# Automated Classification of Parasitic Worm Eggs Based on Transfer Learning and Fine-Tuned CNN Models

Ira Puspita Sari[1], Budi Warsito[2], Oky Dwi Nurhayati[3]
Doctoral Program of Information Systems-School of Postgraduate Studies, Diponegoro University, Semarang, Indonesia[1, 2, 3]
Department of Informatics Engineering-Faculty of Engineering, Universitas Abdurrab, Pekanbaru, Indonesia[1]

***Abstract*—Classification of worm eggs is important for diagnosing worm diseases, but the manual process is time-consuming. This study designs an image classification system using Convolutional Neural Network (CNN), transfer learning, and fine-tuning. The main goal of this study is to create a CNN model to sort parasitic worm eggs into groups. It does this by comparing three CNN architectures: EfficientNetB0, MobileNetV3, and ResNet50; it also creates classification technology for diagnosing worm infections. We applied transfer learning with pre-trained models and fine-tuned them for the IEEE parasitic egg dataset. The results reveal that EfficientNetB0 is superior, with an accuracy of 95.36%, precision of 95.80%, recall of 95.38%, and F1-score of 95.48%. It performs better and more efficiently than the other two architectures. Applying transfer learning and fine-tuning improves model performance, with EfficientNetB0 consistently outperforming. Furthermore, visual similarities between classes in the dataset likely cause prediction errors. Therefore, this system can support the diagnosis of worm diseases with high efficiency and accuracy.**

***Keywords*—*Classification; Convolutional Neural Network; EfficientNetB0; MobileNetV3; ResNet50***

## I. Introduction

Detection of intestinal parasitic infections remains a significant challenge, especially in developing countries with tropical climates such as Indonesia. Conventional diagnosis using a microscope relies heavily on the skills of the laboratory technician, making it prone to errors [1]. The morphological similarity of worm eggs and the presence of faeces in the sample typically cause such errors [2]. In addition, this examination process is rather time-consuming, with an expert technician requiring an average of 8 to 10 minutes to examine one sample [3]. Furthermore, limited diagnostic accuracy also affects the effectiveness of treatment. Therefore, researchers can significantly improve the effectiveness of traditional diagnostics by developing automated diagnostic systems.

In recent years, digital image processing technology has been increasingly used in the medical world to increase the speed and accuracy of diagnosis. Advances in computer vision, particularly Convolutional Neural Networks (CNNs), have offered robust solutions for image classification. CNNs use artificial neural networks to process and analyze images, resulting in significant performance in digital image recognition [4] [5] [6]. One of the superiorities of CNNs is their ability to automatically learn relevant features from large amounts of data, thereby avoiding the need for manual extraction [7].

Researchers have developed various CNN architectures for image classification, such as AlexNet, EfficientNet, LeNet, MobileNet, and ResNet, each offering distinct advantages [8]. This study aims to evaluate three CNN architectures, i.e., EfficientNetB0, MobileNetV3, and ResNet50. These three architectures are trained on large datasets, can produce rich and generalizable feature representations, and allow faster convergence during fine-tuning [9].

To overcome dataset limitations and improve model performance, transfer learning and fine-tuning techniques become effective strategies. Transfer learning enables the use of pre-trained CNN models on large datasets for specific tasks with minimal fine-tuning, reduced training time, and efficient use of limited labelled data, thus being ideal for tasks with little data [10] [11] [12]. Additionally, pre-trained weights also improve model accuracy and performance [13]. Meanwhile, fine-tuning adapts models to recognize specific characteristics of new datasets, such as worm eggs in microscopic images, and improve detection accuracy and diagnostic capabilities [14] [15]. Fine-tuning also helps achieve improved performance on limited data and accelerates training by leveraging knowledge from pre-trained models [16] [17].

A prior study has reported that CNN-based image classification technology can reach high accuracy in identifying three different types of worm eggs, namely *Schistosoma spp*., *Ascaris spp*., and *Trichuris spp*., with accuracy rates of 95.31%, 86.36%, and 80.00%, respectively, indicating the model's ability to handle the complexity of egg morphology and variations in the dataset [18]. Another study found that CNN can detect protozoan cysts and worm eggs in human faeces with accuracy rates of 96.25% and 95.08%, respectively [19].

This study aims to develop a worm egg classification system based on image processing techniques using Convolutional Neural Networks (CNN), transfer learning, and fine-tuning. Specifically, it focuses on building CNN models to classify parasitic worm eggs from digital images, comparing the performance of three architectures—EfficientNetB0, MobileNetV3, and ResNet50—in identifying worm eggs, and enhancing this classification technology to support the diagnosis of human worm infections. Additionally, the study analyzes factors that influence the accuracy and efficiency of CNN-based classification systems.

