CNN Based Transfer Learning for Malaria Parasite Detection Using Thin-Blood Smear Images

Windra Swastika
Research Center For Artificial Intelligence On Digital Images And Technopreneurship, Informatics Engineering, Universitas Ma Chung, Malang
windra.swastika@machung.ac.id

Benedictus Julian Pradana
Informatics Engineering, Universitas Ma Chung, Malang, Indonesia
311810007@student.machung.ac.id

Romy Budhi Widodo
Informatics Engineering, Universitas Ma Chung, Malang, Indonesia
romy.budhi@machung.ac.id

Rehmadanta Sitepu
Pharmaceutical Science And Biomedicine, Universitas Ma Chung, Malang, Indonesia
rehmadanta.sitepu@machung.ac.id

Gregorius Guntur Sunardi Putra
Informatics Engineering, Universitas Ma Chung, Malang, Indonesia
311810015@student.machung.ac.id

ABSTRACT

Transfer learning has been used in computer vision research, including in the health sector. In the health sector, the input image is generally an x-ray image or a microscopic image. In this study, transfer learning for models that have been trained using CNN to detect malaria parasites in red blood cell images. The deep CNN pre-trained model uses 3 architectures, namely ResNet50V2, EfficientNetB0, and InceptionV3. For each architecture, experiments will be carried out and compare which architecture is better in detecting malaria parasites. Based on experiments conducted without fine tune, the accuracy ranges from 0.76 – 0.81 for ResNet50v2, 0.76 – 0.80 for EfficientNetB0, and 0.77 – 0.82 for InceptionV3.

The dataset is a collection of Blood Smear images which have two classes, uninfected and parasitized. The total number of datasets is 27,558, which is divided into two classes with the same number of different image sizes.

CCS CONCEPTS

• **Image Processing**; • **Object Recognition**;

KEYWORDS

malaria detection, CNN, transfer learning

ACM Reference Format:
Windra Swastika, Benedictus Julian Pradana, Romy Budhi Widodo, Rehmadanta Sitepu, and Gregorius Guntur Sunardi Putra. 2022. CNN Based Transfer Learning for Malaria Parasite Detection Using Thin-Blood Smear Images. In *7th International Conference on Sustainable Information Engineering and Technology 2022 (SIET '22), November 22, 23, 2022, Malang, Indonesia.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3568231.3568243

SIET '22, November 22, 23, 2022, Malang, Indonesia

ACM ISBN 978-1-4503-9711-7/22/11…$15.00
https://doi.org/10.1145/3568231.3568243

1 INTRODUCTION

Deep learning is a method of machine learning that is based on the learning process of data representation. The deep learning learning process is divided into several layers, where each layer will transform the input data into an abstract but compact data representation. The use of Deep Learning to perform object recognition requires large resources and long computational time [1]. An important principle used in deep learning is that when a large amount of input data is used, deep learning has the potential to produce models that are able to generalize well. On the other hand, if the amount of input data is small, then there is a potential that the resulting model will be overfitting. The use of large amount of data requires huge resources and the training time required to be long. Another problem that may arise is that obtaining large amounts of data will be costly, especially for patient-related datasets. One popular way to overcome resource limitations is using transfer learning.

Transfer Learning is a method for taking models that are able to generalize to complex tasks to be reused as starting points for different tasks. In transfer learning, the dataset used for the training process does not have to meet independent and identically distributed (i.i.d) criteria as is the case with training data in the deep learning training process [2] Thus, with a small amount of data, the training process using transfer learning can be run well and fast without having to do it from scratch.

Transfer learning has been used in computer vision research, including in the health sector. In the health sector, the input image is generally an x-ray image or a microscopic image. Detecting COVID-19, for example, requires x-ray image data of the lungs. A study conducted by Sahinbas and Catak [3] using transfer learning from pre-trained models VGG16, VGG19, RestNet, DenseNet, and InceptionV3 was able to achieve an accuracy of 0.80, 0.60, 0.50, 0.60, and 0.60.

