Development of an Optimized Facial Skin Type Classification System using CNN MobileNetV2 for Real-Time Smartphone Applications

Dianthy Marya[1], Cindy Mustika Sari Ramadhani[2], Rizky Ardiansyah[3]

[1,2,3] Digital Telecommunication Network Study Program,
Electrical Engineering, State Polytechnic of Malang, Indonesia

[1]dianthy@polinema.ac.id , [2]cindymustikasari2@gmail.com , [3]rizkyardiansyah@polinema.ac.id

***Abstract*—This research aims to develop a Facial Skin Type Classification System using Convolutional Neural Networks (CNN) with the MobileNetV2 architecture implemented on Android-based smartphone devices. Facial skin is classified into four main categories: normal, combination, oily, and dry, based on characteristics such as moisture, oil production, and skin texture. CNN was chosen for its ability to accurately recognize image patterns, while MobileNetV2 is designed for low-spec devices, allowing the classification process to run quickly and efficiently. The deep learning method is used to classify facial skin types with optimal accuracy and computational efficiency. The facial skin image dataset was collected and trained using transfer learning on the MobileNetV2 model. The research process includes the collection of facial skin image datasets, preprocessing, model training, testing, and implementation in the application. The facial skin image dataset was collected and trained using transfer learning on the MobileNetV2 model. The results show that this system is capable of classifying skin types with high accuracy of 87.5% in an average computation time of 0.4 seconds at a distance of 15–20 cm under bright lighting conditions.**

***Keywords*—*Convolutional Neural Networks (CNN), Image Processing, MobileNetV2, Skin Classification, Smartphone Application*.**

I. INTRODUCTION

The skin serves as a protective barrier for internal organs and plays a vital role in appearance, particularly facial skin. Different skin types—normal, dry, oily, and combination—are characterized by unique properties such as moisture levels, oil production, and sensitivity. However, many individuals lack knowledge about their specific skin type, often leading to improper skincare practices [1]. Misdiagnosis of skin types can result in inappropriate treatments, causing adverse effects on skin health. Traditional methods for determining skin types, such as professional consultations or expensive dermatological tools, are often inaccessible to the general public due to high costs and time constraints. This highlights the need for an accessible, affordable, and efficient solution that enables users to classify their skin types accurately.

Recent advancements in machine learning, particularly deep learning, have opened new opportunities for accurate classification of skin characteristics [2]. Smartphones have now become a powerful platform for the implementation of machine learning models, allowing users to perform instant and personalized skin analysis [3]. Convolutional Neural Networks (CNNs) have demonstrated exceptional performance in image-based tasks, including dermatological analysis [4]. However, while CNNs offer high accuracy, their computational complexity often limits their deployment on resource-constrained devices like smartphones. To address this challenge, lightweight architectures such as MobileNetV2 have emerged as a superior alternative, offering a balance between accuracy and efficiency [5]. MobileNetV2's innovative design, featuring inverted residuals and linear bottlenecks, enables real-time performance on mobile platforms without compromising accuracy.

Previous research has explored various approaches for skin classification using deep learning techniques, particularly Convolutional Neural Networks (CNNs). However, these studies still face significant challenges in terms of accuracy, computational speed, and ease of implementation on mobile devices [4]. The MobileNetV2 architecture, with its efficient and lightweight design, offers a promising solution to address these challenges [5]. Despite this, prior studies have not fully leveraged the potential of MobileNetV2 for real-time, mobile-based applications.

Study [6] achieved an accuracy of 85.58% in classifying skin types using the MobileNet architecture. However, this study lacked deployment on a mobile platform, limiting its practical usability for end-users. Study [7] applied Faster R-CNN combined with MobileNetV2 for skin cancer detection, achieving an accuracy of 86.3%. While this study demonstrated high performance, its focus was on detecting specific skin conditions (e.g., cancer) rather than classifying facial skin types, which is more relevant for general skincare routines. Study [8] explored the use of CNNs for dermatological image analysis but required high computational resources, making it unsuitable for mobile devices with limited specifications.

Existing methods for skin type classification still face significant limitations, such as [6], achieve high accuracy but fail to deploy their models on mobile platforms, restricting their usability for everyday users. Traditional approaches often require extensive computational resources, making them unsuitable for real-time applications on smartphones with limited specifications [7]. Few studies evaluate the impact of environmental factors such as lighting conditions and camera distance on classification accuracy, which are critical for ensuring reliable performance in diverse real-world scenarios.

