Unlocking Myopia Detection: Leveraging Deep Learning for Accurate Diagnosis

Arnab Bishakh Sarker[1]$(id: 21-44464-1)^a$ $Zobayer Alam^2(id: 21-44487-1)^a$
$^a Department of Computer Sciences, American International University - Bangladesh$

***Abstract*—This study looks into the usage of deep learning models for myopia detection using retinal pictures from the Retinal Fundus Multi-disease Image Dataset (RFMiD). We assessed the performance of ResNet50, ResNet152, MobileNetV2, DenseNet121, and EfficientNetB7, comparing their accuracies in frozen and unfrozen states. In the frozen state, MobileNetV2 scored the highest accuracy at 96.41%, followed by ResNet152 at 93.75% and ResNet50 at 93.59%, while DenseNet121 achieved 91.25% and EfficientNetB7 lagged severely at 5.00%. After unfreezing and fine-tuning all layers, the models' accuracies increased, with ResNet152 achieving 95.15%, ResNet50 at 94.84%, MobileNetV2 at 94.06%, and DenseNet121 at 92.34%; EfficientNetB7's performance stayed at 5.00%. The data show that fine-tuning improves diagnostic accuracy, with ResNet152 emerging as the top performer. This study demonstrates the possibility of deep learning in automated myopia detection and emphasizes the need for model selection and fine-tuning for the best diagnostic performance.**

Keywords: Myopia, Retinal Diseases, Deep Learning, Transfer Learning, Medical Imaging.

I. Introduction

Eye problems, commonly referred to as eye diseases, are serious threats to eyesight and general well-being that, if unchecked, could result in blindness. Of the five senses, vision is the most vital since it allows people to see and understand their surroundings. The increasing incidence of eye illnesses worldwide highlights the pressing need for efficient detection and intervention methods. Myopia, a condition marked by blurry vision in the distance, is becoming more commonplace globally, raising concerns about its impact on public health. The World Health Organization (WHO) estimates that between 1-4 percent of the world's population suffers from myopia, one of the main causes of blindness. Alarmingly, persons with extreme myopia suffer heightened risks of vision impairment due to linked pathological disorders, including retinal degeneration, cataracts, and glaucoma.

Considering these concerning figures, it is essential to identify and treat myopic eye illness as soon as possible to reduce the likelihood of vision loss and related problems. Earlier studies have looked at a variety of strategies for treating eye disorders, from segmentation techniques to digital imaging for the diagnosis of cataracts and diabetic retinopathy. A key component of image processing, texture analysis has shown promise in identifying minute differences in visual properties, which can help diagnose and categorize eye conditions.

More sophisticated approaches powered by deep learning architectures have been made possible by the advancement of traditional picture classification methods, which relied on manually created features. Deep learning models—like ResNet50, ResNet152, MobileNetV2, EfficientNetB7, and DenseNet121—offer several benefits over traditional methods, such as a decreased need for human feature extraction and increased processing speed for big datasets. By simulating the human visual cortex through complex neural networks, these models allow for the robust interpretation and classification of medical pictures with previously unheard-of speed and precision.

Considering this, the goal of this study is to use the Retinal Fundus Multi-Disease Image Dataset (RFMID) to categorize myopic eye illnesses by utilizing deep learning approaches. We seek to differentiate between normal and abnormal myopic eye diseases by using cutting-edge architectural models such as ResNet50, ResNet152, MobileNetV2, EfficientNetB7, and DenseNet121. This will allow for early diagnosis and focused intervention. RFMID, an extensive collection of both enhanced and non-augmented myopia images, is used to highlight the variety and intricacy of real-world clinical circumstances.

This research's main goal is to use deep learning techniques to classify myopic eye disorders. With the use of the ResNet50, ResNet152, MobileNetV2, EfficientNetB7, and DenseNet121 architectures, we aim to expedite the classification process so that accurate and rapid diagnoses can be made in the field of orthodontia. This research could completely change how eye illnesses are classified, giving doctors a valuable tool to improve patient outcomes and diagnostic precision.

II. Literature Review

Many studies in the field of medical image analysis have focused on the identification and categorization of myopia using a variety of advanced approaches, most notably deep learning techniques. Bismi and Na'am [1] pioneered the application of deep learning in classifying myopia levels using fundus images, showcasing the potential of AI-driven solutions in ophthalmic diagnostics. Meanwhile, investigations into the genetic underpinnings of myopia by Széll et al. [2] have enriched our understanding of its hereditary factors. Texture analysis, as elucidated by Armi and Fekri-Ershad [3], has emerged as a pivotal tool for discerning subtle image features crucial for accurate diagnosis. Surveys conducted by Litjens et al. [4] and Zhang et al. [5] have provided invaluable insights into the expansive landscape of deep learning applications within medical imaging, underscoring its transformative potential. Moreover, studies by Jia et al. [6] and Ni'mah et al. [7] have ventured into diverse domains such as hyperspectral image classification

and herbal plant identification, illustrating the versatility of deep learning paradigms across healthcare applications. Raj et al. [8] proposed innovative feature selection strategies to optimize medical image classification models, while Calik et al. [9] and Hurtik et al. [10] contributed novel approaches for data preprocessing and neural network architecture refinement, respectively. The World Health Organization's reports [11] have underscored the global burden of myopia, reinforcing the urgency for advanced diagnostic methodologies. Recent breakthroughs, exemplified by Du et al.'s [12] automated detection of myopic maculopathy, and the availability of comprehensive datasets like RFMiD [13], have propelled the field towards more precise and efficient diagnostic tools. Collectively, these endeavors signify a concerted interdisciplinary effort aimed at enhancing our ability to diagnose and manage myopia effectively, thereby mitigating its societal impact and improving patient outcomes. The publications from the World Health Organization [11] have shown the prevalence of myopia worldwide, highlighting the need for more sophisticated diagnosis techniques. Advances in the field, such as the automated detection of myopic maculopathy by Du et al. [12] and the availability of large datasets like RFMiD [13], have led to the development of more accurate and effective diagnostic techniques. Altogether, these initiatives represent a coordinated multidisciplinary effort to improve our capacity to identify and treat myopia efficiently, reducing its negative effects on society and promoting better patient outcomes.

III. Research Method

The RFMiD dataset is split into training, testing, and validation subsets (60-20-20 split) as part of the methodology. Training data is supplemented to address class imbalance, and multiple deep learning models (ResNet50, ResNet152, MobileNetV2, EfficientNet B7, DenseNet121) are used to classify myopic and non-myopic images.

