Scientific Journal of Informatics
Vol. 12, No. 1, Feb 2025

p-ISSN 2407-7658 https://journal.unnes.ac.id/journals/sji/index e-ISSN 2460-0040

A Deep Learning Model Comparation for Diabetic Retinopathy Image Classification

Tanzilal Mustaqim[1*], Pima Hani Safitri[2], Daud Muhajir[3]
[1, 2, 3]Informatics Study Program, Surabaya Campus Directorate, Telkom University, Indonesia

Abstract.
Purpose: This study compares the performance of various deep learning models for diabetic retinopathy (DR) classification, emphasizing the impact of different optimization functions. Early detection of DR is vital for preventing blindness, and the research investigates how optimization functions influence the classification accuracy and efficiency of several convolutional neural networks (CNNs). This study fills a gap in the existing literature by examining how optimization functions affect model performance in conjunction with architectural considerations.
Methods: This paper uses the APTOS 2019 dataset, which comprises 3,663 retinal fundus images classified into five classes of diabetic retinopathy severity. Four CNN-based models, including CNN, ResNet50, DenseNet121, and EfficientNet B0, were trained using five optimization techniques: Adam, SGD, RMSProp, AdamW, and NAdam. The performance of the experimental scenarios was evaluated through accuracy, precision, recall, F1-score, training duration, and model size.
Result: EfficientNet B0 demonstrated superior computational efficiency with a minimal model size of 16.16 MB. Subsequently, DenseNet121 with the SGD optimizer achieved the highest test accuracy of 96.86%. The experimental results indicate that the optimizer significantly influences model performance. AdamW and NAdam yield superior outcomes for deeper architectures such as ResNet50 and DenseNet121.
Novelty: This paper offers an analytical examination of deep learning models and optimization techniques for DR classification, helping to clarify the trade-offs between computational efficiency and classification performance. The findings contribute to the development of more accurate and efficient DR detection systems, which could be utilized in real-world, resource-limited settings.

Keywords: Diabetic retinopathy, Deep learning, Image classification, Convolutional neural networks (CNN), Optimization algorithms
Received February 2025 / **Revised** March 2025 / **Accepted** March 2025

INTRODUCTION

Diabetic retinopathy is one of the complications of diabetes mellitus. It can lead to vision impairment and blindness if left undiagnosed or untreated [1], [2]. Diabetic retinopathy (DR) is classified as a severe microvascular condition affecting the eyes of diabetic patients. It begins with progressive damage to the blood vessels in the retina. The more the vessels are damaged, the more they contribute to microaneurysms, hemorrhages, and neovascularization, ultimately leading to retinal detachment and vision loss [3], [4]. The increasing prevalence of diabetes worldwide highlights the urgency of developing efficient and scalable methods for the early detection and classification of DR. Traditional diagnostic methods rely on the manual grading of fundus images by ophthalmologists, which is time-consuming, subjective, and prone to inter-observer variability [5]. The expansion of artificial intelligence, particularly deep learning, has significantly advanced medical imaging, including DR detection, offering automated and highly accurate classification models [6].

Deep learning, particularly convolutional neural networks (CNNs) applied to images, has enhanced medical image performance [7], [8]. CNNs can effectively extract features from retinal fundus images. Various studies have proposed different CNN architectures, such as ResNet, VGG, DenseNet, XceptionNet, and MobileNet, to categorize diabetic retinopathy (DR) severity into distinct stages [9]. Despite demonstrating impressive results, there is still room for improvement regarding model depth, feature extraction, computational efficiency, and generalizability across datasets [10]. The selection of an optimization

*Corresponding author.
Email addresses: derekln14@gmail.com (Mustaqim)
DOI: 10.15294/sji.v12i1.20939

function is a crucial factor in deep learning-based DR classification. It is vital for improving model convergence, minimizing loss, and enhancing classification accuracy.

Although prior research has extensively explored CNN architectures for DR classification, a significant research gap exists in understanding the impact of different optimization algorithms on model performance [11]. The selection of an appropriate optimizer, such as Adam, SGD, RMSprop, or Adagrad, significantly influences model convergence speed, stability, and classification accuracy [12]. Optimization algorithms control how neural networks adjust their weights during backpropagation, ultimately affecting the model's overall performance. However, existing studies primarily focus on architecture comparison without evaluating the contribution of optimizers in enhancing feature extraction and improving classification outcomes [13].

Previous research has highlighted the advantages and limitations of deep learning-based diabetic retinopathy classification. Some prior studies utilizing ResNet and EfficientNet models have shown significant classification accuracy, especially in distinguishing between non-diabetic retinopathy (non-DR) and severe diabetic retinopathy (DR) phases [1], [3], [8]. Nonetheless, these models often suffer from overfitting, increased computational costs, and insufficient generalizability when applied to other datasets [4], [14]. Moreover, research on advanced networks such as ResNet101 and DenseNet-121 demonstrates superior feature extraction capabilities but with heightened computing demands [5]. In contrast, lightweight architectures like SqueezeNet and MobileNet provide computational efficiency; however, they may face challenges in categorizing the severity degrees of DR with precision [9], [15].

