Health, Empathy, and AI Learning (HEAL)
Vol. 1, No. 1, November, 2025, pp. 7−16
E-ISSN: 3163-267X | P-ISSN: 3163-2661

A Novel Deep Learning Technique for Medical Image Analysis Using Improved Optimizer

Vertika Agarwal[1*], M.C. Lohani[2], Qurotul Aini[3], Nesti Anggraini Santoso[4]
[1, 2]Computer Sc. & Engg, Graphic Era Hill University, India
[3,4]Faculty of Economics and Business, University of Raharja, Indonesia
[1]vertika.agarwal21@gmail.com, [2]getmlohani@gmail.com, [3]aini@raharja.info, [4]nesti@raharja.info
***Corresponding Author**

Article Info

Article history:

Submission August 20, 2025
Revised September 25, 2025
Accepted October 27, 2025
Published November 03, 2025

Keywords:

Gradient Centralization
Medical image Analysis
Real Esrgan
GFPGAN

ABSTRACT

Application of Convolutional neural network in spectrum of Medical image analysis are providing benchmark outputs which converges the interest of many researchers to explore it in depth. Availability of clean dataset is one of the significant factor which drives the efficiency of CNN (Convolutional neural network) model. Latest preprocessing technique Real Esrgan and GFPGAN (Generative facial prior GAN) are proving their mettle in providing high resolution dataset. Optimizer also plays a vital role in upgrading the functioning of CNN model. A new optimization technique Gradient Centralization is providing the unparalleled result in terms of generalization and execution time. Our research delves deep into aforementioned factors. Result has already been achieved by taking in account the first factor which is preprocessing technique. In continuation of our effort, our paper explores the next factor which is the employment of new optimization technique, Gradient centralization to our integrated framework (Model with advanced preprocessing technique). Preprocessing integrated Densenet 201, and NasNet are giving encouraging result in terms of training loss and execution time when GC is used as optimizer. Experiments are carried out for three different types of datasets: Skin, Retina and Lung and comparison of loss curves for both preprocessing integrated models is carried out.

1. INTRODUCTION

Integrated framework of deep learning model [1] with advanced preprocessing technique like Real Esrgan [2] and GFPGAN [3] gives significant elevation in terms of efficiency of these models. Efficiency shoots up to 5-10% via pipe lining of these preprocessing technique with base models. On a way to further optimize the deep learning models, usage of optimizer is explored. Optimizer plays an indispensable role in the functioning of CNN. They help in adjusting weights so as to minimize the incurred losses. Many optimizers have been employed in CNN like Gradient descent [4], Stochastic Gradient descent [5], Ada grad [6], Ada delta and Adam etc. for classification and segmentation of Medical image. Gradient descent iteratively reduces a loss function by moving in the direction opposite to that of steepest ascent. As its uses entire training set for calculation which requires large amount of memory and hence slow down process. Stochastic gradient descent is a variant of gradient descent and update model parameter one by one which leads to high variance and make it computationally expensive. Stochastic gradient descent with momentum is an improved version which takes in account the previous update to fine tune the final update direction and hence leads to more stability but extra hyper parameter is added. So far learning rate has been constant but with the introduction of Ada grad (Adaptive Gradient descent) concept of adaptive learning rate emerges i.e. it uses different learning rate for each and every neuron and for each and every hidden layer based on different iteration but it too leads to dead neuron problem. RMS-prop [7] (Root mean square propagation) is a special version of Ada grad in which learning rate is exponential average of gradient instead of cumulative sum of squared gradient but it suffers from slow learning rate. Ada delta is an extension of Adagrad. Here it, does not set default learning rate and uses the ratio of running average of previous time steps to current gradient. It removes decaying rate problem but it is computationally expensive. Adam (Adaptive moment estimator) optimizer is one of the popular gradient descent optimization algorithm. It computes adaptive learning rate for each parameter and stores both the decaying average of past gradients and also the decaying average of past squared gradient. It has little memory requirement and is computationally efficient.