This study also identifies key challenges in parasitic worm egg classification, including high visual similarity among certain egg types, noise and inconsistency in microscopic image quality, and limited dataset diversity, which may affect model generalization. Furthermore, the study explores future directions by evaluating the performance limitations of current architectures and proposing improvements through lightweight models or attention mechanisms suitable for edge deployment.

## II. Research Method

This study employed the architectures of EfficientNetB0, MobileNetV3, and ResNet50. Fig. 1 shows the flowchart of worm egg classification using transfer learning on the CNN models. The research methods cover data collection, preprocessing, model development, transfer learning of the pre-trained CNN models, fine-tuning, and performance evaluation.

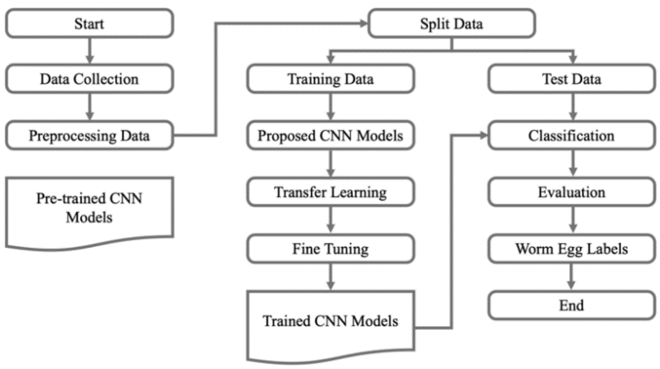



Fig. 1. Flowchart of the research procedure for worm egg classification using the CNN models.

### A. Data Collection

We collected data to obtain the required image dataset and uploaded it to Google Drive. Data were collected systematically using class-based sampling techniques, and sorting images represented each species class. The data used were secondary data in the form of RGB (Red, Green, Blue) images, obtained from the IEEE Data Port website (https://ieee-dataport.org/) [20]. This dataset included eleven categories of worm eggs, i.e., *Ascaris lumbricoides, Capillaria philippinensis, Enterobius vermicularis, Fasciolopsis buski*, Hookworm, *Hymenolepis nana, Hymenolepis diminuta, Opisthrochis viverrine, Paragonimus spp., Taenia spp., and Trichuris trichiura*. The total dataset reached 11,000 images, with each class comprising 1000 images.

### B. Data Preprocessing

Before training, the data were prepared following preprocessing steps, which included resizing with padding, dataset splitting, data augmentation, and input standardization. Resizing with padding is useful for maintaining the image dimensions on each layer and preventing the loss of edge information in the image. First, the dataset was divided into three subsets: training, validation, and testing data [21]. Data augmentation was applied to the training and validation data, using techniques such as rotation, shifting, and zooming and resizing the images to 224 x 224 pixels [22]. These augmentation techniques were employed to increase the diversity of the dataset by generating new variations of the existing dataset and changing the position, scale, and orientation of objects [23]. After augmentation, the dataset was grouped into batches for training.

### C. CNN Models

Convolutional Neural Network (CNN) was recognized as a popular deep learning model for image data analysis [24]. CNN comprised convolutional layers for extracting features from images, pooling layers for reducing matrix dimensions and accelerating computation, and fully connected layers for classification. Pooling layers, such as average and max pooling, were positioned after the convolutional layers to retain important information. In this study, three pre-trained CNN models—EfficientNetB0, MobileNetV3, and ResNet50—were utilized. These models had been pre-trained using ImageNet data [25] and were made available in the TensorFlow library [26].

EfficientNet was a series of CNN models designed to improve accuracy and efficiency using scaling settings. The superiority of EfficientNet was demonstrated by its ability to provide high accuracy while reducing parameters and FLOPS (Floating Point Operations Per Second). A combined scaling method was applied to three network dimensions: width (number of channels per layer), depth (number of CNN layers), and resolution (image size) [27]. The architecture of EfficientNet10 was presented in Fig. 2.