This study aims to assess the impact of various optimization techniques by comprehensively comparing deep learning models for diabetic retinopathy categorization. Despite several studies focusing on model construction, few studies have examined the effect of optimization techniques on model training dynamics and classification efficacy [7]. This study addresses the research gap by systematically comparing CNN-based architectures with different optimizers and analyzing their effect on metric evaluation.

This study contributes to the automated classification of diabetic retinopathy (DR) by systematically evaluating the performance of various deep-learning models using different optimization functions. It aims to comprehensively assess the efficacy of models for classifying diabetic retinopathy images, utilizing CNN-based architectures such as CNN, ResNet50, DenseNet121, and EfficientNet B0, along with various optimizers including Adam, SGD, RMSProp, AdamW, and NAdam. The research emphasizes key performance metrics such as accuracy, precision, recall, F1-score, total training time (s), and model size (MB) to identify the most effective model-optimizer combination for accurate and computationally efficient DR detection. This study enables the paper to present the following primary contributions:

- This study evaluates CNN, ResNet50, DenseNet121, and EfficientNet B0 based on accuracy, precision, recall, F1-score, total training time, and model size. Five optimizers (Adam, SGD, RMSProp, AdamW, NAdam) are used to determine the optimal model-optimizer combination pairing.
- This study examines the trade-offs between classification performance and computational efficiency, considering training time and model size to offer insights for deployment in real-world, resource-constrained situations.

The remainder of this paper is structured as follows: Section 2 presents a literature review on deep learning-based diabetic retinopathy classification, highlighting previous studies, their limitations, and the research gap addressed in this work. Section 3 details the methodology, including a description of the dataset, deep learning models, optimization functions, and evaluation metrics. Section 4 showcases the experimental results, comparing model performance across different optimizers and discussing key findings related to classification accuracy, computational efficiency, and practical implications. Finally, Section 5 concludes the study by summarizing key contributions, discussing limitations, and outlining potential avenues for future research directions.

LITERATURE REVIEW

Deep learning has revolutionized the detection and classification of diabetic retinopathy (DR) through advanced architectures like convolutional neural networks (CNNs). Research has demonstrated that ResNet50, VGG16, DenseNet-121, and XceptionNet, enhanced with transfer learning and hybrid models incorporating attention mechanisms, can significantly improve feature extraction and classification accuracy [7], [9]. These models have been extensively tested and serve as the baseline for experts in image detection and classification based on deep learning. The results have shown high sensitivity and negative predictive value, surpassing the performance of traditional image processing techniques.

Despite these architectural advancements, optimization techniques like Adam, SGD, and RMSProp remain critical for efficient model training and generalization. While Adam is favored for its adaptive learning rate, studies suggest that other optimizers might offer better generalization in specific scenarios [5], [12]. Balancing model complexity and computational efficiency is also crucial, as seen in lightweight models like EfficientNet, which provide a good trade-off between performance and speed [10]. This paper focuses on bridging the gaps between detecting and optimizing deep learning models in image processing, calculated using several performance metrics that address accuracy and computational demands.

METHODS

Dataset description

No_DR (1805 images) Mild (370 images) Moderate (999 images) Severe (193 images) Poliferate_DR (295 images)

Figure 1. Dataset detail

The dataset utilized in this research stems from the APTOS 2019 Blindness Detection competition [10]. It consists of 3,663 retinal fundus images categorized by the degree of diabetic retinopathy. Each image is classified into one of five categories: No_DR (1,805 photos), Mild (370 images), Moderate (999 images), Severe (193 images), and Proliferative_DR (295 images). The images are sourced from the International Clinical Diabetic Retinopathy Disease Severity Scale and are standardized for the progression of diabetic retinopathy. All images have been resized to 224 × 224 pixels to ensure deep learning model training consistency. The dataset exhibits a class imbalance, with the No_DR and Moderate classes containing significantly more images than the Mild, Severe, and Proliferative_DR classes, necessitating handling strategies such as resampling or data augmentation. Figure 1 presents a comprehensive view of the dataset. The images are captured using fundus cameras under various lighting conditions and quality, leading to challenges related to image noise, contrast variations, and illumination discrepancies. The collection lacks additional information, such as patient demographics or clinical history, making categorization reliant solely on retinal image characteristics. Each image is encoded in JPEG format, and the labels are converted into integer values ranging from 0 to 4, corresponding to No_DR to Proliferative_DR.

Model architectures

This study evaluated four architectures of convolutional neural networks (CNN) for classifying the severity of diabetic retinopathy using retinal fundus images: CNN, ResNet50, DenseNet121, and EfficientNet B0. Each model features distinct structural characteristics aimed at optimizing classification performance and computational efficiency. The CNN architecture consists of three convolutional blocks, each containing a convolutional layer, batch normalization, ReLU activation, and max pooling. The network employs 32, 64, and 128 filters in sequential blocks, utilizing a 3×3 kernel and 2×2 max pooling. Following this, the CNN includes a fully connected layer with three dense layers responsible for classification.

