International Journal of Mathematics and Computer Research
ISSN: 2320-7167
Volume 13 Issue 07 July 2025, Page no. –5402-5406
Index Copernicus ICV: 57.55, Impact Factor: 8.615
DOI: 10.47191/ijmcr/v13i7.08

Facial Skin Disease Detection on Android Application Using TensorFlow Lite and Convolutional Neural Network

Masduki Zakarijah[1], Dimas Prasetyo[2]
[1]Electronics & Informatics Edu. Department, Universitas Negeri Yogyakarta, Yogyakarta Indonesia.
[2]Electronics Engineering Edu. Study Program, Universitas Negeri Yogyakarta, Yogyakarta Indonesia.

ARTICLE INFO

Corresponding Author:
Masduki Zakarijah

ABSTRACT

This study focuses on the development of an Android application that utilizes a Convolutional Neural Network (CNN) with the MobileNet V2 architecture and transfer learning methods to detect facial skin diseases. The development process includes requirement specification, planning, modeling, construction, implementation, testing, and maintenance phases. The dataset consists of 700 images categorized into 7 disease classes, with 80% used for training and 20% for validation. Data were collected through direct observation to evaluate the application's accuracy and detection speed. The results show that the application achieved a detection accuracy of 83% and an average detection time of 0.5 seconds, using TensorFlow Lite on the Android platform. Data analysis was conducted using descriptive statistics to assess product design, accuracy, and detection speed.

KEYWORDS: Machine Learning, CNN, Transfer Learning, Android Application, Facial Skin Diseases.

I. INTRODUCTION

Managing facial skin diseases is crucial as skin conditions often reflect overall health and significantly affect quality of life. Facial skin disorders not only cause physical discomfort but also lead to decreased self-confidence and psychological distress. Factors such as habits, lifestyle, diet, gender, skin microbiology, chemical exposure, and environmental conditions influence skin health. Bacterial infections of the skin and soft tissues occur due to an imbalance between pathogenic microorganisms and the human immune system [1]. Hence, accurate and rapid methods for diagnosing facial skin diseases are highly needed.

Recent studies have focused on developing CNN-based detection tools to improve diagnostic accuracy. The application of artificial intelligence in this field aims to prevent the progression of skin conditions and accelerate recovery processes.

A study focused on developing a navigation system for autonomous mobile robots (AMRs) in delivery tasks [2]. It employed an object detection system based on CNN using the SSD MobileNet V2 FPN Lite 320x320 architecture, retrained with a dataset of vehicles in Banda Aceh city. This system could detect objects such as cars, motorcycles, pedestrians, and rickshaws under varying lighting conditions, making it effective for AMRs.

Another study developed a novel method combining CNN and Support Vector Regression (SVR) to predict concrete compressive strength [2][3][4]. A CNN model named MPaConvNet was designed to extract essential features from concrete surface images. These features were then used in SVR algorithms to predict compressive strength, achieving results with R^2 of 99.59%, MAPE of 0.0209, RMSE of 0.6836, MSE of 0.4673, and a-20 of 1.0000 [5].

In a related work titled Facial Skin Disease Detection Using TensorFlow and Convolutional Neural Network, the CNN model was applied to detect 20 types of facial skin diseases, achieving an accuracy of 99.91% through feature extraction of edges, corners, and color variations [6]. Another study used CNN to classify acne from 150 images, split into 80% training and 20% testing, achieving an accuracy of 88% [7]. A pre-trained MobileNet V2 model was also utilized to diagnose five types of skin diseases on smartphones, resulting in 97.5% accuracy, 97.7% sensitivity, and 97.7% precision [8]. Research using Kaggle datasets covering 14 skin disease types resized images to 224x224 pixels and split the data into 85% for training, 10% for validation, and 5% for testing, yielding 96% accuracy [9].

Machine learning implementation on mobile devices has also been previously investigated. One study [10] evaluated the performance of PyTorch Mobile and TensorFlow Lite

(TFLite) across different mobile platforms. The testing scenarios demonstrated that each machine learning framework has different optimization strategies that impact performance. PyTorch Mobile showed significant advantages in multithreaded execution, whereas TFLite outperformed in model quantization. These differences stem from their distinct implementations and optimization strategies.

