The Scientific Temper (2024) Vol. 15 (2): 2231-2237
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2024.15.2.36
https://scientifictemper.com/

RESEARCH ARTICLE

Tuning VGG19 hyperparameters for improved pneumonia classification

K. Kalaiselvi*, **M. Kasthuri**

Abstract

This research focuses on the classification of chest X-ray (CXR) images using powerful VGG19 convolutional neural network (CNN) architecture. The classification task involves distinguishing between various chest conditions present in X-ray images, with the aim of assisting medical professionals in achieving accurate and efficient diagnoses. This research work explores the use of the VGG19 model for classifying CXR images using three optimization algorithms: Stochastic gradient descent with momentum (SGDM), root mean square propagation (RMSprop), and adaptive moment estimation (Adam). This study investigates the impact of various factors on hyperparameter adjustments, including a learning rate (LR), mini-batch size (MBS) and training epochs. Additionally, two dropout layers are introduced for weight decay with an L2 factor, and data augmentation techniques are applied with various activation functions. This study not only helps optimize for medical image analysis but also offers valuable insights into the comparative efficacy of popular optimization algorithms in deep learning (DL) applications.

Keywords: Learning rate, Mini batch size, Epoch, Dropout layer, Data augmentation.

Introduction

Pneumonia classification is crucial for medical intervention, and CNNs are effective for categorizing medical images. VGG19, a 19-layer CNN architecture, can be optimized using algorithms like SGDM, RMSprop, and Adam to improve performance. This study explores the optimization of VGG19 hyperparameters for pneumonia classification, highlighting the importance of tuning parameters like learning rate, batch size, and weight decay for improved accuracy. The researcher uses a CNN algorithm to identify contamination defects and fusion defects in the TIG welding dataset. The accuracy of the CNN approach was 96.1%, demonstrating its efficiency. The real-time dataset was divided into 80% training and 20% testing, and the data was augmented. The study suggests combining the CNN model with a camera for real-time welding input, which could save manpower and costs in automated welding sectors (Kanthalakshmi *et al*., 2023).

The author uses the microcanonical optimization (µO) algorithm to optimize CNN components based on architecture and hyper-parameters. It outperforms the SA meta-heuristic in classification accuracy and network size, especially for smaller networks (Gulcu & Kucs, 2020). The author proposed a genetic algorithm for image classification, outperforming 22 existing methods on nine tasks, demonstrating a superior classification error rate and parameter weights (Sun *et al*., 2019). The author presents a strategy for improving CNN performance by modifying hyperparameters in feature extraction using a parameter-free harmony search algorithm (Lee *et al*., 2018). The you only look once version (YOLOv7) framework is used in a rotation box object detection model, enhancing localization accuracy and precision while reducing parameters and computational resources (Liu *et al*., 2024). The author explores the use of deep learning techniques, specifically convolutional neural networks, for detecting potato leaf disease. The leaky rectifier function was found to be the most effective activation function, with AdaGrad's optimizer showing superior accuracy. The model performed well in the agricultural sector (Asfaw & Temesgen, 2023).

The researcher explores the creation and fine-tuning of CNNs for EEG signal extraction, analyzing signal processing techniques, and evaluating well-known CNN

Bishop Heber College, Tiruchirappalli, Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India.

***Corresponding Author:** K. Kalaiselvi, Bishop Heber College, Tiruchirappalli, Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India., E-Mail: kalaiselvigl1984@gmail.com

How to cite this article: Kalaiselvi, K., Kasthuri, M. (2024). Tuning VGG19 hyperparameters for improved pneumonia classification. The Scientific Temper, **15**(2):2231-2237.

Doi: 10.58414/SCIENTIFICTEMPER.2024.15.2.36

Source of support: Nil

Conflict of interest: None.