Aforementioned optimization techniques operate on activation or weight vectors and hence their adoption to pre-trained models pose a great challenge. So a new method, Gradient centralization [8] is used which acts on gradient of weight vectors and centralizes the gradient vector to have zero mean. Thus introduction of Gradient centralization accelerate the training process and enhances the generalization performance. Our research work explores the functioning of GC (Gradient centralization) on our integrated framework of deep learning models [9]. Section 2. illustrates the related work, Section 3. describes the achieved result via integrated framework and proposed work which includes further optimization with novel Gradient Centralization optimizer. Section 4. tabulates the experimental result and comparison of loss curves and Section 6. winds the research paper with Future work and conclusion.

2. RELATED WORK

ESignificant efforts are made in the direction of improving the efficiency of deep learning models. Application of voting based method to deep learning models helps in strengthening the correct classification of lung disease [10]. Preprocessing techniques plays an important role in boosting efficacy of deep learning models [11]. Other factor like Optimizer too plays an pertinent role in efficiency improvement of Deep learning algorithms as they decide the rate of convergence of algorithm towards an optimal value. Many Researchers are evolving the usage of new optimizers to explore their functioning in deep learning models. [6] proposes method, normalized direction-preserving Adam (ND-Adam), which enables more precise control of the direction and step size for updating weight vectors, which leads to the significant improvement in generalization performance of adam optimizer. [12] brings out another improvement in momentum of adam optimizer by evaluating the gradient after applying the current velocity and called its as Adam. It works efficiently for the high-dimensional dataset and converges rapidly. Usage of appropriate optimizer for particular input dataset greatly affects the efficiency of architecture used for classifying that dataset.

[13] carried out comparison of six optimizer Ada grad, Ada delta, Rms prop, Adam, Nadam, and SGD (Stochastic gradient descent) and identify the best optimizer as Adagrad which estimate potential fire occurrences in given locations and gives precise predictions with less error and processes real-time data. [14] improves the efficiency of its CNN based classifier by comparing six different first-order stochastic gradient-based optimizers to select the adaptative based optimizers as best for classifying histopathology imagesand Ada grad as low performer. [15] perform a comparative analysis of 10 different gradient descent-based optimizers, Adaptive Gradient (Adagrad), Adaptive Delta (AdaDelta), Stochastic Gradient Descent (SGD), Adaptive Momentum (Adam), Cyclic Learning Rate (CLR), Adaptive Max Pooling (Ada max), Root Mean Square Propagation (RMS Prop), Nesterov Adaptive Momentum (Nadam), and Nesterov accelerated gradient (NAG) for CNN for brain tumor classification and segmentation and Adam optimizer helps in attaining the greatest efficiency of 99.2%. [16] explores theCNN with one, two, three, four and five hidden layers with combination of three optimizers namely, Rms prop, Adam and SGD and finds CNN with four hidden layer using SGD optimizer having greatest testing accuracy of 91%. [17] performs Hyper spectral image (HSI) classification with seven different optimizers SGD, Adagrad, Adadelta, Rms prop, Adam, AdaMax, and Nadam using Deep CNN model [18] and establishes the superiority of Adam optimizer. [19] illustrates that hyper parameter optimizer plays indispensable role in in deep learning-based side-channel analysis and strongly supports Adagrad for longer training phases. [20] used CNN based model Resnet 50 and Inception v3 for kaggle cat versus dog classification and carried out comparative analysis for three optimizer namely Adam, SGD, RMS prop and RMS prop performed extremely well with training accuracy of 99%.