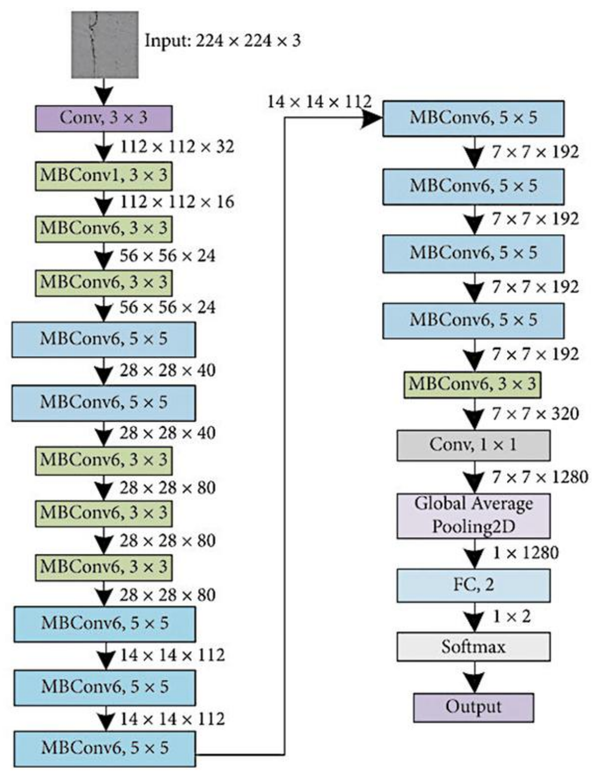



Fig. 2. Architecture of EfficientNet10.

MobileNet was an artificial neural network architecture that Google developed for image processing and object recognition on resource-constrained devices. MobileNetV3 was divided into two models: MobileNetV3-Large for high-resource environments and MobileNetV3-Small for low-resource environments [28]. This architecture was formed by combining depthwise separable convolutions from MobileNetV1, linear bottleneck, and inverted residuals from MobileNetV2, and lightweight attention modules based on squeeze and excitation from MnasNet to enhance accuracy. The architecture of MobileNetV3-Large was presented in Fig. 3.

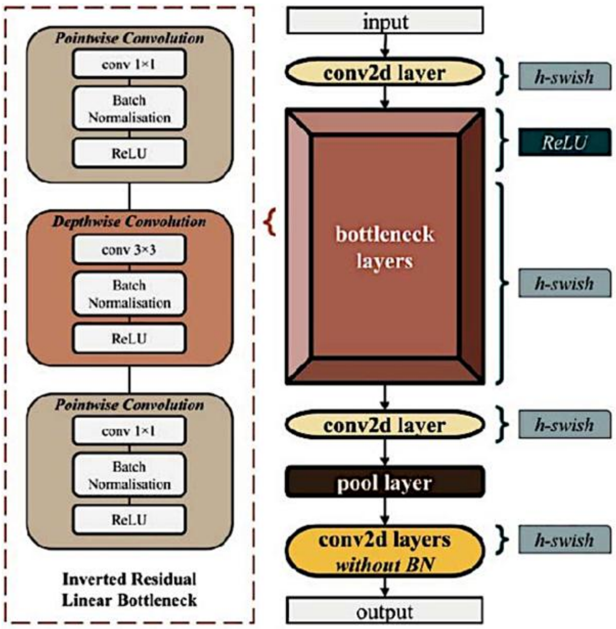



Fig. 3. Architecture of MobileNetV3-Large.

Several versions of ResNet were developed, one of which was ResNet-50, which used 50 layers of a neural network. ResNet-50 introduced the concept of shortcut connections to address the vanishing gradient problem, which occurred when increasing the depth of the network. With shortcut connections, gradients could pass through deeper layers without being significantly reduced, improving performance and accuracy [29]. The architecture of ResNet-50 was presented in Fig. 4.

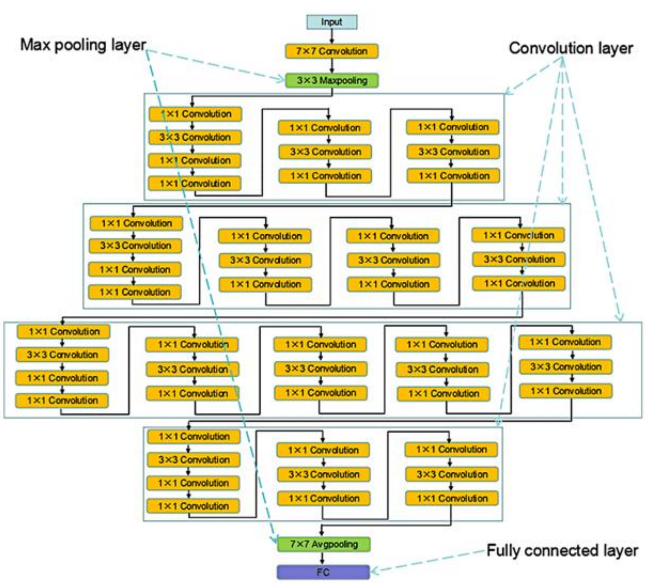



Fig. 4. Architecture of ResNet50.

