International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 31, No. 1, January, 2026, pp. 225-232
Journal homepage: http://iieta.org/journals/isi

Multi-Model Convolutional Neural Network Architecture for Cervical Cell Image Classification

Budanis Dwi Meilani[1*], Siti Nurmuslimah[1], Lisetyo Ariyanti[2], Miswanto[3], Yuli Panca Asmara[4], Aeri Rachmad[5]

[1] Department of Informatics, Institut Teknologi Adhi Tama Surabaya, Surabaya 60117, Indonesia
[2] English Literature Study Program, Faculty of Languages and Arts, Universitas Negeri Surabaya, Surabaya 60213, Indonesia
[3] Department of Mathematics, Faculty of Science and Technology, University of Airlangga, Surabaya 60115, Indonesia
[4] Faculty of Engineering and Quantity Surveying, INTI International University, Nilai 71800, Malaysia
[5] Department of Informatics, Faculty of Engineering, University of Trunojoyo Madura, Bangkalan 69162, Indonesia

Corresponding Author Email: budanis@itats.ac.id

https://doi.org/10.18280/isi.310121

Received: 17 August 2025
Revised: 15 December 2025
Accepted: 20 January 2026
Available online: 31 January 2026

Keywords:
cervical cancer detection, cervical cell image classification, deep learning, multi-model architecture, convolutional neural networks, medical image analysis, optimization algorithms

ABSTRACT

Cervical cancer remains one of the leading causes of cancer-related mortality among women worldwide, particularly in low- and middle-income countries. Early and accurate detection is therefore essential for improving patient outcomes. However, traditional cytological examination methods are often time-consuming and subject to inter-observer variability. This study proposes a deep learning–based framework for cervical cell image classification using a multi-model convolutional neural network architecture. The proposed framework integrates three lightweight convolutional neural network (CNN) architectures—MobileNet V3, EfficientNet V2, and ShuffleNet V2—as parallel feature extractors to capture complementary visual representations. Feature maps generated by the individual models are fused to form a richer representation before the final classification stage. To evaluate the effectiveness of the proposed approach, four experimental scenarios were designed using different optimization algorithms, including Adam, Root Mean Square Propagation (RMSProp), Stochastic Gradient Descent (SGD), and Adamax, with a fixed learning rate of 0.01. The dataset was divided using an 80:20 training and testing strategy to ensure reliable performance evaluation. Experimental results demonstrate that the proposed multi-model architecture significantly improves classification performance. The best results were achieved using the SGD optimizer, reaching a validation accuracy of 94.44% and an F1-score of 94.50%. These findings indicate that combining multiple CNN architectures with appropriate optimization strategies can enhance feature representation and improve the accuracy of cervical cancer image classification.

1. INTRODUCTION

Breast, cervical, and ovarian cancers are some of the most common malignancies affecting women and are major contributors to early female mortality globally [1, 2]. Cervical cancer is a malignancy that develops in the cervix when cervical cells grow abnormally and proliferate uncontrollably, exceeding the normal regulatory processes of cell division [3]. WHO data shows that this disease claims the lives of over 270,000 women annually, and approximately 85% of the fatalities are reported in countries with limited to moderate income levels. Global estimates suggest that roughly 444,500 new instances of cervical cancer are diagnosed annually [4]. Early detection and accurate diagnosis are essential to reduce the mortality rate caused by this disease. Unfortunately, traditional diagnostic methods that still rely on manual cytological analysis by pathologists are often time-consuming, subjective, and prone to human error [5].

With the advancement of information technology and artificial intelligence, deep learning-based approaches have shown significant potential in improving the accuracy and efficiency of cervical cancer detection. In particular, cervical cell image classification methods using convolutional neural network (CNN) models have been employed to identify various types of abnormal cells, such as dyskeratotic, koilocytotic, metaplastic, parabasal, and intermediate-superficial cells [5].

In computer vision research, a single CNN architecture often fails to capture the full range of complex image features, particularly in medical or agricultural images with diverse textures, patterns, and scales. Therefore, multi-model CNN approaches, or ensemble learning, are widely adopted to improve model accuracy, stability, and generalization. The combination of MobileNet V3, EfficientNet V2, and ShuffleNet V2 was chosen because they are lightweight CNN architectures with complementary characteristics in feature extraction, computational efficiency, and generalization capabilities.

In this study, a multi-model CNN architecture optimized with several popular optimization algorithms, including Adam, Root Mean Square Propagation (RMSProp), Stochastic Gradient Descent (SGD), and Adamax, is utilized. Each model is tasked with extracting features from cervical cell images, which are then concatenated to provide a richer feature representation before passing through the final classification layer. This approach aims to enhance classification performance by leveraging the strengths of each architecture.

2. METHODOLOGY AND METHODS

2.1 System architecture

Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNN) are widely recognized as one of the most applied deep learning models, especially in tasks involving image analysis and pattern detection. The main advantage of CNNs over previous algorithms lies in their ability to automatically extract important features from data without requiring human intervention [5, 6]. CNNs are inspired by the visual systems of living organisms, particularly the structure of the visual cortex in a cat's brain, which is simulated in these networks to recognize patterns in images [6, 7].

Architecturally, CNNs are composed of several main layers: convolutional layers, pooling layers, and fully connected (FC) layers. The process begins with an input image consisting of height, width, and depth dimensions (for example, 3 channels for RGB images), which is then processed by the convolutional layer using a number of kernels or filters to generate feature maps. These filters are responsible for detecting local features such as lines, edges, or textures [7, 8]. It is at this stage that the concepts of parameter sharing and sparse connections are applied to reduce the number of trainable parameters and accelerate the computation process [6, 8].