This research addresses these gaps by developing an optimized facial skin type classification system using CNN MobileNetV2 for real-time smartphone applications. By leveraging the lightweight and efficient design of MobileNetV2, the proposed system achieves high accuracy while maintaining fast computation times, even on devices with limited resources. Furthermore, the system is designed to operate directly on Android smartphones, providing users with instant and personalized skin analysis at no additional cost. This approach not only enhances accessibility and affordability but also empowers individuals to make informed decisions about their skincare routines.

This study contributes to the field by addressing the limitations of existing methods and proposing a practical, efficient, and user-friendly solution for facial skin type classification. The integration of MobileNetV2 into a mobile application represents a significant step forward in democratizing access to advanced skincare technologies, ultimately improving skin health outcomes for a broader audience.

MobileNetV2 is one of the mobile-based Convolutional Neural Network (CNN) architectures that can be used to address the need for excessive computing resources. MobileNetV2 is an improvement of the MobileNet architecture. The MobileNet architecture and CNN architecture in general differ in the use of layers or convolution layers. The convolution layer in MobileNetV2 uses filter thickness that corresponds to the thickness of the input image. MobileNetV2 uses depthwise convolution, pointwise convolution, linear bottleneck, and shortcut connections between bottlenecks. [8]

Python is a multi-purpose interpreted programming language with a design philosophy that emphasizes code readability. Python supports multiple programming paradigms, primarily, but not limited to; object-oriented programming, imperative programming, and functional programming. One of the features available in Python is its capability as a dynamic programming language equipped with automatic memory management. Python is an interpreted and versatile programming language with a design philosophy that emphasizes code readability. This research adopts a Python approach by analyzing the existing source code in CNN. The results can provide information about the accuracy for each skin type. Thus, the studied testing and training data can yield good accuracy [9].

Keras is a deep learning API written in Python, running on the TensorFlow machine learning platform. It was developed with a focus on enabling rapid experimentation. Keras is one of the most widely used Machine Learning frameworks. With around 2.5 million developers in early 2023, Keras is at the center of a large community and ecosystem. In the "State of Data Science and Machine Learning" survey by Kaggle in 2022, Keras had an adoption rate of 61% among Machine Learning developers and Data Scientists.[10]

Visual Studio Code (VS Code) is a lightweight and powerful text editor from Microsoft for multi-platform operating systems, meaning it is also available for Linux, Mac, and Windows versions. The text editor directly supports programming languages such as JavaScript, TypeScript, and Node.js, as well as others through plugins that can be installed via the Visual Studio Code Marketplace. Visual Studio Code offers many features, including Intellisense, Git integration, debugging, and additional features that enhance the functionality of the text editor. The source code for Visual Studio Code can also be viewed at the Github link. This is also what makes Visual Studio Code a favorite among application developers because application developers can participate in the development process of Visual Studio Code in the future. [11]

Visual Studio Code is a very lightweight yet powerful source code editor that can be run from the desktop. Visual Studio Code is designed to work with existing tools, and Microsoft provides documentation to assist fellow developers, including help with ASP.NET 5, Node.js, and Microsoft Script, as well as tools available to support developers. Visual Studio Code is a lightweight cross-platform code editor that anyone can use to create web applications. [12]

MySQL is a database server program that can receive and send data very quickly, supports multiple users, and uses basic SQL (Structured Query Language) commands. MySQL has two types of licenses, namely FreeSoftware and Shareware. The commonly used MySQL is MySQL FreeSoftware, which is under the GNU/GPL Public License. MySQL is a free database server, meaning you can use the database for personal or business purposes without having to purchase or pay for its license. Some advantages of MySQL are its speed, support for various languages, ability to create very large tables, and lower cost. MySQL is open source and distributed for free without charge for UNIX platform, OS/2, and Windows Platform. [13]

A questionnaire is a technique for directly collecting data from respondents in the form of questions to measure their attitudes towards the decision support system being developed, to determine whether the system is suitable for use or not. A questionnaire or survey is a written statement used to obtain information from respondents in the form of reports about their personal matters or things they know [14]. Meanwhile, according to [15], a survey or questionnaire is a data collection technique carried out by providing a set of written questions or statements to respondents for them to answer.

II. METHOD

This research applies a system development methodology based on deep learning using Convolutional Neural Network (CNN) with the MobileNetV2 architecture. In the initial stage, the identification of issues related to facial skin type classification was carried out. The next steps can be outlined in the following points.