[21] implemented the usage of seven optimizers namely Stochastic Gradient Descent (SGD), RmsProp, Adam, Adamax, Adagrad, Adadelta, and Nadam are implemented in CNN on Indian Pines Dataset Adamax excels with 99.58% accuracy. [22] supports the Adam optimizer for Brain tumor classification and segmentation among the ten optimizer namely namely Adaptive Gradient (Adagrad), Adaptive Delta (AdaDelta), Stochastic Gradient Descent (SGD), Adaptive Momentum (Adam), Cyclic Learning Rate (CLR), Adaptive Max Pooling (Adamax), Root Mean Square Propagation (RMS Prop), Nesterov Adaptive Momentum (Nadam), and Nesterov accelerated gradient (NAG) employed to carry out the experiment. [23] works out on step size taken for each parameter and proposed an Diffgrad optimization technique which takes larger step size for faster gradient changing parameter and lower step size for lower gradient changing parameter. This method outperforms optimizer technique like SGDM, AdaGrad, AdaDelta, RMSProp, AMSGrad, and Adam. [24] proposes Alzheimer Disease (AD) classification based by using four different optimizers Adagrad, ProximalAdagrad, Adam, and RMSProp and Rms prop works with 100% accuracy. [25] experimented with various optimizers such as ADAM, SGD, RMSprop, Adadelta, Adamax, Adagrad, Nadam to detect bot accounts with the help of deep learning model from CRESCI-2017 twitter dataset issued by Indiana University and establishes the highest accuracy of 98.90% with RMS prop. [26] bring out the role of optimizer in improving the performance of deep neural network for image classification problem and analyzes three standard first-order optimizers like stochastic gradient descent with momentum (SGDM), adaptive moment estimation (Adam), and root mean square propagation (RMSProp) for detecting glaucoma employing architectures like Alex net, VGG-19 and Resnet 101. Adam optimizer shows better result in this.

3. PROPOSED WORK

Our proposed work is the continuation of our effort to optimize the deep learning models. Imbibing Advanced preprocessing technique like Real Esrgan and GFP Gan with base deep learning model like Nas net and Densenet 201 produces an significant results in context of validation accuracy. Results achieved via this pipe lining is tabulated in Table 1 for Real Esrgan and GFP Gan respectively.

Table 1. Result of Integrated Model

REAL ESR-GAN	**SKIN**		**LUNG**		**RETINA**	
Model Used	**Without Prepro-cessing**	**With Prepro-cessing**	**Without Prepro-cessing**	**With Prepro-cessing**	**Without Prepro-cessing**	**With Prepro-cessing**
DENSENET 201	28%	72%	85%	89%	74%	88%
NASNET	35%	63%	77%	78%	73%	89%
GFPGAN	**SKIN**		**LUNG**		**RETINA**	
Model Used	**Without Prepro-cessing**	**With Prepro-cessing**	**Without Prepro-cessing**	**With Prepro-cessing**	**Without Prepro-cessing**	**With Prepro-cessing**
DENSENET 201	28%	62%	85%	86%	74%	89%
NASNET	35%	73%	77%	78%	73%	84%

For skin dataset massive improvement is achieved for validation accuracy. Retina dataset also responds well for integrated framework. Minimal enhancement is observed for lung dataset. But overall positive trends are observed using images preprocessed with Real Esrgan and GFPGAN as input to our base models.

Our next step to further optimize the integrated deep learning models with the use of novel optimizer Gradient centralization technique.

3.1. Gradient Centralization

Gradient centralization can enhance DNN (Deep neural network) ultimate generalisation performance while also accelerating the training process. By centralizing the gradient vectors to have a mean of zero, it directly manipulates gradients. Gradient Centralization further enhances the loss function and its gradient Lipschitzness, making the training process more effective and stable. Batch Normalization and Weight standardization technique also reduces the Lipschitzness of loss function but they operate on activation and weight vectors and were not able to adapt to pre trained models but Gradient Centralization takes in account the gradients vectors to achieve improvement. Z-score standardization can also be used to normalize the gradient, but normalising gradient does not increase training's stability. Instead, calculating the gradient vectors' means and then centralizing the gradients so that their means are zero effectively helps in achieving the desired result. This is the underlying principle of Gradient centralization method. It can be easily embedded into gradient based current optimization algorithm like Adam, SGDM with one line of code and helps in increasing training process, improves generalization capability and better tune it to pre trained models.