After the convolutional process, the results are passed through an activation function such as ReLU to introduce non-linearity, followed by a pooling layer. Pooling, such as max pooling, aims to reduce the dimensionality of the data and minimize sensitivity to shifts and distortions in the image [7-9]. This not only reduces the computational burden but also helps preserve important features from the data [10].

The data that have passed through several stages of convolution and pooling are then flattened and passed on to the fully connected layer. Here, the data are processed similarly to a multilayer perceptron (MLP) and end with an activation function such as softmax to produce probability values for each target class [11]. CNNs have proven to be highly effective in recognizing both low-level features such as color and shape, and high-level features such as object parts in a hierarchical manner [11].

In addition, during CNN training, a loss function is used to calculate prediction errors, and the backpropagation process is carried out to adjust the network's weights, thereby improving model accuracy over time [7]. The addition of batch normalization can also help stabilize the learning process and accelerate convergence [7].

With all these advantages, CNNs have been widely applied in various fields such as facial recognition, object detection, speech processing, and even in the medical field for disease image classification—for example, in cervical cancer detection [1, 6]. In implementation, several popular architectures widely used in both research and real-world applications include Adam, RMSprop, SGD, and Adamax, each offering advantages in parameter efficiency, network depth, and high performance on devices with limited resources.

1) Adam

The Adaptive Moment Estimation (Adam) algorithm is an optimization technique developed to address various challenges in training machine learning models, such as sparse gradients and noisy data. This method was introduced by Kingma and Ba in 2014, combining the advantages of RMSProp and SGD with momentum [12, 13]. Adam computes two moments of the gradient—the exponential moving average of the first and second moments—and corrects the initialization bias using bias correction techniques [12-14].

Mathematically, Adam defines the first and second moment estimates as:

$$m_t = p_1 m_{t-1} + (1 - p_1) g_t \tag{1}$$

$$u_t = p_2 u_{-1} + (1 - \ p_2) g_1^2 \tag{2}$$

where, p_1 dan p_2 represent the exponential decay rates (typically 0.9 and 0.999), and g_t is the gradient at time t [13]. To avoid underestimation due to initial zero initialization, bias corrections are applied [15]:

$$\widehat{m}_t = \frac{m_t}{(1 - p_1^t)} \tag{3}$$

$$\widehat{u}_t = \frac{u_t}{(1 - p_1^t)} \tag{4}$$

Model parameters are then updated using the formula:

$$\theta_t = \theta_{t-1} - \lambda \frac{\widehat{m}_t}{\sqrt{\hat{u}_t} + \ \epsilon} \tag{5}$$

where, λ is the learning rate (typically 0.001), and ϵ is a small constant (around10^{-8s}) to prevent division by zero [15].

In practice, this algorithm—developed at the University of Toronto—adjusts adaptive learning rates for each parameter [16]. The optimization process proceeds iteratively, beginning with gradient computation, moment estimation, bias correction, and weight updates based on the estimates [14].

Adam's primary advantage lies in its ability to automatically adjust the step size for each parameter update, making it more efficient than other algorithms such as SGD and AdaDelta [17]. Moreover, Adam is effective in minimizing errors and preventing overfitting by maintaining balanced performance between training and validation data [14]. For large and complex models, Adam has been shown to improve accuracy and significantly reduce loss values [18]. Algorithm 1 illustrates the working structure of the Adam method in detail [12].

2) RMSprop

RMSprop is one of the commonly used optimization algorithms in neural network training, particularly in non-convex settings. This algorithm is an improvement of AdaGrad, with the key difference being the use of an exponential moving average of gradients rather than full accumulation, helping to alleviate the vanishing gradient problem [14, 19]. RMSprop computes the mean square of the gradients for each parameter to adjust the weight updates, as described in the corresponding formula [14]. Although effective, this method was first introduced informally through

Geoff Hinton's Coursera lectures, rather than a formal publication [16]. In practice, RMSprop is often compared with Adam: while RMSprop relies on gradients adjusted with momentum, Adam implements direct estimation of the first and second moments. Unlike Adam, RMSprop does not include a bias correction mechanism, which makes Adam more effective in handling sparse gradients [20].

3) SGD

In the gradient descent method [20], the total residual is computed using a summation formula, where the residual is the difference between the actual label and the predicted label. This residual indicates the model's error, which is then minimized through optimization. The commonly used formula, the sum of squared residuals, is expressed as:

To reduce this error, derivatives of the function are calculated. However, in large-scale data scenarios (big data), this formula becomes inefficient due to high memory requirements for storing and processing the entire dataset. To overcome this, a stochastic approach is applied by randomly selecting values from the available data [20].

$$Sum\ of\ squared\ residuals = \sum_{n-1}^{m} (y - \hat{y})^2 \tag{6}$$

The parameter update equation in the SGD approach is written as:

$$w^{(r+1)} = w^r - \alpha \nabla f_{i_r}(w^r) \tag{7}$$

Random sampling, as suggested in references [21, 22] helps conserve memory by avoiding full summations. Gradient updates are performed iteratively, allowing the model to find the optimal point more efficiently [23]. While it reduces computational complexity and memory usage, this method can result in non-zero gradient noise [20].

