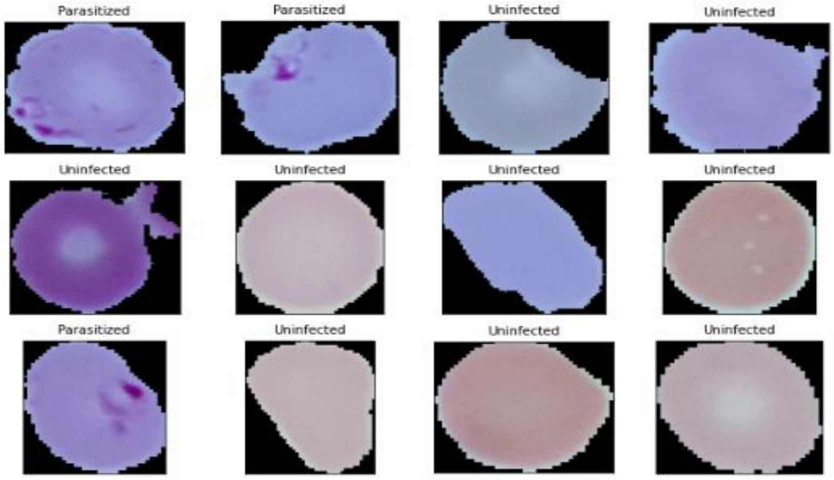

Figure 2. Image samples for each malaria-stained image label

2.2. Preprocessing Data

The dataset contains two classes, parasitized and uninfected, with 27,558 images. Data preprocessing was done, including data splitting into train and validation folders. The train data to validation data ratio was defined as 80% to 20%. Data augmentation was performed using ImageDataGenerator from the Keras library. The CNN model was trained using the train data. The EfficientNetB0 and EfficientNetB7 models were used, and the hyperparameter technique was added to acquire the best parameters for the suggested model.

2.3. Augmentation

Augmentation data is a technique used to enrich the diversity of images to enhance the model's accuracy. Augmentation encompasses a range of techniques that expand the dataset by either modifying existing data copies or synthetically generating new instances using the available dataset. This involves introducing subtle alterations to the dataset or leveraging deep learning methods to create fresh data points. This technique involves slightly modifying the original data, such as altering its geometry and color properties through rotation, zooming, flipping, and many more to expand and diversify the training dataset. Data augmentation can enhance the robustness and performance of machine learning models, especially when obtaining quality data is challenging in various fields, including healthcare [24]. In this research, the augmentation parameters employed included a rotation range of 30, a zoom range of 0.1, width and height shift range set to 0.1, horizontal flip set to True, vertical flip set to False, and rescale at 1 of 255.

2.4. Model Architecture

This research concentrates on the CNN model's structure for identifying malaria in blood smear cell images with EfficientNetB0 and EfficientNetB7 architecture models. These models are recognized for being more precise and quicker than competing pre-trained models. [25]. EfiicientNetB0 consists of 230 layers, whereas EfficientNetB7 boasts a more extensive architecture with 813 layers [26]. The model is composed of various elements, including an input layer (128×128) from the pre-trained

EfficientNet model, such as GlobalAveragePooling2D, Dropout layers (0.2, 0.5), and dense layers (128, 512, and 1024).

Pooling layers splits the image feature map into smaller parts and combines them into new sections. The MaxPool layer picks the highest values from the feature map, whereas the AveragePool calculates the average values to capture image information thoroughly [21]. The GlobalAveragePooling2D layer extracts the average feature value from an image. Techniques are used to enhance image diversity and improve the model's precision. The outcomes of the model's structure are illustrated in Table 1.