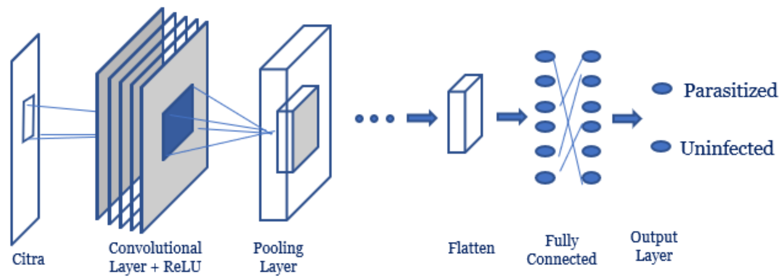

Figure 1: General CNN architecture consists of input image, convolution layer, pooling layer and output layer

The blood smear method on red blood cells can detect malaria infection [4]. Transfer learning can work well on microscopic cell-images as did by Reddy and Juliet [5] able to achieve 95.91% accuracy using the RestNet50 pre-trained model on Malaria Cell Images data. The fine-tuning treatment on the pre-trained model was proven to increase accuracy compared to without fine tuning, the pre-trained model VGG-19 was able to increase accuracy higher than RestNet50, RestNet34, and VGG 16 in the study [6] with an increase from 0.9609 to 0.9720.

In this study, we will utilize transfer learning for models that have been trained using CNN to detect malaria parasites in red blood cell images. The deep CNN pre-trained model uses 3 architectures, namely ResNet50V2, EfficientNetB0, and InceptionV3. For each architecture, experiments will be carried out and compare which architecture is better in detecting malaria parasites. The types of optimizers and tuning parameters will also be tested to obtain optimal parameters in malaria detection using transfer learning.

2 TECHNICAL OVERVIEW

2.1 CNN

Convolutional Neural Networks (CNN) can be defined as a deep learning method that mimics how the brain works in recognizing an image [7]. In CNN, there are two main processes, feature learning and classification. Feature learning consists of convolutional and pooling layers which perform feature extraction from the input image. Whereas in classification, it has a fully-connected block that performs feature extraction for the final output. In a digital image, the input in the form of pixels will be stored into a two-dimensional array consisting of an array of numbers and an array of kernel values. Feature extraction is performed at every pixel in the image which makes image processing very efficient. The result of feature learning is a feature map which will then be converted into a vector using flatten. Each vector will be mapped by fully connected to calculate the probability of each class. The output layer will produce the expected number of class probability nodes.The diagram of CNN is shown in the Figure 1.

CNN has performed in various fields in computer vision. Radiological tasks such as x-ray images can be solved with CNN such as Yadav's study [8] conducted image classification to detect pneumonia using the VGG16, InceptionV3, and Capsule Neural Network architectures resulting in accuracy of 0.923, 0.869, and 0.824. The architecture of VGG166 is not too complex so that it is overfitting compared to InceptionV3 which is experiencing overfitting. Object detection using CNN for mask detection results in up to 99% accuracy on the MobileNetV2 Architecture with Adam Optimizer [1]. Combining LSTM with Architectural Inception CNN can perform American Sign Language Recognition with performance accuracy approaching 90% [9]. A similar study using CNN-LSTM to detect natural frequencies of different beams in modal frequency detection tax was able to produce the highest accuracy of 96.6% on aluminum-long [10].

2.2 RestNet

The RestNet architecture was developed by Microsoft with the aim of solving the vanishing gradient problem in the training process. RestNet developed a concept called ResidualBlock. The second version of Resiudal Neural Network (RestNet) 50 layer has better performance than the previous version [11]. This model has been trained using the ImageNet dataset which consists of 1000 different classes. Summary for pre-trained model RestNet50V2 is shown in Figure 2.