A. DATASET

We used the Retinal Fundus Multi-Disease Image Dataset (RFMiD) *Fig.1*, which contained a total of 3200 images classified into 46 classes. Our focus was on the Myopic class. We divided the dataset into 1920 training images, 640 test images, and 640 validation images, adhering to a 60-20-20 split ratio *Fig.2*. Initially, the dataset was distributed into these three portions: train, test, and validation. Each portion was further categorized into two groups: Myopic and Non-Myopic. This step ensured that both Myopic and Non-Myopic images were represented in each dataset split, allowing for balanced and comprehensive model training, testing, and validation. By maintaining a consistent ratio and categorization, we aimed to enhance the model's ability to generalize well on unseen data, ensuring robust performance across different subsets of the dataset.

B. AUGMENTATION

There were just 101 myopic photos in the training dataset. We used data augmentation approaches to correct this imbalance and improve the training procedure. With the use of

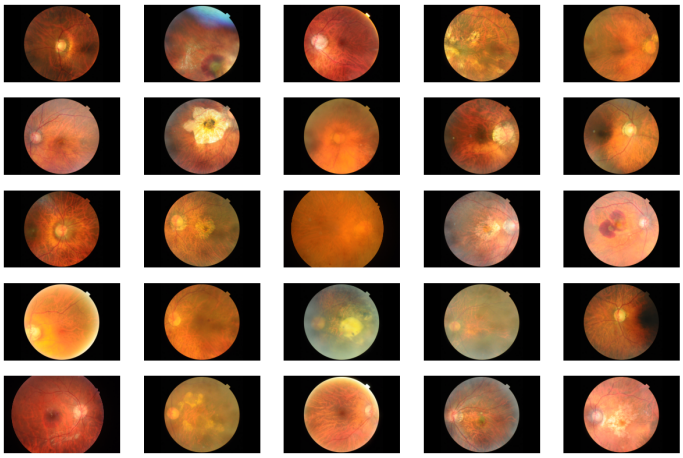

Fig. 1: RFMiD Dataset

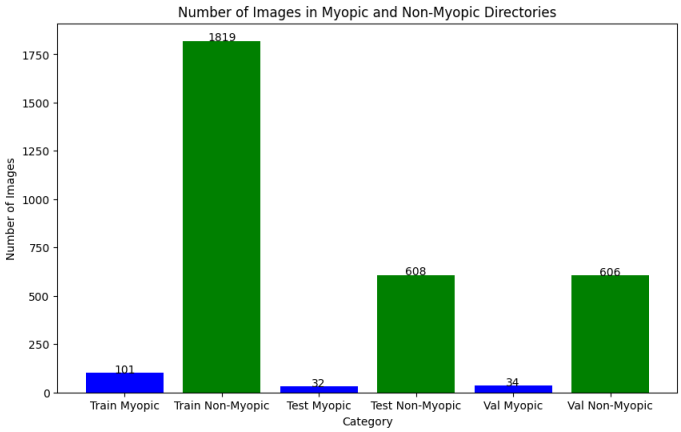

Fig. 2: Sorted RFMiD Dataset Size

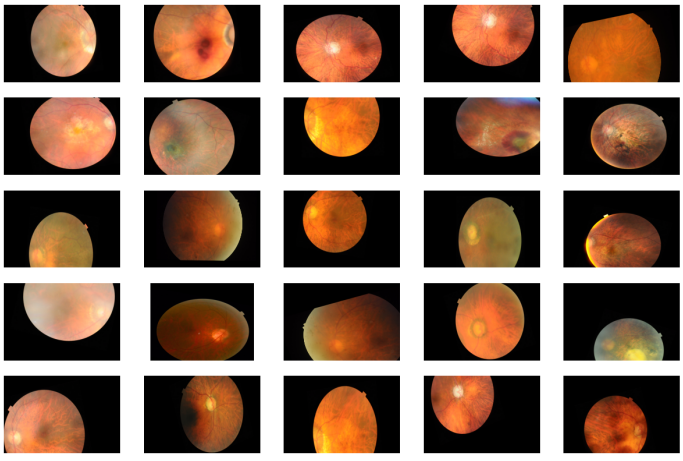

Fig. 3: Augmented RFMiD Dataset

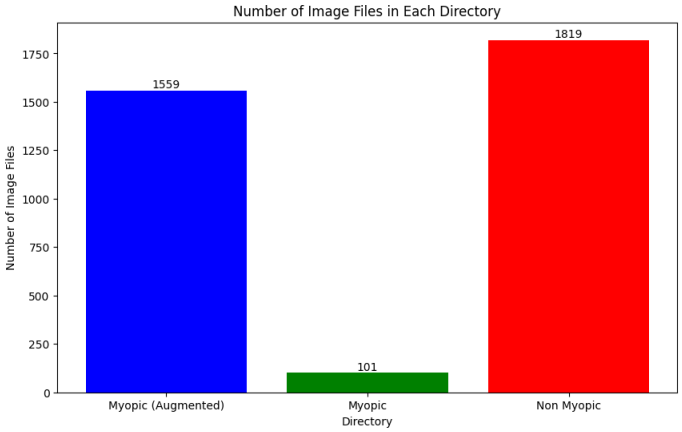

Fig. 4: Augmented Dataset Size Comparison

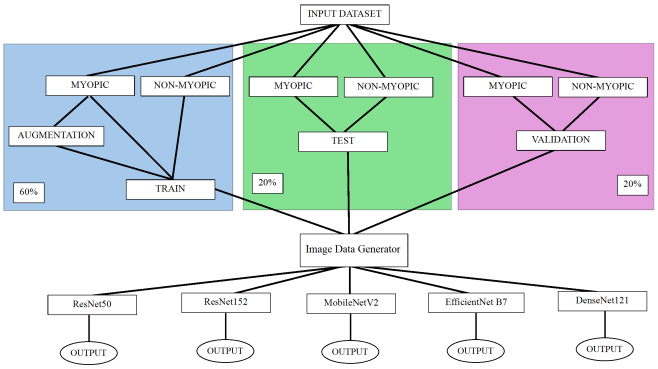

Fig. 5: Methodological Overview

these techniques which included rotation, width shift, height shift, zooming, shearing, and horizontal flip 1706 augmented images were produced. By mimicking variances that the model would experience in real-world situations, augmentation helps to reduce overfitting, boost the variety of the training dataset, and strengthen the model's resilience.