Table 1. Architectural Design of the CNN Model

Layer	Filter	Kernel Size	Activation
EfficientNet_Input (128, 128)	-	-	-
GlobalAveragePooling2D	-	-	-
Dropout	0.2,0.5	-	-
Dense	512, 1024	-	relu
Dropout	0.2, 0.5	-	-
Dense	128	-	relu
Dense	2	-	sigmoid

This study is proposed model that makes use of pre-trained models from EfficientNetB0 and EfficientNetB7. By integrating a Hyperparameter Tuning process, the model's optimal parameters are determined, thereby enhancing its performance. The study presents two testing scenarios for the model with the specified parameters. The check point function in callback is used to save the top-performing model during training, determined by a selected performance metric on the validation dataset. This callback can be used flexibly, but in this context, it's solely used to save the best model found during training. It can also be used in conjunction with early stopping to halt neural network training at the appropriate juncture [27]. The table below outlines the testing scenarios for the proposed model along with a description of the mode, as detailed in Table 2.

Table 2. Model scenario description

Scenario	Description
Model 1	Model EfficientNetB0
Model 2	EfficientNetB7 model with the best hyperparameter results

2.4.1 Model 1 Scenario Testing

Model 1 scenario testing employs the proposed EfficientNetB0 model and uses the parameters listed in Table 3 for this scenario model.

Table 3. Scenario architecture model 1

Layer	**Filter**	**Kernel Size**	**Activation**
EfficientNet_Input (128, 128)	-	-	-
GlobalAveragePooling2D	-	-	-
Dropout	0.5	-	-
Dense	1024	-	relu
Dropout	0.5	-	-
Dense	128	-	relu
Dense	2	-	sigmoid

Table 3 includes specifications with Adam Optimizer, such as the Dropout layer with range 0.5, the Dense layer with relu 1024 activation, the Dropout layer with range 0.5, the Dense layer with relu 128 activation, and the Dense layer with sigmoid two activations.

2.4.2 Model 2 Scenario Testing

Scenario testing in Model 2 involves utilizing the EfficientNetB7 model with modified hyperparameters, and Table 4 concisely displays the outcomes of these hyperparameter adjustments.

Table 4. EfficientNetB7 Hyperparameter tuning results

Parameter	**Optimizer**	**Dropout**	**Dense Layer**	**accuracy**
1	Adamax	0.2	1024	92%
2	RMSprop	0.2	1024	71%

Based on the outcomes of Hyperparameter tuning for EfficientNetB7 in Table 4, it is evident that parameter 1, with an accuracy of 92%, is the most optimal. In this instance, the model utilizes the parameters derived from the outcomes of Hyperparameter Tuning, as detailed in Table 5.

Table 5. Architectural Design of the CNN Model

Layer	Filter	Kernel Size	Activation
Efficient Net Input (128, 128)	-	-	-
Global Average Pooling 2D	-	-	-
Dropout	0.2	-	-
Dense	512	-	ReLu
Dropout	0.2	-	-
Dense	128	-	ReLu
Dense	2	-	Sigmoid

The architecture for scenario 2 is outlined in Table 5. It consists of specifications such as Dropout set to 0.2, a Dense Layer of 512 using Relu activation, another Dropout of 0.2, a Dense Layer of 128 with Relu activation, a Dense layer of 2 using sigmoid activation, using the Adamax Optimizer, and spanning 50 epochs. Binary_crossentropy classification is used.

2.5. Hyperparameter Tuning

Hyperparameter tuning is a procedure designed to discover the optimal parameter combination for a model to attain the best possible outcomes. When employing the CNN network model in conjunction with hyperparameter tuning, various parameters, including kernel size, stride, channel count, and dropout rates, are adjusted [18]. By adjusting these hyperparameters, the model can be fine-tuned to yield the most favorable results [19]. In this research, hyperparameter tuning was applied to identify the most suitable parameter values by comparing proposed parameter variations. These parameter comparisons are detailed in Table 6.