Received: 26/04/2024 **Accepted:** 16/05/2024 **Published** : 15/06/2024

architectures into standard implementation, recurrent convolutional, decoder architecture, and combined architectures (Rakhmatulin *et al*., 2024). The authors investigate evolutionary computation strategies for optimizing CNN designs, verifying their effectiveness against benchmark datasets (Ma *et al*., 2020). Generative adversarial networks (GANs) can address privacy concerns and class imbalances in medical applications by creating artificial images of underrepresented classes for accurate chest MRIs (Al-qaness *et al*., 2024). Researchers use deep learning techniques to enhance early detection of *pneumonia* and lung cancer, training CXR datasets on ANN, CNN, and VGG19 models, achieving high accuracy and reduced memory usage. A model called chest X-ray COVID network (CXRVN) is proposed for analyzing and evaluating grayscale CXR images, achieving high accuracy in a few milliseconds. The lightweight architecture reduces memory usage and processing time, with an average accuracy of 94.5% (Elzeki *et al*., 2021). The researcher proposes a convolutional neural network approach called COVID-CCD-Net for early detection of COVID-19, normal, and viral pneumonia in chest X-ray images, using tissue microarray for colorectal cancer analysis and demonstrating high classification performance (Kiziloluk & Sert, 2022). The Chest X-Ray COVID Network (CXRVN) is a robust DCNN architecture used for COVID-19 classification. The architecture processes extracted features from convolutional layers, demonstrating its robustness (Sharma & Guleria, 2023). The study explored three optimizers and showed that using GANs enhanced accuracy by 96.7 and 93.07% respectively (Bhosale & Yadav, 2024).

The CXR-DLM model for chest X-ray image classification employs a modified VGG19 architecture for feature extraction and classification, but its effectiveness is limited by data imbalance and image limitations (Kalaiselvi & Kasthuri, 2023). This study simplifies CNN design for binary and subject-dependent emotional valence classification using modular network architecture and a procedural tuning method. It includes twelve hyperparameters and validates their significance through statistical variance analysis (Khan *et al*., 2020).

The research introduces a deep earning-based method for detecting furcation defects on periapical radiographs, achieving 95% accuracy and 94.97% overall, potentially improving periodontal diagnosis, treatment planning, and patient outcomes. The chest X-ray COVID network model is proposed for early detection of COVID-19 (Mezzah & Tari, 2023). The author presents a cricket activity dataset and a deep learning model for cricket activity recognition, outperforming recurrent neural networks with 92% accuracy, recall, and F1 score (Mao *et al*., 2023).

Materials and Methods

The *pneumonia* CXR image dataset consists of diverse images from a wide range of categories, including *pneumonia* and normal. Each image is labeled with one of the predefined classes, and the challenge is to develop models that can accurately classify images into these categories. The images were taken from the Kaggle database; 3875 images *pneumonia* and 1341 images are normal (https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia).

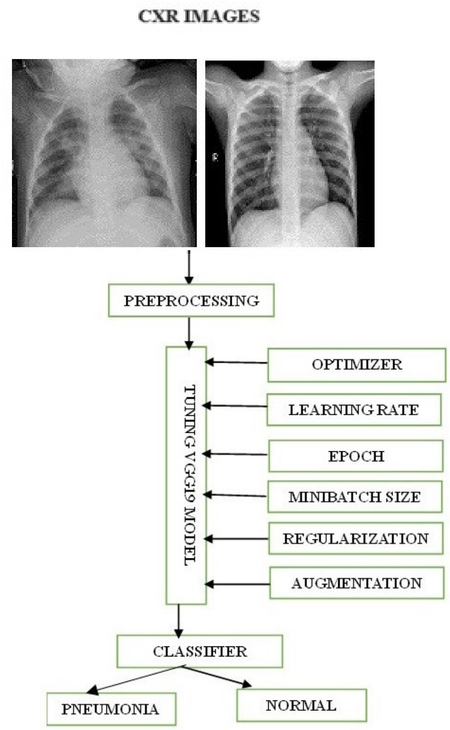

Figure 1: A framework of FOBIC-VGG19 model

The VGG19 model was chosen due to its efficacy in image feature extraction tasks, making it ideal for medical image classification. The suggested study improves the binary image classification work by using the VGG19 model. Figure 1 shows a framework for the FOBIC-VGG19 model. This model optimizes a VGG19 model on a CXR dataset by performing data collection, pre-processing, model fine-tuning, hyperparameter tuning, data augmentation, model evaluation, and deployment.