The CNN model has 25,817,605 parameters and a size of 98.49 MB [16]. The ResNet50 architecture incorporates 50 layers with residual connections to address vanishing gradients, improving memory efficiency. It has 23,518,277 parameters and a size of 89.72 MB [17]. DenseNet121, with 6,958,981 parameters and a size of 26.55 MB, improves feature propagation through dense connectivity and reduces overfitting [18]. The EfficientNet B0 architecture is a lightweight model designed for computational efficiency. It employs compound scaling to balance accuracy and computational cost. It includes MBConv and Squeeze-and-Excitation (SE) blocks, followed by a global average pooling and fully connected layer. With only 4,013,953 parameters and a size of 15.31 MB, EfficientNet B0 is the most compact model in this study, offering high efficiency [19].

Optimization algorithms

This study utilized five optimization algorithms to train the deep learning models for diabetic retinopathy classification: Adam, SGD, RMSProp, AdamW, and NAdam. Each optimizer offers unique characteristics to enhance training efficiency and model performance, mainly when dealing with complex and large datasets. The Adam optimizer integrates momentum optimization and adaptive learning rate to keep steady convergence and avoid excessive oscillations in weight updates [20]. It helps avoid local minima. Hence, it is computationally economical, handles sparse gradients successfully, and is extensively utilized for tasks including diabetic retinopathy classification. Simplicity and efficiency abound in the SGD optimizer [21]. Using single training samples or small batches to update parameters causes noise in gradient updates to increase generalization and prevent local minima. While Nesterov momentum increases this impact by predicting future updates, adding momentum speeds convergence.

The RMSProp optimizer adjusts its rate based on the data's gradient values [22]. It improved the limited amount that occurred from SGD and AdaGrad. It maintains a decaying average of squared gradients, ensuring stable and efficient convergence, particularly for non-stationary functions. Next, the AdamW optimizer is improved from the Adam optimizer. It has separated weight decay gradient updates and amplifies the model's stability and generalization [23]. Last, NAdam combines Adam with Nesterov momentum for faster and more stable convergence, improving performance in high-dimensional tasks [24]. Each optimizer has unique advantages depending on the architecture and task requirements.

Metric evaluation

The evaluation of deep learning models for distinguishing diabetic retinopathy images is based on metrics such as Accuracy, Precision, Recall, F1-Score, Total Training Time, and Model Size. Accuracy measures the percentage of correct classifications, covering both true positives and true negatives, as shown in Eq. (1) [25]. Precision focuses on the proportion of true positive predictions among all predicted positives, which is calculated in Eq. (2). The F1-Score, particularly useful for imbalanced datasets, calculates the harmonic mean of Precision and Recall, providing a balanced measure, as described in Eq. (4). Recall evaluates the proportion of true positives correctly identified, as shown in Eq. (3). Computational efficiency is assessed by analyzing both the total training time and model size. The optimal configuration for diabetic retinopathy detection is determined by comparing models such as CNN, ResNet50, DenseNet121, and EfficientNet B0, using various optimizers like Adam, SGD, RMSProp, AdamW, and Adam.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Samples}} \tag{1}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

RESULTS AND DISCUSSIONS

We utilized 3,663 retinal images from the APTOS 2019 dataset to perform tests categorizing them into five classifications: No_DR, Mild, Moderate, Severe, and Proliferate_DR. Each data point employed scaled images of 224 × 224 pixels. Each testing program comprised 100 epochs with a batch size of 32. We utilized prevalent deep learning architectures, including EfficientNet B0, DenseNet121, Vanilla CNN, and ResNet50. Each design was optimized using five algorithms: Adam, SGD, RMSProp, AdamW, and NAdam. We assessed the performance of these models with different optimizers by utilizing accuracy, precision, recall, and F1-score metrics for each system.