## D. Transfer Learning and Fine-Tuning

Transfer learning was an approach in machine learning that used pre-trained models to solve new problems, either in the same or different domains. In transfer learning, a base model with general knowledge from large datasets, such as ImageNet, was used as a feature extractor to overcome data limitations and accelerate the convergence process during model training [30]. Furthermore, a classification head was added to the base model and trained using a smaller dataset to solve a specific task. Only the classification layer was trained to adapt to the task to be solved, as the base model layers were typically frozen since they already had a good representation of general features.

After the initial stage, fine-tuning was performed to improve the performance of the pre-trained model on new tasks or datasets. In this stage, previously frozen layers in the base model were reactivated (unfrozen) to allow for adjustments during training. This method aimed to refine the feature representation generated by the base model to suit the new dataset's characteristics better. Fine-tuning was performed using a smaller learning rate to optimize model accuracy [31].

## E. Evaluation Metrics

The performance of a multiclass classification model was evaluated using various metrics, including accuracy, precision, recall, and F1-score [32]. These metrics were calculated based on information from the confusion matrix, which compared the model’s predicted results and the actual data. The formulas for accuracy, precision, recall, and F1-score were presented in Eqs. (1), (2), (3), and (4), respectively. Parameters used in this calculation were: TP (True Positive): correct prediction for the positive class; TN (True Negative): correct prediction for the negative class; FP (False Positive): wrong prediction for the positive class; and FN (False Negative): wrong prediction for the negative class. Evaluation using these metrics allowed for a comprehensive assessment of model performance in classifying multiclass data.

$$Accuracy = \frac{\mathrm{TP+TN}}{\mathrm{TP+FP+FN+TN}} \tag{1}$$

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

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

$$F1\ Score = 2\left(\frac{\mathrm{precision.recall}}{\mathrm{precision+recall}}\right) \tag{4}$$

# III. Result and Discussion

This study proposes a method for classifying worm egg images with a transfer learning approach using CNN models. The proposed method is developed using Python programming language and trained on Google Colab by utilizing GPU.

## A. Dataset

The dataset used in this study covers eleven types of worm eggs, each consisting of 1000 images, making a total of 11,000 images. This dataset varies in size, magnification level, lighting conditions, blur level, and background. Image samples from the worm egg dataset can be seen in Fig. 5.

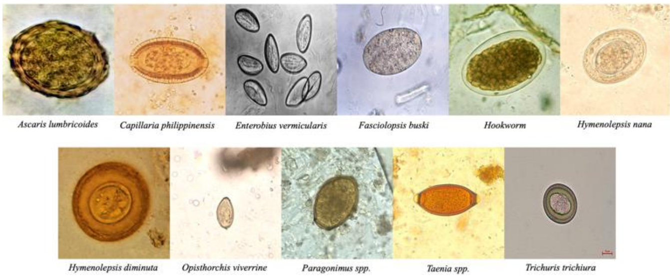



Fig. 5. Image samples from the worm egg dataset.

One of the main challenges identified in the dataset is the variation in image quality due to differences in lighting, magnification, background, and resolution. These inconsistencies can introduce bias and reduce the model's generalization ability. Additionally, certain species of parasitic eggs show high morphological similarity, which complicates classification. Misclassifications often occur due to subtle differences that are difficult to distinguish even by an expert.

## B. Preprocessing Result

The preprocessing stage was carried out to ensure uniformity in the size of all images in the dataset by changing them to dimensions of 224x224x3 through resizing. However, this method has the potential to blur or even eliminate the image's main object. A cropping technique, which involves cutting the part of the image that contains the main object to a certain size and saving the results in the desired dimensions, was used to overcome this problem. In addition, a color scheme conversion was performed using the cvtColor function to ensure that the image conforms to the RGB (Red, Green, Blue) format, which is compatible with the models used. This preprocessing stage utilized several Python modules, such as NumPy, glob, OpenCV (cv2), and Pickle. The final result of data preprocessing is shown in Fig. 6.