Table 6. Comparative parameters in hyperparameter tuning

Parameter	Comparative Value
Dropout	0.2, 0.5
Dense Layer	1024, 512
Optimizer	Adam, Adamax, RMSprop

3. RESULT AND DISCUSSION

The results of the architectural model developed in this study for classifying malaria in blood smear images have been achieved in this phase. The initial step involved data collection and splitting, with an 80% portion allocated for training and 20% for testing. Subsequently, the training data underwent augmentation, applying parameters like rotation range of 30, a zoom range of 0.2, width and height shift range of 0.1, horizontal flip set to True, vertical flip set to False, and rescale of 255. Data augmentation is a technique employed to increase image variance and enhance model accuracy artificially. At this stage, the callback technique is utilized, incorporating ModelCheckpoint to monitor real-time model performance. This method retains only the best models across epochs, reducing memory consumption. It encompasses making subtle alterations to the dataset or generating new data points through deep learning. Augmented data finds application in various machine learning domains, especially when obtaining high-quality data, is challenging, and it aids in enhancing model robustness and performance across different fields, including healthcare.

In scenario 1, the architecture details, as outlined in Table 3, include specifications with Adam Optimizer such as Dropout layer with range of 0.5, Dense layer with Relu activation of 1024, Dropout layer with range 0.5, Dense layer with Relu activation of 128, and Dense layer with sigmoid activation of 2. The training duration is 50, and binary_crossentropy is employed for classification. After conducting the tests, a line graph illustrates the training process results. This graphical representation helps monitor changes over each iteration and identify potential overfitting issues in the model. The plotting results on the EfficientNetB0 model are presented in Figure 3 and Figure 4.

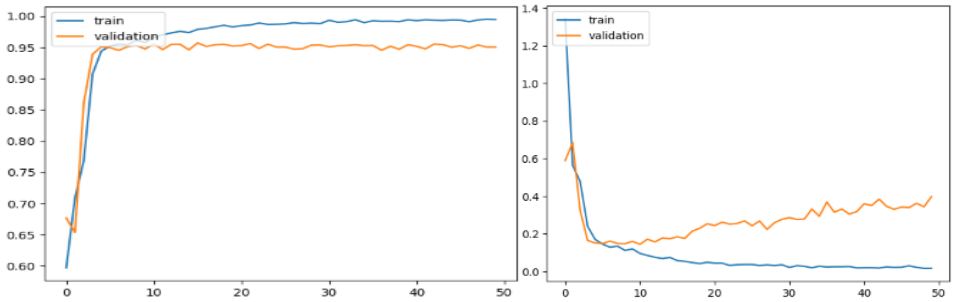

Figure 3 and 4. Results of model scenario accuracy plot 1

Figure 3 displays the accuracy plot results for scenario model 1. Initially, from epoch 0 to 5, the validation accuracy exhibits erratic fluctuations. However, from epoch 6 to 50, it stabilizes and approaches a value of 1. The erratic accuracy at the outset is expected because the model is still learning from the dataset. So, after epoch 6, the model's performance improves as it has acquired knowledge from the dataset.

In contrast, Figure 5 illustrates the validation loss plot, where initial instability is observed from epoch 10 to 50. This initial loss variability is also attributed to the model's learning phase with the dataset. Unfortunately, at the 10th to 50th epoch, the model still demonstrates overfitting issues.

Following the acquisition of graphical results from the completed training, the next stage involves assessing the constructed models. Performance metrics are presented in a classification report, visualized in Table 7, as outlined below.

Table 7. Model 1 scenario classification report

Classification Report	
Accuracy	95 %
Precision	95 %
Recall	96 %

In scenario 1, the architectural outcomes show a 95% accuracy, 95% precision, 96% recall, and a 95% f1-score. Additionally, model evaluation can be further understood by examining the confusion matrix table, which quantifies the machine learning method's ability to correctly or incorrectly predict outcomes from the entire dataset [20]. Confusion matrix’s result for the scenario test of model 1 are illustrated in the captivating of Figure 5.