This research focuses on the development of an Android-based application for diagnosing facial skin diseases. The dataset used includes several skin conditions such as acne, actinic keratosis, milia, eczema, herpes, wrinkles, and healthy skin. A pre-trained MobileNet V2 model is employed as the CNN architecture for classifying facial skin diseases.

II. METHOD

A. Dataset

The dataset was collected manually from various sources, including Google Images and the Kaggle repository. Each selected image has high resolution to ensure optimal data quality. The types of images collected represent various skin conditions such as acne, actinic keratosis, milia, herpes, eczema, wrinkles, and healthy skin. The use of high-resolution images is intended to ensure accuracy and reliability in further analysis.

B. Data Preparation

The dataset consists of a collection of image files in .jpeg or .jpg format, collected manually. Some images originate from the same individual but are taken from different angles, while still representing the same disease class. Figure 1 illustrates sample images from the seven disease classes included in the dataset.

Acne Actinic Keratosis Eczema Kulit Sehat

Herpes Kerutan Milia

Figure 1. Representative Samples from the Dataset

C. Dataset Splitting

The dataset was evenly split according to each skin disease class. A total of 80% of the dataset was used as training data, while the remaining 20% was used for validation. This split ensures that each disease class is proportionally represented in both datasets. Table 1 presents the details of the dataset distribution.

Table 1. Dataset

Disease Class	Number of Images	Training Data (80%)	Validation Data (20%)
Acne	100	80	20
Actinic Keratosis	100	80	20
Eczema	100	80	20
Herpes	100	80	20
Milia	100	80	20
Wrinkles	100	80	20
Healthy Skin	100	80	20
Total	700	560	140

D. Training Algorithm

To ensure both efficiency and accuracy, this study employed the MobileNet V2 pre-trained model using transfer learning. The model was retrained in three experimental setups with different optimizers (Adam, SGD, RMSprop) to identify the most effective configuration. All images were resized to 160x160 pixels prior to training, which was conducted for 100 epochs with a learning rate of 0.0001. An early stopping technique was applied to prevent overfitting on the training data.

Table 2. First Experiment Configuration

Learning Type	**Adam Optimizer**	
Learning Rate	**0.0001**	
Layer Type	**Output Shape**	**Parameter**
Input	(160, 160, 3)	0
MobileNetV2 Base Model	(5, 5, 1280)	2,257,984
GlobalAveragePooling2D	(1280)	0
Dropout (rate=0.2)	(1280)	0
Dense (128 units, ReLU)	(128)	163,968
Dropout (rate=0.2)	(128)	0
Dense (32 units, ReLU)	(32)	4,128
Dropout (rate=0.2)	(32)	0
Dense (7 units, Softmax)	(7)	231
Total Parameters		2,426,311

Table 3. Second Experiment Configuration

Learning Type	**Adam Optimizer**	
Learning Rate	**0.0001**	
Layer Type	**Output Shape**	**Parameter**
Input	(160, 160, 3)	0
MobileNetV2 Base Model	(5, 5, 1280)	2,257,984

GlobalAveragePooling2D	(1280)	0
Dropout (rate=0.2)	(1280)	0
Dense (128 units, ReLU)	(128)	163,968
Dropout (rate=0.2)	(128)	0
Dropout (rate=0.2)	(32)	0
Dense (7 units, Softmax)	(7)	903
Total Parameters		2,426,311

Table 4. Third Experiment Configuration

Learning Type	**SGD (Stochastic Gradient Descent)**	
Momentum	**0.9**	
Learning Rate	**0.0001**	
Layer Type	**Output Shape**	**Parameter**
Input	(160, 160, 3)	0
MobileNetV2 Base Model	(5, 5, 1280)	2,257,984
GlobalAveragePooling2D	(1280)	0
Dropout (rate=0.2)	(1280)	0
Dense (128 units, ReLU), kernel_reguralizer: L2(0.0001)	(128)	163,968
Dropout (rate=0.2)	(128)	0
Dropout (rate=0.2)	(32)	0
Dense (7 units, Softmax)	(7)	903
Total Parameters		2,422,855

E. Android Application

The development of a facial skin disease detection Android application using TensorFlow and CNN involves three main stages: image capture using the camera, analysis via CNN, and displaying the classification result, as illustrated in Figure 2, with the user interface shown in Figure 3.