C. Concept of Deep Learning Models

Deep learning, a subset of machine learning, has emerged as a transformative paradigm that involves training artificial neural networks with numerous layers to automatically learn hierarchical data representations. Deep learning models excel in detecting subtle patterns directly from raw input by leveraging large datasets and powerful computational resources, enabling breakthroughs in a variety of domains such as image identification, natural language processing, and speech recognition. This paper highlights deep learning's widespread impact across industries, from healthcare and finance to autonomous vehicles and entertainment, emphasizing its ability to revolutionize problem-solving by extracting nuanced insights from massive amounts of data and driving unprecedented levels of accuracy and performance.

1) ResNet50: ResNet50 Fig.6 is a 50-layer deep convolutional neural network created to address the degradation problem in deep networks using residual learning. It features shortcut connections within residual blocks to improve gradient flow during training. This technique allows for effective training of very deep networks while avoiding vanishing gradients and finding a compromise between depth and computational efficiency. ResNet50 is commonly used in image classification jobs.

2) DenseNet121: DenseNet121 Fig.7 has 121 layers with direct connections from any layer to all subsequent layers, which promotes efficient feature reuse and improves gradient flow. This extensive connection alleviates the vanishing gradient problem and enables the model to learn more robust features with fewer parameters. DenseNet121 is known for its small and efficient design, which makes it suitable for a variety of image classification tasks.

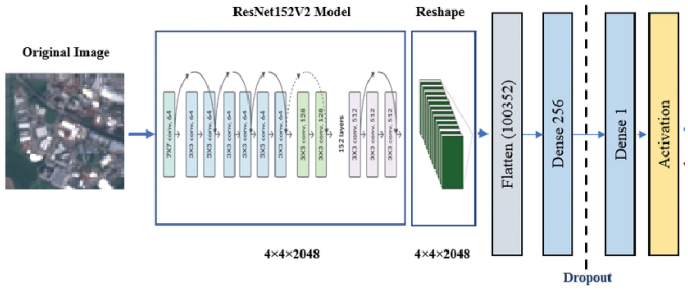

Fig. 6: ResNet50 Model

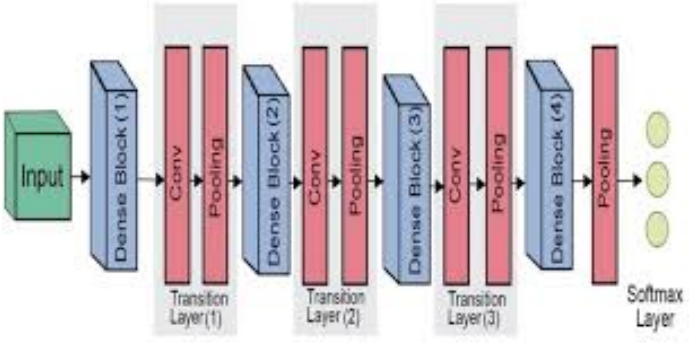

Fig. 7: DenseNet121 Model

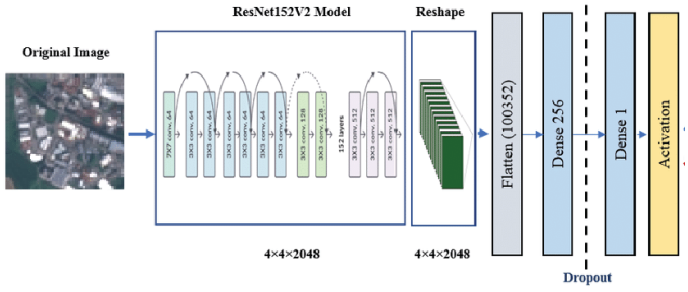

Fig. 8: ResNet152 Model

3) ResNet152: ResNet152 Fig.8 is an expansion of ResNet50, with 152 layers. This more advanced architecture enables the model to learn more complicated and abstract features, resulting in higher accuracy on difficult image categorization tasks. ResNet152 captures complicated patterns by integrating more residual blocks, however, this comes at the cost of increased processing resources and longer training duration.

4) MobileNetV2: MobileNetV2 Fig.9 is designed for mobile and embedded applications that value efficiency and speed. It uses depthwise separable convolutions and inverted residuals with linear bottlenecks to reduce parameters and computing effort. This design ensures that MobileNetV2 delivers outstanding performance while remaining lightweight and suitable for real-time applications.

5) EfficientNet B7: EfficientNet B7 Fig.10 employs a compound scaling approach to uniformly scale network breadth, depth, and resolution. This balanced approach enables EfficientNet B7 to achieve great accuracy while using fewer parameters and processing resources than standard models. Advanced approaches, such as squeeze-and-excitation blocks, improve feature re-calibration.

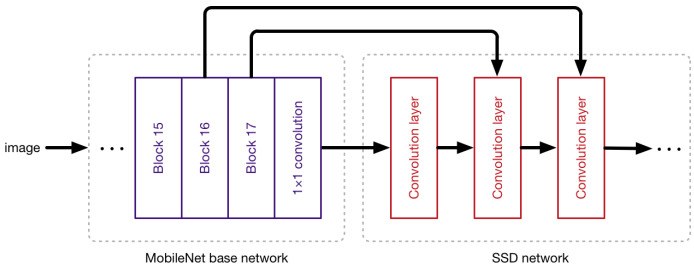

Fig. 9: MobileNetV2 Model

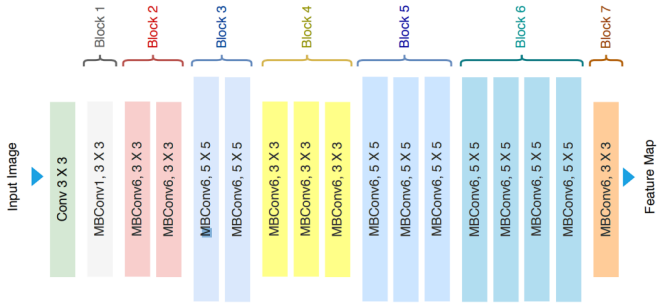

Fig. 10: EfficientNet B7 Model

D. Training Process

A convolutional neural network (CNN) is trained using two stages: feedforward and backpropagation. To categorize picture data, the feedforward approach generates layers with updated weights and biases. These weights are determined using iterative optimization approaches such as gradient descent. Backpropagation is the process of tracking mistakes from the output layer back to the first layer and modifying weights and biases to minimize them. Gradient calculation is important during training because it generates new bias and weight values required for optimization.

E. Testing Process

The testing procedure uses the weights and biases learned during the training process to classify. It examines classification accuracy, identifies misclassified data, and evaluates network performance. The output layer is fully coupled to the labels and generates the final categorization results.