4) Adamax

Adamax represents an enhanced variation of the Adam optimizer, distinguished by its reliance on the infinity norm (L∞) of gradients to update model parameters. Like Adam, it utilizes an exponential moving average of the first-order moment, but replaces the second moment with the highest absolute gradient value. This mechanism enables dynamic adjustment of the learning rate per parameter, guided by both the mean and the maximum gradient values. Such flexibility improves its ability to sustain stable learning, even when large variations in gradients occur—something commonly encountered during deep learning processes. In contrast to Adam, one of Adamax's primary benefits is its improved reliability when working with noisy or sparsely distributed gradients. This resilience makes it especially suitable for applications where gradient intensities differ greatly across parameters. In addition, Adamax minimizes memory usage by omitting the storage of squared gradient values, which lowers the computational burden [24]. Within CNN structures, Adamax also supports faster model convergence by modulating learning rates based on gradient strength—a method that proves advantageous for complex datasets containing features of unequal importance [25, 26].

3. RESULT

3.1 Experimental scenarios

The proposed method employs a multi-model CNN architecture that integrates MobileNet V3, EfficientNet V2, and ShuffleNet V2 as parallel feature extractors which is shown in Figure 1 and Figure 2. This multi-model configuration is designed to leverage the complementary characteristics of lightweight CNN architectures, enabling more robust feature representation while maintaining computational efficiency. The total number of parameters of the proposed multi-model CNN architecture is 26,472,039, which remains feasible for practical implementation.

The dataset is partitioned using an 80:20 data splitting strategy, consisting of training, testing, and validation subsets. Specifically, 2,591 images are allocated for training, 810 images for testing, and 648 images for validation. This data partitioning strategy ensures effective model training while allowing for reliable and unbiased performance evaluation on unseen data.

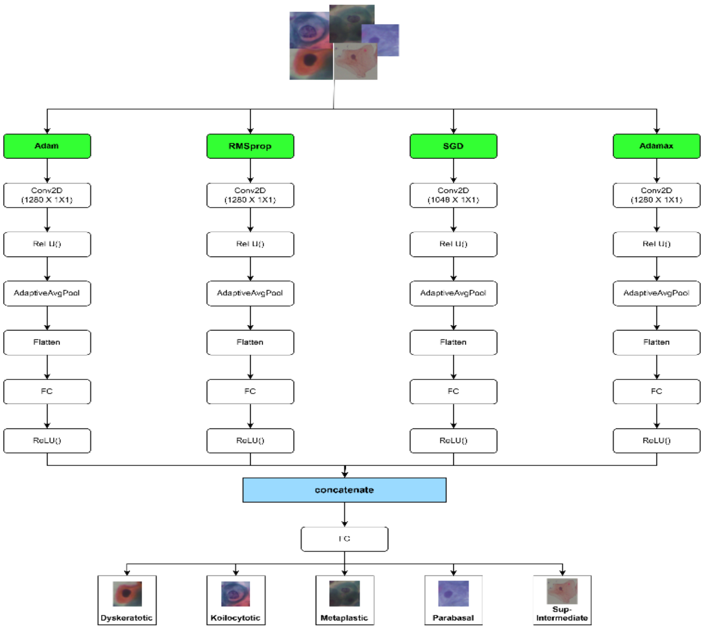


Figure 1. System flow

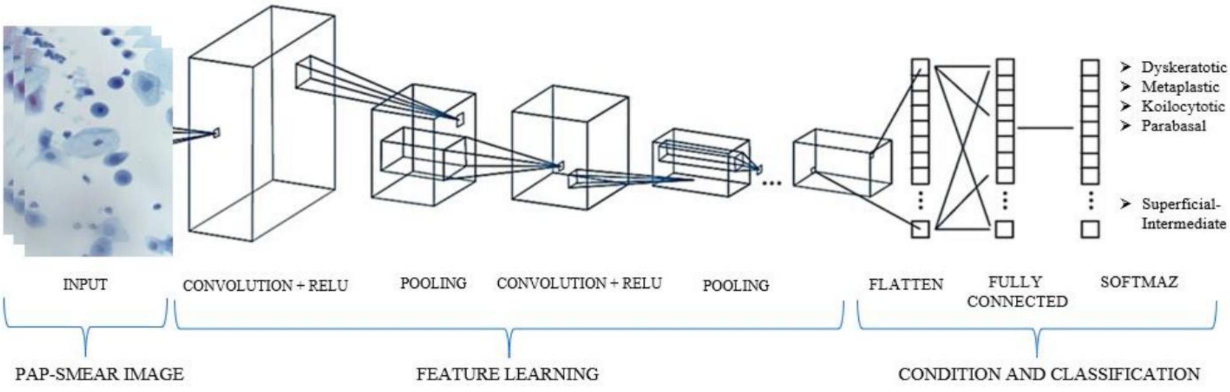


Figure 2. CNN architecture for cervical cancer classification
Note: CNN = convolutional neural network.

Table 1. Classification scenario testing with optimizer variations

Scenario	Learning Rate (LR)	Optimizer
Sc 1	0.01	Adam
Sc 2	0.01	RmsProp
Sc 3	0.01	SGD
Sc 4	0.01	Adamax

Note: RMSProp = Root Mean Square Propagation; SGD = Stochastic Gradient Descent.

To obtain the best performance from the cancer classification model, several test scenarios were conducted using different optimizer algorithms with a fixed learning rate of 0.01. The four tested scenarios are shown in Table 1.

3.2 Model training results

The four scenarios were tested using a preprocessed cancer dataset. The objective of these tests was to assess the impact of optimizer selection on model accuracy and overall performance. The results of each scenario were compared based on evaluation metrics such as accuracy, precision, recall, and F1-score, as explained below.

