Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 3, Juni 2025, Page. 1589-1604
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

Impact of Optimizer Selection on MobileNetV1 Performance for Skin Disease Detection Using Digital Images

Khairul Fathan Habie[1], Murinto*[2], Sunardi[3]

[1]Master Program of Informatics, Universitas Ahmad Dahlan, Indonesia
[2]Informatics, Universitas Ahmad Dahlan, Indonesia
[3]Electrical Engineering, Universitas Ahmad Dahlan, Indonesia

Email: murintokusno@tif.uad.ac.id

Received : May 5, 2025; Revised : Jun 24, 2025; Accepted : Jul 1, 2025; Published : Jul 9, 2025

Abstract

Automatic detection of skin diseases using digital images is a growing field in the application of deep learning in the medical world, especially to help the early diagnosis process. One of the most widely used models is MobileNetV1 because it is lightweight and efficient in image processing. However, the performance of the model is greatly affected by the training configuration, including the type of optimizer used. This study aims to compare the effectiveness of six types of optimizers, namely SGD, RMSprop, Adam, Adadelta, Adagrad, Adamax, and Nadam in training MobileNetV1 models for human skin disease image classification. The model was trained on annotated skin image dataset with predetermined training parameters: batch size 32, learning rate of 0.0001, and 10 epochs. Performance evaluation was performed using accuracy metrics. The results obtained demonstrate that RMSprop performs best, with 99.10% accuracy, 99.14% precision, 99.10% recall, and a 99.10% F1-score. Adadelta showed the lowest performance consistently, with only 22.22% accuracy, 20.34% precision, 22.22% recall, and 18.42% F1-score. This finding confirms that the type of optimizer affects the effectiveness of model training, especially in medical image classification tasks. This research provides empirical insights that are useful in selecting the optimal optimizer for MobileNetV1 model implementation in the healthcare domain.

Keywords : *CNN, Hyperparameter, Image Classification, MobileNetV1, Optimizer, Skin Disease.*

1. INTRODUCTION

Skin disease is a health problem that has a high prevalence in the global community [1]. Early diagnosis of skin diseases is essential to ensure prompt and appropriate treatment to prevent the development of more serious conditions [2]. Diagnosis of skin diseases is currently still dominated by conventional methods, which is by utilizing direct visuals by dermatologists [3]. Diagnosis is done by evaluating clinical symptoms such as the color, shape, size, and distribution of skin lesions [4]. However, this method relies heavily on the expertise and experience of a specialist dermatologist [5], and sometimes requires additional time and cost for in-depth laboratory examinations [6]. Many regions, especially developing countries, experience challenges in the process of diagnosing skin diseases, as the unequal distribution of experts prevents patients from getting fast and proper treatment [7]. This is exacerbated by the increasing number of skin disease cases that occur due to various factors [8], such as climate change, pollution, environmental hygiene, and unhealthy lifestyles [9].

Patients with skin diseases will experience complex challenges if they do not get the right diagnosis and treatment from the start [10]. Skin conditions that do not receive immediate treatment will become more severe [11] [12]. Not only that, psychological impacts such as embarrassment, low self-esteem, and depression are often experienced by sufferers [13], especially when skin conditions occur

P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

in visible areas of the body. In some cases, skin diseases located in vital areas cause sufferers to be reluctant to seek further medical assistance [14]. In the long run, people with skin diseases will experience profound changes, not only in physical conditions but also in mental health and quality of life, even after the disease is cured [15].

Given the complex impact caused by skin diseases, the need for fast and accurate diagnosis is essential, especially in areas with a shortage of dermatologists. The development of artificial intelligence technology, especially within the domain of machine learning [16], has played a pivotal role in enhancing healthcare services, including assisting in disease diagnosis. Machine learning enables systems to extract insights directly from data without requiring explicit programming instructions [17]. Meanwhile, deep learning [18], which is a specialized branch of machine learning, utilizes deep artificial neural networks to detect intricate patterns in data [19]. Convolutional Neural Networks (CNNs) are among the most effective architectures for classification problems, specifically designed to process visual data through an automatic spatial feature extraction mechanism.

MobileNetV1 is one of the CNN architectures known for its lightweight nature while still delivering competitive performance. It was developed to function on devices with limited resources, making it well-suited for mobile-based diagnostic applications or integrated systems in regions with limited healthcare infrastructure [20].

In designing MobileNet-based models, selecting the appropriate optimizer algorithm and fine-tuning the hyperparameters are key elements that significantly impact the model's overall performance [21]. Optimizers are essential in the training process of neural networks, as they govern how the model's weights are adjusted to reduce error [22]. Various types of optimizers such as Stochastic Gradient Descent (SGD), Adam, RMSprop, Adagrad, Adadelta, Nadam, and Adamax each offer their own advantages in terms of convergence speed, training stability, and the ability to avoid local minima traps.

Several previous studies have discussed the use of CNN for skin disease classification. For example, the study "Deep Learning Based Automated Diagnosis of Skin Disease Using Dermoscopy" [23] used CNN to classify skin disease types into seven classes and obtained a model with the best accuracy reaching 96%. Research entitled "Atopic Dermatitis and Psoriasis Skin Disease Classification by Using Convolutional Neural Network" [24] aims to classify two types of commonly encountered skin diseases, namely atopic dermatitis and psoriasis. This study compares CNN architectures and obtains the best accuracy results using the InceptionV3 architecture with a value of 82%. The research entitled "Six Skin Disease Classification Using Deep Convolutional Neural Network" [25] aims to classify skin disease types into six disease classes using the CNN algorithm and obtained an accuracy of 81.75%. On the other hand, studies on the effect of determining the optimizer on CNN models, especially MobileNetV1 for human skin disease classification tasks, are still relatively limited. Therefore, this study aims to address the existing research gap by thoroughly evaluating the impact of different optimizers on enhancing the performance of the MobileNetV1 model for skin disease image classification.

The novelty of this research lies in the comprehensive exploration of seven popular optimizer types, namely SGD, RMSprop, Adam, Adadelta, Adagrad, Adamax and Nadam, that are systematically compared in the context of MobileNetV1 for skin disease diagnosis. This research also focuses on analyzing the effect of optimizer selection on training stability and model accuracy, two important aspects that have rarely been reviewed in depth in previous studies. This research differs significantly from earlier works in that it focuses on integrating a lightweight CNN architectures such as MobileNetV1 with a thorough evaluation of several optimizers, as well as direct application to the medical skin disease image classification domain. Most of the previous studies have only focused on improving accuracy through model architecture selection or data preprocessing stages, while the optimization aspect through optimizer selection has not been explored in a structured manner. It is

Vol. 6, No. 3, Juni 2025, Page. 1589-1604
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

assumed that optimizer variations make a significant contribution to the model, so the selection of the right optimizer can be the key in optimizing the performance of the MobileNetV1 model in the task of skin disease image classification. With this approach, the research is expected to be able to produce more precise technical guidance in building MobileNet-based early diagnosis systems, as well as enrich the literature on lightweight CNN model optimization for medical applications.