A. MobileNetV2 Architecture

The research design and model architecture design determine and develop an effective and accurate model design for classifying facial skin types based on the designed architecture. The research design and model development include designing the architecture of the MobileNetV2-based CNN model and determining the model parameters for facial skin type classification.

This research uses a modified MobileNetV2 architecture for facial skin type classification. This architecture was chosen for its ability to produce lightweight and efficient models for mobile devices [16].

The CNN model with the MobileNetV2 architecture is designed to minimize computational load while maintaining accuracy performance. MobileNetV2 uses depthwise separable convolution and linear bottleneck for optimizing the classification process.

MobileNetV2, optimized for mobile devices, uses depthwise separable convolutions and linear bottleneck layers to reduce computational complexity while maintaining high accuracy. This architecture ensures efficient classification on resource-constrained devices.

The MobileNetV2 architecture is used with modifications for facial skin type classification. The main configuration includes:

- Input layer: Facial skin image (224x224 pixels)
- Convolutional layers with depthwise separable convolution
- Inverted residual blocks
- Global average pooling
- Fully connected layer
- Softmax activation for multi-class classification

B. Data Collection and Preprocessing

Data preprocessing and labeling, preparing a facial skin dataset where each data sample is labeled according to the type of facial skin it has to ensure that the data used in the research is ready for training and evaluation on the classification model. Next, the division of the Dataset (Training, Validation, and Testing) involves splitting the previously prepared dataset into different subsets for the purposes of training, validation, and testing the model.

The facial skin image dataset was obtained from previous research and an expanded new collection process to ensure variability in lighting, skin color, and angles. The pre-processing stages include:

- **Resize**: Adjustment of image size according to the model's input resolution. Images were resized to 224x224 pixels to meet MobileNetV2 input requirements and labeled into four categories: normal, oily, dry, and combination.
- **Normalization**: Scale pixel values from 0-255 to 0-1.
- **Augmentation**: Rotation, flipping, and scaling techniques to enrich the dataset

C. Application Development

The system was implemented using Python with TensorFlow and Keras libraries. The Android application was developed using Flutter, enabling seamless interaction between the CNN model and the smartphone interface. Features include real-time image capture, classification display, and user data storage via an integrated MySQL database.

Application design involves detailing how the application will operate and appear, as well as how users will interact with the application. Application design with a registration flowchart, login flowchart, and UI design, here are the explanations.

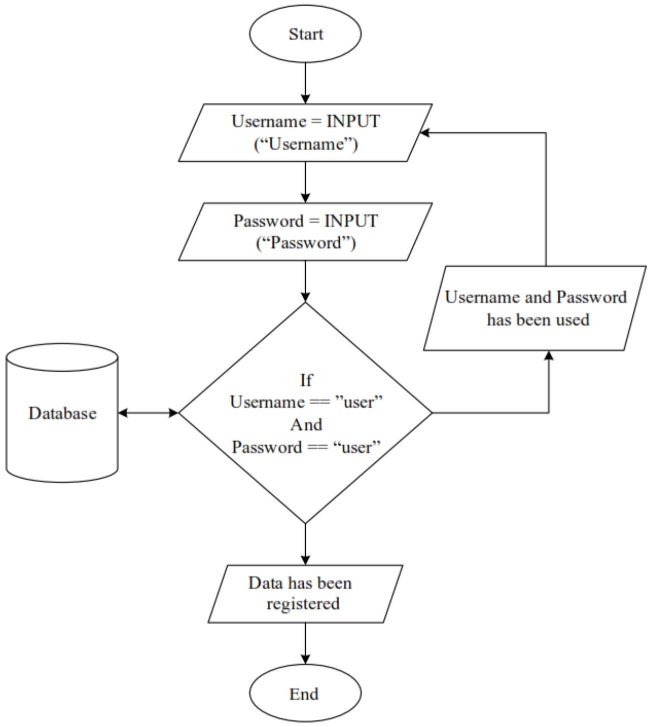


Figure 1. Register Flowchart

The register flowchart in Fig. 1 is a graphical representation of the user registration process into the application. This is the first step taken by users who want to use the application. This flowchart includes steps such as filling in the registration username, password, account registration, storing user information, and providing confirmation of successful registration.