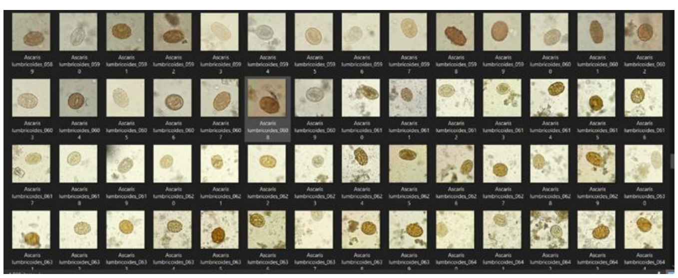


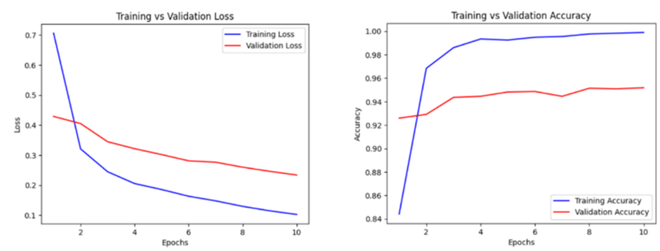
Fig. 6. Results of data augmentation.

## C. Transfer Learning and Fine-Tuning Result

The proposed method used pre-trained EfficientNetB0, MobileNetV3, and ResNet50 models from ImageNet, the base model used as a feature extraction layer. The training process was carried out in two stages: the transfer learning phase and the fine-tuning phase. In transfer learning, the models were trained for ten epochs by monitoring the best performance based on the lowest validation loss value. Only the classification layer was trained at this stage, while the other layers remained frozen. In the second phase, namely fine-tuning, the models were re-trained for ten epochs by unfreezing the base model layers. This allows all layers, including the feature extraction layer, to be tuned with weight layers relevant to the worm egg dataset. The training process in this phase used a lower learning rate to ensure that parameter adjustments run stably.

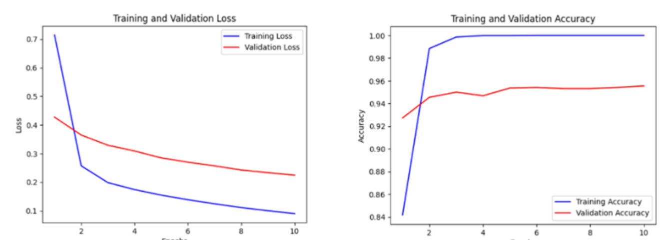
Based on the training and testing results, an analysis was carried out on the main performance metrics: training loss, validation loss, training accuracy, and validation accuracy. The comparison graph of these metrics was visualized using the matplotlib module. The loss graph shows a gradual decrease in value as the number of epochs increases, indicating an increase in the model's ability to predict until convergence. The accuracy graph illustrates a similar trend, showing a steady rise throughout training, with the highest accuracy achieved when the models successfully identify patterns in the training data. Meanwhile, the validation loss and validation accuracy graphs were used to evaluate the generalization ability of the models to unused validation data during training. These graphs help identify potential problems, such as overfitting or underfitting, which can be observed if there is a significant difference between training and validation metrics.

Training process graphs of EfficientNetB0, MobileNetV3, and ResNet50 models are shown in Figs. 7, 8, and 9, with panel (a) describing the transfer learning phase and panel (b) describing the fine-tuning phase.



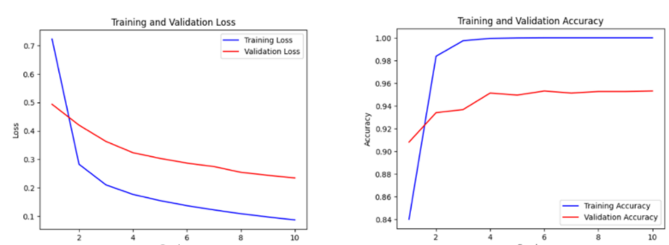

(a) Transfer learning phase (b) Fine-tuning phase

Fig. 7. Training graph of EfficientNetB0 model.




(a) Transfer learning phase (b) Fine-tuning phase

Fig. 8. Training graph of MobileNetV3 model.




(a) Transfer learning phase (b) Fine-tuning phase

Fig. 9. Training graph of ResNet50 model.

Based on the graphs displayed, it can be concluded that the models developed in this study have good learning abilities. This can be seen from the consistent increase in accuracy values and the steady decrease in loss values as the number of epochs rises. The difference between training loss/accuracy and validation loss/accuracy is relatively small, indicating that the models can adequately generalize unused data during training. Further analysis reveals that each model architecture produces varied accuracy and loss performance with the same training parameters, although the difference is only a few per cent. This reflects the influence of architectural characteristics in capturing data patterns in the classification task being performed.

### D. Model Evaluation

Evaluations of the three models, namely EfficientNetB0, MobileNetV3, and ResNet50, were made based on the models trained after the transfer learning and fine-tuning phases using precision, recall, accuracy, and F1-score values (Table I).