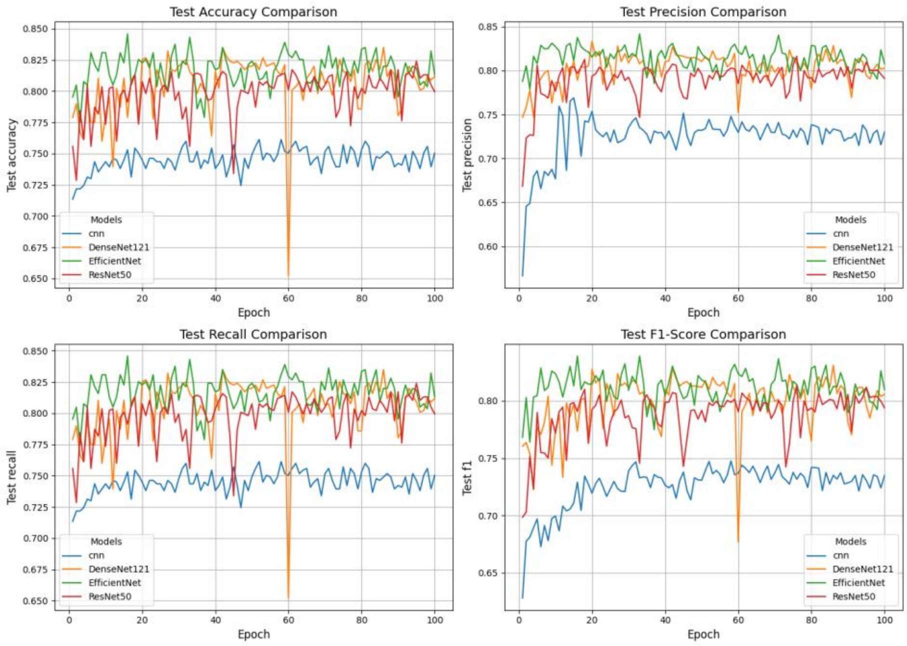

Figure 2. Comparison of metric evaluations with the adam optimizer

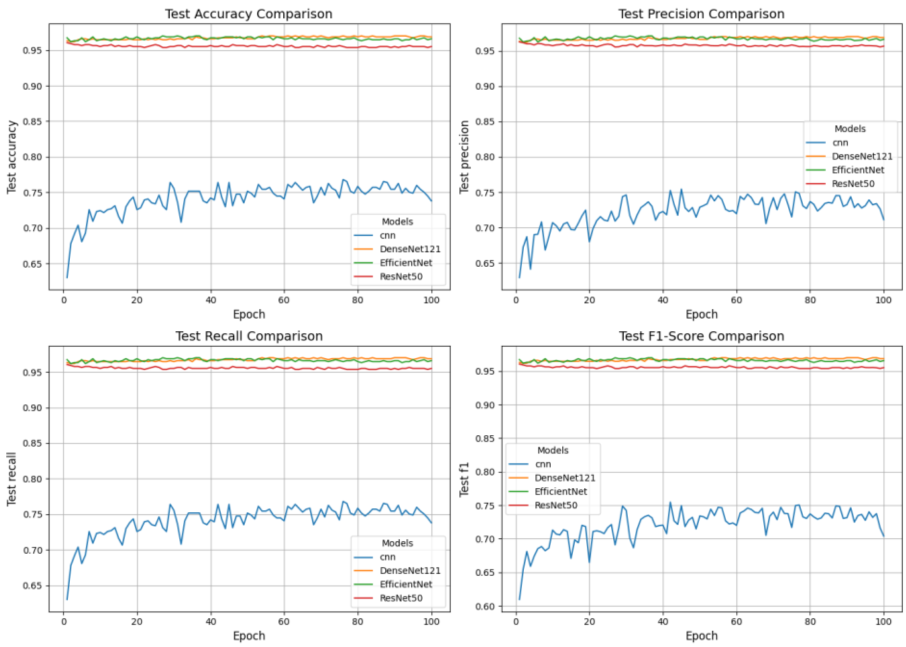

Figure 3. Comparison of metric evaluations with the stochastic gradient descent (SGD) optimizer

Figure 2 illustrates that the performance comparison of models trained using the Adam optimizer demonstrates distinct variations in validation measures. DenseNet121 achieves superior validation accuracy, precision, recall, and F1 score with robust generalization and balanced predictions. This model is optimal for tasks necessitating continuous predictions, since its high accuracy and recall indicate proficient detection of true positives while minimizing false positives. The CNN model is not superior in all the tests, as can be seen from lower validation accuracy, precision, recall, and F1 scores. It is not good at generalization and prediction when compared to more complex systems. The lower values of memory and accuracy indicate that there are more false positives, or real positives are not being picked up well.

Figure 2 illustrates that the performance comparison of models trained using the Adam optimizer demonstrates distinct variations in validation measures. DenseNet121 achieves superior performance in validation accuracy, precision, recall, and F1 score with robust generalization and balanced predictions. This model is optimal for tasks necessitating continuous predictions, since its high accuracy and recall indicate proficient detection of true positives while minimizing false positives.The CNN model is not superior in all the tests, as can be seen from lower validation accuracy, precision, recall, and F1 scores. It is not good at generalization and prediction when compared to more complex systems. The lower values of memory and accuracy indicate that there are more false positives or real positives are not being picked up well. The results demonstrate the significant impact of the Adam optimizer, which provides steady and efficient optimization by integrating adaptive learning rates with momentum. The Adam optimizer’s adaptation to sparse gradients and automated learning rate adjustment helps it excel in advanced models such as DenseNet121, EfficientNet-B0, and ResNet50. Its advantages are less pronounced in simpler models like CNN, which may not fully leverage its capabilities. These results suggest the necessity of carefully selecting the model architecture and optimizer to enhance machine learning performance.

The evaluation of models optimized by Stochastic Gradient Descent (SGD), illustrated in Figure 3, shows notable variation in performance across validation metrics. EfficientNet-B0 can outperform other models, resulting in the optimal validation accuracy, precision, recall, and F1 score. These results demonstrate robust generalization capability and equitable predictive performance, with improved accuracy and recall achieved by reducing false positives and properly classifying real positives. Thus, EfficientNet-B0 is recognized as the most reliable model for tasks requiring accurate predictions when trained using SGD. The CNN model exhibits the lowest performance across all metrics, indicating that simpler architectures struggle to identify complex patterns, particularly under SGD. The decreased precision of CNN signifies a higher incidence of false positives, thereby highlighting the challenges posed by SGD's reliance on smaller, stochastic updates. This example emphasizes the importance of selecting appropriate designs based on the dataset and optimization method. While the stochastic nature of SGD may improve generalization, it can lead to delayed convergence for simpler models, whereas more advanced designs like EfficientNet-B0, DenseNet121, and ResNet50 are better equipped to manage this, resulting in superior accuracy and more balanced predictions.