In the implantation phase of hyperparameter tuning for the FOBIC-VGG19 model, various parameters were meticulously adjusted to optimize performance. The minibatch size was systematically varied to determine its impact on model convergence and generalization. Different optimizers, including SGD and Adam, were tested to evaluate their efficacy in minimizing the loss function. A fixed LR was set initially, followed by fine-tuning to ensure stable training dynamics. The number of epochs was also experimented with to find the optimal point where the model achieves the best validation accuracy without overfitting. Additionally, weight factors were adjusted to balance the contribution of different loss components. Data augmentation techniques were employed, manipulating the rows, columns, and channels of input images to enhance the model's robustness and ability to generalize. This comprehensive tuning process was critical in leveraging the full potential of VGG19 for the given task.

Optimizer

DNNs often utilize SGDM, RMSprop, and Adam optimizers. These strategies are used to update the weights of a neural network during training in order to lower the loss function. SGDM is a basic optimization algorithm that updates the model parameters in the opposite direction of the gradient of the loss function with respect to the parameters. It achieves a certain level of accuracy but struggles to converge or oscillate. RMS is an extension of SGD that adapts the learning rates for each parameter individually. It maintains a moving average of the squared gradients for each parameter and divides the gradient by the square root of this moving average. The adaptive learning rate improves convergence compared to SGDM but shows sensitivity to certain hyperparameters. Adam combines the ideas of momentum and the square of past gradients (RMS). The author uses a graph neural network (GNN) model trained using the Adam optimizer over 100 epochs, evaluating its predictive capabilities and performance metrics, emphasizing the need for continuous refinement and adaptation (Dandale *et al.*, 2024) learning rate impact

In DL, various optimizers are employed to update the model parameters during training, each with its own approach to adjusting the learning rate. Gradient descent, stochastic gradient descent (SGD), typically use fixed or adaptive learning rates ranging from 10^{-3} to 10^{-1}, but values outside this range can be explored based on the problem and architecture. Optimizers like Adam and RMSprop adaptively adjust the learning rate for each parameter, commonly starting with values around 0.001 or 0.0001. These learning rate parameters are not set and may require adjustment based on factors like the features of the dataset and the complex structure of the neural network being trained.

A fixed LR of 0.0001 was optimal for all optimization algorithms, preventing overshooting and ensuring stable convergence. The choice of mini-batch size (MBS) is a hyperparameter that can affect the training dynamics of a neural network, including VGG19. The MBS represents the number of training examples utilized in one iteration of the gradient descent optimization algorithm. In practice, common MBS for training VGG19 often falls in the range of 16 to 128. The number of epochs decides when to stop training to avoid overfitting or sign to stop training or adjust hyperparameters. The model may include running experiments with different epoch numbers and analyzing performance on a validation set to discover the optimal number of epochs.

Dropout is a regularization technique in VGG19 for image classification tasks, temporarily ignoring randomly selected neurons during training to improve generalization performance and prevent overfitting, especially in limited datasets or complex models. Weight decay in VGG19 training prevents overfitting by adding an L2 -regularization term to the loss function, penalizing large weights. This regularization encourages VGG19 to learn simpler decision boundaries, improving generalization performance on unseen data. The parameter is chosen through experimentation or cross-validation. An epoch is a single pass through the entire training dataset, where the model makes predictions, computes loss, and updates parameters using an optimizer. The number of epochs is a hyperparameter for VGG19, determining the number of times the entire dataset is fed through the network. Overtraining can lead to underfitting or overfitting, so finding the optimal number of epochs is a challenging one. In VGG19, these parameters are tuned depending on factors such as the available computational resources, the size and complexity of the dataset, and the specific optimization algorithm being used. Experimentation and tuning are often necessary to find the optimal minibatch size for a given task.

Results

The FOBIC-VGG19 model was trained on CXR image dataset. Experimenting with different optimizers, LR, MBS and Epoch, is often part of the process of training VGG19. Accuracy, sensitivity, specificity and F1-score were used to evaluate the performance of a classifier on a set of test data. Pre-trained weight factors from a merged dataset are used, data pre-processing enhances classification accuracy (Nguyen & Lee, 2018). It is essential to experiment proposed model with different optimizers for LR = 0.0001, 10 epochs and 128 MBS. Performance of model was measured by training time (TT) in minutes (M) and seconds (S), accuracy (V1), sensitivity (V2), specificity (V3), precision (V4) and F1-score (V5).

Results of the performance of the FOBIC-VGG19 model pneumonia classification are presented in Table 1. It is observed that Adam's optimizer is better than another optimizer.