TABLE I. COMPARISON OF PERFORMANCE EVALUATION

| Model | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| EfficientNetB0 | 95.36% | 95.80% | 95.38% | 95.48% |
| MobileNetV3 | 94.54% | 94.85% | 94.60% | 94.65% |
| ResNet50 | 94.09% | 94.94% | 94.10% | 94.31% |

As seen in Table I, the EfficientNetB0 model shows the best performance on the validation dataset during the fine-tuning phase compared to other models, with an accuracy of 95.36%. Moreover, this model has better precision, recall, and F1-score than the other two, whose values reach 95.80%, 95.38%, and 95.48%, respectively. These results show that EfficientNetB0 can better recognize and classify both positive and negative classes accurately.

The superiority of EfficientNetB0 evaluation metrics can be attributed to its efficient architecture design and powerful feature extraction capability through the MBConv (Mobile Inverted Residual Bottleneck Convolution) block. The MBConv structure enables the model to adaptively extract important features, improving classification accuracy on the parasitic worm egg dataset. A careful approach to scalability also contributes to model performance, allowing efficient computational and parameter optimization without sacrificing accuracy.

Higher evaluation results on accuracy, precision, recall, and F1-score metrics indicate that EfficientNetB0 is a superior architecture for parasitic worm egg classification. This better performance proves that EfficientNetB0 addresses the classification challenges more effectively than ResNet50 and MobileNetV3Large on the same dataset. In addition to the training evaluation, the testing evaluation on parasitic worm egg image data that the models have never seen before detects two prediction errors. Due to limitations in the generalization capabilities of classification models, these errors are normal during testing.

The application of EfficientNet-B0 in classifying parasitic worm eggs demonstrates significant potential in enhancing the accuracy, efficiency, and accessibility of parasite infection diagnostics. Butploy et al. [33] successfully identified three types of *Ascaris lumbricoides* eggs using the EfficientNet-B0 deep learning architecture, achieving an accuracy of 93.33%. Furthermore, Mirzaei et al. [34] reported that EfficientNet-B0 effectively extracts relevant features for helminth egg identification, reducing misclassification rates commonly observed in conventional microscopy-based methods.

Aldahoul et al. [35] also found that combining EfficientNet with parasite detection techniques significantly improved the classification performance of microscopic images. Meanwhile, Kumar et al. [36] emphasized that integrating efficient models such as YOLOv5 can accelerate healthcare system responses to parasitic infections, highlighting the synergy between rapid detection and precise classification.

Although further studies are needed to support implementation on edge devices, lightweight models like EfficientNet-B0 offer a promising solution for fast and accurate detection, particularly in resource-limited or remote areas.

## IV. CONCLUSION

Based on the results of this study, the EfficientNetB0 architecture shows the best performance with an accuracy of 95.36%, precision of 95.80%, recall of 95.38%, and F1-score of 95.48%, reflecting high ability in worm egg classification. This study also reveals that applying transfer learning and fine-tuning can significantly improve model performance, with variations in CNN architecture having different impacts on performance, where EfficientNetB0 consistently outperforms the other two architectures. The prediction errors are most likely caused by visual similarities between classes in the dataset, making it difficult for the models to identify the class correctly. For future work, we propose integrating attention mechanisms, deeper exploration of lightweight CNN models like EfficientNet-Lite, and validation of the classification system in real clinical environments using edge devices.

## REFERENCES

[1] H. R. Hadi, K. Ghazali, I. Khalidin, and M. Zeehaida, "Human parasitic worm detection using image processing technique," in *2012 International Symposium on Computer Applications and Industrial Electronics (ISCAIE)*, 2012, pp. 196–200. doi: 10.1109/ISCAIE.2012.6482095.

[2] Y. Yang, D. Park, H. Kim, M. Choi, and J. Chai, "Automatic identification of human helminth eggs on microscopic fecal specimens using digital image processing and an artificial neural network," *IEEE Transactions on Biomedical Engineering*, vol. 48, pp. 718–730, 2001. doi: 10.1109/10.923789.

[3] O. Holmstrom, N. Linder, B. Ngasala, A. Martensson, E. Linder, M. Lundin, H. Moilanen, A. Suutala, V. Diwan, and J. Lundin, "Point-of-care mobile digital microscopy and deep learning for the detection of soil-transmitted helminths and schistosoma haematobium," *Global Health Action*, vol. 10, pp. 49–57, 2017. doi: 10.1080/16549716.2017.1337325.

[4] S. Dargan, M. Kumar, M. R. Ayyagari, and G. Kumar, "A Survey of Deep Learning and Its Applications: A New Paradigm to Machine Learning," *Archives of Computational Methods in Engineering*, vol. 27, no. 4, pp. 1071–1092, 2020. doi: 10.1007/s11831-019-09344-w.