F. Validation Process

Separate image data, such as the Retinal Fundus Multi-Disease Image Dataset (RFMID), are used during the validation process to test the model's performance and assure its dependability in interpreting images. This procedure serves to validate CNN's accuracy and usefulness in detecting certain disorders, such as myopic eye disease in retinal fundus images.

IV. Results & Discussion

A. Deep Learning Models Architecture

Layer (type)	Output Shape	Param #
resnet50 (Functional)	(None, 7, 7, 2048)	23587712
global_average_pooling2d (GlobalAveragePooling2D)	(None, 2048)	0
dense (Dense)	(None, 256)	524544
batch_normalization (Batch Normalization)	(None, 256)	1024
dense_1 (Dense)	(None, 128)	32896
batch_normalization_1 (Bat chNormalization)	(None, 128)	512
dense_2 (Dense)	(None, 32)	4128
dense_3 (Dense)	(None, 1)	33

Total params: 24150849 (92.13 MB)
Trainable params: 562369 (2.15 MB)
Non-trainable params: 23588480 (89.98 MB)

Fig. 11: ResNet50 Architecture Model

During model training, the convolutional base was frozen and unfrozen. Freezing these layers allowed for the preservation of learned features, which was useful for tasks involving transfer learning. When these layers were made trainable, fine-tuning became easier, allowing the model to respond to new input. The option to freeze or unfreeze the convolutional base was determined based on dataset size and task similarity, to find a balance between using pre-trained knowledge and adjusting to new data.

The same structure was used for all models: a sequential configuration with a base model, a GlobalAveragePooling2D layer, and multiple tightly connected layers. Each dense layer was followed by batch normalization. The base model, usually a pre-trained convolutional neural network, acted as the

Layer (type)	Output Shape	Param #
densenet121 (Functional)	(None, 7, 7, 1024)	7037504
global_average_pooling2d (GlobalAveragePooling2D)	(None, 1024)	0
dense (Dense)	(None, 256)	262400
batch_normalization (BatchNormalization)	(None, 256)	1024
dense_1 (Dense)	(None, 128)	32896
batch_normalization_1 (BatchNormalization)	(None, 128)	512
dense_2 (Dense)	(None, 32)	4128
dense_3 (Dense)	(None, 1)	33

Total params: 7338497 (27.99 MB)
Trainable params: 7254081 (27.67 MB)
Non-trainable params: 84416 (329.75 KB)

Fig. 12: DenseNet121 Architecture Model

Model: "sequential"

Layer (type)	Output Shape	Param #
mobilenetv2_1.00_224 (Functional)	(None, 7, 7, 1280)	2257984
global_average_pooling2d (GlobalAveragePooling2D)	(None, 1280)	0
dense (Dense)	(None, 256)	327936
batch_normalization (BatchNormalization)	(None, 256)	1024
dense_1 (Dense)	(None, 128)	32896
batch_normalization_1 (BatchNormalization)	(None, 128)	512
dense_2 (Dense)	(None, 32)	4128
dense_3 (Dense)	(None, 1)	33

Total params: 2,624,513
Trainable params: 365,761
Non-trainable params: 2,258,752

Fig. 13: MobileNetV2 Architecture Model

Layer (type)	Output Shape	Param #
efficientnetb7 (Functional)	(None, 19, 19, 2560)	64097687
global_average_pooling2d (GlobalAveragePooling2D)	(None, 2560)	0
dense (Dense)	(None, 256)	655616
batch_normalization (BatchNormalization)	(None, 256)	1024
dense_1 (Dense)	(None, 128)	32896
batch_normalization_1 (BatchNormalization)	(None, 128)	512
dense_2 (Dense)	(None, 32)	4128
dense_3 (Dense)	(None, 1)	33

Total params: 64791896 (247.16 MB)
Trainable params: 693441 (2.65 MB)
Non-trainable params: 64098455 (244.52 MB)

Fig. 14: EfficientNetB7 Architecture Model

Model: "sequential"

Layer (type)	Output Shape	Param #
resnet152v2 (Functional)	(None, 7, 7, 2048)	58331648
global_average_pooling2d (GlobalAveragePooling2D)	(None, 2048)	0
dense (Dense)	(None, 256)	524544
batch_normalization (BatchNormalization)	(None, 256)	1024
dense_1 (Dense)	(None, 128)	32896
batch_normalization_1 (BatchNormalization)	(None, 128)	512
dense_2 (Dense)	(None, 32)	4128
dense_3 (Dense)	(None, 1)	33

...
Total params: 58,894,785
Trainable params: 562,369
Non-trainable params: 58,332,416

Fig. 15: ResNet152 Architecture Model

foundation, offering basic feature extraction capabilities. Subsequent layers, such as the pooling layer and densely linked layers, improved the model's ability to learn hierarchical representations and detect complicated patterns in the data. Batch normalization was used to stabilize and accelerate training by normalizing each layer's activations. This same configuration was used across all models to guarantee uniformity and allow for a comparative examination of their performance.

1) Distribution of Training, Testing, and Valid Data: Table I shows a breakdown of data distribution across distinct datasets, with samples allocated for training, testing, and validation. The original dataset, which contains unaltered data, includes

TABLE I: Distribution Of Train, Test And Valid Data

No	Dataset	Train	Test	Val	Total
1	Original	1920	640	640	3200
2	Augmented	3478	640	640	4758

TABLE II: Freezed Model Accuracy Table

Models	Freezed Model			
	Correct	Incorrect	Support	Accuracy
ResNet50	599	41	640	0.9359
MobileNetV2	599	41	640	0.9641
ResNet152	600	40	640	0.9375
DenseNet121	584	56	640	0.9125
EfficientNetB7	32	608	640	0.0500

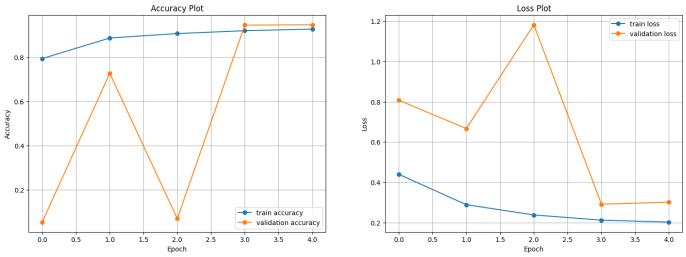

Fig. 16: ResNet50 Accuracy Curve (Freezed)