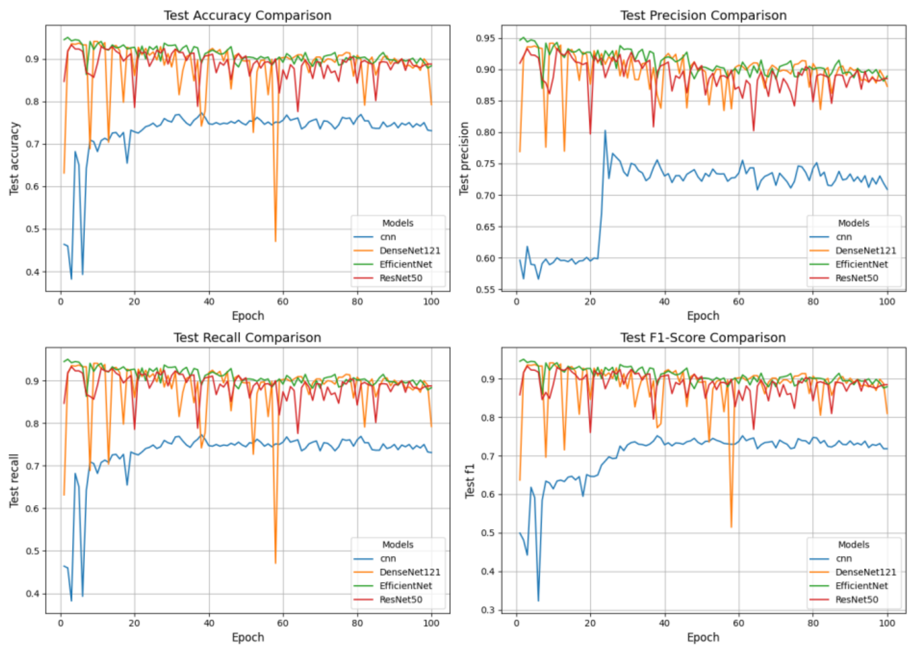

Figure 4. Comparison of metric evaluations with the root mean square propagation (RMSProp) optimizer

Figure 4 illustrates the significant performance differences observed when models are trained with the RMSProp optimizer. Overall, DenseNet121 consistently outperforms other models in validation accuracy, precision, recall, and F1 score. Its exceptional recall and accuracy when optimized with RMSProp highlight its effectiveness in reducing false positives by accurately identifying real positives. Across all metrics, the CNN model demonstrates low performance; thus, it struggles with the complexity of the dataset, underscoring the necessity for more advanced architectures like DenseNet121, EfficientNet-B0, and

ResNet50. Simpler models fail to leverage the advantages offered by RMSProp, resulting in suboptimal outcomes.

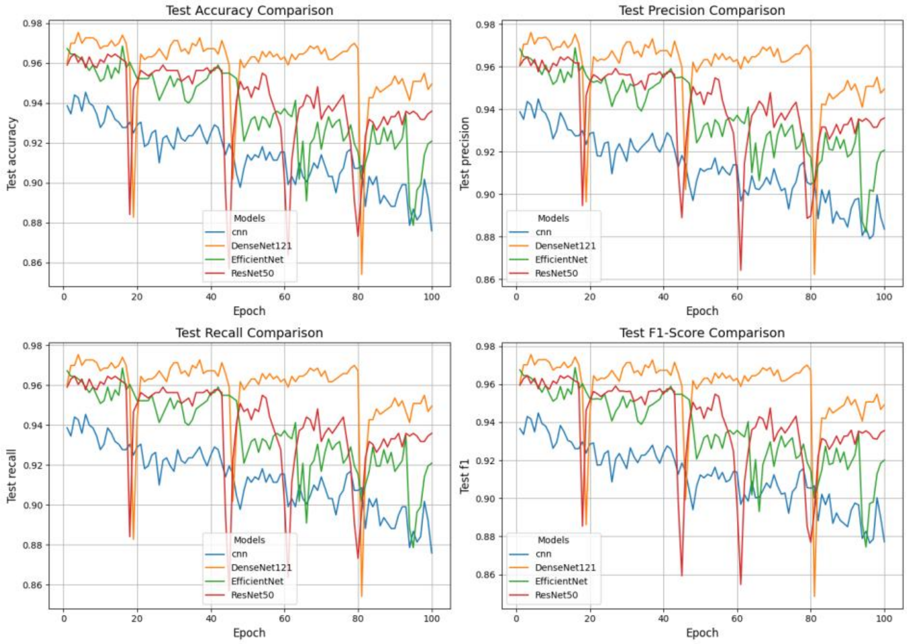

Figure 5. Comparison of metric evaluations with the adamw optimizer

Figure 5 displays the clear variations in validation metrics shown by models tuned with the AdamW optimizer. Regularly achieving the highest accuracy, precision, recall, and F1 score, EfficientNet-B0 stands out for its excellent generalization and balanced predictive capacity. AdamW optimizes this model to minimize false positives and identify actual positives. In contrast, the CNN model continues to underperform with declining accuracy and recall, indicating its limited ability to generalize and produce accurate forecasts. The results emphasize the importance of pairing AdamW with advanced models for optimal performance.