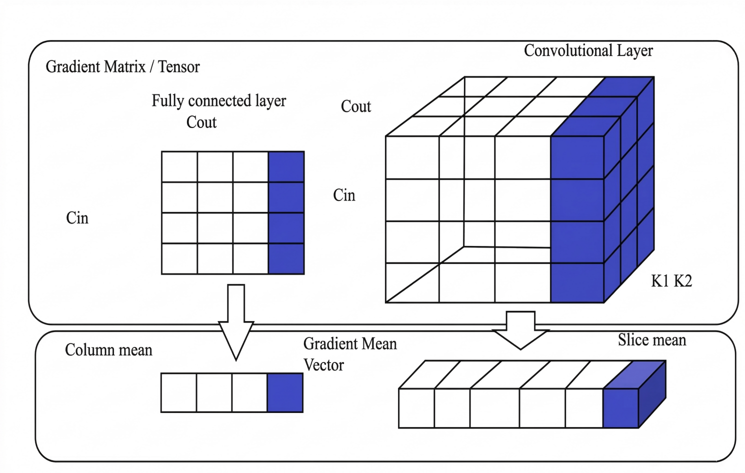

Figure 1. Mechanism of Gradient Centralization (GC) in Neural Networks

Note: GC computes the column/slice mean of the gradient matrix/tensor and centralizes each column/slice to have a zero mean. This operation is applied to the weight matrix in the fully connected layer (left) and the weight tensor in the convolutional layer (right).

3.2. Formula of GC

Let Gradient of a weight vector obtained through back propagation for a Fully connected layer or Convolutional layer be $\nabla w_i L$ $(i = 1, 2, \ldots, N)$ and GC operator be denoted by Φ_{GC}, then formulation of Gradient centralization is given by:

$$\Phi_{GC}(\nabla w_i L) = \nabla w_i L - \mu \nabla w_i L \tag{1}$$

where

$$\mu \nabla w_i L = \frac{1}{M} \sum_{j=1}^{M} \nabla W_{i,j} L \tag{2}$$

Mean value of each column of gradient matrix is subtracted from its each column value. In this way each gradient of loss function w.r.t to weight vector is transform so that its mean becomes zero. Computation of GC is quite simple and efficient. Matrix formulation of equation 1 is also given by:

$$\Phi_{GC}(\nabla w L) = P \nabla w L \tag{3}$$

where

$$P = I - ee^T \tag{4}$$

and P is known as the **centering operator**, I is the identity matrix of order $M \times M$, and

$$e = \frac{1}{\sqrt{M}} \begin{bmatrix} 1 & 0 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ 0 & 0 & 0 & \ldots & 1 \end{bmatrix} \tag{5}$$

4. EXPERIMENTAL RESULTS AND DISCUSSION

In our proposed research work, integrated deep model(model pipeline with advanced preprocessing technique) is further enhanced by employing an new optimizer Gradient Centralization. GC can be easily embedded into existing optimization algorithm like SGDM, Adam and RMS prop by adding a single line of code. We have embedded GC with Rms prop. Densenet 201 and Nasnet deep models are used as base models. Three datasets have been used to carry out the experiment: Lung, Retina and skin. Inculcating GC in our integrated model brings out improvement in three parameters of model which are Training accuracy, Training Loss and Execution time. Result of our experiment for GFPGAN integrated models are tabulated in table 2, 3, and 4 for all three datasets.

Table 2. Result for Optimized GFPGAN for Lung Dataset

Model Used	**Accuracy**		**Loss**		**Execution Time (Sec.)**	
	Without GCTF	With GCTF	Without GCTF	With GCTF	Without GCTF	With GCTF
DENSENET201	91.875	94.0625	.214071	.05539	1393.98	647.126
NASNET	86.9565	90.625	.37491	.11726	1042.77	638.959

Table 3. Result for Optimized GFPGAN for Retina Dataset

Model Used	**Training Accuracy / Validation Accuracy**		**Training Loss / Validation Loss**		**Execution Time(Sec.)**	
	Without GCTF	With GCTF	Without GCTF	With GCTF	Without GCTF	With GCTF
DENSENET201	92.0792	99.0099	.227322	.0357533	1199.61	812.791
NASNET	90.625	95.3125	.3010	.101164	1654.65	842.769

Table 4. Result for Optimized GFPGAN for Skin Dataset

Model Used	**Training Accuracy / Validation Accuracy**		**Training Loss / Validation Loss**		**Execution Time(Sec.)**	
	Without GCTF	With GCTF	Without GCTF	With GCTF	Without GCTF	With GCTF
DENSENET201	98.3031	100	.04873	.009683	776.049	505.114
NASNET	96.6102	99.661	.09873	.0304052	738.592	511.332

Enhancement in Accuracy has been observed for all three datasets but retina dataset responds remarkably better as compared to other datasets. Training loss has also been reduced which appends to our positive result. Improvement in execution time is the major break through achieved via this new optimizer algorithm. Significant reduction has been observed as clear from table. Our result supports the underlying principle of GC which states of accelerating the training process.