3.2.1 Scenario 1: Optimizer adam

In Scenario 1, the cancer classification model was trained and evaluated using several metrics such as loss, accuracy, precision, recall, and F1-score on both training and validation data can be seen in the Table 2. The generated graphs and metrics showed a performance improvement trend as training epochs progressed. At the beginning of training (Figure 3), the training loss was relatively high at 1.238, while the validation loss peaked at 2104.658, indicating that the model had not yet learned effectively. Over time, these losses decreased, with the minimum training loss reaching 0.226 and the validation loss dropping to 0.529, suggesting increased model stability and better generalization.

As shown in Figure 4, the training accuracy improved steadily, reaching a peak of 92.24%, while validation accuracy achieved 87.35%, indicating that the model could recognize patterns in new data. The highest training precision was 92.28%, reflecting accurate identification of positive cases. The recall peaked at 92.24%, showing the model's capability to detect most positive instances. The F1-score reached 92.26%, indicating a good balance between precision and recall.

Table 2. Test results-Adam optimizer

	Training	Validation
Loss	0.226%	0.529%
Accuracy	92.24%	87.35%
Precision	92.28%	87.48%
Recall	92.24%	87.26%
F1_Score	92.26%	87.11%

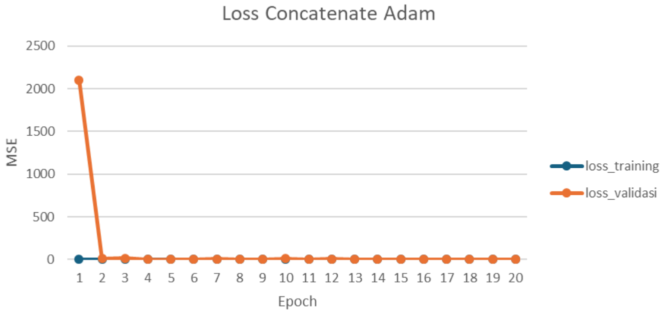


Figure 3. Adam concatenate loss graph results

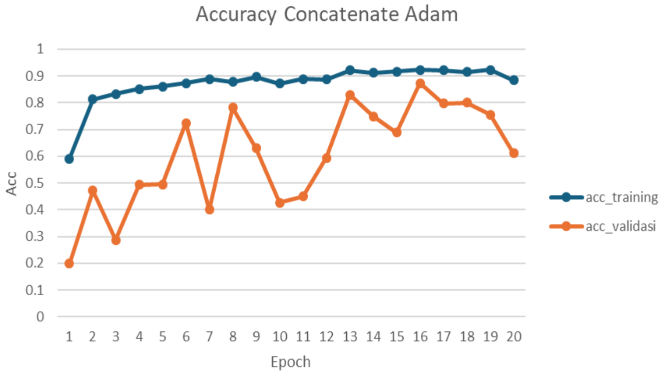


Figure 4. Accuracy concatenate adam graph results

Despite fluctuations during several epochs, these maximum values indicate that the model generalized well, as seen in Figure 4. The confusion matrix visualization also supported these results, showing that most classifications were accurate. For validation data in Figure 5, the maximum precision recorded was 89.48%, with a recall of 87.26% and an F1-score of 87.11%.

In summary, Scenario 1 demonstrated strong cancer classification results, especially after several training epochs, achieving high performance with metrics close to ideal.

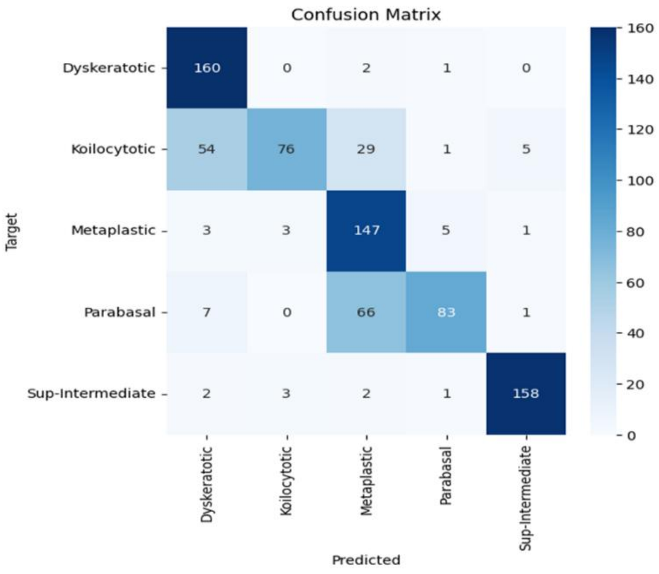


Figure 5. Adam's confusion matrix results

3.2.2 Scenario 2: Optimizer RMSProp

In Scenario 2, the cancer classification model was re-evaluated using various performance metrics, including loss, accuracy, precision, recall, and F1-score, on both the training and validation datasets. At the beginning of training, the model exhibited signs of instability, as reflected by relatively high loss values. However, as the number of epochs increased, the model demonstrated significant improvement. By the end of the training phase, the training loss had been successfully reduced to 0.212, while the validation loss decreased to 0.261, as shown in Figure 6. This reduction indicates that the model gradually improved its understanding of data patterns and enhanced its predictive capability.