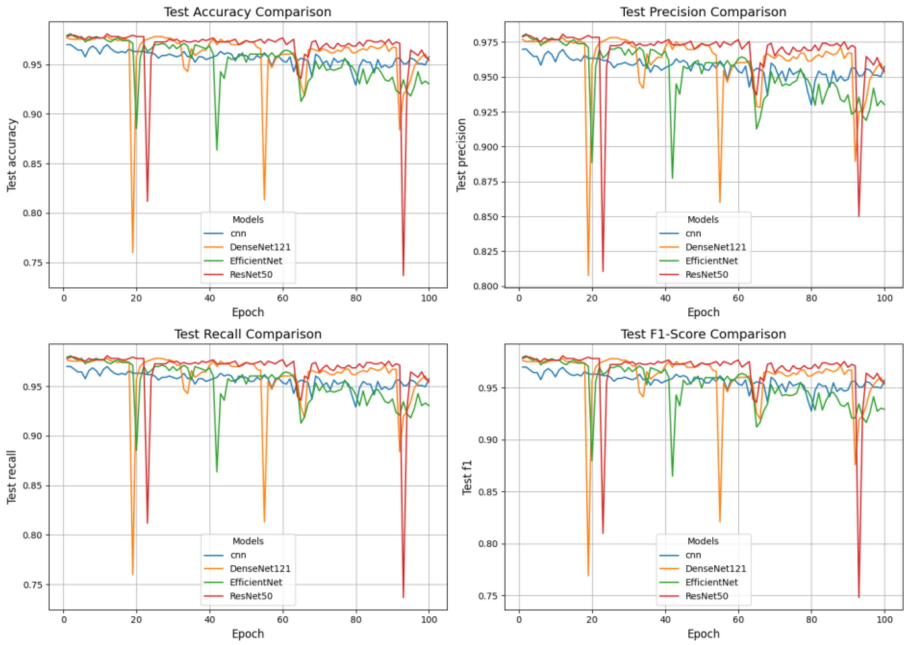

Figure 6. Comparison of metric evaluations with the NAdam optimizer

Figure 6 illustrates the notable differences in validation measures when models are tuned with NAdam. ResNet50 demonstrates a strong ability to generalize and accurately identify true positives, as it ranks highest in accuracy, precision, recall, and F1 score, indicating its impressive capabilities. DenseNet121, on the other hand, underperforms across all measures, likely due to insufficient interaction with NAdam or challenges in convergence. These results highlight the importance of pairing specific architectures with appropriate optimizers to ensure optimal model performance. Utilizing Nesterov momentum, NAdam significantly enhances models like ResNet50; however, its effectiveness is contingent on the complexity of the architecture.

Table 1. Comparison of metric evaluations across models and optimization functions

Model	Optimizer	Test			
		Accuracy	Precision	Recall	F1
CNN	Adam	75.03%	73.00%	75.03%	73.48%
CNN	AdamW	87.59%	88.35%	87.59%	87.72%
CNN	NAdam	95.77%	95.74%	95.77%	95.74%
CNN	RMSProp	73.12%	70.90%	73.12%	71.82%
CNN	SGD	73.81%	71.12%	73.81%	70.41%
DenseNet121	Adam	81.17%	80.28%	81.17%	80.58%
DenseNet121	AdamW	94.95%	94.95%	94.95%	94.92%
DenseNet121	NAdam	95.36%	95.29%	95.36%	95.29%
DenseNet121	RMSProp	79.26%	87.28%	79.26%	80.97%
DenseNet121	SGD	96.86%	96.87%	96.86%	96.85%
EfficientNet B0	Adam	81.31%	80.78%	81.31%	80.99%
EfficientNet B0	AdamW	92.09%	92.07%	92.09%	92.01%
EfficientNet B0	NAdam	93.04%	93.01%	93.04%	92.93%
EfficientNet B0	RMSProp	88.40%	88.91%	88.40%	87.92%
EfficientNet B0	SGD	96.59%	96.61%	96.59%	96.56%
ResNet50	Adam	79.95%	79.10%	79.95%	79.39%
ResNet50	AdamW	93.59%	93.58%	93.59%	93.56%
ResNet50	NAdam	95.50%	95.42%	95.50%	95.44%
ResNet50	RMSProp	88.81%	88.51%	88.81%	88.45%
ResNet50	SGD	95.50%	95.65%	95.50%	95.52%