Experiments for another pre-preprocessing technique Real Esrgan integrated deep learning model is carried out which also employed a new optimized method of Gradient Centralization and the results are tabulated in table 5, 6 and 7 respectively for three datasets: Skin, Retina and Lung.

Table 5. Result of Optimized Real Esrgan for skin dataset

Model Used	**Accuracy**		**Loss**		**Execution Time(Sec.)**	
	Without GCTF	With GCTF	Without GCTF	With GCTF	Without GCTF	With GCTF
DENSENET201	99.3548	100	.027379	.00645161	1613.32	1112.79
NASNET	98.38	99.3548	.0562655	.026688	1639.68	1113.87

Table 6. Result of Optimized Real Esrgan for Retina dataset

Model Used	**Accuracy**		**Loss**		**Execution Time(Sec.)**	
	Without GCTF	With GCTF	Without GCTF	With GCTF	Without GCTF	With GCTF
DENSENET201	97.11	98.75	.0714245	.0235976	1427.4	1049.87
NASNET	94.23	95	.17624	.10362	1877.56	1015.65

Table 7. Result of Optimized Real Esrgan for Lung dataset

Model Used	**Accuracy**		**Loss**		**Execution Time(Sec.)**	
	Without GCTF	With GCTF	Without GCTF	With GCTF	Without GCTF	With GCTF
DENSENET201	95.315	93.4375	.145507	.0640958	1849.38	848.866
NASNET	96.3021	97.704	.25616	.08136	2483.02	819.201

Upward trends are observed for integrated Densenet 201 in terms of accuracy for skin and retina dataset but slight down trend is noticed for lung dataset which shows an deviation from our expectation. Nasnet exhibit an similar trend for all three datasets. Training loss comes out with positive result and significant drop is observed for three datasets. Execution time too shows substantial drop which further strengthen the concept of imbibing this new optimized technique in our integrated model.

Evaluation of loss curve on imbibing Gradient Centralization technique for Dense net model pipe lined with Real esrgan and GfpGan for all three datasets: Retina, Lung and Skin brings out an interesting predictions as indicated by diagrams shown below:

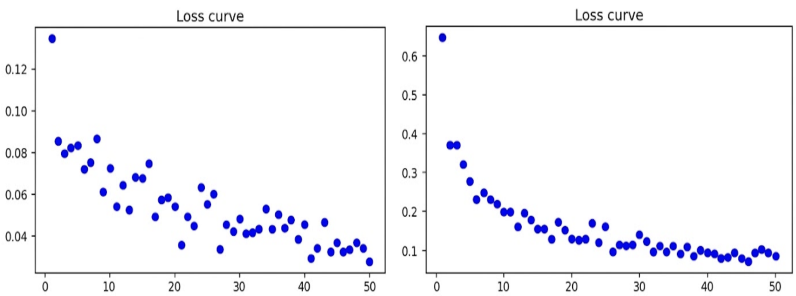

Figure 2. a)Loss curve for real esrgan pipe lined Densenet model for retina dataset b)Loss curve for GFPGAN pipe lined Densenet model for retina dataset

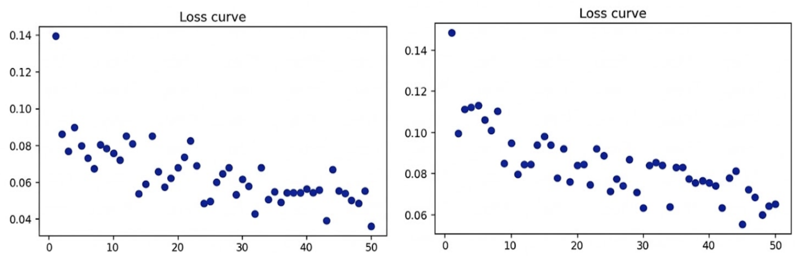

Figure 3. a)Loss curve for real esrgan pipe lined Densenet model for lung dataset b)Loss curve for GFPGAN pipe lined Densenet model for lung dataset