2. METHOD

This study focuses on analyzing human skin images as the main input for computational processing. For the classification task, it employs the convolutional neural network approach by leveraging the MobileNetV1 architecture, which is recognized for its efficiency and suitability in lightweight deep learning applications. A learning rate of 0.0001 is selected as a key hyperparameter for comparative evaluation across experiments. All computational processes and model training activities are conducted on the Google Colab platform, which provides cloud-based GPU resources. Additionally, the study leverages a variety of image processing libraries to enhance data preprocessing, model implementation, and result visualization.

2.1. Research Steps

Beginning with the acquisition of human skin disease images, Figure 1 shows the research workflow, first preprocessed by splitting the dataset into three main subsets: training, validation, and testing. Images are homogenized and are converted into tensor form before being used for model training. Built on the MobileNetV1 architecture, the model is trained using SGD, RMSprop, and others among other optimizers. The model is tested following the training phase with reference to the validation set. The most effective optimizer was determined based on accuracy metrics, highlighting the CNN model's ability to accurately classify different types of human skin diseases.

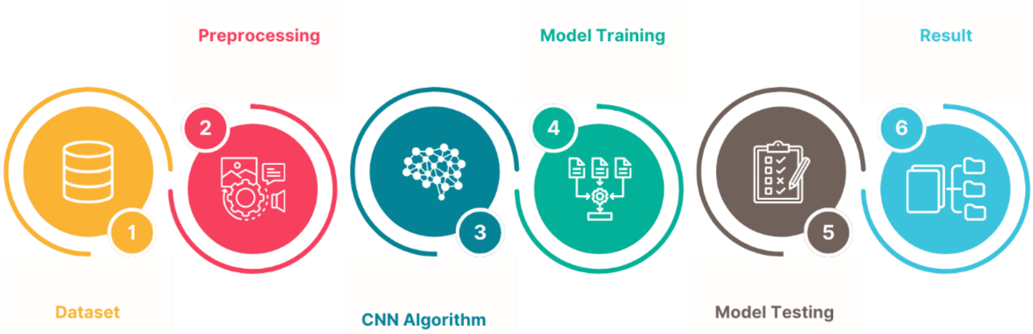


Figure 1 Research Steps

2.2. Dataset

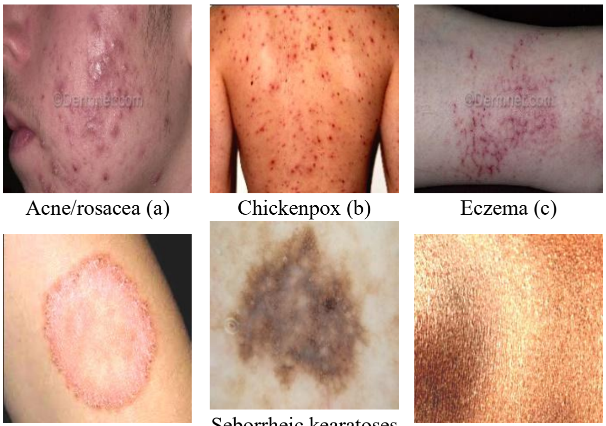
This study uses a dataset of images of skin disease types obtained from Kaggle, as illustrated in Figure 2. This dataset contains 3,295 images categorized into six classes, namely acne/rosacea, chickenpox, eczema, ringworm, seborrheic keratoses, and normal skin. Detailed distribution of the number of each class can be seen in Table 1.

Table 1 Class and number of images in the datasets

Class	Number of Image
Acne/Rosacae	580
Chickenpox	371
Eczema	877
Ringworm	370
Seborrheic Keratoses	434

P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 3, Juni 2025, Page. 1589-1604
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

Class	Number of Image
Normal/Healthy	663
Total	**3,295**



Acne/rosacea (a) Chickenpox (b) Eczema (c)

Ringworm (d) Seborrheic kearatoses (e) Normal (f)

Figure 2 Image of Skin Disease

2.3. Preprocessing

Preprocessing is one of the most crucial stages in preparing images for use in machine learning models [26]. Split data is the initial phase in this procedure, which entails splitting the datasets into three primary groups, 80% for training, 10% for validations, and 10% for testing [27]. Furthermore, argumentation is carried out with the aim of increasing the size of the dataset so that it becomes richer by transforming the image [28]. The next step is resizing, which is done to ensure a uniform image size before training. This is very important because differences in image size can affect the results [29]. Figure 3 (a) shows an image that is 720x472 pixels converted to 224x224 pixels in Figure 3 (b). The last stage of preprocessing is normalization by changing the value of each tensor by multiplying by 1/255. This aims to make the model more efficient [30].

(a) (b)

Figure 3 Resize Image, Before Resize (a), After Resize (b)

2.4. CNN Algorithm

CNN is a form of artificial neural network that is very effective and often used in classification tasks on objects, as shown in Figure 4. CNN works well because of its ability to recognize patterns and features from input data, such as images, through a series of interconnected layers that learn together. The CNN algorithm consists of three main parts, namely the convolution layer, the pooling layer, and the fully connected layer, each of which has a specific role in the training and pattern recognition process

Vol. 6, No. 3, Juni 2025, Page. 1589-1604
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

[31]. The convolution layer is tasked with convolving the input and extracting important features, while the pooling layer serves to reduce the dimensionality of the data and retain important information. Finally, the Fully Connected Layer is responsible for classification based on the features extracted and compacted by the previous layers. Each layer performs training and produces outputs that are then used as inputs in the next layer, forming a deep and iterative learning process [32]. With this structure, CNNs are able to provide excellent performance in various classification and object recognition tasks.

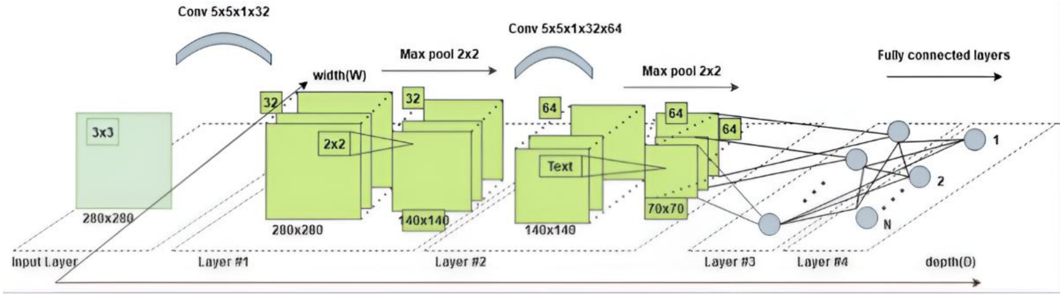


Figure 4 Convolutional Neural Network [source: Researchgate.net]

MobileNet is a neural network model designed to be suitable for computing in resource-constrained environments [33]. It adopts an efficiently structured depth convolution architecture to produce a lightweight and low-complexity model, making it suitable for mobile phone use. In the MobileNet architecture, the convolution process begins with feature map denoted as F, which has dimensions DF × DF × M. Here, DF indicates the spatial resolution of the input, while number of input channels denoted by M. After applying the convolution operation, the layer generate an output feature map represented as G, with dimension DG × DG × N, where DG is the output spatial resolution, while N is the number of output channels.