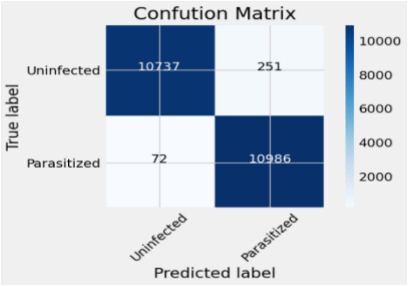

Figure 5. Confusion matrix scenario model 1

Figure 5 presents the results of the confusion matrix for model 1. It shows that for the “Uninfected” category, 10,737 images data were correctly predicted, while 251 were incorrectly predicted. In the “Parasitized” category, 10,986 images data were correctly predicted, with 72 incorrectly predicted.

Following the training phase, a line graph is generated to visualize the training process results. These graphs help monitor changes in each iteration and assess potential overfitting. The plotting results, conducted without using hyperparameters on the EfficientNetB7 model, are illustrated in Figure 6 and Figure 7.

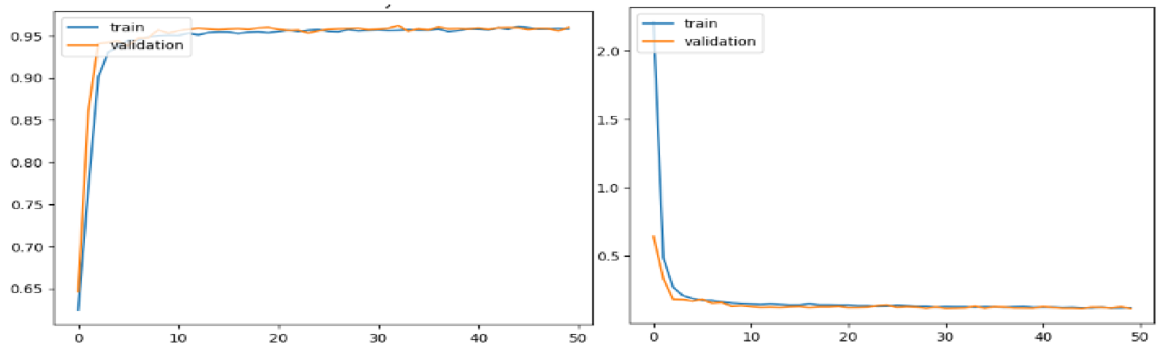

Figure 6 and 7. The results of the model scenario accuracy plot 2

Figure 6 illustrates the accuracy plot results for scenario 2. In this graph, the initial validation accuracy fluctuates from epochs 0 to 8 but stabilizes and approaches 1 from epochs 9 to 50. The initial inconsistency in accuracy is attributed to the model's learning phase within the dataset, and it becomes more consistent from the 9th epoch onward as the model gains a better understanding of the data.

On the other hand, Figure 7 showcases the validation loss plot, where initial instability is observed from epochs 0 to 4. However, from epochs 5 to 50, the loss stabilizes and approaches 0. The initial erratic loss movements can be attributed to the model's early dataset learning phase, and the stability in loss from the 5th epoch onwards indicates that the model has learned from the dataset.

Once the graphical results from the completed training are obtained, the subsequent phase involves assessing the constructed models. Performance metrics are presented in a classification report, visualized in Table 8, as outlined below.

Table 8. Summary of the Classification. Report for Model 2

Classification Report	
Accuracy.	99%
Precision.	99%
Recall.	98%
F1-Score.	99%

In the context of model scenario 2, the architectural results indicate an accuracy of 99%, precision of 99%, recall of 98%, and an f1-Score of 99%. Additionally, model evaluation can be further assessed using the confusion matrix, which measures the machine learning method's ability to predict correct and incorrect outcomes from the entire dataset [20]. The confusion matrix results for the scenario test of model 2 are presented in Figure 8.

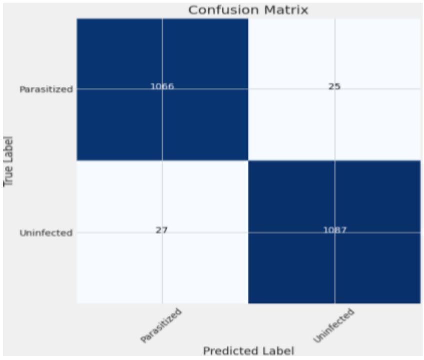

Figure 8. Confusion matrix scenario model 2

Figure 8 displays the confusion matrix for model 2, revealing that in the "Uninfected" class, 1,066 images were correctly predicted, while 25 were incorrect. In the "Parasitized" class, 1,087 photos were correctly predicted, but 27 were misclassified.