We apply the Adam optimizer with learning rates 0.01,0.001 and 0. 0001 on the FOBIC-VGG19 model. Table 2 summarizes the results of the performance of the FOPIC-VGG19 model for varying learning rates.

The appropriate number of epochs can vary depending on the complexity of the task, the size of the dataset, and other factors. Table 3 shows the results performance of the FOPIC-VGG19 model for 10,20 and 30 epochs with 0.0001 LR and Adam optimizer.

The experiments are conducted with different MBS and monitor the results of performance for 0.0001 LR. 10 epoch with for Adam optimizer. Results are shown in Table 4.

The regularization strength is controlled by the weight

Table 1: Performance of FOPIC-VGG19 model for various optimizer

Optimizer	*Epoch*	*MBS*	*LR*	*TT*	*V1*	*V2*	*V3*	*V4*	*V5*
SGDM	10	128	0.0001	55M&37S	0.971	0.998	0.925	0.944	0.952
RMSprop	10	128	0.0001	33M&59S	0.997	0.998	0.995	0.998	0.998
Adam	10	128	0.0001	16M&22S	0.998	0.998	1.000	1.000	0.999

Table 2: Performance of FOPIC-VGG19 model for various learning rates

Optimizer	*Epoch*	*MBS*	*LR*	*TT*	*V1*	*V2*	*V3*	*V4*	*V5*
Adam	10	128	0.01	144M&46S	0.953	1.000	0.000	0.963	0.967
Adam	10	128	0.001	72M&59S	0.995	0.999	0.987	0.993	0.996
Adam	10	128	0.0001	24M&18S	0.998	0.998	1.000	1.000	0.999

Table 3: Performance of FOPIC-VGG19 model for different epoch with 0.0001 LR

Optimizer	*Epoch*	*MBS*	*LR*	*TT*	*V1*	*V2*	*V3*	*V4*	*V5*
Adam	10	128	0.0001	16M&22S	0.998	0.998	1.000	1.000	0.999
Adam	20	128	0.0001	84M&57S	0.998	0.997	1.000	1.000	0.999
Adam	30	128	0.0001	87M&59S	0.998	0.998	1.000	1.000	0.999

Table 4: Performance of FOPIC-VGG19 model for different MBS

Optimizer	*Epoch*	*MBS*	*LR*	*TT*	*V1*	*V2*	*V3*	*V4*	*V5*
Adam	10	32	0.0001	19M&49S	0.997	0.995	1.000	1.000	0.997
Adam	10	64	0.0001	19M&67S	0.997	0.999	0.992	0.996	0.997
Adam	10	128	0.0001	17M&59S	0.998	0.998	1.000	1.000	0.999

Table 5: Performance of FOPIC-VGG19 model for regularization parameters

Optimizer	*WL2F*	*Epoch*	*MBS*	*LR*	*TT*	*V1*	*V2*	*V3*	*V4*	*V5*
Adam	0	10	128	0.0001	16M&11S	0.998	0.998	1.000	1.000	0.999
Adam	0.25	10	128	0.0001	37M&48S	0.998	0.999	0.998	0.999	0.999
Adam	0.5	10	128	0.0001	40M&53S	0.990	0.984	1.000	1.000	0.992

Table 6: Performance of FOPIC-VGG19 model for dropout layer

Optimizer	*Dl*	*Epoch*	*MBS*	*LR*	*TT*	*V1*	*V2*	*V3*	*V4*	*V5*
Adam	2Dl include	10	128	0.0001	18M&40S	0.998	0.998	1.000	1.000	0.999
Adam	1Dl removed	10	128	0.0001	36M&31S	0.995	0.999	0.989	0.994	0.996
Adam	2Dl removed	10	128	0.0001	31M&60S	0.998	0.998	0.999	1.000	0.999

decay parameter, which is set when defining the optimizer with weight L2 factor (WL2F) such as (0, 0.25, 0.5) to avoid overfitting. Table 5 shows the results of the performance of the FOBIC-VGG19 model for various regularization parameters.