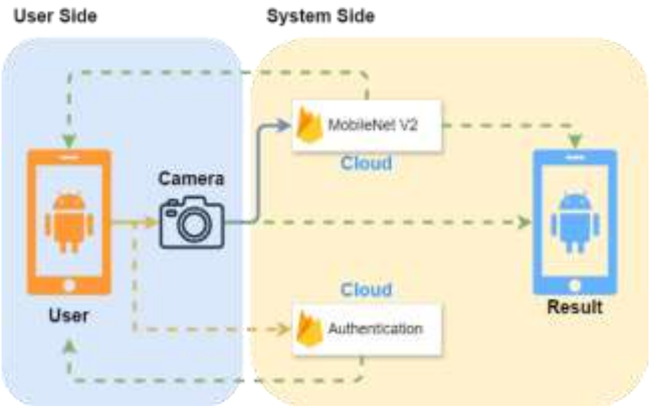
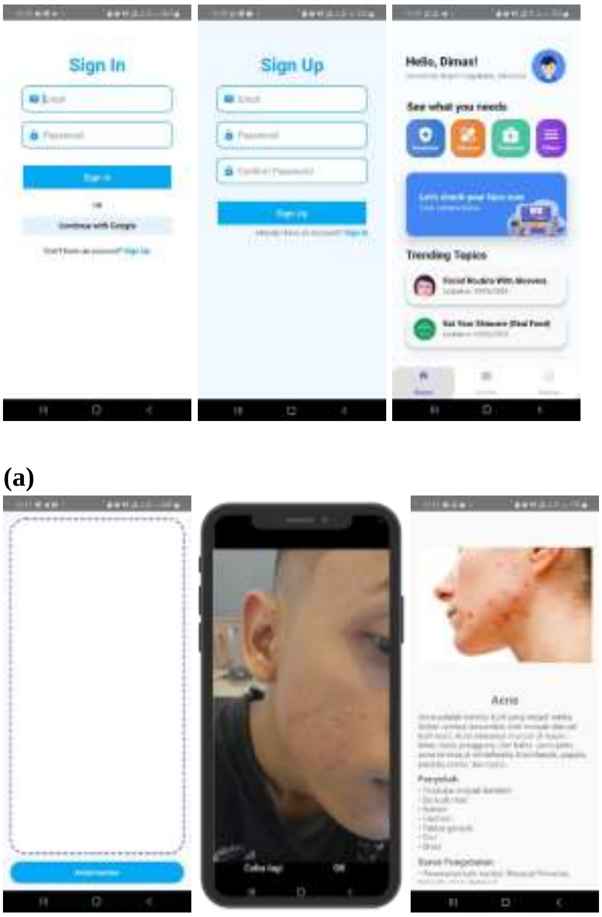


Figure 2. System Diagram



(a)

(b)

Figure 1 (a & b) Application User Interface Display

III. RESULTS AND DISCUSSION

A. Model Accuracy Testing

Three experiments were conducted using the pre-trained MobileNet V2 model on the prepared dataset. Figures 4 to 6 present the confusion matrices of the model, with respective accuracy scores of 69%, 73%, and 83%.

Figures 4 and 5 show confusion matrices from models trained with limited data augmentation and without fine-tuning, achieving a maximum accuracy of 73%.

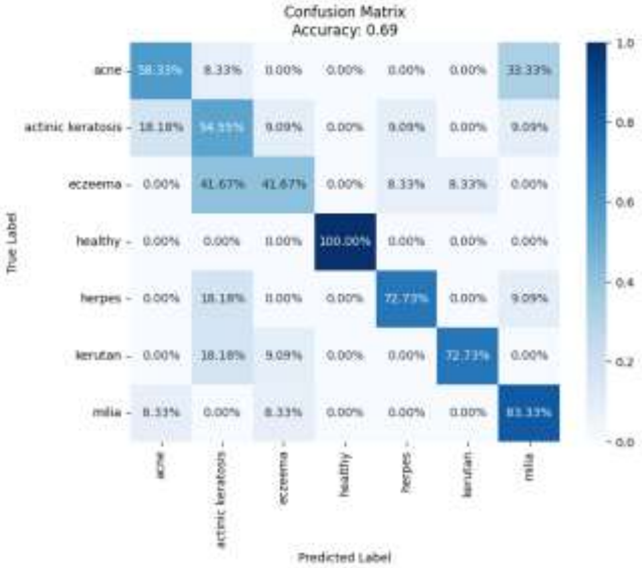


Figure 2 Confusion matrix of the first experiment.