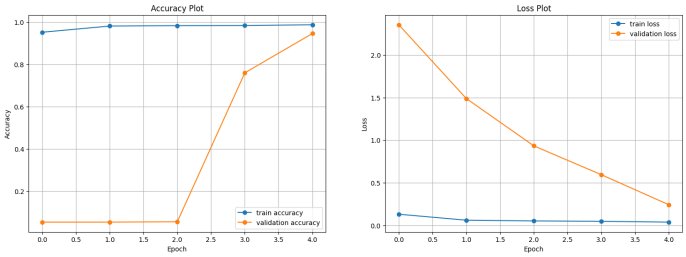

Fig. 17: ResNet50 Accuracy Curve (Unfreezed)

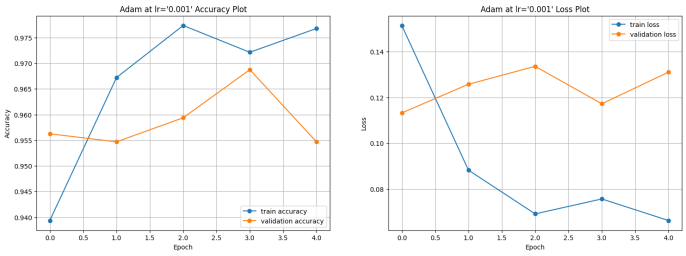

Fig. 18: MobileNetV2 Accuracy Curve (Freezed)

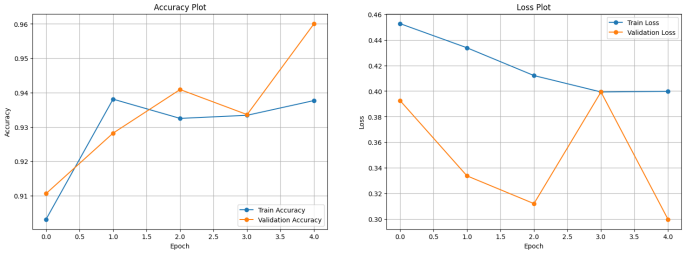

Fig. 19: MobileNetV2 Accuracy Curve (Unfreezed)

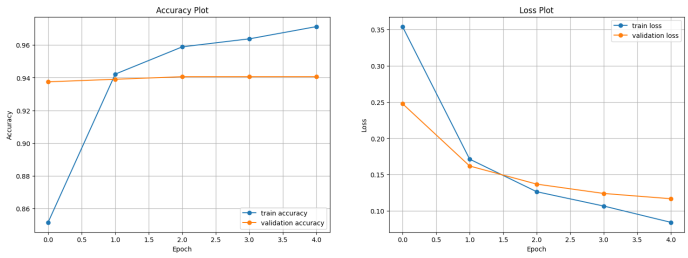

Fig. 20: ResNet152V2 Accuracy Curve (Freezed)

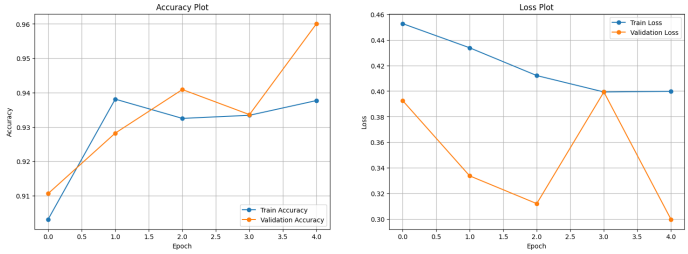

Fig. 21: ResNet152V2 Accuracy Curve (Unfreezed)

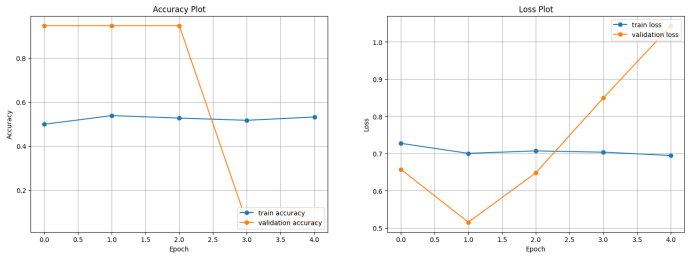

Fig. 22: EfficientNetB7 Accuracy Curve (Freezed)

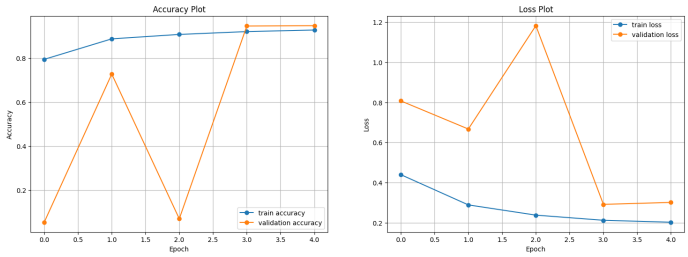

Fig. 23: EfficientNetB7 Accuracy Curve (Unfreezed)

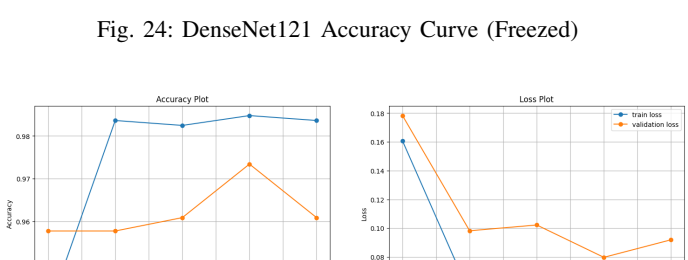

Fig. 24: DenseNet121 Accuracy Curve (Freezed)

Fig. 25: DenseNet121 Accuracy Curve (Unfreezed)

TABLE III: Unfreezed Model Accuracy Table

Models	Unfreezed Model			
	Correct	Incorrect	Support	Accuracy
ResNet50	607	33	640	0.9484
MobileNetV2	602	38	640	0.9406
ResNet152	609	31	640	0.9515
DenseNet121	591	49	640	0.9234
EfficientNetB7	32	608	640	0.0500

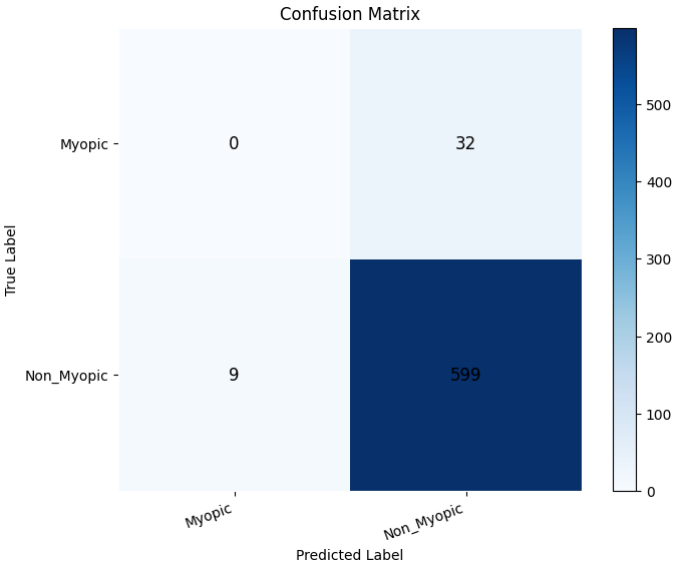

Fig. 26: ResNet50 Confusion Graph (Freezed)