RestNet50V2 architecture is divided into five convolution blocks and one classifier block. This architecture adds 3 Residual Blocks from RestNet34V2. RestNet50V2 extends the previous version by adding a pre-activation variant of the residual block. This pre-activation is able to make the gradient run without a hitch via a shortcut connection.

RestNet50 was used as a pre-trained model by Reddy and Juliet [5] in 2019 to solve the problem of malaria detection with an accuracy of 95.91%. By utilizing RestNet50 as a pre-trained model, it is able to achieve an accuracy of 84.1% in Brain Tumor Detection [12]

2.3 EfficientNet

EfficientNet architecture is a CNN architecture developed by Google's Brain Team to increase efficiency in scaling and improve performance accuracy [13]. Scaling is done on the channel width of the layer, the height of the number of layers, and the resolution of the digital image.EfficientNet automates scaling up using AutoML from the MobileNet model architecture. EfficientNet has 8 different models with different number of parameters named sequentially

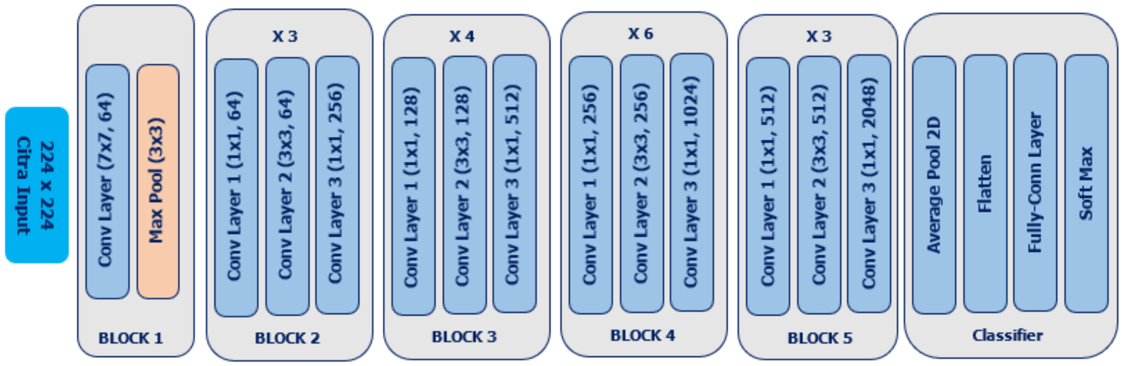

Figure 2: RestNet50V2 architecture

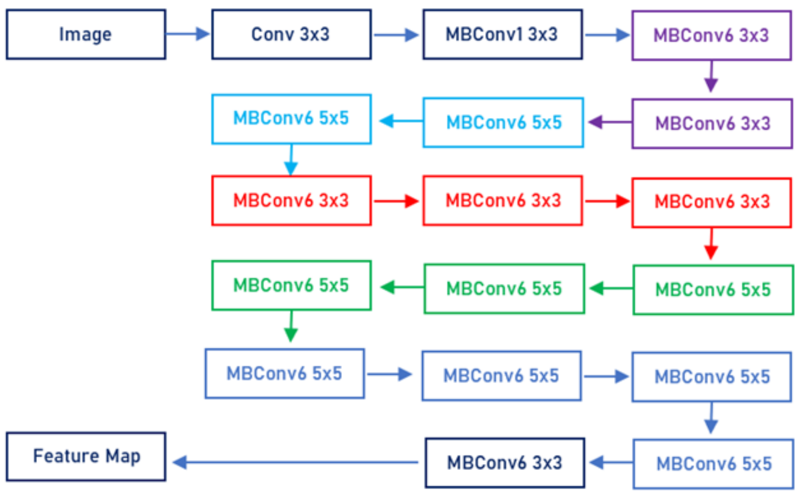

Figure 3: EfficientNetB0 architecture

from B0 to B7 [14]. EfficientNet architecture requires faster computation time by applying the concept of inverted residual block [15]. The EfficientNetB0 architecture is shown in Figure 3.