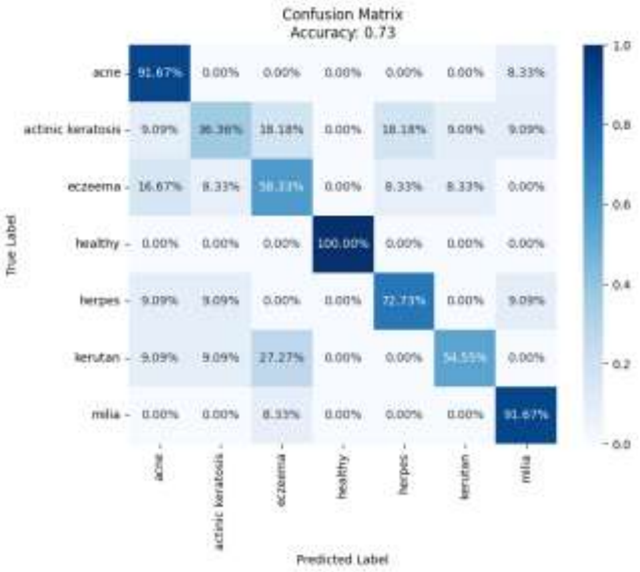


Figure 3. Confusion matrix of the second experiment

In the third experiment, shown in Figure 6, fine-tuning was performed after transfer learning for 50 epochs, using the RMSprop optimizer. Additionally, the "unfreeze" method was applied to unlock layers of the base model, allowing for retraining of those layers. Figures 7 and 8 display the training and validation accuracy and loss graphs.

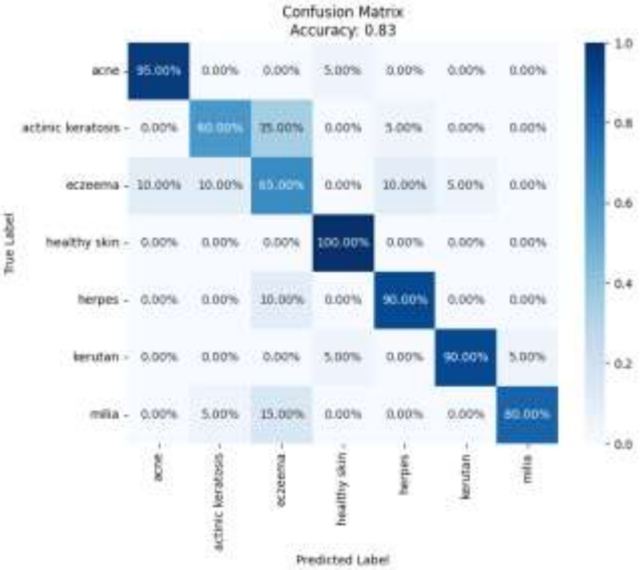


Figure 4. Confusion Matrix of the Third Experiment

Figure 5. Training and validation accuracy graph

Figure 6 Graph of Training and Validation Loss

The evaluation metrics used in the third configuration included accuracy, precision, recall, and F1-score for each class. The results showed satisfactory accuracy levels across all classes. However, there remains room for improvement, particularly in reducing misclassification and enhancing other metrics such as precision, recall, and F1-score. Table 5 presents the computed accuracy, precision, recall, and F1-score metrics.

Table 5. Accuracy, precision, recall, and F1-score values

Class	Accuracy	Precision	Recall	F1-Score
Acne	0.85	0.908	0.95	0.92
Actinic Keratosis	0.87	0.8	0.6	0.68
Eczeema	0.92	0.5	0.65	0.56
Healthy Skin	0.84	0.9	1	0.95
Herpes	0.85	0.9	0.9	0.9
Wrinkles	0.84	0.9	0.9	0.9
Milia	0.86	0.88	0.8	0.84

B. *Detection Speed Testing*

Detection speed testing was performed over 10 trials by end-users to evaluate real-time classification accuracy. Table 6 presents the results of detection speed and accuracy testing.