Based on the results of the scenario trials that tried to classify malaria derived from images depicting blood cells, there were differences in the trial results. The difference lies in using Hyperparameters, which significantly increase model accuracy. This can be proven in Table 9, which summarizes the results of accuracy tests in these two model scenarios.

Tabel 9. Model Testing Results

Scenario	Accuracy	Precision	Recall	F1-Score
Model 1	95%	96%	95%	96%
Model 2	99%	99%	98%	99%

.

Nur Ibrahim et al. [1] have researched the same dataset using a pure CNN model with accuracy results of 95%. After testing the scenarios in this research, it was found that the model in scenario one, which used the EfficientNetB0 model, also achieved the same accuracy. However, accuracy on the same dataset classification can increase rapidly when using the EfficientNetB7 model with the addition of Hyperparameter tuning in scenario 2 with an accuracy 99%. Specifics regarding the comparisons made in previous studies are presented in Table 10 below.

Table 10. Summary of the Comparison Table with Prior Research

Scenario	Dataset	Model	Accuracy
Nur Ibrahim et al.[1]	*Malaria Cell Images Dataset*	CNN	95%
Model 1	*Malaria Cell Images Dataset*	*EfficientNetB0*	95%
Model 2	*Malaria Cell Images Dataset*	*EfficientNetB7 using Hyperparameter Tuning*	99%

.

CONCLUSION

This research proposes 2 CNN architectural model scenarios in EfficientNetB0 and EfficientNetB7 for classifying malaria based on blood cell data. The EfficientNetB7 model, enhanced with Hyperparameter Tuning, can improve the model's work quality performance. Refining the EfficientNetB7 model with Hyperparameter Tuning can increase the results of predicting the sensitivity of training data. This increases accuracy by 4% compared to previous research [1] and scenario 1. Apart from the proposed model, the increase in classification accuracy is also influenced by any existing data using blood cell images. The recommended layer arrangement in EfficientnetB0 includes specifications with Adam optimizer, such as a Dropout layer with a range 0.5, a Dense layer with relu 1024 activation, a Dropout layer with a range of 0.5, a Dense layer with relu 128 activation, and Dense layer with sigmoid activation as presented in Table 3.

Meanwhile, EfficientNetB7 includes specifications like a Dropout layer with a range of 0.2, Dense layer 128 with relu activation, and Dense layer 2 using sigmoid activation, using the Adamax Optimizer, as shown in Table 5. This model also provides better prediction results than previous research [1]. The initial stage of this research before implementing the model architecture includes data collection and separation, with 80% allocated for training and 20% for testing. After undergoing augmentation with parameters rotation range of 30, zoom range of 0.2, width and height shift range of 0.1, horizontal flip set to true, vertical flip set to false, and rescaling to 255, the training data is further enhanced with Hyperparameter Tuning and ModelCheckPoint to monitor real-time model performance. This research addresses the challenge of achieving the highest accuracy in classifying malaria based on blood cells. It demonstrates that the EfficientNetB7 model, including Hyperparameter Tuning, outperforms EfficientNetB0 and previous research [1].

Future research on similar topics, it is advisable to explore the dataset further. This study introduced various adjustments to the dataset's preprocessing and incorporated augmentation techniques. Subsequent research should assess which preprocessing and augmentation methods yield optimal performance and results.

ACKNOWLEDGEMENTS

The University of Muhammadiyah Malang's Department of Informatics provided valuable support for this research, enabling our study.

REFERENCES

[1] N. K. C. PRATIWI, N. IBRAHIM, Y. N. FU'ADAH, and S. RIZAL, "Deteksi Parasit Plasmodium pada Citra Mikroskopis Hapusan Darah dengan Metode Deep Learning," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 9, no. 2, p. 306, 2021, doi: 10.26760/elkomika.v9i2.306.