Some studies use transfer learning with pretrained model efficientnetb0. A study conducted by Pangkasidhi [16] carried out transfer learning using the EfficientNet model versions B0. B1, and B2 resulting in an accuracy of 92.54%, 93.19%, and 92.03%. In study [17], the EfficientNetB0 architecture performed transfer learning and hyperparameter tuning on Dense, Batch Size, and Learning Rate resulting in the highest accuracy of 95.17% on the ADAM optimizer. Transfer Learning using EfficientNetB0 produces 98.33% accuracy and MobileNetV3 provides 98.75% accuracy in detecting puppets [18].

2.4 InceptionNet

Google Researcher in 2014 introduced the GoogleLeNet architecture or known as Inception to complete the ImageNet Recognition Challenge. InceptionV1 has 27 deep layers and there is an inception layer or Inception module, this layer does many convolutional layers and pooling layers in parallel. The development of the Inception architecture has reached the third version which is better than the previous version [19]. The third version of Inception adds batch normalization to the Inception layer. When the InceptionV3 architecture is transferred using training data, ImageNet can be used as a pre-trained model in transfer learning.The summary of pre-trained model InceptionV3 is shown in Figure 4.
The study by Sriporn et al. [20] using InceptionV3 as one of the pre-trained models to analyze malaria was able to produce an accuracy of 95%. Wang et al. [21] using InceptionV3 as a pre-trained model for pulmonary image classification obtained an accuracy of 95.41%. Based on two studies that have been carried out, InceptionV3 has high accuracy but is not the best in solving image classification problems.

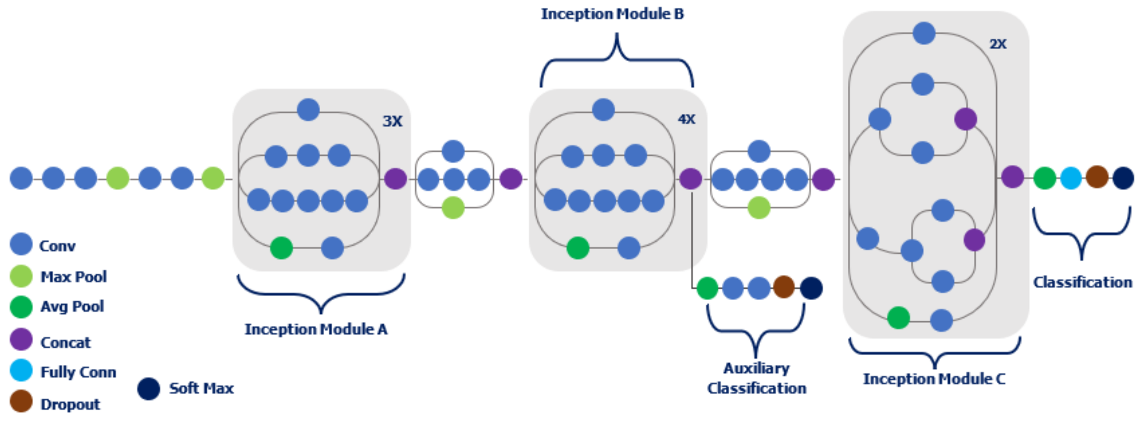

Figure 4: InceptionV3 architecture

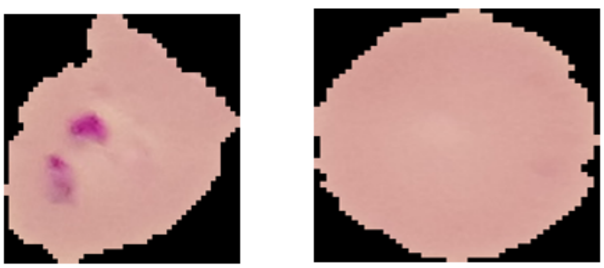

Figure 5: Images of Parasitized cell (left) and Uninfected cell (right)