1920 samples for training, 640 for testing, and an additional 640 for validation, for a total of 3200 samples. In contrast, after using augmentation techniques to enrich the dataset, the enhanced dataset has a larger scale, with 3478 samples allotted for training and 640 samples for both testing and validation,

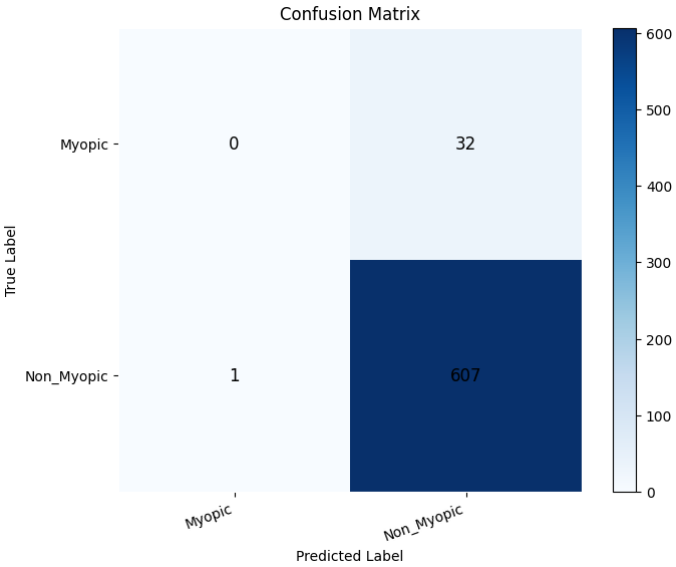

Fig. 27: ResNet50 Confusion Graph (Unfreezed)

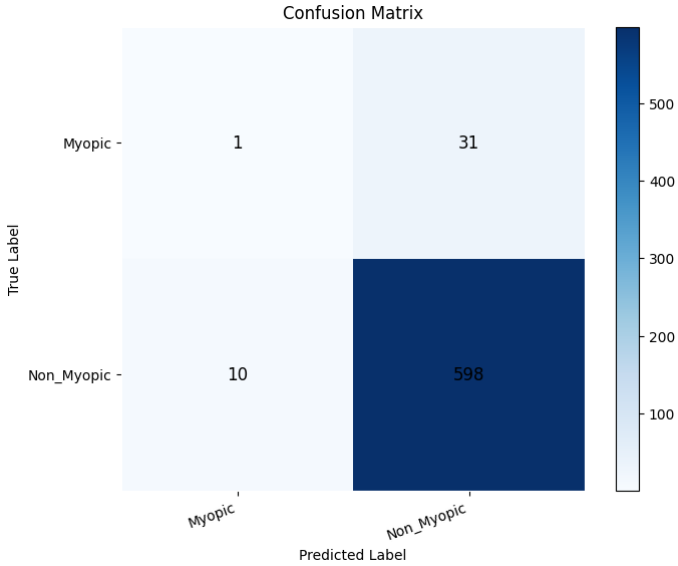

Fig. 28: MobileNetV2 Confusion Graph (Freezed)

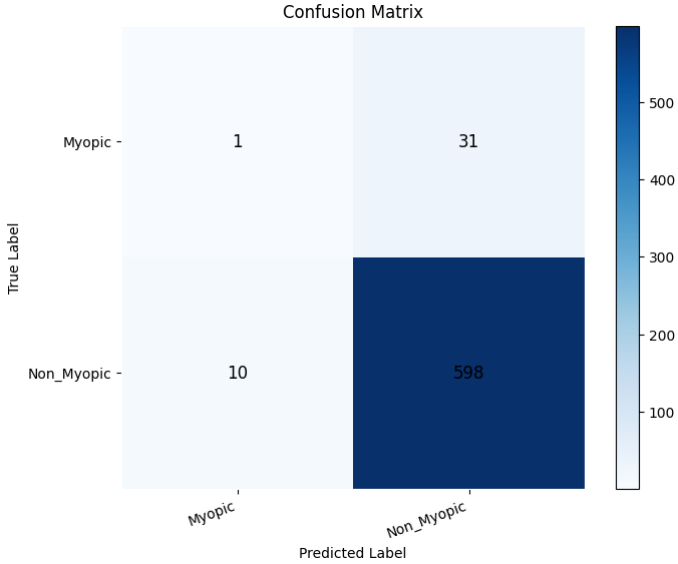

Fig. 29: MobileNetV2 Confusion Graph (Unfreezed)

for a total of 4758. This tabular depiction explains the strategic separation of data for model construction and evaluation, emphasizing the importance of balanced and appropriately sized datasets in robust and dependable machine-learning endeavors.

Table II summarises the accuracy metrics for various freezed models in a classification test, including the number of accurate and incorrect predictions, support, and total accuracy. ResNet50 achieved 93.59% accuracy with 599 correct and 41 incorrect predictions from 640 samples. MobileNetV2 achieved the maximum accuracy of 96.41%, with an equal distribution of correct and incorrect predictions. ResNet152 performed similarly to ResNet50, with a slightly better accuracy of 93.75%, having 600 correct and 40 wrong predictions. The DenseNet121 model's findings are not shown in the

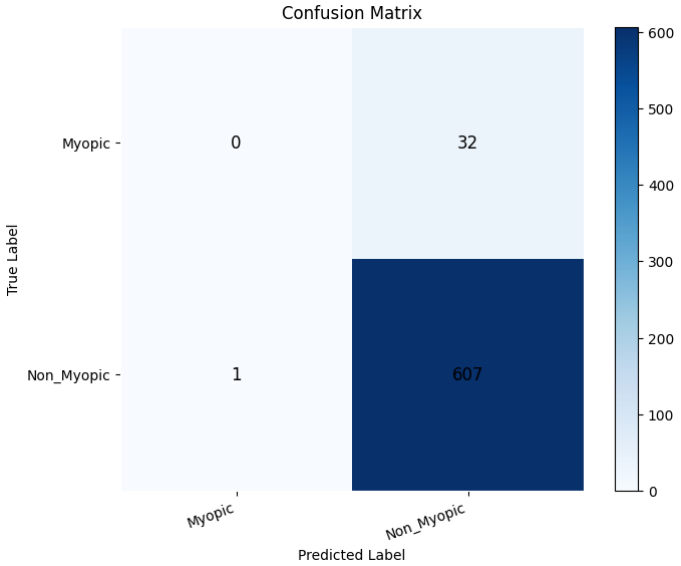

Fig. 30: ResNet152V2 Confusion Graph (Freezed)