For a fixed LR, MBS and epoch, monitor the training performance to determine the most suitable setting for our model. We made specific changes, including LR of 0.0001, MBS of 128, training for 10 epochs, and incorporating a dropout layer. Results are presented in Table 6. The results of introducing pre-processing using data augmentation techniques and activation functions (AF) are exhibited in Table 7. Figure 2 shows a sample diagram of the training progress of the FOPIC-VGG19 model.

Discussion

CNN design involves optimizing parameters like bias, learning rate, and filters, known as hyperparameters. Accurate hyperparameter tuning is experimental, as incorrect settings can affect model accuracy (Wang & Gang,

Table 7: Performance of FOPIC-VGG19 model for augmentation with different activation

Optimizer	*AF*	*Epoch*	*MBS*	*LR*	*TT*	*V1*	*V2*	*V3*	*V4*	*V5*
Adam	Row	10	128	0.0001	35M&59S	0.999	1.000	0.998	0.999	1.000
Adam	Column	10	128	0.0001	29M&49S	0.997	0.998	0.995	0.998	0.998
Adam	Channel	10	128	0.0001	36M&50S	0.997	0.995	1.000	1.000	0.997

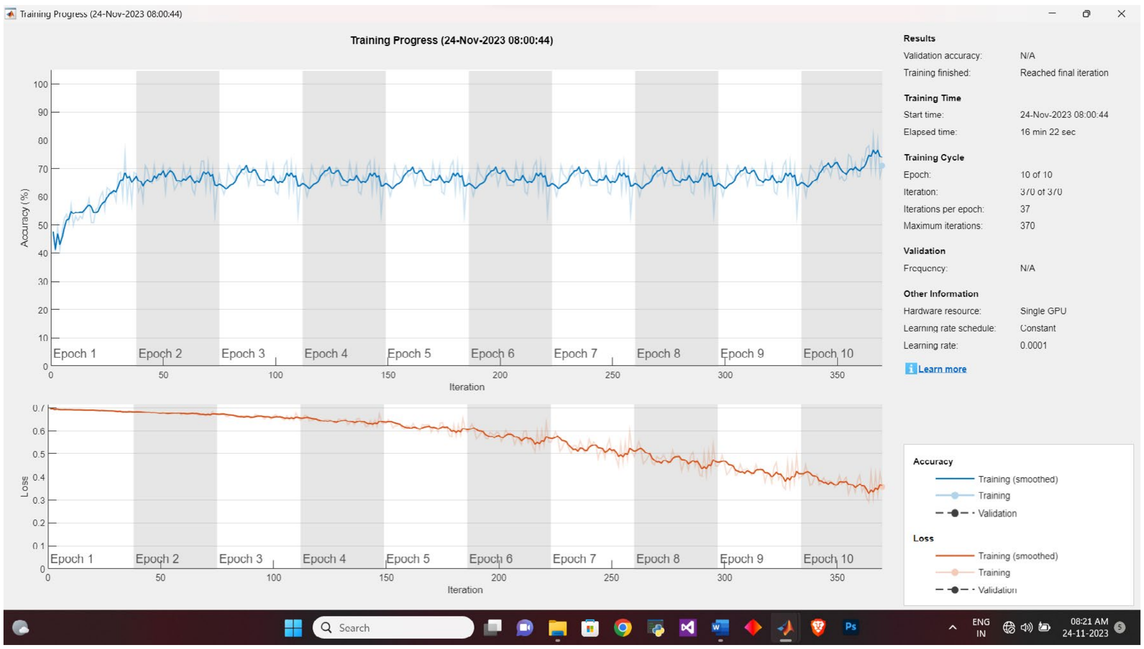

Figure 2: Training progress of FOPIC-VGG19 model

2018). The researcher compares various classifier techniques for leaf disease, including NB, KNN, DT, SVM, RF, and MLP, focusing on accuracy and execution time. SVM is found to be more accurate and faster. The study highlights the recent advancements in deep learning technologies for plant disease diagnosis and highlights research gaps for efficient strategies (Nagila & Mishra, 2023). This research explores the use of gradient descent optimization algorithms in real-life issues and software product creation, comparing different variants and presenting widely used parameters like momentum (Wang *et al*., 2023). The study introduces an adaptive learning rate algorithm for CNNs, enhancing classification accuracy and time-efficient training, making it comparable to gradient descent algorithms with fixed learning rates (Ahmad *et al*., 2023). The author compares the performance of CNN in image classification using different batch sizes and learning rates. Results show a high correlation between learning rate and batch size, with large batch sizes performing better. The optimal batch size is 32 or 64, with a power of 2 to maximize GPU processing. Increasing batch size can be done until satisfactory results are achieved (Jepkoech *et al*., 2021). The author offers an adaptive learning rate algorithm for CNNs, enhancing classification accuracy and time-efficient training, making it comparable to gradient descent algorithms with fixed learning rates (Kandel & Castelli, 2020). Hyperparameter-tuning is done step-wise, dividing hyperparameters into architecture and learning categories and using an L27 orthogonal array for both types of tuning. The author explores the impact of batch size on the effectiveness of a deep learning model for enset leaf disease detection in Ethiopia. Results show that even smaller batch sizes improve the model's efficiency, providing heuristic knowledge for boosting enset production in agriculture (Asfaw & Temesgen, 2024).