In the depthwise convolution architecture, there are two main stages: first, the depthwise convolution layer, which applies one filter to each input channel individually, and second, the pointwise convolution, which is responsible for combining the results from depthwise through a 1×1 convolution operation. Equation (1) is used to describe this depthwise convolution mechanism. An illustration of the concept of depthwise and pointwise convolution is shown in Figure 5.

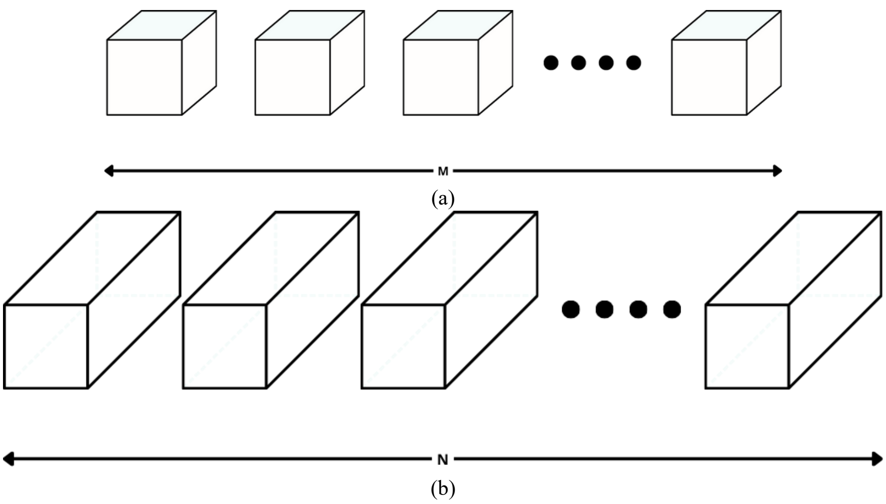


Figure 5 MobileNet Architecture Convolutional Techniques, Depthwise Convolution (a), Pointwise Convolution (b)

Vol. 6, No. 3, Juni 2025, Page. 1589-1604
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

Overall, MobileNet factors the standard convolution operation into a combination of depthwise and pointwise convolutions. Each input channel is processed using one filter in the depthwise stage, then the entire result is combined using pointwise convolution. After that, the output is separated into two layers, namely the filtering stage and the merging stage, both of which remain based on the principle of depthwise convolution.

$$\widehat{G_{k,l,m}} = \sum_{i,j} \widehat{K_{\iota,J,m}} F_{k+i-1,l+j-1,m} \qquad (1)$$

2.5. Model Training

Model training is the phase in which the developed model is trained using data that has undergone preprocessing. During this stage, the model is exposed to the training data to learn patterns and relationships within it. To maintain consistency, all models are trained using identical parameter settings, with the only variation being the type of optimizer applied. Table 2 shows the parameters used in the model.

Table 2 Training Parameter Configuration

Parameter	Value
Batch Size	32
Learning Rate	0.0001
Activation	Softmax
Epoch	10

2.6. Hyperparameter

Developing machine learning models necessitates careful consideration of hyperparameters at each stage. Hyperparameters are predefined settings that are established prior to training and remain constant throughout the training process [34]. Unlike model parameters, which are calculated from the dataset, hyperparameters require prior evaluation of the complexity and requirements of the model along with the problem being solved in question [35]. These hyperparameters significantly affect learning behaviors, how quickly the model converges or adapts, its final performance level, and its capability to specialize to unseen data [36]. A few of the commonly accepted ones include learning rate, momentum in optimization, neural network architecture (number of layers and neurons), batch size, and dropout used for regularization [37]. It, however, needs to be defined that setting incorrect hyperparameters can be counterproductive. Therefore, methods such as grid search, random search, or Bayesian-based optimization to calculate the right hyperparameters are essential for predictive analysis.

2.7. Optimizer

An optimizer, also known as an optimization algorithm, is a crucial element in the machine learning training process by updating the model wights according to the gradients derived from the loss function [38]. Its primary objective is to determine the most effective combination of weights that minimizes the model's error [39].

This study specifically uses only the learning rate of 0.0001, while other parameters in each optimizer use the default values provided from the TensorFlow library. The selection of this value aims to maintain the consistency of each model to be compared.

Various optimizer algorithms have been developed to improve the efficiency and convergence speed of artificial neural network training. Stochastic Gradient Descent (SGD) performs iterative parameter updates using random subsets of training data, making it very efficient for large-scale datasets, although it takes longer to reach the optimal point [40]. RMSprop overcomes the drawbacks of SGD through dynamic learning rate adaptation via the exponential moving average of squared gradients,

Jurnal Teknik Informatika (JUTIF) Vol. 6, No. 3, Juni 2025, Page. 1589-1604
P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

thereby making updates stable and convergence quicker [41]. Adaptive Moment Estimation (Adam) accumulates the advantage of both momentum and RMSprop by estimating the first-order moment (average gradient) in addition to the second-order moment (uncentered variance) for more stable and quicker parameter optimization [42]. The Adaptive Gradient Algorithm (AdaGrad) adjusts the learning rate for every individual parameter based on the cumulative sum of past gradients, making it especially well-suited for sparse data sets, but it often suffers from an ever-decreasing learning rate. To address this drawback [43], AdaDelta introduces an exponential decay factor aimed at limiting the accumulation of previous gradients, thus maintaining the adaptability of the learning rate without manual tuning [44]. Adamax, being a variation of the Adam optimization algorithm, employs infinity norm during the update step, which enhances numerical stability under circumstances that involve large gradients or large datasets [45]. Moreover, Nestrov-accelerated Adaptive Moment Estimation (Nadam), integrates Nesterov's momentum technique with the adaptive learning approach of Adam, enabling parameter updates that account for anticipated future gradients. This results in faster convergence and improved training efficiency [46].

2.8. Model Testing

Accuracy serves as one of the most widely applied metrics for evaluating the performance of classification models. It describes the extent to which the model can correctly predict the label or class of the data by calculating the proportion of correct predictions compared to the total number of predictions made. Moreover, as the main focus of this research is the comparative evaluation of the use of optimizers on MobileNetV1 performance, the use of accuracy allows for a consistent and simple comparison between models without the interpretation complexity of other metrics. Although metrics such as precision, recall, or F1-score are also important, especially in the case of data imbalance, accuracy remains a good choice in this context as it provides a fairly representative global performance indicator of a model's classification effectiveness. The accuracy metric ranges between 0 and 1, with a score of 1 representing ideal classification performance, meaning the model has correctly predicted all instances without any errors [47].