Table 1 compares deep learning models and optimizers for classifying diabetic retinopathy and evaluating metrics such as Test Accuracy, Precision, Recall, and F1 Score. DenseNet121, optimized with SGD, outperforms all other configurations, achieving the highest accuracy of 96.87%, alongside precision and recall rates of 96.86% and an F1 Score of 96.85%. This model excels at extracting subtle retinal features like microaneurysms and hemorrhages, which are vital for accurate diagnosis in clinical settings. Its ability to reduce false positives enhances its effectiveness in preventing overdiagnosis, an essential aspect of medical imaging. The CNN model optimized with RMSProp shows notably weaker performance across all metrics, achieving a precision of 70.90%, a recall of 73.12%, and an F1 Score of 71.82%. This highlights the limitations of simpler architectures and less adaptive optimization techniques in handling the complexities of diagnosing diabetic retinopathy. The significant performance gap emphasizes the need for advanced models like DenseNet121, which are better equipped to manage the intricacies of medical imaging and effectively diagnose diabetic retinopathy.

Table 2. Comparison of total training time and model size for different models and optimization functions

Model	Optimizer	Total Training Time (s)	Model Size (MB)
CNN	Adam	818.95	98.54
CNN	AdamW	928.69	98.54
CNN	NAdam	1025.74	98.54
CNN	RMSProp	875.25	98.54
CNN	SGD	831.18	98.54
DenseNet121	Adam	1786.85	27.93
DenseNet121	AdamW	1922.98	27.94
DenseNet121	NAdam	2115.97	27.94
DenseNet121	RMSProp	1754.13	27.94
DenseNet121	SGD	1605.10	27.94
EfficientNet B0	Adam	1175.98	16.16
EfficientNet B0	AdamW	1325.65	16.17
EfficientNet B0	NAdam	1396.20	16.17
EfficientNet B0	RMSProp	1179.30	16.17
EfficientNet B0	SGD	1083.57	16.16
ResNet50	Adam	1466.05	90.30
ResNet50	AdamW	1557.05	90.31
ResNet50	NAdam	1604.29	90.31
ResNet50	RMSProp	1451.88	90.31
ResNet50	SGD	1390.56	90.30

Table 2 evaluates the computational efficiency and memory requirements of various models and optimizers, showing significant differences in training time and model sizes. The training time for DenseNet121 optimized with NAdam is 2115.97 seconds, reflecting its complex architecture and adaptive learning approach. The CNN model using Adam exhibits the shortest training period at 818.95 seconds, emphasizing its efficiency. These findings underscore the need to balance computational costs with model complexity in diabetic retinopathy classification, as longer training times may lead to better performance, albeit at the cost of efficiency.

EfficientNet-B0 is remarkable for its memory efficiency, featuring a compact model size of approximately 16.16 MB across various optimizers, which makes it particularly suitable for environments with limited resources. In contrast, the model size of CNN is 98.54 MB, and its less-than-ideal performance limits its effectiveness in applications where memory is a critical factor. EfficientNet-B0 offers a lightweight design that enables strong performance while ensuring portability, making it ideal for edge computing applications in diabetic retinopathy classification. The findings emphasize the trade-offs among training efficiency, model size, and performance, offering vital insights for selecting optimal configurations for various operations.

CONCLUSION

This research conducted a systematic comparison of four deep learning models, including CNN, ResNet50, DenseNet121, and EfficientNet B0, utilizing five optimization algorithms (Adam, SGD, RMSProp, AdamW, and NAdam) for the classification of diabetic retinopathy, using the APTOS 2019 dataset. The findings highlight that DenseNet121, paired with SGD, achieved the highest classification accuracy of 96.86%, demonstrating its exceptional capability in feature extraction and generalization. The CNN using RMSProp showed the weakest performance, suggesting that simpler architectures struggle to address complex challenges in medical image classification. EfficientNet B0 exhibited remarkable computational efficiency, striking an ideal balance between performance and resource constraints, which makes it highly applicable in real-world scenarios. The findings indicate that both the model's architecture and the optimization strategy significantly affect classification effectiveness. Therefore, selecting the right combination is essential for developing scalable, efficient, and reliable systems for detecting diabetic retinopathy. This study advocates for the use of artificial intelligence in medical imaging to enhance early detection and intervention, aiming to prevent vision loss in diabetic patients.

REFERENCES

[1] N. Gupta, "Deep Learning Applications in Healthcare: Diagnosing Diabetic Retinopathy," *Int. J. Res. Publ. Semin.*, vol. 14, no. 5, pp. 278–283, 2024, doi: 10.36676/jrps.v14.i5.1558.

[2] L. Math and R. Fatima, "Adaptive machine learning classification for diabetic retinopathy," *Multimed. Tools Appl.*, vol. 80, no. 4, pp. 5173–5186, 2021.

[3] D. E. Similié, J. Andersen, S. Dinesen, T. R. Savarimuthu, and J. Grauslund, "Grading of diabetic retinopathy using a pre-segmenting deep learning classification model: Validation of an automated algorithm," *Acta Ophthalmol.*, 2024, doi: 10.1111/aos.16781.

[4] G. T. Reddy *et al.*, "An ensemble based machine learning model for diabetic retinopathy classification," in *2020 International Conference on Emerging Trends in Information Technology and Engineering (IC-ETITE)*, 2020, pp. 1–6.