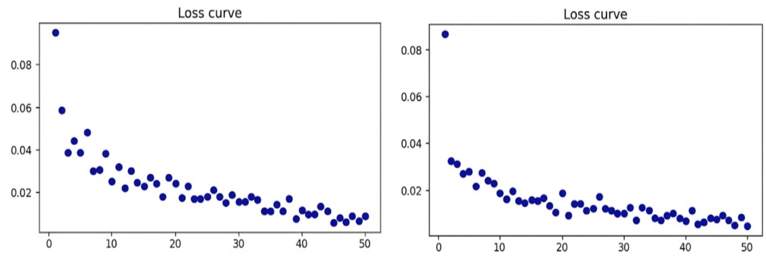

Figure 4. a)Loss curve for real esrgan pipe lined Densenet model for skin dataset b)Loss curve for GFPGAN pipe lined Densenet model for skin dataset

Experiments shows thatDensenet model when pipe lined with Real esrgan preprocessing technique for skin dataset gives optimum results for losses incurred during training which is .006451 and is minimum as compared with other combinations of dataset and technique.

Loss curves for combination of Gradient Centralization with Nasnet model pipelined with Real esrgan and Gfpgan technique are shown below for all three datasets.

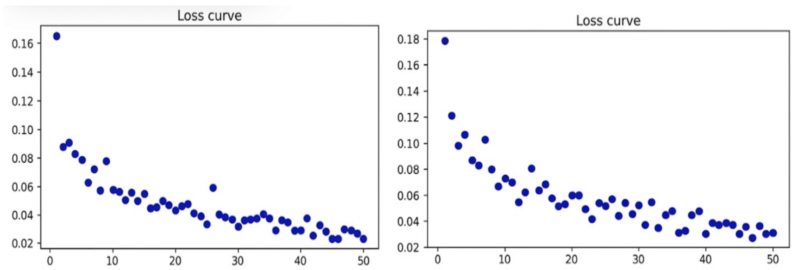

Figure 5. a)Loss curve for real Esrgan pipe lined Nasnet model for skin dataset b)Loss curve for GFPGAN pipe lined Nasnet t model for skin dataset

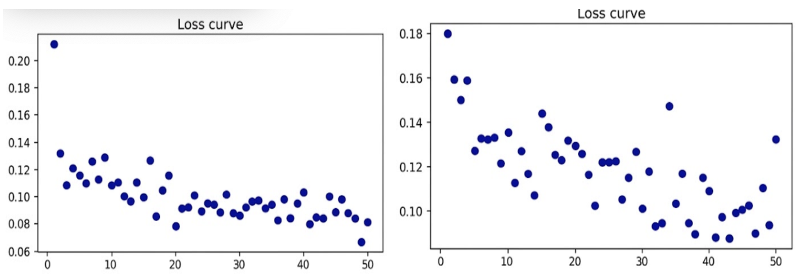

Figure 6. a)Loss curve for real Esrgan pipe lined Nasnet model for Lung dataset b)Loss curve for GFPGAN pipe lined Nasnet t model for lung dataset

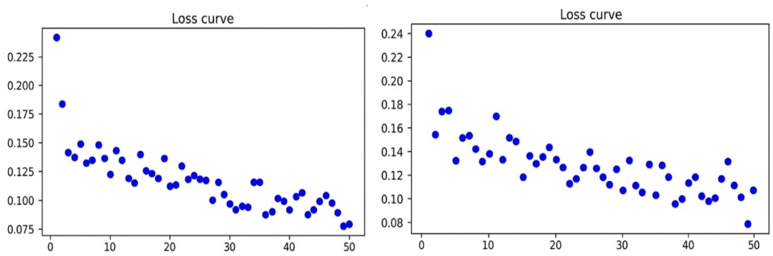

Figure 7. a)Loss curve for real Esrgan pipe lined Nasnet model for Retina dataset b)Loss curve for GFPGAN pipe lined Nasnet t model for Retina dataset

Smooth loss curves are observed for skin dataset when Nasnet pipe lined with Real esrgan is employed with GC is used. Though downward trend is shown by all datasets with both the preprocessing techniques but integrated real esrgan with GC shows better result than GFPGAN.