[2] R. Puasa, A. A. H, and A. Kader, "Identifikasi Plasmodium Malaria Didesa Beringin Jaya Kecamatan Oba Tengah Kota Tidore Kepulauan," J. Ris. Kesehat., vol. 7, no. 1, p. 21, 2018, doi: 10.31983/jrk.v7i1.3056.

[3] M. R. Islam *et al.*, "Explainable Transformer-Based Deep Learning Model for the Detection of Malaria Parasites from Blood Cell Images," *Sensors*, vol. 22, no. 12, pp. 1–20, 2022, doi: 10.3390/s22124358.

[4] P. Chanda-Kapata, N. Kapata, and A. Zumla, “COVID-19 and malaria: A symptom screening challenge for malaria endemic countries,” *Int. J. Infect. Dis.*, vol. 94, pp. 151–153, 2020, doi: 10.1016/j.ijid.2020.04.007.

[5] Y. Yohannes, S. Devella, and K. Arianto, “Deteksi Penyakit Malaria Menggunakan Convolutional Neural Network Berbasis Saliency,” JUITA J. Inform., vol. 8, no. 1, p. 37, 2020, doi: 10.30595/juita.v8i1.6671.

[6] WHO, W. H. O. (2020) World Malaria Report 2020. Diambil 1 april 2023, dari https://www.who.int/publications/i/item/9789240015791

[7] J. E. Mosso and C. Song, “Distribusi prevalensi infeksi Plasmodium serta gambaran kepadatan parasit dan jumlah limfosit absolut pada penderita malaria di RSUD Kabupaten Manokwari periode Januari – Maret 2019,” *Tarumanagara Med. J.*, vol. 2, no. 2, pp. 320–330, 2020, doi: 10.24912/tmj.v3i1.9735.

[8] A. Alim, A. Adam, and B. Dimi, “Prevalensi Malaria Berdasarkan Karakteristik Sosio Demografi,” J. Ilm. Kesehat., vol. 19, no. 01, pp. 4–9, 2020, doi: 10.33221/jikes.v19i01.399.

[9] J. Talapko, I. Škrlec, T. Alebić, M. Jukić, and A. Včev, “Malaria: The past and the present,” Microorganisms, vol. 7, no. 6, 2019, doi: 10.3390/microorganisms7060179.

[10] Y. Gustiani, K. Pramudho, and D. Sadik, “Analisis Faktor Yang Berhubungan Dengan 38 Pencegahan Penyakit Malaria,” J. Ilm. Permas J. Ilm. STIKES Kendal, vol. 11, pp. 1– 22, 2021, [Online]. Available: http://journal.stikeskendal.ac.id/index.php/PSKM.

[11] M. Harahap, J. Jefferson, S. Barti, S. Samosir, and C. A. Turnip, “Implementation of Convolutional Neural Network in the classification of red blood cells have affected of malaria,” SinkrOn, vol. 5, no. 2, pp. 199–207, 2020, doi: 10.33395/sinkron.v5i2.10713.

[12] A. W. Setiawan et al., “Deteksi Malaria Berbasis Segmentasi Warna Citra Dan Malaria Detection Using Color Image Segmentation and Machine,” vol. 8, no. 4, pp. 769–776, 2021, doi: 10.25126/jtiik.202184377.

[13] L. A. Andika, H. Pratiwi, and S. S. Handajani, “Klasifikasi Penyakit Pneumonia Menggunakan Metode Convolutional Neural Network Dengan Optimasi Adaptive Momentum,” Indones. J. Stat. Its Appl., vol. 3, no. 3, pp. 331–340, 2019, doi: 10.29244/ijsa.v3i3.560.

[14] M. Z. Ersyad, K. N. Ramadhani, A. Arifianto, and L. Belakang, “Pengenalan Bentuk Tangan dengan Convolutional Neural Network (CNN),” e-Proceeding Eng., vol. 7, no. 2, pp. 8212–8222, 2020.