[5] A. Vaisman, N. Linder, J. Lundin, A. Orchanian-Cheff, J. T. Coulibaly, R. K. Ephraim, and I. I. Bogoch, "Artificial intelligence, diagnostic imaging and neglected tropical diseases: Ethical implications," *Bulletin of the World Health Organization*, vol. 98, pp. 288–289, 2020. doi: 10.2471/BLT.19.237560.

[6] S. Kumar, T. Arif, A. S. Alotaibi, M. B. Malik, and J. Manhas, "Advances towards automatic detection and classification of parasites microscopic images using deep convolutional neural network: Methods, models and research directions," *Archives of Computational Methods in Engineering*, vol. 30, pp. 2013–2039, 2023. doi: 10.1007/s11831-022-09858-w.

[7] M. I. Razzak, S. Naz, and A. Zaib, "Deep learning for medical image processing: Overview, challenges and the future," in *Classification in BioApps*, 2008, pp. 323–350. doi: 10.1007/978-3-319-65981-7_12.

[8] S. Patel, "A comprehensive analysis of convolutional neural network models," *International Journal of Advanced Science and Technology*, vol. 29, no. 4, pp. 771–777, 2020.

[9] T. Suwannaphong, S. Chavana, S. Tongsom, D. Palasuwan, T. H. Chalidabhongse, and N. Anantrasirichai, "Parasitic egg detection and classification in low-cost microscopic images using transfer learning," *SN Computer Science*, vol. 5, no. 82, pp. 1-10, 2023. doi: 10.1007/s42979-023-02406-8.

[10] Z. Zhao, L. Alzubaidi, J. Zhang, Y. Duan, and Y. Gu, "A comparison review of transfer learning and self-supervised learning: Definitions, applications, advantages and limitations," *Expert Systems with Applications*, vol. 242, Elsevier Ltd, 2024. doi: 10.1016/j.eswa.2023.122807.

[11] K. Liu, Q. Peng, Y. Che, Y. Zheng, K. Li, R. Teodorescu, D. Widanage, and A. Barai, "Transfer learning for battery smarter state estimation and ageing prognostics: Recent progress, challenges, and prospects," *Advances in Applied Energy*, vol. 9, 100117, 2023. doi: 10.1016/j.adapen.2022.100117.

[12] H. A. Al-Iiedane and A. I. Mahameed, "Satellite images for roads using transfer learning," *Measurement: Sensors*, vol. 27, 100775, 2023. doi: 10.1016/j.measen.2023.100775.

[13] H. D. Jahja and N. Yudistira, "Mask usage recognition using vision transformer with transfer learning and data augmentation," *Intelligent Systems with Applications*, vol. 17, 200186, 2023. doi: 10.1016/j.iswa.2023.200186

[14] S. Jiang, Q. Chen, Y. Xiang, Y. Pan, X. Wu, and Y. Lin, "Confounder balancing in adversarial domain adaptation for pre-trained large models fine-tuning," *Neural Networks*, vol. 173, 2024. doi: 10.1016/j.neunet.2024.106173.

[15] M. A. Talukder, M. A. Layek, M. Kazi, M. A. Uddin, and S. Aryal, "Empowering covid-19 detection: Optimizing performance through fine-tuned efficientnet deep learning architecture," *Computers in Biology and Medicine*, vol. 168, 107789, 2024. doi: 10.1016/j.compbiomed.2023.107789.

[16] A. Rahdar, M. Chahoushi, and S. A. Ghorashi, "Efficiently improving the Wi-Fi-based human activity recognition, using auditory features, autoencoders, and fine-tuning," *Computers in Biology and Medicine*, vol. 108232, 2024. doi: 10.1016/j.compbiomed.2024.108232.

[17] M. A. Talukder, M. M. Islam, M. A. Uddin, A. Akhter, M. A. J. Pramanik, S. Aryal, M. A. A. Almoyad, K. F. Hasan, and M. A. Moni, "An efficient deep learning model to categorize brain tumor using reconstruction and fine-tuning," *Expert Systems with Applications*, vol. 230, 120534, 2023. doi: 10.1016/j.eswa.2023.120534.

[18] K. E. Delas Peñas, E. A. Villacorte, P. T. Rivera, and P. C. Naval, "Automated detection of helminth eggs in stool samples using convolutional neural networks," in *Proceedings of the 2020 IEEE Region 10 Conference (TENCON)*, Osaka, Japan, 16–19 Nov. 2020. doi: 10.1109/TENCON50793.2020.9293746.