Table 6. Detection Speed Test

No.	Detection Speed (Hours : Minutes : Seconds : Milliseconds)	Description	
		Accurate	**Inaccurate**
1	00:00:00,42	✓	
2	00:00:00,34		✓
3	00:00:00,49		✓
4	00:00:00,52	✓	
5	00:00:00,44	✓	
6	00:00:00,49	✓	
7	00:00:00,57	✓	
8	00:00:00,68	✓	
9	00:00:00,50	✓	
10	00:00:00,68	✓	

The model demonstrated excellent detection speed performance. The average detection time across 10 trials was 0.5 seconds, with 8 correct classifications and 2 misclassifications.

IV. CONCLUSION

The Android application for facial skin disease detection, developed using TensorFlow Lite and CNN, features six interface screens, from the splash screen to the disease classification result. The training algorithm achieved an accuracy of 83% in classifying seven disease classes, with evaluation metrics for precision, recall, and F1-score indicating good performance. The application responded quickly during image capture and classification, although further data augmentation—such as varying image angles, luminance, and brightness—is recommended to enhance overall accuracy.

REFERENCES

1. A. N. Hidayati, Damayanti, M. Sari, M. D. Alinda, N. R. Reza, S. Anggraeni and Y. Widia, Infeksi Bakteri di Kulit, Surabaya: Airlangga University Press, 2019.
2. S. Ghimire, T. Nguyen-Huy, R. C. Deo, D. C. Perez and S. S. Sanz, "Efficient daily solar radiation prediction with deep learning 4-phase," Sustainable Material and Technologies, vol. 32, pp. 1-24, 2022.
3. M. Panahi, N. Sadhasivan, H. R. Pourghasemi, F. Rezaie and S. Lee, "Spatial prediction of groundwater potential mapping based on convolutional," Journal of Hydrology, vol. 588, pp. 1-12, 2020.
4. Y. Teng, Y. Chen, X. Chen, S. Zuo, X. Li, Z. Pan, K. Shao, J. Du and Z. Li, "Revealing the adulteration of sesame oil products by portable Raman," Food Chemistry, vol. 436, pp. 1-12, 2024.
5. G. Celik and M. Ozdemir, "Determination of concrete compressive strength from surface images with," Measurement, pp. 1-15, 2024.
6. I. W. Prastika and E. Zuliarso, "DETEKSI PENYAKIT KULIT WAJAH MENGGUNAKAN TENSORFLOW DENGAN METODE CONVOLUTIONAL NEURAL NETWORK," Jurnal Manajemen Informatika dan Sistem Informasi, vol. 4, no. 2, pp. 84-91, 2021.
7. S. Dewi, F. Ramadhani and S. Djasmayena, "Klasifikasi Jenis Jerawat Berdasarkan Gambar Menggunakan Algoritma CNN (Convolutional Neural Network)," Jurnal Ilmu Komputer, vol. 3, no. 2, pp. 68-73, 2024.
8. K. A. Muhaba, K. Dese, T. M. Aga, F. T. Zewdu and G. L. Simegn, "Automatic skin disease diagnosis using deep learning," Skin Health and Disease, vol. 2, no. 1, pp. 1-12, 2021.
9. F. A. Febriyanti, "IMAGE PROCESSING DENGAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN) UNTUK DETEKSI PENYAKIT KULIT PADA MANUSIA," Jurnal Multidisiplin Saintek, vol. 3, no. 10, pp. 1-12, 2024.
10. Z. Li, M. Paolieri and L. Golubchik, "Inference latency prediction for CNNs on heterogeneous mobile," Journal Performance Evaluation, vol. 165, pp. 1-26, 2024.
11. U. Aulia, I. Hasanuddin, M. Dirhamsyah and N. Nasaruddin, "A new CNN-BASED object detection system for autonomous," Heliyon, pp. 1-22, 2024.
12. M. Ath-Thariq and T. N. Suharsono, "Deteksi Penyakit Kulit Serupa Pada Wajah Berbasis Mobile dengan Metode Convolutional Neural Network," Innovative: Journal of Social Science Research, vol. 3, no. 5, pp. 876-887, 2023.