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{2}$$

3. RESULT

This study explores the performance of deep learning models, particularly MobileNetV1, in classifying images of human skin diseases by employing a variety of optimization methods. The dataset comprises several types of skin lesions, each characterized by unique visual traits such as color differences, textures, and growth patterns. All models were trained using consistent settings, including a batch size of 32, a learning rate set to 0.0001, and a total of 10 training epochs. Seven optimizers were tested in this experiment, namely SGD, RMSprop, Adam, Adadelta, Adagrad, Adamax, and Nadam. The core aim of this study is to evaluate the impact of each optimizer type on the model's classification performance.

In general, the results obtained, as shown in Figure 6, show that the choice of optimizer has a significant impact on model performance, even when all other variables are strictly controlled. The RMSprop optimizer achieved the highest overall performance, with accuracy, precision, recall, and F1-score values all reaching 99.09%. The success of RMSprop is attributed to its ability to adaptively adjust the learning rate based on the exponential mean of the squared gradient. This strategy allows RMSprop to maintain stable and efficient learning even on lightweight architectures such as MobileNetV1. In the context of skin disease classification, where visual variations between classes can be subtle, learning stability is key to detecting meaningful features.

P-ISSN: 2723-3863
https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

Optimizers from the Adam family also performed very well, with Nadam being the best variant among the three. Nadam recorded an accuracy of 98.21%, followed by Adamax (97.91%) and Adam (97.60%). The advantage of Adam and its derivatives lies in using a combination of momentum and gradient scaling to update the model weights. Nadam adds a Nesterov momentum component that provides more accurate gradient estimates at points that have been accelerated by momentum. This makes the model more sensitive to changes in the landscape of the loss function and speeds up the convergence process without causing overshoot. In the context of medical images, especially skin diseases, this ability allows the model to learn to distinguish lesions that are visually similar but belong to different categories clinically.

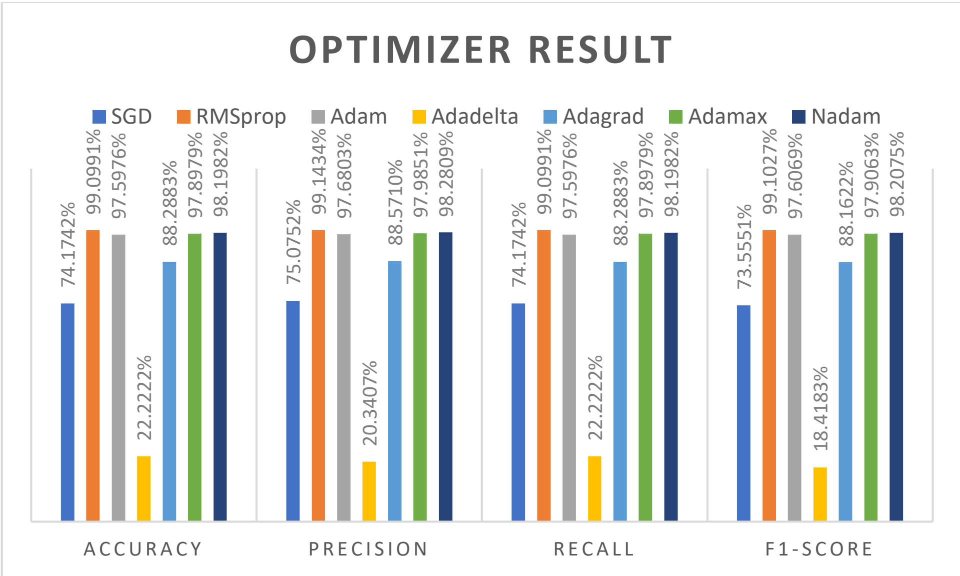


Figure 6 Model Training Result

Adamax is a significantly better version of standard Adam as it employs infinite norms for parameter changes. The precision, recall, and F1-score values achieved by Adamax are consistently around 97.9%, indicating that the infinite norm approach provides robustness against large gradient anomalies and improves learning stability. Adam itself remains a strong choice, but its performance is slightly below the other two variants due to its sensitivity to hyperparameter values that may be less than optimal for these conditions.

Adagrad showed decent performance, achieving an accuracy of 88.29% and an F1-score of 88.16%. This optimizer dynamically modifies the learning rate for each parameter based on how often it gets updated. Parameters that are frequently updated experience a gradual reduction in their learning rates. Such a mechanism is particularly useful in scenarios where certain features appear infrequently. Nonetheless, a key limitation of Adagrad is its continuously decreasing learning rate caused by the accumulation of squared gradients, which can significantly slow down training in the later stages. In the case of skin disease classification, where the model must adapt to diverse visual patterns, this reduction in learning rate may restrict the model's capacity to fully refine its feature representations.

Stochastic Gradient Descent (SGD), as the most basic optimization method, recorded lower performance than other adaptive optimizers. The accuracy value of 74.17% and F1-score of 73.56% show that without any adaptive learning rate or momentum mechanism, the model struggles to explore the parameters efficiently within the limited 10 epochs. SGD tends to perform slow parameter updates

Vol. 6, No. 3, Juni 2025, Page. 1589-1604
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

and can get stuck in local valleys of the loss function. In the context of complex skin disease images, this approach is not robust enough to capture an accurate visual representation in a short training time.

Most striking is the performance of Adadelta, which produces the lowest accuracy and F1-score values of 22.22% and 18.41%, respectively. Although Adadelta is designed to overcome the shortcomings of Adagrad by limiting gradient accumulation through an exponential moving window, the experimental results show that in its default configuration, Adadelta is unable to produce meaningful learning on the MobileNetV1 model for this classification task. This shows that Adadelta is very sensitive to the settings of hyperparameters such as ρ and ε, which, if not fine-tuned, can cause the parameter updates to be too small, rendering the learning process almost non-existent.

When compared overall, it can be seen that the more complex and adaptive the mechanism used by the optimizer, the higher the model performance on all evaluation metrics. Optimizers such as RMSprop and Nadam, which not only adjust the learning rate but also add a momentum component, provide excellent results even in a limited number of epochs. This is important in real applications such as skin disease detection, where training time is often limited by computational resources, and high accuracy is required given the sensitivity of clinical decisions made based on the model's predictive results.

In addition, the high precision and recall metrics in RMSprop and the Adam family optimizer demonstrate the model's ability to minimize classification errors, both in the form of false positives and false negatives. This is important in a medical context, as such errors can have a direct impact on patient diagnosis and treatment. The high F1-score values of the best optimizers show that they are able to maintain a balance between the two metrics, which is an important indicator in the evaluation of classification models facing imbalanced or complex data.