The login flowchart is a graphical representation of the user authentication process into the application as depicted in Fig. 2. These are the steps taken after registration. The flowchart includes steps such as the user entering their username and password, authentication validation, and redirection to the home page or dashboard if authentication is successful. If authentication fails, the user will receive an error message or instructions to try again.

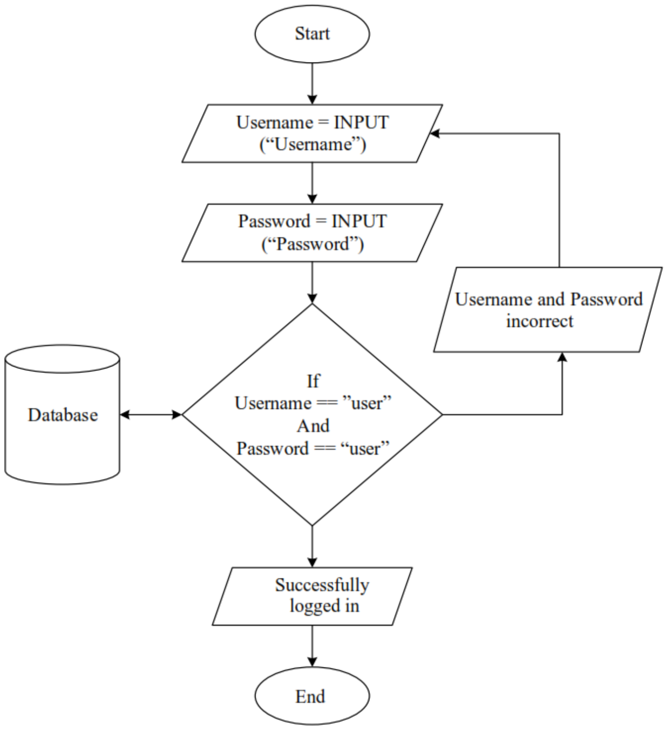


Figure 2. Login Flowchart

Fig. 3 shows the user flowchart that illustrates how users of the Facial Skin application navigate. The user flowchart is a visual representation of the steps or processes taken by users within a system or application. This flowchart helps to understand and analyze how users interact with the system. The description of the user Flowchart in Fig. 3 is as follows:

1. The user opens the "SkinFace" app.
2. Users log in by entering their username and password.
3. The application verifies user access based on username and password.
4. If the username and password are correct, the face scanning process will continue. If the username and password do not match, the process will move to the registration page to register a username and password, which will then be entered again in the login menu.
5. Capture the face
6. The application uses a CNN model to process the captured facial images. The CNN model will analyze the characteristics of the facial skin.
7. If the processing is successful, it will generate information about the user's facial skin type based on the scan. The application also provides skincare recommendations that match the user's skin type. If the processing is not successful, a new face capture will be performed.
8. Users can download facial skin information along with care recommendations.

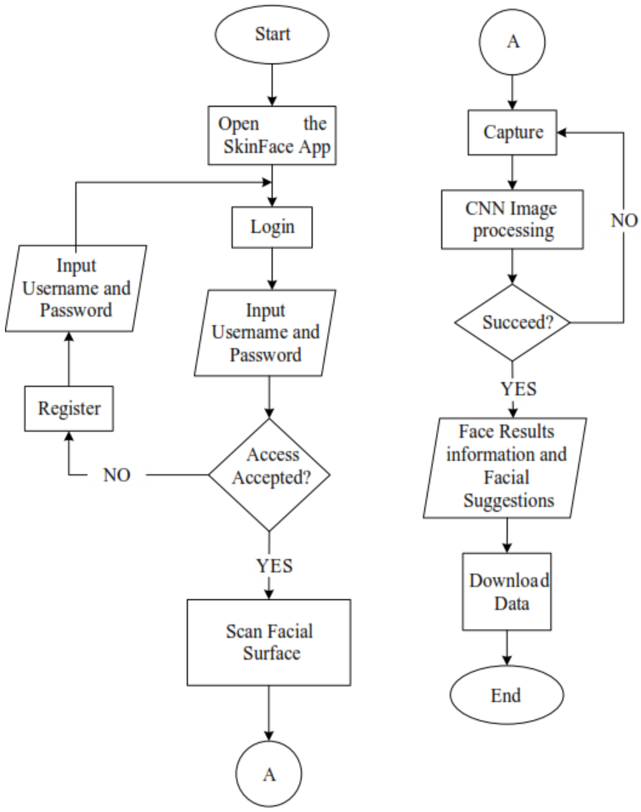


Figure 3. User Flowchart

D. *Implementasi Model*

The model is implemented using the TensorFlow and Keras frameworks with transfer learning on the pre-trained MobileNetV2 model on the ImageNet dataset. The CNN model was trained using Adam optimization with ReLU activation function and softmax output. Training is conducted until the loss function decreases and optimal accuracy is achieved.