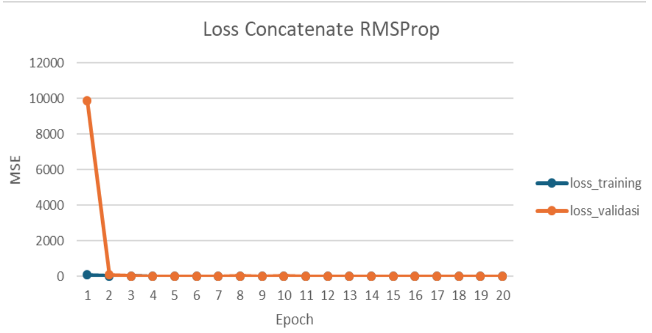


Figure 6. RMSProp concatenate loss graph results
Note: RMSProp = Root Mean Square Propagation.

The model's accuracy, as illustrated in Figure 7, was also quite impressive. The training accuracy reached 93.17%, and the validation accuracy climbed to 90.90%. These results suggest that the model was not only capable of recognizing patterns in the training data but also able to generalize well to previously unseen data. In terms of other metrics, the model achieved a training precision of 93.16%, a recall of 93.19%, and an F1-score of 93.17%, reflecting a very well-balanced performance between precision and sensitivity in detecting positive cases.

Table 3. Results of the RMSProp optimizer test scenario

	Training	Validation
Loss	0.212%	0.261%
Accuracy	93.17%	90.90%
Precision	93.16%	91.18%
Recall	93.19%	90.91%
F1_Score	93.17%	90.63%

Note: RMSProp = Root Mean Square Propagation.

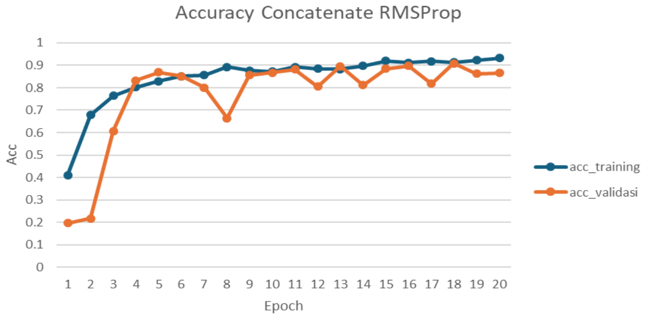


Figure 7. RMSProp concatenate accuracy graph results
Note: RMSProp = Root Mean Square Propagation.

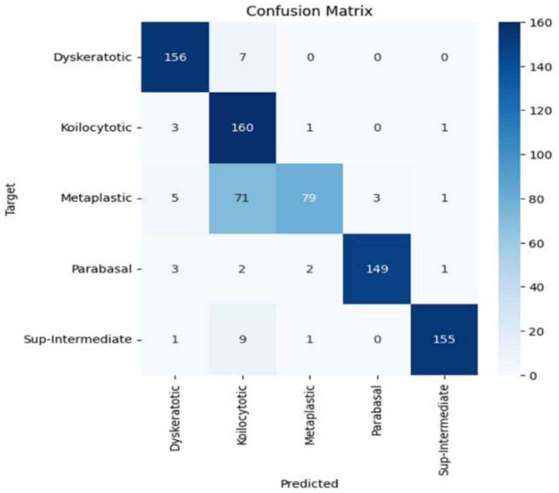


Figure 8. RMSProp confusion matrix results
Note: RMSProp = Root Mean Square Propagation.

The performance on the validation set was similarly strong, with a precision of 91.18%, recall of 90.91%, and F1-score of 90.63%. These values demonstrate that, despite being trained on a limited dataset, the model was still able to generalize effectively and identify new patterns with commendable accuracy. This is further supported by the confusion matrix in Figure 8, which shows that the majority of predictions were correctly classified.

Overall, Scenario 2 produced excellent results. With consistently high metrics across both training and validation datasets, the model proved to be accurate and reliable in classifying cancer data, as shown in Table 3.

3.2.3 Scenario 3: Optimizer SGD

In Scenario 3, the cancer classification model was further evaluated using key performance metrics such as loss, accuracy, precision, recall, and F1-score on both the training and validation datasets. From the beginning of the training phase, the model demonstrated a consistent upward trend in performance. As shown in Figure 9, the training loss, which was initially high, was successfully reduced to 0.113 by the end of training. Meanwhile, the validation loss also dropped significantly to 0.202, indicating that the model became increasingly effective at recognizing data patterns.

The model's accuracy, illustrated in Figure 10, was particularly impressive. Training accuracy reached 96.06%, while validation accuracy achieved 94.44%. These results confirm that the model not only learned effectively from the training data but also applied its knowledge successfully to new, unseen data.

Table 4. Results of SGD optimizer test scenarios

	Training	Validation
Loss	0.113%	0.202%
Accuracy	96.06%	94.44%
Precision	96.07%	94.74%
Recall	96.06%	94.48%
F1_Score	96.07%	94.50%

Note: SGD = Stochastic Gradient Descent.

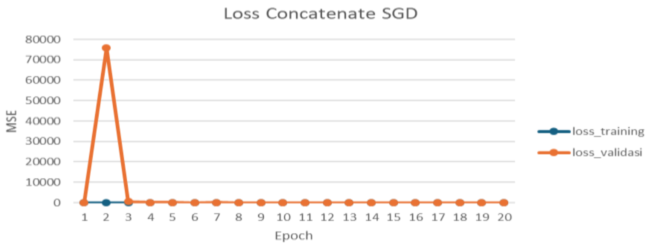


Figure 9. SGD concatenate loss graph results
Note: SGD = Stochastic Gradient Descent.