Figure 7 presents a set of graphs that illustrate the performance comparison of various optimizers applied to the MobileNetV1 model. Each graph depicts the development of accuracy and loss values during the training process. The four graph lines represent training accuracy (blue), training loss (green), validation accuracy (orange), and validation loss (red).

The experimental results reveal the unique characteristics of each optimizer. Classical SGD shows the smoothest and most consistent loss reduction, with training loss and validation loss moving parallel without significant fluctuations. However, due to its non-adaptive nature, the convergence rate of SGD is relatively slow, so that by the 10th epoch, the validation accuracy only reaches about 82%. However, the small gap between training and validation accuracy (<3%) indicates good generalization ability. Meanwhile, RMSprop, which adjusts the parameter learning rate based on the mean square of the gradient, presents a drastic loss reduction in the first two epochs. Validation accuracy rises rapidly to about 88% by the 5th epoch, although small oscillations in validation loss appear in the middle epochs, indicating fluctuations in the adaptive learning rate that need to be watched out for.

Vol. 6, No. 3, Juni 2025, Page. 1589-1604
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

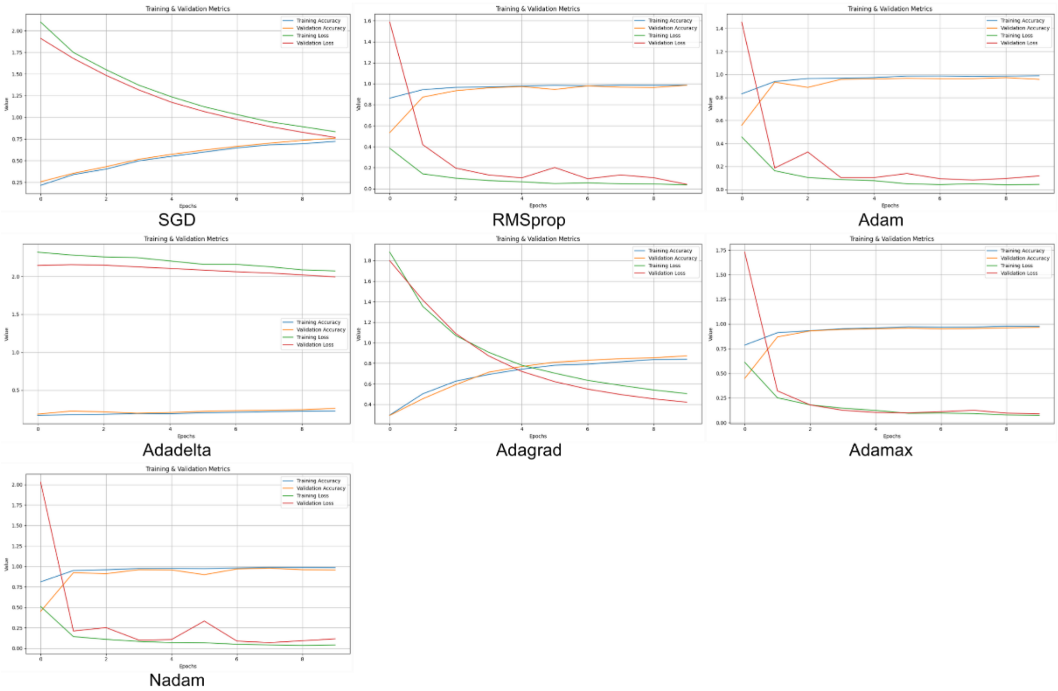


Figure 7 Accuracy and Loss Graph

Adaptive optimizers such as Adam and Nadam stand out as the fastest convergers. Adam, which combines momentum and step-size adaptation, reduces loss aggressively in the first three epochs and then maintains a plateau at the lowest loss level. Adam's validation accuracy curve jumps to 92% in the third epoch and stabilizes until the end of training. Nadam, as a variant of Adam with Nesterov momentum, even showed the sharpest loss reduction in the first and second epochs, followed by the highest validation accuracy of 93%. However, Nadam experienced mild oscillations in validation loss in the 4th and 6th epochs due to anticipatory updates from Nesterov, indicating the need for tuning the momentum hyperparameter to dampen fluctuations. Adamax, which is a variant of Adam, experienced a decrease in loss and an increase in accuracy almost equal to Adam, with validation accuracy reaching 91% and less oscillation, making it suitable for models with many parameters.

The adagrad and Adadelta optimizers have limitations in short-duration training. Adagrad regulates the learning rate by considering the accumulation of squared gradient values, is able to reduce the loss quickly in the early epochs but starts to stagnate from the 4th epoch because the learning rate becomes to small. This makes Adagrad's validation accuracy only reach about 80%. Adadelta, while eliminating the need to specify an initial learning rate, proved to be too conservative in the context of MobileNetV1 such that the loss decreased very slowly and plateaued at relatively high loss levels, resulting in validation accuracy stalling at around 75%, indicating relative underfitting.

Comparatively, under short training conditions, Adam and Nadam are recommended as they are able to achieve fast convergence and the highest accuracy in a short time, although they require parameter adjustments to reduce oscillations in Nadam. RMSprop and Adamax also performed well with a small generalization gap and good stability. For tests with more epochs or long-term generalization needs, classical SGD combined with learning rate scheduling and Adamax are effective choices to deepen the exploration of minima and reduce the risk of overfitting. Optimizer selection should consider the trade-off between convergence rate, learning stability, and generalization ability on unseen data.

Vol. 6, No. 3, Juni 2025, Page. 1589-1604
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

Table 3 shows the evaluation results of precision, recall, and f1-score matrices for each class using the best optimizer. Based on the test, the model combining the MobileNetV1 architecture with the RMSprop optimizer obtained the best results by achieving f1-score above 90.00% for all classes. Two classes, chickenpox and ringworm, recorded precision, recall, and F1-score of 100.00%, indicating no misclassification in both conditions. The Normal and Seborrheic Keratoses classes also performed almost flawlessly with a precision of 100.00% and recall of 98.51% and 97.73%, respectively, resulting in a very high f1-score of 99.25% and 98.85%, respectively. For classes with more complex visual variations such as acne and rosacea, the model obtained a precision of 85.94% and a recall of 94.83%, resulting in an f1-score of 90.16%, which shows a small number of false positives but strong detection sensitivity. In the Eczema class, the precision was 95.29% and the recall was 91.01%, resulting in an f1-score of 93.10%, indicating a small number of false positives and false negatives but still within high tolerance limits. Overall, these results confirm the effectiveness of MobileNetV1 optimized with RMSprop in recognizing diverse visual features in skin disease images, especially after selecting the best optimizer from the seven options tested.