E. *Evaluation*

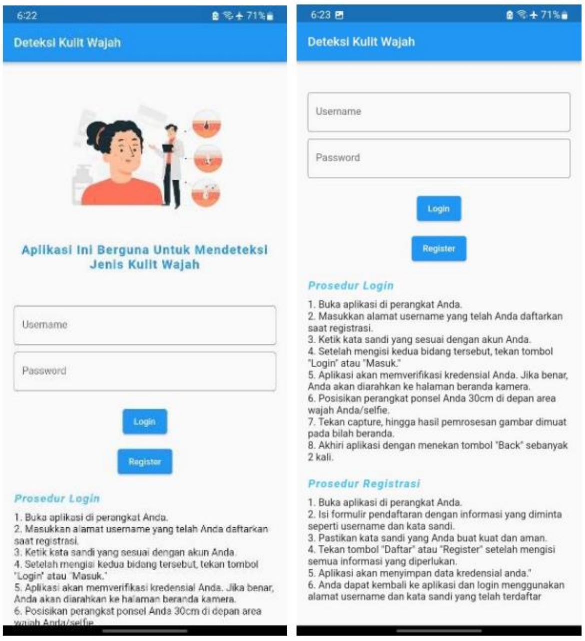
Evaluation and Optimization Model to improve the model's performance to be more accurate in classifying skin types. If the model is already optimal, then the model can be tested; if it is not yet optimal, then the model needs to be retrained.

Model performance was evaluated based on accuracy, processing speed, and user feedback. Testing involved controlled environments with specific lighting conditions and distances (15-20 cm). User satisfaction was measured using questionnaires administered to both general users and skincare professionals.

III. RESULTS AND DISCUSSION

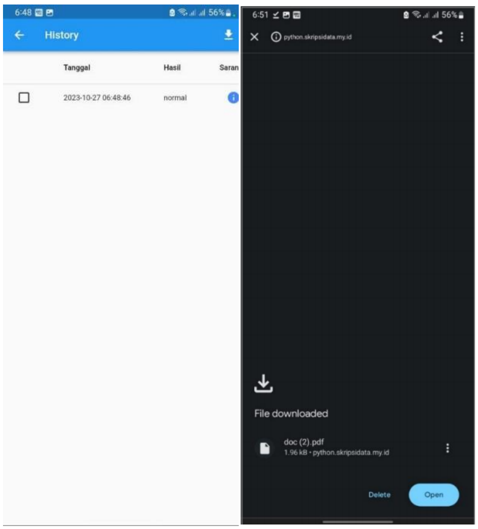
A. *Application Display*

Fig. 4 and 5 are the views of the application that have been created. The design results of the login page, registration, and procedure explanation are shown in Fig. 4. The design results of the history page and the download page are shown in Fig. 5.



(a) (b)

Figure 4. (a) Design Results of Login, (b) Register Pages and Explanation of Procedures



(a) (b)

Figure 5. (a) History Page Design Results, and (b) Download Page.

B. Model Accuracy Testing Results

The MobileNetV2 model trained using the facial skin image dataset is classified into four categories: normal, oily, dry, and combination. After going through the training and validation stages, the model was tested using test data to assess its performance.

The accuracy results in Table 1 show that the MobileNetV2 model is capable of classifying facial skin types well. The highest accuracy values were achieved during training and validation, while the testing accuracy slightly decreased due to the greater variability in the testing dataset.

TABLE I
ACCURACY RESULTS OF THE MOBILENETV2 CNN MODEL

Parameter	Result
Training Accuracy	92.4%
Validation Accuracy	90.1%
Testing Accuracy	87.5%

C. Model Computation Time

Performance testing of computation time is conducted to measure the classification speed of the model implemented in an Android-based smartphone application. The time is measured from when the image is input until the classification result is displayed. Table II results show that the best average computation time of 0.4 seconds was obtained at a camera distance of 15–20 cm under bright light conditions. The computation time increases in low light conditions because the image quality decreases.