Regarding other metrics, the training precision was 96.07%, recall was 96.06%, and F1-score was 96.07%—demonstrating an excellent balance between the model's precision and sensitivity in identifying cancer cases. Similar outcomes were observed in the validation set, with precision at 94.74%, recall at 94.48%, and F1-score at 94.50%. These values indicate that even with a limited dataset, the model maintained high generalization capability and delivered reliable predictions.

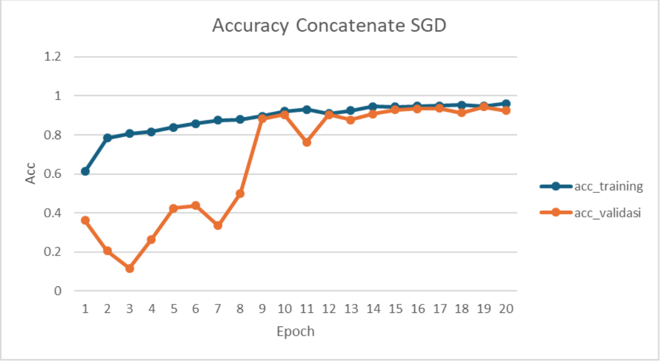
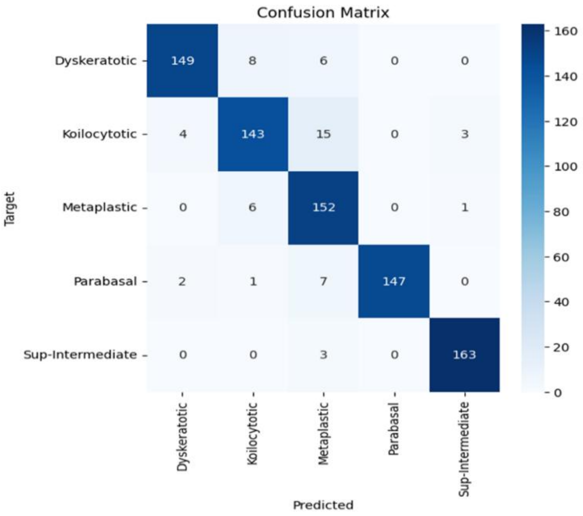


Figure 10. SGD concatenate accuracy graph results

The trend of performance metrics throughout training and validation further supports the model's effectiveness. As shown in Figure 11, the steady decline in loss and consistent rise in accuracy across epochs reflect an efficient and stable learning process. By the final epoch, the validation accuracy surpassed 94%, underscoring the model’s stability and dependability.

Overall, Scenario 3 demonstrates a highly solid and promising model performance. With very high and consistent metrics across both training and validation datasets, the model achieved excellent accuracy and precision in cancer classification, while keeping overfitting to a minimum. This makes it one of the best-performing models tested in this study so far, as shown in Table 4.



	Dyskeratotic	Koilocytotic	Metaplastic	Parabasal	Sup-Intermediate
Dyskeratotic	149	8	6	0	0
Koilocytotic	4	143	15	0	3
Metaplastic	0	6	152	0	1
Parabasal	2	1	7	147	0
Sup-Intermediate	0	0	3	0	163

Figure 11. SGD confusion matrix results

3.2.4 Scenario 4: Optimizer adamax

In Scenario 4, the cancer classification model was evaluated using various performance metrics, including loss, accuracy, precision, recall, and F1-score on both training and validation datasets. Throughout the training process, the model exhibited excellent progress. As shown in Figure 12, the training loss was initially relatively high at approximately 0.773, but it decreased significantly as the number of epochs increased, reaching 0.043 by the end of training. This substantial reduction in loss indicates that the model became increasingly effective at learning and recognizing data patterns.

In terms of accuracy (Figure 13), the model’s performance was remarkable. The training accuracy steadily increased, eventually reaching 98.53%, while the validation accuracy also remained high at 96.30%. This suggests that the model was not only capable of learning from the training data but also generalized well to new, unseen data.

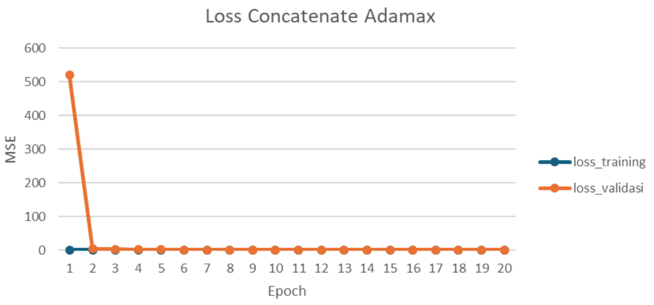


Figure 12. Adamax concatenate loss graph results

The precision, recall, and F1-score metrics also demonstrated consistently high values. On the training dataset, all three metrics reached 98.54%, indicating a well-balanced ability to accurately predict and detect positive cases. For the validation data, precision, recall, and F1-score values ranged from 96.30% to 96.33%, confirming the model’s reliability and effectiveness in broader application scenarios as shown in

Figure 14.