5. MANAGERIAL IMPLICATIONS

The integration of Gradient Centralization (GC) into deep learning frameworks provides critical advantages for healthcare technology management, primarily through the substantial reduction in execution time. This efficiency allows medical institutions to process large-scale diagnostic data more rapidly, optimizing computational resource allocation. Furthermore, the ease of embedding GC into existing optimizers like Adam or SGDM with minimal code adjustments facilitates seamless technological adoption. By enhancing generalization and reducing training loss, this approach offers a more stable and reliable foundation for deploying AI-driven diagnostic tools in clinical environments.

6. CONCLUSION

This research demonstrates that employing Gradient Centralization optimization on integrated architectures using NasNet and DenseNet 201 significantly improves performance in medical image analysis. Experimental results across skin, lung, and retina datasets show motivating progress in reducing training loss and enhancing computational speed. While minor accuracy deviations occurred in specific datasets, the findings suggest that this optimization approach is a vital step toward more effective medical imaging solutions. Future work should expand this framework to include alternative models such as ResNet, advanced preprocessing techniques like Swin Transformers, and a wider spectrum of medical datasets.

7. DECLARATIONS

7.1. About Authors

Vertika Agarwal (VA) https://orcid.org/0000-0003-1160-0610

M.C. Lohani (ML) https://orcid.org/0000-0002-3017-164X

Qurotul Aini (QA) https://orcid.org/0000-0002-7546-5721

Nesti Anggraini Santoso (NA) https://orcid.org/0009-0005-4568-9715

7.2. Author Contributions

Conceptualization: VA; Methodology: ML; Software: QA; Validation: VA and ML; Formal Analysis: NA and QA; Investigation: VA; Resources: ML; Data Curation: ML; Writing Original Draft Preparation: QA and NA; Writing Review & Editing: QA and NA; Visualization: ML; All authors, VA, ML, QA, and NA, have read and agreed to the published version of the manuscript.

7.3. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

7.4. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

7.5. Declaration of Conflicting Interest

The authors declare that they have no conflicts of interest, known competing financial interests, or personal relationships that could have influenced the work reported in this paper.

REFERENCES

[1] N. Nuryani, A. B. Mutiara, I. M. Wiryana, D. Purnamasari, and S. N. W. Putra, "Artificial intelligence model for detecting tax evasion involving complex network schemes," *Aptisi Transactions on Technopreneurship (ATT)*, vol. 6, no. 3, pp. 339–356, 2024.

[2] X. Wang, L. Xie, C. Dong, and Y. Shan, "Real-esrgan: Training real-world blind super-resolution with pure synthetic data," *arXiv preprint arXiv:2107.10833*, 2021.

[3] X. Wang, Y. Li, H. Zhang, and Y. Shan, "Towards real-world blind face restoration with generative facial prior," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9168–9178.

[4] E. Bisong, "Optimization for machine learning: Gradient descent," in *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Apress, 2019, pp. 203–207.

[5] I. Gitman, H. Lang, P. Zhang, and L. Xiao, "Understanding the role of momentum in stochastic gradient methods," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[6] Z. Zhang, "Improved adam optimizer for deep neural networks," in *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*. IEEE, 2018, pp. 1–2.

[7] D. V. Babu, C. Karthikeyan, and A. Kumar, "Performance analysis of cost and accuracy for whale swarm and rmsprop optimizer," *IOP Conference Series: Materials Science and Engineering*, vol. 993, no. 1, p. 012080, 2020.

[8] H. Yong, J. Huang, X. Hua, and L. Zhang, "Gradient centralization: A new optimization technique for deep neural networks," in *European Conference on Computer Vision*. Springer, 2020, pp. 635–652.

[9] E. T. Rusmiati, L. Febrina, Y. Sari, and E. M. S. Sakti, "Adoption of ai driven ecological preaching systems using sem pls analysis," *Aptisi Transactions on Technopreneurship (ATT)*, vol. 8, no. 1, pp. 284–295, 2026.