Table 3 Result of Precision, Recall, and F1-Score

Class	Precision	Recall	F1-Score
Acne and Rosacae	0.8594	0.9483	0.9016
Chickenpox	1.0000	1.0000	1.0000
Eczema	0.9529	0.9101	0.9310
Normal	1.0000	0.9851	0.9925
Ringworm	1.0000	1.0000	1.0000
Seborrheic Keratoses	1.0000	0.9773	0.9885

Thus, this part of the results confirms that optimizer selection is not only a matter of training efficiency but also has a direct impact on the generalization quality of the model, especially in critical domains such as human skin disease detection. The results of this study indicate that in the context MobileNetV1, RMSprop, Nadam, and Adamax are the most superior options, with a combination of stability, convergence speed, and high prediction accuracy. In contrast, optimizers such as SGD and Adadelta require additional tuning or replacement to be competitive in complex medical image classification tasks.

4. DISCUSSIONS

The findings of this study show that the choice of optimizer type plays a major role in determining the performance of MobileNetV1 models for the human skin disease image classification task. Although all models were trained with a uniform configuration, the evaluation results using accuracy metric showed significant variations. Optimizers RMSprop, Nadam, and Adamax provided superior performance with accuracy exceeding 97%. In contrast, Adadelta showed very low performance with an accuracy of around 22%, indicating its inability to converge the model in this configuration.

The difference in performance between these optimizers is closely related to the way each algorithm updates the weights during the training process. Adam and its variants (Adamax, Nadam) utilize the first and second momentum of the gradient, which speeds up convergence and makes the training process more stable, especially on more complex data, in this case medical images. RMSprop adapts the learning rate dynamically based on the estimated mean square of the gradient, which is helpful in avoiding extreme updates. In contrast, Adadelta, which relies on an exponential averaging window, can suffer from stagnation of weight updates, especially when the learning rate used is not large enough to trigger significant changes. In the context of skin disease classification, where visual patterns can vary greatly between classes, adaptability and update stability are crucial to model performance.

P-ISSN: 2723-3863
https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

This finding is in line with a recent study by Taqiyuddin et al. [48], who compared various optimizers for drone signal classification using CNN. They found that RMSprop and Adam were the best choices, with Adadelta giving the worst performance due to stagnation of weight updates during the training process. In the context of medical image classification, Hussain et al. [49], reported that the combination of Xception architecture with Adam optimizer was able to achieve an accuracy of up to 99.67%, much higher than the combination with SGD. This supports the finding that Adam has advantages in complex image classification tasks, including medical and skin images.

Adadelta's poor performance in this study is also reinforced by the findings of Kumari et al. [50], which stated that optimizer performance is highly dependent on parameters and architecture, where Adadelta failed to show stability and convergence in weed classification using transfer learning. In addition, Sahin et al. [51] found that in monkeypox skin disease classification, Adam and its variants (such as AdamW) showed superior performance, while SGD and Adadelta consistently underperformed.

The high performance of Nadam in this study is also reinforced by the approach that combines Nesterov momentum and learning rate adaptation, which allows for smoother and faster weight updates. Research by Mortazi et al. [52] showed that a Nesterov-based strategy, especially when combined with a cyclic learning rate, can provide more accurate and stable medical image segmentation results.

However, this study comes with several important caveats. The dataset remains small and lacks a wide range of skin conditions, meaning results cannot yet be confidently applied to the larger population. Also, only the MobileNetV1 model was tested, leaving its newer siblings-MobileNetV2 or MobileNetV3-unexamined and their possible benefits untapped. Trying only ten training epochs may also mean that some optimizers have not yet reached peak performance. Finally, issues of data distribution and class imbalance were not analyzed, even though they could introduce bias into the findings.

In application, these findings provide important directions for the development of deep learning-based skin disease diagnosis systems. In real-life situations, especially in regions with limited healthcare infrastructure, choosing the right optimizer, such as Adam or Nadam, can result in a fast, stable, and resource-efficient diagnosis system.

5. CONCLUSION

This study has evaluated the effect of optimizer selection on the performance of MobileNetV1 models in human skin disease image classification. Each model was trained under uniform conditions, including a batch size of 32, a learning rate set to 0.0001, and a total of 10 training epochs, the results show that optimizer selection significantly affects the accuracy, precision, recall, and F1-score of the model. Optimizers such as RMSprop, Nadam, Adamax, and Adam produced the highest performance with accuracy above 97%, while Adadelta showed very low performance and failed to achieve good convergence.

These findings are reinforced by various recent studies that demonstrate the superiority of adaptive optimizers, especially Adam and its variants, in complex image classification and segmentation tasks. Considering the training efficiency and prediction quality, Adam and RMSprop can be recommended as the default optimizers in MobileNetV1 training for skin disease image classification tasks or similar cases.

However, this research has some limitations. The dataset used is limited in number and type of skin diseases, which results in limitations in assessing the performance of the model when applied to clinical data or real situations. In addition, the use of the MobileNetV1 architecture without exploration of other variants, as well as limiting the number of epochs to 10, may not give an overview of the maximum performance of all optimizers tested.

P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

For future research, it is recommended to explore the combination of optimizers with adaptive learning rate setting techniques such as cyclic learning rate or learning rate scheduler, as well as test the model performance on larger and more diverse datasets to improve the generalizability of the model.

ACKNOWLEDGEMENT

This research was supported by Directorate of Research, Technology, and, Community Service (DRTPM) Indonesia.

REFERENCES

[1] M. A. Richard *et al.*, "Prevalence of most common skin diseases in Europe: a population-based study," *J. Eur. Acad. Dermatology Venereol.*, vol. 36, no. 7, pp. 1088–1096, Jul. 2022, doi: 10.1111/jdv.18050.

[2] K. Vayadande, A. A. Bhosle, R. G. Pawar, D. J. Joshi, P. A. Bailke, and O. Lohade, "Innovative approaches for skin disease identification in machine learning: A comprehensive study," *Oral Oncol. Reports*, vol. 10, no. April, p. 100365, Jun. 2024, doi: 10.1016/j.oor.2024.100365.

[3] G. Nikolakis, A. G. Vaiopoulos, I. Georgopoulos, E. Papakonstantinou, G. Gaitanis, and C. C. Zouboulis, "Insights, Advantages, and Barriers of Teledermatology vs. Face-to-Face Dermatology for the Diagnosis and Follow-Up of Non-Melanoma Skin Cancer: A Systematic Review," *Cancers (Basel).*, vol. 16, no. 3, p. 578, Jan. 2024, doi: 10.3390/cancers16030578.

[4] T. G. Debelee, "Skin Lesion Classification and Detection Using Machine Learning Techniques: A Systematic Review," *Diagnostics*, vol. 13, no. 19, 2023, doi: 10.3390/diagnostics13193147.

[5] J. A. Fee, F. P. McGrady, and N. D. Hart, "Dermoscopy use in primary care: a qualitative study with general practitioners," *BMC Prim. Care*, vol. 23, no. 1, pp. 1–10, 2022, doi: 10.1186/s12875-022-01653-7.