In the hyperparameter tuning for the VGG19 model, various parameters were meticulously adjusted to optimize performance, with their values varying depending on the dataset used. In our study, the Adam optimizer was employed due to its efficiency and adaptive learning rate properties. A learning rate of 0.0001 was found to be optimal, providing a stable training process. The mini-batch size was set to 128, balancing between training speed and model performance. No weight L2 regularization factor was used, as it contributes significantly to improving the model in this context. Dropout was included to prevent overfitting, enhancing the model's generalization capabilities. Additionally, row augmentation was applied to the images to further improve the model's robustness. Training the model for 10 epochs yielded the best results, striking a balance between sufficient learning and avoiding overfitting. These hyperparameter values have been fixed and can be utilized for further studies in similar contexts, providing a robust framework for pneumonia image classification.

Conclusion

This work contributes to the understanding of optimization algorithms in the context of medical image classification using the VGG19 model. By comparing SGDM, RMSprop, and Adam, and exploring hyperparameter configurations, we pursue to identify the most effective strategy for improving classification accuracy and robustness in CXR image analysis. The combination of Adam optimization, specific hyperparameter settings, regularization techniques, and data augmentation strategies contributes to a robust and accurate model for medical image analysis. These findings can guide future research in the development of efficient DL models for healthcare applications, emphasising the importance of optimization algorithms and hyperparameter tuning in medical image classification tasks.

Acknowledgment

The authors express gratitude to all the contributors who have advanced the development of MATLAB and deep learning technologies, whose innovative efforts and unwavering dedication have significantly enriched my study. The authors extend their sincere gratitude to the reviewers and editors for their valuable suggestions, which have significantly improved the manuscript.

References

Ahmad, W., Munsif, M., Ullah, H., Ullah, M., Alsuwailem, A. A., Saudagar, A. K. J., Muhammad, K., & Sajjad, M. (2023). Optimized deep learning-based cricket activity focused network and medium scale benchmark. *Alexandria Engineering Journal*, *73*, 771–779.

Al-qaness, M. A., Zhu, J., AL-Alimi, D., Dahou, A., Alsamhi, S. H., Abd Elaziz, M., & Ewees, A. A. (2024). Chest x-ray images for lung disease detection using deep learning techniques: a comprehensive survey. *Archives of Computational Methods in Engineering*, 1–35.

Asfaw, & Temesgen, A. (2023). Deep learning hyperparameter's impact on potato disease detection. *The Scientific Temper*, *14*(03), 582–590.

Asfaw, & Temesgen, A. (2024). Batch size impact on enset leaf disease detection. *The Scientific Temper*, *15*(01), 1613–1618.

Bhosale, R. D., & Yadav, D. (2024). Customized convolutional neural network for pulmonary multi-disease classification using chest x-ray images. *Multimedia Tools and Applications*, *83*(6), 18537–18571.

Dandale, M. N., Yadav, A. P., Reddy, P., Kadu, S. G., Madhusudana, T., & Manavadaria, M. S. (2024). Deep learning enhanced drug discovery for novel biomaterials in regenerative medicine utilizing graph neural network approach for predicting cellular responses. *The Scientific Temper*, *15*(01), 1588–1594.

Elzeki, O. M., Shams, M., Sarhan, S., Abd Elfattah, M., & Hassanien, A. E. (2021). COVID-19: A new deep learning computer-aided model for classification. *PeerJ Computer Science*, *7*, e358.