[10] V. Agarwal, M. C. Lohani, A. S. Bist, and D. Julianingsih, "Application of voting based approach on deep learning algorithm for lung disease classification," in *2022 International Conference on Science and Technology (ICOSTECH)*. IEEE, 2022, pp. 1–7.

[11] K. K. Pal and K. S. Sudeep, "Preprocessing for image classification by convolutional neural networks," in *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*. IEEE, 2016, pp. 1778–1781.

[12] U. M. Khaire and R. Dhanalakshmi, "High-dimensional microarray dataset classification using an improved adam optimizer (iadam)," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 11, pp. 5187–5204, 2020.

[13] M. N. Halgamuge, E. Daminda, and A. Nirmalathas, "Best optimizer selection for predicting bushfire occurrences using deep learning," *Natural Hazards*, vol. 103, no. 1, pp. 845–860, 2020.

[14] I. Kandel, M. Castelli, and A. Popovič, "Comparative study of first order optimizers for image classification using convolutional neural networks on histopathology images," *Journal of Imaging*, vol. 6, no. 9, p. 92, 2020.

[15] M. Yaqub, J. Feng, M. S. Zia, K. Arshid, K. Jia, Z. U. Rehman, and A. Mehmood, "State-of-the-art cnn optimizer for brain tumor segmentation in magnetic resonance images," *Brain Sciences*, vol. 10, no. 7, p. 427, 2020.

[16] Y. S. Chowdhury, R. Dasgupta, and S. Nanda, "Analysis of various optimizer on cnn model in the application of pneumonia detection," in *2021 3rd International Conference on Signal Processing and Communication (ICPSC)*. IEEE, 2021, pp. 417–421.

[17] S. Bera and V. K. Shrivastava, "Analysis of various optimizers on deep convolutional neural network model in the application of hyperspectral remote sensing image classification," *International Journal of Remote Sensing*, vol. 41, no. 7, pp. 2664–2683, 2020.

[18] I. P. Gustiah, N. Lutfiani, M. R. Kusuma, E. P. Lestari, and T. Green, "Grant submission monitoring system based on orange technology using laravel 12 and vue.js," *Journal of Orange Technology*, vol. 1, no. 1, pp. 75–90, 2025.

[19] G. Perin and S. Picek, "On the influence of optimizers in deep learning-based side-channel analysis," in *International Conference on Selected Areas in Cryptography*. Springer, 2020, pp. 615–636.

[20] R. Poojary and A. Pai, "Comparative study of model optimization techniques in fine-tuned cnn models," in *2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA)*. IEEE, 2019, pp. 1–4.

[21] S. Vani and T. M. Rao, "An experimental approach towards the performance assessment of various optimizers on convolutional neural network," in *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*. IEEE, 2019, pp. 331–336.

[22] M. Yaqub, J. Feng, M. S. Zia, K. Arshid, K. Jia, Z. U. Rehman, and A. Mehmood, "State-of-the-art cnn optimizer for brain tumor segmentation in magnetic resonance images," *Brain Sciences*, vol. 10, no. 7, p. 427, 2020.

[23] S. R. Dubey, S. Chakraborty, S. K. Roy, S. Mukherjee, S. K. Singh, and B. B. Chaudhuri, "diffgrad: An optimization method for convolutional neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 11, pp. 4500–4511, 2019.

[24] A. M. Taqi, A. Awad, F. Al-Azzo, and M. Milanova, "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)*. IEEE, 2018, pp. 140–145.

[25] K. Thavasimani and N. K. Srinath, "Deep learning techniques: A case study on comparative analysis of various optimizers to detect bots from cresci-2017 dataset," *International Journal of Advanced Science and Technology*, vol. 29, no. 4, pp. 10 040–10 053, 2020.

[26] P. Elangovan and M. K. Nath, "Performance analysis of optimizers for glaucoma diagnosis from fundus images using transfer learning," in *Machine Learning, Deep Learning and Computational Intelligence for Wireless Communication*. Springer, 2021, pp. 507–518.