[15] V. M. P. Salawazo, D. P. J. Gea, R. F. Gea, and F. Azmi, “Implementasi Metode Convolutional Neural Network (CNN) Pada Peneganalan Objek Video Cctv,” J. Mantik Penusa, vol. 3, no. 1, pp. 74–79, 2019.

[16] C. Garbin, X. Zhu, and O. Marques, “Dropout vs. batch normalization: an empirical study of their impact to deep learning,” Multimed. Tools Appl., vol. 79, no. 19–20, pp. 12777–12815, 2020, doi: 10.1007/s11042-019-08453-9.

[17] K. Sriporn, C.-F. Tsai, C.-E. Tsai, and P. Wang, “Analyzing Malaria Disease Using Effective Deep Learning Approach,” Diagnostics, pp. 1–22. doi.org/10.3390/diagnostics10100744

[18] I. G. S. M. Diyasa, A. Fauzi, A. Setiawan, M. Idhom, R. R. Wahid, and A. D. Alhajir, “Pre-Trained Deep Convolutional Neural Network for Detecting Malaria on the Human Blood Smear Images,” 3rd Int. Conf. Artif. Intell. Inf. Commun. ICAIIC 2021, pp. 235– 240, 2021, doi: 10.1109/ICAIIC51459.2021.9415183.

[19] Mbanefo A, Kumar N. Evaluation of Malaria Diagnostic Methods as a Key for Successful Control and Elimination Programs. Trop Med Infect Dis. 2020 Jun 19;5(2):102. doi: 10.3390/tropicalmed5020102.

[20] I. G. S. M. Diyasa, A. Fauzi, A. Setiawan, M. Idhom, R. R. Wahid, and A. D. Alhajir, “Pre-Trained Deep Convolutional Neural Network for Detecting Malaria on the Human Blood Smear Images,” 3rd Int. Conf. Artif. Intell. Inf. Commun. ICAIIC 2021, pp. 235– 240, 2021, doi: 10.1109/ICAIIC51459.2021.9415183.

[21] M. M. QANBAR and S. TASDEMIR, “Detection of Malaria Diseases with Residual Attention Network,” IJISAE, vol. 7, no. 4, pp. 238–244, 2019, doi: https://doi.org/10.18201/ijisae.2019457677

[22] D. Shah, K. Kawale, M. Shah, S. Randive, and R. Mapari, “Malaria Parasite Detection Using Deep Learning: (Beneficial to humankind),” *Proc. Int. Conf. Intell. Comput. Control Syst. ICICCS 2020*, no. Iciccs, pp. 984–988, 2020, doi: 10.1109/ICICCS48265.2020.9121073.

[23] kaggle datasets download -d iarunava/cell-images-for-detecting-malaria

[24] X. Xiao, M. Yan, S. Basodi, C. Ji, and Y. Pan, "Efficient Hyperparameter Optimization in Deep Learning Using a Variable Length Genetic Algorithm," arXiv, 2020. https://doi.org/10.48550/arXiv.2006.12703

[25] M. Aji. Purnama Wibowo, Muhammad Bima Al Fayyadl, Yufis Azhar, and Zamah Sari, “Classification of Brain Tumors on MRI Images Using Convolutional Neural Network Model EfficientNet,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 4, pp. 538–547, 2022, doi: 10.29207/resti.v6i4.4119.

[26] D. R. Nayak, N. Padhy, P. K. Mallick, M. Zymbler, and S. Kumar, “Brain Tumor Classification Using Dense Efficient-Net,” *Axioms*, vol. 11, no. 1, 2022, doi: 10.3390/axioms11010034.

[27] Ulfah Nur Oktaviana and Yufis Azhar, “Klasifikasi Sampah Menggunakan Ensemble DenseNet169,” *Resti*, vol. 1, no. 1, pp. 19–25, 2021, doi**:** https://doi.org/10.29207/resti.v5i6.3673