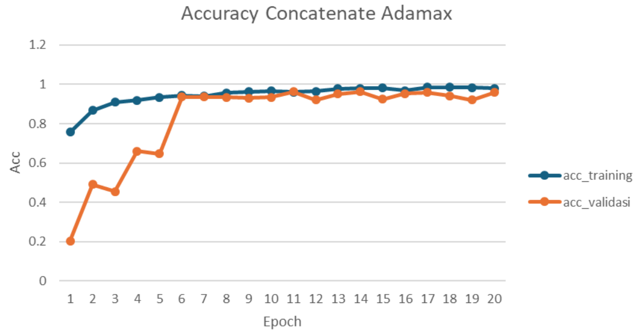


Figure 13. Accuracy concatenate adamax graph results

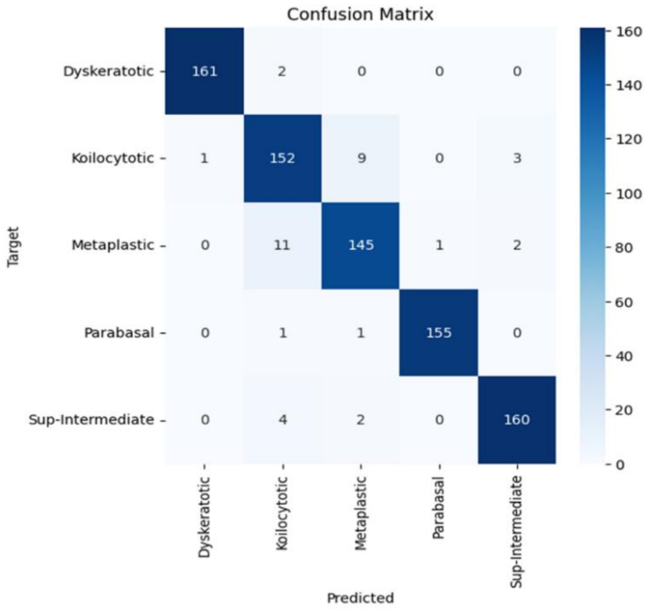


Figure 14. Adamax confusion matrix results

Table 5. Results of the adamax optimizer test scenario

	Training	Validation
Loss	0.043%	0.092%
Accuracy	98.53%	96.30%
Precision	98.54%	96.30%
Recall	98.54%	96.33%
F1_Score	98.54%	96.30%

Overall, Scenario 4 showed a very strong and stable model performance, with consistently high metrics across both training and validation phases. These results indicate that the developed cancer classification model is capable of producing accurate and reliable predictions, making it a valuable tool for supporting data-driven medical diagnosis, as shown in Table 5.

4. CONCLUSIONS

Cervical cancer is one of the leading causes of premature death among women worldwide, particularly in low- to middle-income countries. Early detection and accurate diagnosis are key factors in reducing the mortality rate associated with this disease. However, traditional diagnostic methods such as manual cytology analysis still face numerous limitations, including subjectivity and susceptibility to human error. Therefore, the use of artificial intelligence technology—particularly deep learning—has emerged as a promising solution.

In this study, a multi-model approach was implemented by combining four popular CNN architectures—Adam, RMSProp, SGD, and Adamax—to enhance the performance of cervical cell image classification. This approach enables the extraction of richer and more in-depth feature representations, which are then classified to detect various types of abnormal cells. Four experimental scenarios were conducted to evaluate the model's performance using different optimizer algorithms (Adam, RMSProp, SGD, and Adamax), all with a fixed learning rate of 0.01. The experimental results revealed that the model using the SGD optimizer (Scenario 3) achieved the best performance, with the highest validation accuracy of 94.44%, as well as exceptionally high and balanced precision, recall, and F1-score values (each exceeding 94%).

The experimental results demonstrate that the combination of a multi-model CNN architecture (MobileNet V3, EfficientNet V2, and ShuffleNet v2) and appropriate optimization techniques significantly enhances both accuracy and reliability in cervical cancer classification. This deep learning approach can assist clinicians by providing efficient, consistent, and objective diagnostic support, thereby showing considerable potential for early detection and improved women's healthcare outcomes.

ACKNOWLEDGMENT

For guiding this study endeavor, we are grateful to the University of Trunojoyo Madura. For supporting this research endeavor, we are grateful to INTI University Malaysia, University of Airlangga, and Institut Teknologi Adhi Tama Surabaya. We also express our gratitude to the Publication Assistance Program by the Directorate General of Research and Development of the Ministry of Education and Science and Technology No. 090/SPK/C3/DT.05.00/BP/2025 dated December 12, 2025.

REFERENCES

[1] Ghoneim, A., Muhammad, G., Hossain, M.S. (2020). Cervical cancer classification using convolutional neural networks and extreme learning machines. Future Generation Computer Systems, 102: 643-649. https://doi.org/10.1016/j.future.2019.09.015

[2] Nithya, B., Ilango, V. (2019). Evaluation of machine learning based optimized feature selection approaches and classification methods for cervical cancer prediction. SN Applied Sciences, 1(6): 641. https://doi.org/10.1007/s42452-019-0645-7

[3] Kusumawardani, L.A., Rulaningtyas, R., Winarno, W. (2023). Classification of cervical cancer cells using the K-nearest neighbor (KNN) method based on geometric feature extraction. AIP Conference Proceedings, 2858(1): 030003. https://doi.org/10.1063/5.0167165

[4] Mustari, A., Ahmed, R., Tasnim, A., Juthi, J.S., Shahariar, G.M. (2023). Explainable contrastive and cost-sensitive learning for cervical cancer classification. In 2023 26th International Conference on Computer and

Information Technology (ICCIT), Cox's Bazar, Bangladesh, pp. 1-6. https://doi.org/10.1109/ICCIT60459.2023.10441352

[5] Hapsari, D.P., Rochman, E.M.S., Asmara, Y.P., Rachmad, A., Setiawan, W. (2025). Automated detection of knee osteoarthritis using CNN with adaptive moment estimation. Mathematical Modelling of Engineering Problems, 12(4): 1126-1136. https://doi.org/10.18280/mmep.120404