[19] K. M. Naing, S. Boonsang, S. Chuwongin, V. Kittichai, T. Tongloy, S. Prommongkol, P. Dekumyoy, and D. Watthanakulpanich, "Automatic recognition of parasitic products in stool examination using object detection approach," *PeerJ Comput. Sci.*, vol. 8, p. e1065, 2022. doi: 10.7717/peerj-cs.1065.

[20] Palasuwan, D.; Naruenatthanaset, K.; Kobchaisawat, T.; Chalidabhongse, T. H.; Nunthanasup, N.; Boonpeng, K.; Anantrasirichai, N. Parasitic Egg Detection and Classification in Microscopic Images. IEEE Data Port. Available online: https://ieee-dataport.org/competitions/parasitic-egg-detection-and-classification-microscopic-images#files

[21] F. Kong and R. Henao, "Efficient Classification of Very Large Images with Tiny Objects," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2022-June, pp. 2374–2384, 2022, doi: 10.1109/CVPR52688.2022.00242.

[22] L. F. Sánchez-Peralta, A. Picón, F. M. Sánchez-Margallo, and J. B. Pagador, "Unravelling the effect of data augmentation transformations in polyp segmentation," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 15, no. 12, pp. 1975–1988, 2020, doi: 10.1007/s11548-020-02262-4.

[23] L. Alzubaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00444-8.

[24] J. D. Kelleher, *Deep learning*. MIT Press, 2019. [Online]. Available: https://books.google.co.id/books?hl=id&lr=&id=b06qDwAAQBAJ.

[25] Z. Zhu *et al.*, "Webface260m: A benchmark unveiling the power of million-scale deep face recognition," *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10492–10502, 2021, doi: 10.1109/CVPR46437.2021.01035.

[26] N. Kumar, M. Rathee, N. Chandran, D. Gupta, A. Rastogi, and R. Sharma, "CrypTFlow: Secure TensorFlow Inference," *2020 IEEE Symposium on Security and Privacy (SP)*, pp. 336–353, 2020, doi: 10.1109/SP40000.2020.00092.

[27] M. A. Basyir, *"Application of Convolutional Neural Network Method With EfficientNet-B4 Architecture for Pneumonia Disease Classification,"* Universitas Islam Negeri Sultan Syarif Kasim, 2021.

[28] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for MobileNetV3," *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1314–1324, 2019. doi: 10.1109/ICCV.2019.00140.

[29] F. Nashrullah, S. A. Wibowo, and G. Budiman, " Epoch Parameter Investigation On ResNet-50 Architecture For Pornography Classification," *Journal of Computer, Electronic, and Telecommunication*, vol. 1, no. 1, 2020. doi: 10.52435/complete.v1i1.51.

[30] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," Artificial Neural Networks and Machine Learning--ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III, vol. 27, pp. 270–279, 2018. doi: 10.1007/978-3-030-01424-7_27.

[31] M. Banjaransari and A. Prahara, "Image classification of wayang using transfer learning and fine-tuning of CNN models," Buletin Ilmiah Sarjana Teknik Elektro, vol. 5, no. 4, pp. 632–641. doi: 10.12928/biste.v5i4.9977.

[32] M. Grandini, E. Bagli, and G. Visani, "Metrics for Multi-Class Classification: an Overview," arXiv, pp. 1–17, Aug. 2020. [Online]. Available: https://arxiv.org/abs/2008.05756v1.

[33] N. Butploy, W. Kanarkard, and P. Intapan, "Deep learning approach for Ascaris lumbricoides parasite egg classification," *Journal of Parasitology Research*, vol. 2021, pp. 1–8, 2021. doi: 10.1155/2021/6648038.

[34] O. Mırzaeı, A. İlhan, E. Güler, K. Süer, and B. Şekeroğlu, "Comparative evaluation of deep learning models for diagnosis of helminth infections," *Journal of Personalized Medicine*, vol. 15, no. 3, p. 121, 2025. doi: 10.3390/jpm15030121.

[35] N. AlDahoul, H. Karim, S. Kee, and M. Tan, "Localization and classification of parasitic eggs in microscopic images using an EfficientDet detector," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2022, pp. 4253–4257. doi: 10.1109/icip46576.2022.9897844.

[36] S. Kumar, T. Arif, G. Ahamad, A. Chaudhary, S. Khan, and M. Ali, "An efficient and effective framework for intestinal parasite egg detection using YOLOv5," *Diagnostics*, vol. 13, no. 18, p. 2978, 2023. doi: 10.3390/diagnostics13182978.