[6] K. A. Muhaba, K. Dese, T. M. Aga, F. T. Zewdu, and G. L. Simegn, "Automatic skin disease diagnosis using deep learning from clinical image and patient information," *Ski. Heal. Dis.*, vol. 2, no. 1, 2022, doi: 10.1002/ski2.81.

[7] T. D. Nigat, T. M. Sitote, and B. M. Gedefaw, "Fungal Skin Disease Classification Using the Convolutional Neural Network," *J. Healthc. Eng.*, vol. 2023, 2023, doi: 10.1155/2023/6370416.

[8] Y. Yang, L. Qu, I. Mijakovic, and Y. Wei, "Advances in the human skin microbiota and its roles in cutaneous diseases," *Microb. Cell Fact.*, vol. 21, no. 1, pp. 1–14, 2022, doi: 10.1186/s12934-022-01901-6.

[9] A. M. Thawabteh, A. Jibreen, D. Karaman, A. Thawabteh, and R. Karaman, "Skin Pigmentation Types, Causes and Treatment—A Review," *Molecules*, vol. 28, no. 12, 2023, doi: 10.3390/molecules28124839.

[10] B. Seshakagari and H. Reddy, "Deep Learning-Based Detection of Hair and Scalp Diseases Using CNN and Image Processing," pp. 145–155, 2025, doi: 10.5281/zenodo.14965660.

[11] N. Hasan *et al.*, "Skin cancer: understanding the journey of transformation from conventional to advanced treatment approaches," *Mol. Cancer*, vol. 22, no. 1, pp. 1–70, 2023, doi: 10.1186/s12943-023-01854-3.

[12] B. Yang *et al.*, "DrHouse: An LLM-empowered Diagnostic Reasoning System through Harnessing Outcomes from Sensor Data and Expert Knowledge," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 8, no. 4, pp. 1–29, Nov. 2024, doi: 10.1145/3699765.

[13] E. I. Adesanya *et al.*, "Factors associated with depression, anxiety and severe mental illness among adults with atopic eczema or psoriasis: A systematic review and meta-analysis," *Br. J. Dermatol.*, vol. 188, no. 4, pp. 460–470, 2023, doi: 10.1093/bjd/ljac132.

[14] F. Jafarzadeh, F. Rahmani, F. Azadmehr, M. Falaki, and M. Nazari, "Different applications of telemedicine - assessing the challenges, barriers, and opportunities- a narrative review," *J. Fam. Med. Prim. Care*, vol. 11, no. 3, pp. 879–886, Mar. 2022, doi: 10.4103/jfmpc.jfmpc_1638_21.

[15] A. Yakupu *et al.*, "The burden of skin and subcutaneous diseases: findings from the global burden of disease study 2019," *Front. Public Heal.*, vol. 11, 2023, doi: 10.3389/fpubh.2023.1145513.

[16] R. I. Mukhamediev *et al.*, "Review of Artificial Intelligence and Machine Learning Technologies: Classification, Restrictions, Opportunities and Challenges," *Mathematics*, vol. 10, no. 15, pp. 1–25, 2022, doi: 10.3390/math10152552.

[17] D. Touretzky, C. Gardner-McCune, and D. Seehorn, *Machine Learning and the Five Big Ideas in AI*, vol. 33, no. 2. Springer New York, 2023. doi: 10.1007/s40593-022-00314-1.

[18] S. Garg, K. Arora, S. Singh, and K. Nagarajan, "Artificial intelligence and machine learning in drug discovery and development," *Artif. Intell. Age Nanotechnol.*, pp. 42–61, 2023, doi: 10.4018/9798369303689.ch003.

[19] R. Obuchowicz, M. Strzelecki, and A. Piórkowski, "Clinical Applications of Artificial Intelligence in Medical Imaging and Image Processing—A Review," *Cancers (Basel).*, vol. 16, no. 10, p. 1870, May 2024, doi: 10.3390/cancers16101870.

[20] U. Kulkarni, S. M. Meena, S. V. Gurlahosur, and G. Bhogar, "Quantization Friendly MobileNet (QF-MobileNet) Architecture for Vision Based Applications on Embedded Platforms," *Neural Networks*, vol. 136, pp. 28–39, 2021, doi: 10.1016/j.neunet.2020.12.022.

[21] A. Pfob, S. C. Lu, and C. Sidey-Gibbons, "Machine learning in medicine: a practical introduction to techniques for data pre-processing, hyperparameter tuning, and model comparison," *BMC Med. Res. Methodol.*, vol. 22, no. 1, pp. 1–15, 2022, doi: 10.1186/s12874-022-01758-8.

[22] R. Abdulkadirov, P. Lyakhov, and N. Nagornov, "Survey of Optimization Algorithms in Modern Neural Networks," *Mathematics*, vol. 11, no. 11, p. 2466, May 2023, doi: 10.3390/math11112466.

[23] V. Anand, S. Gupta, D. Koundal, S. Mahajan, A. K. Pandit, and A. Zaguia, "Deep learning based automated diagnosis of skin diseases using dermoscopy," *Comput. Mater. Contin.*, vol. 71, no. 2, pp. 3145–3160, 2022, doi: 10.32604/cmc.2022.022788.

[24] D. M. R. Sari, S. Nurmaini, D. P. Rini, and A. I. Sapitri, "Dermatitis Atopic and Psoriasis Skin Disease Classification by using Convolutional Neural Network," *Comput. Eng. Appl. J.*, vol. 12, no. 1, pp. 1–10, Feb. 2023, doi: 10.18495/comengapp.v12i1.419.

[25] R. Saifan and F. Jubair, "Six skin diseases classification using deep convolutional neural network," *Int. J. Electr. Comput. Eng.*, vol. 12, no. 3, p. 3072, Jun. 2022, doi: 10.11591/ijece.v12i3.pp3072-3082.

[26] K. Park, M. Chae, and J. H. Cho, "Image Pre-Processing Method of Machine Learning for Edge Detection with Image Signal Processor Enhancement," *Micromachines*, vol. 12, no. 1, p. 73, Jan. 2021, doi: 10.3390/mi12010073.

[27] V. R. Joseph, "Optimal ratio for data splitting," *Stat. Anal. Data Min.*, vol. 15, no. 4, pp. 531–538, 2022, doi: 10.1002/sam.11583.

[28] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Glob. Transitions Proc.*, vol. 3, no. 1, pp. 91–99, 2022, doi: 10.1016/j.gltp.2022.04.020.

[29] R. Sarki, K. Ahmed, H. Wang, Y. Zhang, J. Ma, and K. Wang, "Image Preprocessing in Classification and Identification of Diabetic Eye Diseases," *Data Sci. Eng.*, vol. 6, no. 4, pp. 455–471, 2021, doi: 10.1007/s41019-021-00167-z.