[5] P. Anilkumar, "Automated Diabetic Retinopathy Detection Using Convolutional Neural Networks For Feature Extraction And Classification (ADRFEC)," *Int. J. Multidiscip. Res.*, vol. 6, no. 5, 2024, doi: 10.36948/ijfmr.2024.v06i05.27731.

[6] N. Jiwani, K. Gupta, and N. Afreen, "A convolutional neural network approach for diabetic retinopathy classification," in *2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT)*, 2022, pp. 357–361.

[7] Y. Guo and S. Li, "Small-scale cross-layer fusion network for classification of diabetic retinopathy," *J. Med. Imaging*, vol. 41, no. 5, pp. 861–868, 2024, doi: 10.7507/1001-5515.202403016.

[8] G. Kalyani, B. Janakiramaiah, A. Karuna, and L. V. N. Prasad, "Diabetic retinopathy detection and classification using capsule networks," *Complex Intell. Syst.*, vol. 9, no. 3, pp. 2651–2664, Jun. 2023, doi: 10.1007/s40747-021-00318-9.

[9] K. R. Devi, "Classification of Diabetic Retinopathy Using Regularized Pre-Trained Models," in *Advances in Computational Intelligence and Robotics Book Series*, 2024, pp. 139–164. doi: 10.4018/979-8-3693-6150-4.ch007.

[10] D. Praneeth and S. Kumar, “Diabetic retinopathy detection and categorizing using a lightweight deep learning approach,” *ARPN J. Eng. Appl. Sci.*, pp. 854–863, Sep. 2024, doi: 10.59018/072416.
[11] M. T. Al-Antary and Y. Arafa, “Multi-scale attention network for diabetic retinopathy classification,” *IEEE Access*, vol. 9, pp. 54190–54200, 2021.
[12] D. Muthusamy and P. Palani, “Deep learning model using classification for diabetic retinopathy detection: an overview,” *Artif. Intell. Rev.*, vol. 57, no. 7, 2024, doi: 10.1007/s10462-024-10806-2.
[13] M. Alhalabi, N. Eissa, H. M. Ahmed, A. Sheibani, A. Amin, and M. Ghazal, “A Mobile Feature-Based Cascaded Deep Learning Approach for Diabetic Retinopathy Classification,” in *Proceedings of the IEEE International Conference on Future Internet of Things and Cloud (FiCloud)*, 2024, pp. 183–188. doi: 10.1109/ficloud62933.2024.00035.
[14] A. Gopi, L. . Sudha, and J. S. Iwin Thanakumar, “Diabetic Detection from Images of the Eye,” *AG Salud*, vol. 3, p. 197, Jan. 2025, doi: 10.62486/agsalud2025197.
[15] A. Mehboob, M. U. Akram, N. S. Alghamdi, and A. Abdul Salam, “A Deep Learning Based Approach for Grading of Diabetic Retinopathy Using Large Fundus Image Dataset,” *Diagnostics*, vol. 12, no. 12, p. 3084, 2022, doi: 10.3390/diagnostics12123084.
[16] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, “A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 12, pp. 6999–7019, Dec. 2022, doi: 10.1109/TNNLS.2021.3084827.
[17] Suyahman, Sunardi, and Murinto, “Comparative Analysis of CNN Architectures in Siamese Networks with Test-Time Augmentation for Trademark Image Similarity Detection,” *Sci. J. Informatics*, vol. 11, no. 4, pp. 949–958, Dec. 2024, doi: 10.15294/sji.v11i4.13811.
[18] Y. Zhu and S. Newsam, “DenseNet for dense flow,” in *2017 IEEE International Conference on Image Processing (ICIP)*, IEEE, Sep. 2017, pp. 790–794. doi: 10.1109/ICIP.2017.8296389.
[19] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.
[20] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” *CoRR*, vol. abs/1412.6, 2014.
[21] R. M. Gower, N. Loizou, X. Qian, A. Sailanbayev, E. Shulgin, and P. Richtárik, “SGD: General analysis and improved rates,” in *International conference on machine learning*, 2019, pp. 5200–5209.
[22] F. Zou, L. Shen, Z. Jie, W. Zhang, and W. Liu, “A sufficient condition for convergences of adam and rmsprop,” in *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 2019, pp. 11127–11135.
[23] I. Loshchilov and F. Hutter, “Fixing Weight Decay Regularization in Adam,” *ArXiv*, vol. abs/1711.0, 2017.
[24] L. Fan and Z. Long, “Optimization of Nadam algorithm for image denoising based on convolutional neural network,” in *2020 7th International Conference on Information Science and Control Engineering (ICISCE)*, IEEE, Dec. 2020, pp. 957–961. doi: 10.1109/ICISCE50968.2020.00197.
[25] M. A. Muslim *et al.*, “An Ensemble Stacking Algorithm to Improve Model Accuracy in Bankruptcy Prediction,” *J. Data Sci. Intell. Syst.*, Mar. 2023, doi: 10.47852/bonviewJDSIS3202655.