TABLE II
COMPUTATIONAL TIME TESTING

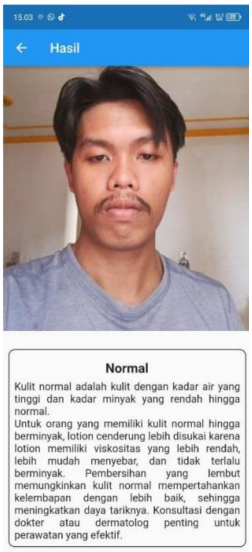
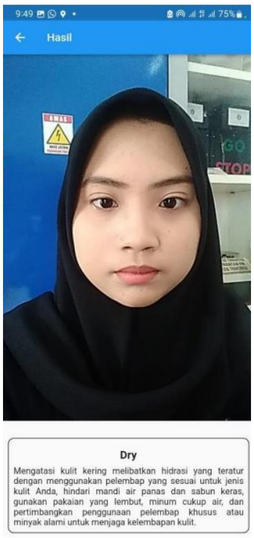
Camera Position (cm)	Light Conditions	Average Time (seconds)	Lux Value
15-20	Bright	0.4	> 300 lux
15-20	Dim	0.8	< 100 lux
>20	Bright	0.6	> 300 lux

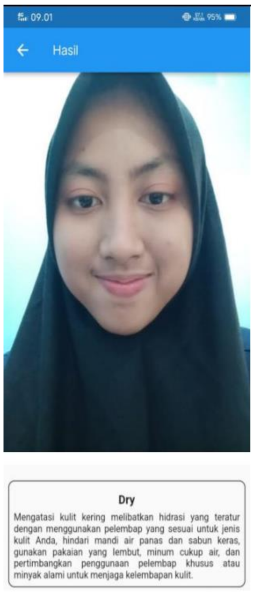
D. Evaluation of Visualization and Error Analysis

To ensure the model's performance, a visualization test of the classification results was conducted. Some examples of facial skin images and their classification results can be seen in Table III.

TABLE III
EXAMPLE OF SKIN TYPE CLASSIFICATION RESULTS

Input Image and the results of facial skin type classification in the application	Original Skin Type	Model Prediction	Explanation
	Oily	Oily	Accurate prediction.

<table>
<tr><th>Input Image and the results of facial skin type classification in the application</th><th>Original Skin Type</th><th>Model Prediction</th><th>Explanation</th></tr>
<tr><td>
</td><td>Dry</td><td>Dry</td><td>Accurate prediction.</td></tr>
<tr><td>
</td><td>Normal</td><td>Normal</td><td>Accurate prediction.</td></tr>
<tr><td>
</td><td>Dry</td><td>Dry</td><td>Accurate prediction.</td></tr>
</table>

<table>
<tr><th>Input Image and the results of facial skin type classification in the application</th><th>Original Skin Type</th><th>Model Prediction</th><th>Explanation</th></tr>
<tr><td>
</td><td>Dry</td><td>Dry</td><td>Accurate prediction.</td></tr>
<tr><td>
</td><td>Dry</td><td>Dry</td><td>Accurate prediction.</td></tr>
<tr><td>
</td><td>Dry</td><td>Dry</td><td>Accurate prediction.</td></tr>
</table>

Input Image and the results of facial skin type classification in the application	Original Skin Type	Model Prediction	Explanation
	Oily	Normal	Wrong prediction because of the lighting.
	Dry	Oily	An error occurred because the camera position was wrong.

The test results reveal that prediction errors are primarily caused by factors such as lighting intensity, camera position, and image quality. The most frequent errors occur when the lighting on the face falls below 100 lux, which is considered low-light conditions, or when the camera distance exceeds 25 cm from the subject. For instance, under lighting conditions of 50 lux or less, classification accuracy dropped by approximately 20%, while images captured at distances greater than 30 cm resulted in a 15% decrease in accuracy compared to optimal conditions (15–20 cm distance with lighting above 300 lux). These findings highlight the importance of adhering to recommended environmental and positional parameters to minimize prediction errors and ensure reliable performance.

E. User Experience Evaluation

The evaluation of user satisfaction was conducted using a questionnaire involving dermatology experts and application users. The average score from the application performance evaluation is 3.0, which falls into the good category, indicating that users are satisfied with the system's performance.