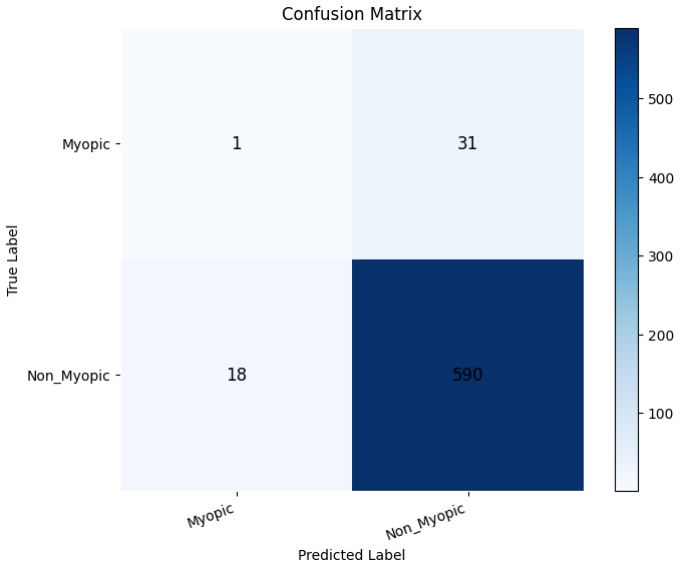

Fig. 31: ResNet152V2 Confusion Graph (Unfreezed)

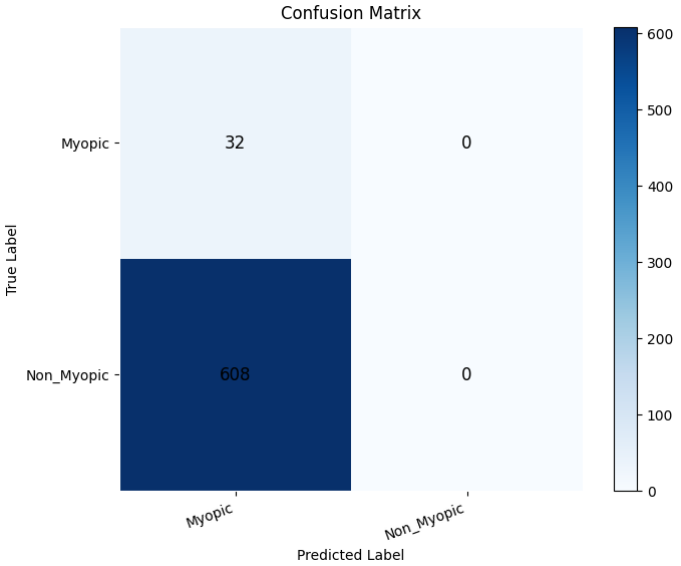

Fig. 32: EfficientNetB7 Confusion Graph (Freezed)

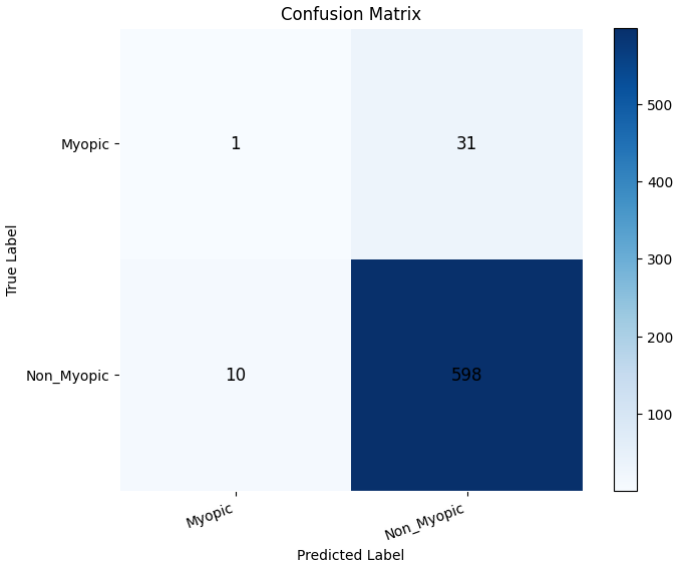

Fig. 33: EfficientNetB7 Confusion Graph (Unfreezed)

table. In sharp contrast, EfficientNetB7 scored substantially worse, with only 32 correct predictions and an accuracy of 5.00%, indicating probable problems with its frozen state or application in this particular challenge.

Based on the supplied accuracy measures, MobileNetV2 emerges as the most effective model, with an accuracy of 96.41%. This shows that it is robust and reliable for the specified classification task. ResNet50 and ResNet152 demonstrate strong performance, with accuracies of 93.59% and 93.75%, respectively, making them viable options. However, EfficientNetB7's very poor performance, with an accuracy of only 5.00%, suggests substantial limitations or potential flaws in its frozen implementation for this task, necessitating further examination prior to deployment. Based on these findings, MobileNetV2 is recommended as the primary model for this classification assignment, with ResNet50 and ResNet152 as backup possibilities. DenseNet121 requires further evaluation to determine its applicability.

Table III summarises the performance metrics of various unfrozen models on a classification test, with emphasis on the number of accurate and wrong predictions, support, and total accuracy. ResNet50 achieved 94.84% accuracy with 607 correct and 33 incorrect predictions from 640 samples. ResNet152 did somewhat better, with an accuracy of 95.15% based on 609 correct and 31 wrong predictions. DenseNet121 also performed well, but with a lesser accuracy of 92.34%, resulting in 591 correct and 49 incorrect predictions. This table does not include findings for MobileNetV2 and EfficientNetB7.