[30] S. Boskovic, L. Svadlenka, S. Jovcic, M. Dobrodolac, V. Simic, and N. Bacanin, "An Alternative Ranking Order Method Accounting for Two-Step Normalization (AROMAN) - A Case Study of the Electric Vehicle Selection Problem," *IEEE Access*, vol. 11, no. March, pp. 39496–39507, 2023, doi: 10.1109/ACCESS.2023.3265818.

[31] 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.

[32] V. S. Dhaka *et al.*, "A survey of deep convolutional neural networks applied for prediction of plant leaf diseases," *Sensors*, vol. 21, no. 14, 2021, doi: 10.3390/s21144749.

[33] R. Sadik, S. Anwar, and L. Reza, "AutismNet: Recognition of Autism Spectrum Disorder from Facial Expressions using MobileNet Architecture," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 10, no. 1, pp. 327–334, Feb. 2021, doi: 10.30534/ijatcse/2021/471012021.

[34] B. Bischl *et al.*, "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," *WIREs Data Min. Knowl. Discov.*, vol. 13, no. 2, pp. 1–43, Mar. 2023, doi: 10.1002/widm.1484.

[35] T. Elansari, M. Ouanan, and H. Bourray, "Mixed Radial Basis Function Neural Network Training Using Genetic Algorithm," *Neural Process. Lett.*, vol. 55, no. 8, pp. 10569–10587, 2023, doi: 10.1007/s11063-023-11339-5.

[36] C. Vengaiah and S. R. Konda, "A Comparative Study of Convolutional Neural Network Architectures for Enhanced Tomato Leaf Disease Classification Using Refined Statistical Features," *Trait. du Signal*, vol. 41, no. 1, pp. 201–212, Feb. 2024, doi: 10.18280/ts.410116.

[37] S. Roy, R. Mehera, R. K. Pal, and S. K. Bandyopadhyay, "Hyperparameter optimization for deep neural network models: a comprehensive study on methods and techniques," *Innov. Syst. Softw. Eng.*, pp. 0–20, Oct. 2023, doi: 10.1007/s11334-023-00540-3.

[38] M. Reyad, A. M. Sarhan, and M. Arafa, "A modified Adam algorithm for deep neural network optimization," *Neural Comput. Appl.*, vol. 35, no. 23, pp. 17095–17112, Aug. 2023, doi: 10.1007/s00521-023-08568-z.

[39] K. M. Hamdia, X. Zhuang, and T. Rabczuk, "An efficient optimization approach for designing machine learning models based on genetic algorithm," *Neural Comput. Appl.*, vol. 33, no. 6, pp. 1923–1933, Mar. 2021, doi: 10.1007/s00521-020-05035-x.

[40] Y. Tian, Y. Zhang, and H. Zhang, "Recent Advances in Stochastic Gradient Descent in Deep Learning," *Mathematics*, vol. 11, no. 3, p. 682, Jan. 2023, doi: 10.3390/math11030682.

[41] Y.-L. Peng and W.-P. Lee, "Practical guidelines for resolving the loss divergence caused by the root-mean-squared propagation optimizer," *Appl. Soft Comput.*, vol. 153, p. 111335, Mar. 2024, doi: 10.1016/j.asoc.2024.111335.

[42] N. Attrapadung *et al.*, "Adam in Private: Secure and Fast Training of Deep Neural Networks with Adaptive Moment Estimation," *Proc. Priv. Enhancing Technol.*, vol. 2022, no. 4, pp. 746–767, Oct. 2022, doi: 10.56553/popets-2022-0131.

[43] R. Elshamy, O. Abu-Elnasr, M. Elhoseny, and S. Elmougy, "Enhancing colorectal cancer histology diagnosis using modified deep neural networks optimizer," *Sci. Rep.*, vol. 14, no. 1, p. 19534, Aug. 2024, doi: 10.1038/s41598-024-69193-x.

[44] R. S. Kale and S. Shitole, "Deep learning optimizer performance analysis for pomegranate fruit quality gradation," in *2022 IEEE Bombay Section Signature Conference (IBSSC)*, Dec. 2022, pp. 1–5. doi: 10.1109/IBSSC56953.2022.10037429.

[45] P. Das, S. Gupta, J. Patra, and B. Mondal, "ADAMAX optimizer and CATEGORICAL CROSSENTROPY loss function-based CNN method for diagnosing Lung cancer," in *2023 7th International Conference on Trends in Electronics and Informatics (ICOEI)*, Apr. 2023, pp. 806–810. doi: 10.1109/ICOEI56765.2023.10126046.

[46] J. Sharma *et al.*, "A novel long term solar photovoltaic power forecasting approach using LSTM with Nadam optimizer: A case study of India," *Energy Sci. Eng.*, vol. 10, no. 8, pp. 2909–2929, Aug. 2022, doi: 10.1002/ese3.1178.

[47] K. Ali, Z. A. Shaikh, A. A. Khan, and A. A. Laghari, "Multiclass skin cancer classification using EfficientNets – a first step towards preventing skin cancer," *Neurosci. Informatics*, vol. 2, no. 4, p. 100034, 2022, doi: 10.1016/j.neuri.2021.100034.

[48] M. Taqiyuddin, K. Adi, O. Dwi Nurhayati, and H. Ochi, "Comparison of Optimizers for Drone Signal Detection Using Convolutional Neural Networks (CNN)," *E3S Web Conf.*, vol. 448, 2023, doi: 10.1051/e3sconf/202344802025.

[49] M. Z. Hussain *et al.*, "A fine-tuned convolutional neural network model for accurate Alzheimer's disease classification," *Sci. Rep.*, vol. 15, no. 1, pp. 1–24, 2025, doi: 10.1038/s41598-025-86635-2.

[50] Deepika Kumari, Santosh Kumar Singh, Sanjay Subhash Katira, Inumarthi V Srinivas, and Uday Salunkhe, "Benchmarking Optimizers in Transfer Learning for Automated Weed Image Recognition," *Metall. Mater. Eng.*, vol. 31, no. 4, pp. 1–10, Apr. 2025, doi: 10.63278/1376.

[51] V. H. Sahin, I. Oztel, and G. Yolcu Oztel, "Human Monkeypox Classification from Skin Lesion Images with Deep Pre-trained Network using Mobile Application," *J. Med. Syst.*, vol. 46, no. 11, p. 79, Oct. 2022, doi: 10.1007/s10916-022-01863-7.

[52] A. Mortazi, V. Cicek, E. Keles, and U. Bagci, "Selecting the best optimizers for deep learning–based medical image segmentation," *Front. Radiol.*, vol. 3, no. September, pp. 1–17, Sep. 2023,

Vol. 6, No. 3, Juni 2025, Page. 1589-1604
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.3.4685

doi: 10.3389/fradi.2023.1175473.