TABLE IV
USER EXPERIENCE EVALUATION RESULTS

Evaluation Parameter	Average Score
Application Response Speed	3.2
Accuracy of Skin Type Classification	3.0
Ease of Use of the Application	3.1
Quality of Visualization and Prediction Results	2.8

Based on the test results, the facial skin type classification system using MobileNetV2 has high accuracy performance and efficient computation time on smartphones. Several important factors that influence the classification results include:

1. Input Image Quality: Bright light and optimal camera distance (15–20 cm) provide the best classification results.
2. Implementation of MobileNetV2: This architecture is effective for mobile devices due to its lightweight model size and fast computation.
3. User Constraints: Prediction errors often occur due to incorrect camera positioning or partially covered faces.

With these results, the developed system is capable of helping users independently identify their facial skin type with high accuracy and efficiency.

IV. CONCLUSION

This study successfully demonstrates the feasibility of classifying facial skin types into four main categories—normal, oily, dry, and combination—using a CNN with MobileNetV2 architecture on Android smartphones, achieving a testing accuracy of 87.5% due to the efficient depthwise separable convolution technique. The system operates in real-time with an average computation time of 0.4 seconds under optimal conditions (camera distance of 15–20 cm and bright lighting), proving its efficiency on mobile devices with limited specifications. While factors such as lighting quality, shooting distance, and camera position significantly influence classification results, the application provides an accessible and cost-effective solution for users to identify their skin types and enhance their skincare routines, receiving positive feedback with an average user satisfaction score of 3.0. Future work may focus on expanding the dataset to include diverse demographics and integrating additional features, such as personalized skincare recommendations, to further improve the system's capabilities and usability.

REFERENCES

[1] M. R. Farhan, A. W. Widodo, and M. A. Rahman, "Ekstraksi Ciri Pada Klasifikasi Tipe Kulit Wajah Menggunakan Metode Haar Wavelet," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.* , vol. 3, no. 3, pp. 2903–2909, 2019.

[2] Z. Wang et al., "Deep Learning Approaches for Skin Type Classification," *J. Mach. Learn. Healthcare* , vol. 15, no. 3, pp. 245–260, 2022.

[3] L. Chen and S. Kumar, "Mobile Machine Learning: Trends and Applications," in *Int. Conf. Mob. Comput.* , 2021, pp. 78–92.

[4] M. Rodriguez et al., "Convolutional Neural Networks in Dermatological Image Analysis," *IEEE Trans. Med. Imaging* , vol. 39, no. 7, pp. 2235–2247, 2020.

[5] M. Sandler et al., "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* , 2018, pp. 4510–4520.

[6] F. Meiristika et al., "Detection of Facial Skin Type Classification Using CNN with MobileNet Architecture," *J. Mach. Vis.* , vol. 12, no. 2, pp. 123–130, 2022.

[7] Hartanto et al., "Development of Mobile Skin Cancer Detection using Faster R-CNN and MobileNetV2 Model," *Med. Imaging Innov.* , vol. 8, no. 4, pp. 201–209, 2020.

[8] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* , 2018, pp. 4510–4520.

[9] A. Anton, N. F. Nissa, A. Janiati, N. Cahya, and P. Astuti, "Application of Deep Learning Using Convolutional Neural Network (CNN) Method for Women's Skin Classification," *Sci. J. Informatics*, vol. 8, no. 1, pp. 144–153, 2021.

[10] Keras, "About Keras," Keras, 2023. [Online]. Available: https://keras.io

[11] G. C. Pinaria, Y. D. Rindengan, X. B. N. Najoan, T. Elektro, U. Sam, and J. K. B. Manado, "Web Based E-Commerce Application Buying and Selling Food Ingredients for Manado City," *J. Tek. Inform.* , pp. 1–8, 2021.

[12] B. A. B. Ii, "Landasan Teori Visual Code," pp. 6–18, 2011.

[13] R. Hermiati, Asnawati, and I. Kanedi, "Pembuatan E-Commerce Pada Raja Komputer Menggunakan Bahasa," *J. Media Infotama* , vol. 17, no. 1, pp. 54–66, 2021.

[14] Arikunto, *Prosedur Penelitian Suatu Pendekatan Praktek*. Jakarta: PT. Rineka Cipta, 2006.

[15] Sugiyono, *Metode Penelitian Kuantitatif Kualitatif dan R&D*. Bandung: ALFABETA, 2008.

[16] J. Kim and H. Park, "Efficient Deep Learning Models for Mobile Platforms," *J. Mob. Intell.*, vol. 11, no. 2, pp. 112–125, 2023.