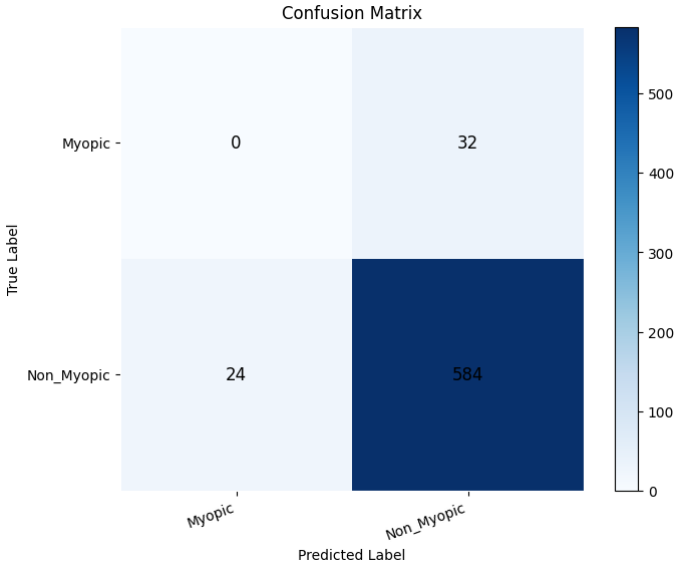

Fig. 34: DenseNet121 Confusion Graph (Freezed)

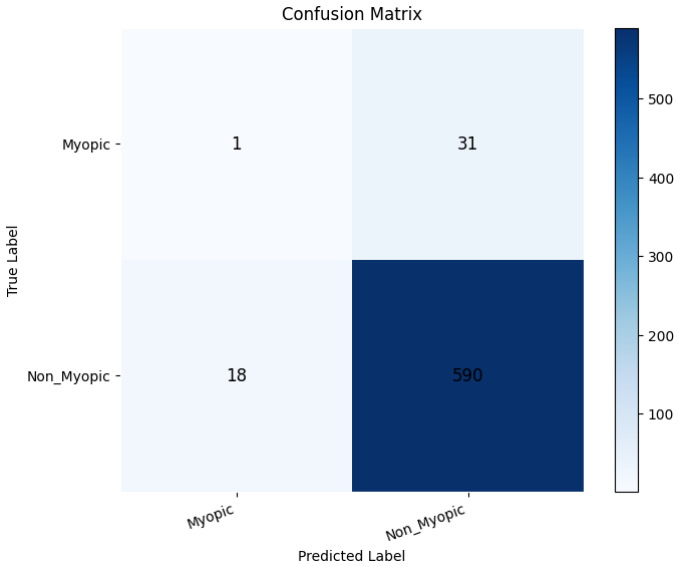

Fig. 35: DenseNet121 Confusion Graph (Unfreezed)

When analyzing the performance of unfrozen models, ResNet152 leads with an accuracy of 95.15%, followed by ResNet50 at 94.84%. Both models demonstrate a slight improvement over their frozen counterparts. DenseNet121, which has a respectable accuracy of 92.34%, is slightly less effective. The lack of data for MobileNetV2 and EfficientNetB7 in the unfrozen state prevents a thorough comparison. Based on the available data, ResNet152 is recommended as the principal model for the classification task, with ResNet50 serving as a viable option. MobileNetV2 and EfficientNetB7 require further evaluation to complete the assessment.

B. Conclusion

In summary, unfreezing the models generally improves accuracy, with ResNet152 emerging as the best performer after fine-tuning. The large boost in model performance achieved through fine-tuning emphasises the necessity of allowing models to completely adapt to the unique subtleties of fresh datasets. EfficientNetB7's poor performance suggests potential problems or incompatibilities that require additional examination. MobileNetV2's great performance in the freezed state makes it an excellent contender, particularly where computational economy is critical. DenseNet121, while effective, demonstrates that architecture selection should be based on the individual requirements of the task. Further examination and adjustment of these models, particularly those with lacking unfrozen data, would provide a more complete picture of their capabilities and limitations.

REFERENCES

[1] W. Bismi and J. Na'am, "Classification of myopia levels using deep learning methods on fundus image," *Medical Informatics Technology, University of Nusa Mandiri Jakarta, Indonesia*, 2023, article History: Received 08 June 23; Final Revision: 10 June 23; Accepted: 20 June 23; Online Publication: 30 June 23.

[2] N. Széll *et al.*, "Myopia-26, the female-limited form of early-onset high myopia, occurring in a european family," *Orphanet J. Rare Dis.*, vol. 16, no. 1, pp. 1–13, 2021.

[3] L. Armi and S. Fekri-Ershad, "Texture image analysis and texture classification methods - a review," *ArXiv*, vol. 2, no. 1, pp. 1–29, 2019. [Online]. Available: http://arxiv.org/abs/1904.06554

[4] G. Litjens *et al.*, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, no. December 2012, pp. 60–88, 2017.

[5] J. Zhang, Y. Xie, Q. Wu, and Y. Xia, "Medical image classification using synergic deep learning," *Med. Image Anal.*, vol. 54, pp. 10–19, 2019.

[6] S. Jia, S. Jiang, Z. Lin, N. Li, M. Xu, and Y. Shiqi, "A survey: Deep learning for hyperspectral image classification with few labeled samples," *Neurocomputing*, vol. 448, pp. 179–204, 2021.

[7] F. S. Ni'mah, T. Sutojo, and D. R. I. M. Setiadi, "Identification of herbal medicinal plants based on leaf image using gray level co-occurence matrix and k-nearest neighbor algorithms," *J. Teknol. dan Sist. Komput.*, vol. 6, no. 2, pp. 51–56, 2018.

[8] R. J. S. Raj, S. J. Shobana, I. V. Pustokhina, D. A. Pustokhin, D. Gupta, and K. Shankar, "Optimal feature selection-based medical image classification using deep learning model in internet of medical things," *IEEE Access*, vol. 8, pp. 58 006–58 017, 2020.

[9] N. Calik, M. A. Belen, and P. Mahouti, "Deep learning base modified mlp model for precise scattering parameter prediction of capacitive feed antenna," *Int. J. Numer. Model. Electron. Networks, Devices Fields*, vol. 33, no. 2, 2020.

[10] P. Hurtik, V. Molek, and J. Hula, "Data preprocessing technique for neural networks based on image represented by a fuzzy function," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 7, pp. 1195–1204, 2020.

[11] W. H. Organization, "The impact of myopia and high myopia: report of the joint world health organisation-brian holden vision institute global scientific meeting on myopia," 2015.

[12] R. Du *et al.*, "Deep learning approach for automated detection of myopic maculopathy and pathologic myopia in fundus images," *Ophthalmol. Retin.*, pp. 1–10, 2021.

[13] S. Pachade *et al.*, "Retinal fundus multi-disease image dataset (rfmid)," 2020, [Online]. Available: https://dx.doi.org/10.21227/s3g7-st65.