Gulcu, A., & Kucs, Z. (2020). Hyper-parameter selection in convolutional neural networks using microcanonical optimization algorithm. *IEEE Access*, *8*, 52528–52540.

Jepkoech, J., Mugo, D. M., Kenduiywo, B. K., & Too, E. C. (2021). The effect of adaptive learning rate on the accuracy of neural networks. *International Journal of Advanced Computer Science and Applications*, *12*(8).

Kalaiselvi, K., & Kasthuri, M. (2023). Refining the accuracy of chest x-ray image classification through layer and activation function optimization. *Indian Journal of Science and Technology*, *16*(13), 1030–1037.

Kandel, I., & Castelli, M. (2020). The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset. *ICT Express*, *6*(4), 312–315.

Kanthalakshmi, S., Pradeepa, G. (2023). Classification of weld defects using machine vision using convolutional neural network. *The Scientific Temper*, *14*(01), 172–177.

Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, *53*, 5455–5516.

Kiziloluk, S., & Sert, E. (2022). COVID-CCD-Net: COVID-19 and colon cancer diagnosis system with optimized CNN hyperparameters using gradient-based optimizer. *Medical & Biological Engineering & Computing*, *60*(6), 1595–1612.

Lee, W.-Y., Park, S.-M., & Sim, K.-B. (2018). Optimal hyperparameter tuning of convolutional neural networks based on the parameter-setting-free harmony search algorithm. *Optik*, *172*, 359–367.

Liu, Y., Ma, Y., Chen, F., Shang, E., Yao, W., Zhang, S., & Yang, J. (2024). YOLOv7oSAR: A lightweight high-precision ship detection model for SAR images based on the YOLOv7 algorithm. *Remote Sensing*, *16*(5), 913.

Ma, B., Li, X., Xia, Y., & Zhang, Y. (2020). Autonomous deep learning: A genetic DCNN designer for image classification. *Neurocomputing*, *379*, 152–161.

Mao, Y.-C., Huang, Y.-C., Chen, T.-Y., Li, K.-C., Lin, Y.-J., Liu, Y.-L., Yan, H.-R., Yang, Y.-J., Chen, C.-A., Chen, S.-L. (2023). Deep learning for dental diagnosis: a novel approach to furcation involvement detection on periapical radiographs. *Bioengineering*, *10*(7), 802.

Mezzah, S., & Tari, A. (2023). Practical hyperparameters tuning of convolutional neural networks for EEG emotional features classification. *Intelligent Systems with Applications*, *18*, 200212.

Nagila, A., & Mishra, A. K. (2023). The effectiveness of machine learning and image processing in detecting plant leaf disease. *The Scientific Temper*, *14*(01), 8-13.

Nguyen, H. N., & Lee, C. (2018). Effects of hyper-parameters and dataset on CNN Training. *Journal of IKEEE*, *22*(1), 14-20.

Rajeev, C., & Natarajan, K. (2023). Data augmentation in classifying chest radiograph images (CXR) using DCGAN-CNN. In *GANs for Data Augmentation in Healthcare,* Springer, 91-110.

Rakhmatulin, I., Dao, M.S., Nassibi, A., & Mandic, D. (2024). Exploring convolutional neural network architectures for EEG feature extraction. *Sensors*, *24*(3), 877.

Sharma, S., & Guleria, K. (2023). A deep learning based model for the detection of pneumonia from chest X-ray images using VGG-16 and neural networks. *Procedia Computer Science*, *218*, 357–366.

Sun, Y., Xue, B., Zhang, M., & Yen, G. G. (2019). Evolving deep convolutional neural networks for image classification. *IEEE Transactions on Evolutionary Computation*, *24*(2), 394–407.

Wang, W., & Gang, J. (2018). Application of convolutional neural network in natural language processing. *2018 International Conference on Information Systems and Computer Aided Education (ICISCAE)*, 64–70.

Wang, W., Pei, Y., Wang, S.-H., manuel Gorrz, J., & Zhang, Y.-D. (2023). PSTCNN: Explainable COVID-19 diagnosis using PSO-guided self-tuning CNN. *Biocell: Official Journal of the Sociedades Latinoamericanas de Microscopia Electronica... et. al*, *47*(2), 373.