[6] Hamouda, M., Saheb Ettabaa, K., Bouhlel, M.S. (2019). Hyperspectral imaging classification based on convolutional neural networks by adaptive sizes of windows and filters. IET Image Processing, 13(2): 392-398. https://doi.org/10.1049/iet-ipr.2018.5063

[7] Li, Y.S., Xie, W.Y., Li, H.Q. (2017). Hyperspectral image reconstruction by deep convolutional neural network for classification. Pattern Recognition, 63: 371-383. https://doi.org/10.1016/j.patcog.2016.10.019

[8] Nawrocka, A., Nawrocki, M., Kot, A. (2023). Research study of image classification algorithms based on convolutional neural networks. In 2023 24th International Carpathian Control Conference (ICCC), Miskolc-Szilvásvárad, Hungary, pp. 299-302. https://doi.org/10.1109/ICCC57093.2023.10178933

[9] Patil, A., Rane, M. (2020). Convolutional neural networks: An overview and its applications in pattern recognition. In International Conference on Information and Communication Technology for Intelligent Systems, pp. 21-30. https://doi.org/10.1007/978-981-15-7078-0_3

[10] Khozaimi, A., Mahmudy, W.F. (2024). New insight in cervical cancer diagnosis using convolution neural network architecture. arXiv Preprint arXiv: 2410.17735. https://doi.org/10.48550/arXiv.2410.17735

[11] Lakshmi, G.K., Krishnaveni, K. (2016). Feature extraction and feature set selection for cervical cancer diagnosis. Indian Journal of Science and Technology, 9(19): 1-7. https:/.doi.org.10.17485/ijst/2016/v9i19/93881

[12] Albattah, W., Javed, A., Nawaz, M., Masood, M., Albahli, S. (2022). Artificial intelligence-based drone system for multiclass plant disease detection using an improved efficient convolutional neural network. Frontiers in Plant Science, 13: 808380. https://doi.org/10.3389/fpls.2022.808380

[13] Chandriah, K.K., Naraganahalli, R.V. (2021). RNN/LSTM with modified Adam optimizer in deep learning approach for automobile spare parts demand forecasting. Multimedia Tools and Applications, 80(17): 26145-26159. https://doi.org/10.1007/s11042-021-10913-0

[14] Yi, D., Ahn, J., Ji, S. (2020). An effective optimization method for machine learning based on ADAM. Applied Sciences, 10(3): 1073. https://doi.org/10.3390/app10031073

[15] Ashqar, B.A., Abu-Naser, S.S. (2019). Identifying images of invasive hydrangea using pre-trained deep convolutional neural networks. International Journal of Academic Engineering Research (IJAER), 3(3): 28-36. https://ssrn.com/abstract=3369016.

[16] Patterson, J., Gibson, N.A. (2016). Deep learning a practitioner's approach, 29(7553). http://deeplearning.net/.

[17] Saurabh, N. (2020). LSTM-RNN model to predict future stock prices using an efficient optimizer. International Research Journal of Engineering and Technology, 7(11): 672-677.

[18] Andika, L.A., Pratiwi, H., Handajani, S.S. (2019). Klasifikasi penyakit pneumonia menggunakan metode convolutional neural network dengan optimasi adaptive momentum. Indonesian Journal of Statistics and Its Applications, 3(3): 331-340. https://doi.org/10.29244/ijsa.v3i3.560

[19] Rachmad, A., Chamidah, N., Rulaningtyas, R. (2020). Mycobacterium tuberculosis images classification based on combining of convolutional neural network and support vector machine. Communications in Mathematical Biology and Neuroscience. https://doi.org/10.28919/cmbn/5035

[20] Bera, S., Shrivastava, V.K. (2020). Analysis of various optimizers on deep convolutional neural network model in the application of hyperspectral remote sensing image classification. International Journal of Remote Sensing, 41(7): 2664-2683. https://doi.org/10.1080/01431161.2019.1694725

[21] Lei, Y., Hu, T., Tang, K. (2021). Generalization performance of multi-pass stochastic gradient descent with convex loss functions. Journal of Machine Learning Research, 22(25): 1-41.

[22] Smith, S.L., Dherin, B., Barrett, D.G., De, S. (2021). On the origin of implicit regularization in stochastic gradient descent. arXiv Preprint arXiv: 2101.12176. https://doi.org/10.48550/arXiv.2101.12176

[23] Wojtowytsch, S. (2023). Stochastic gradient descent with noise of machine learning type part i: Discrete time analysis. Journal of Nonlinear Science, 33(3): 45. https://doi.org/10.1007/s00332-023-09903-3

[24] El-Shafai, W., Mahmoud, A.A., El-Rabaie, E.S.M., Taha, T.E., Zahran, O.F., El-Fishawy, A.S., Abd-Elnaby, M., Abd El-Samie, F.E. (2022). Efficient deep CNN model for covid-19 classification. Computers, Materials & Continua, 70(3): 4373-4391. https://doi.org/10.32604/cmc.2022.019354

[25] Taqi, A.M., Awad, A., Al-Azzo, F., Milanova, M. (2018). The impact of multi-optimizers and data augmentation on TensorFlow convolutional neural network performance. In 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, FL, USA, pp. 140-145. https://doi.org/10.1109/MIPR.2018.00032

[26] Manoj, R.K., Batumalay, M. (2025). A hybrid CNN-transformer model with quantum-inspired fourier transform for accurate skin disease classification. Journal of Applied Data Sciences, 6(3): 1940-1952. https://doi.org/10.47738/jads.v6i3.782