3 PROPOSED WORK

3.1 Data collection

The dataset obtained comes from the Lister Hill National Center for Biomedical Communication and was first introduced by Rajaraman et al,. [22]. The dataset is a collection of Blood Smear images which have two classes, uninfected and parasitized. The total number of datasets is 27,558, which is divided into two classes with the same number of different image sizes. Different image sizes will be resized according to the needs of the required CNN architecture. The image in the parasitized class has a pale bluish purple with black spots from infected blood, while in the uninfected class it does not have a purple color. The dataset is divided into 3 parts, ie. data train, validation, and test data. Ratio between data train, validation, and data test is 70%:15%:15%, respectively. The sample images of two classes can be seen in Figure 5.

3.2 Model Training

In this study, 15 experiments were carried out to get the best results. Experiments were carried out on each model consisting of 5 experiments to explore transfer learning performance. The first experiment was carried out without retraining the model. The second experiment conducted an experiment for the input image and the learning-rate used. The third experiment reduces the value of the learning rate and fine-tune the model. Experiments on the four models were fine-tuned by increasing the learning-rate and epoch. The fifth experiment was the same as the fourth experiment but did not augment the data.

3.3 Evaluation

Model from each experiment will be evaluated on test dataset using F1-Score and Confusion Matrix (CM). CM is a standard tool to evaluate model performance on test data (data that has never been seen before). CM has corresponding rows and columns representing ground truth class and predicted class. CM consists of True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). TP is positive class that predicted as positive, FP is non-positive class that predicted as positive, TN is negative class that predicted as negative, and FN is non-negative class that predicted as negative. F1-Score is weighted average of precision and recall

$$\mathrm{Pr}\,ecision = \frac{TP}{TP+FP} \tag{1}$$

$$\mathrm{Recall} = \frac{TP}{TP+FN} \tag{2}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{3}$$

$$F1-Score = \frac{2TP}{2TP+FP+FN} \tag{4}$$

4 RESULTS AND DISCUSSION

The result of experiments is split into two parts. First is the result of model training and second is the result of prediction on test dataset.

4.1 Training Result

The results of accuracy and loss using training data and validation data can be seen in table 2.

Based on the experiment in table 2, in term of accuracy the overall model does not have a significant difference. We can conclude that the choice of architecture for training has little effect on increasing accuracy.

Table 1: Experiments and hyper-parameters for training

No	Experiments	Optimizer	Augmentation	Citra Input	Learning Rate	Epochs	Fine Tune
1	RestNet50V2_01	ADAM	Yes	64 x 64	0.001	10	No
2	RestNet50V2_02	ADAM	Yes	96 x 96	0.0001	20	No
3	RestNet50V2_03	ADAM	Yes	96 x 96	0.001	20	Yes
4	RestNet50V2_04	ADAM	Yes	96 x 96	0.0001	30	Yes
5	RestNet50V2_05	ADAM	No	96 x 96	0.0001	30	Yes
6	EfficientNetB0_01	ADAM	Yes	64 x 64	0.001	10	No
7	EfficientNetB0_02	ADAM	Yes	96 x 96	0.0001	20	No
8	EfficientNetB0_03	ADAM	Yes	96 x 96	0.001	20	Yes
9	EfficientNetB0_04	ADAM	No	96 x 96	0.0001	30	Yes
10	EfficientNetB0_05	ADAM	Yes	96 x 96	0.0001	30	Yes
11	InceptionV3_01	ADAM	Yes	64 x 64	0.001	10	No
12	InceptionV3_02	ADAM	Yes	96 x 96	0.0001	20	No
13	InceptionV3_03	ADAM	Yes	96 x 96	0.001	20	Yes
14	InceptionV3_04	ADAM	No	96 x 96	0.0001	30	Yes
15	InceptionV3_05	ADAM	Yes	96 x 96	0.0001	30	Yes

Table 2: Training Result

No	Experiments	Train loss	Train acc	Val loss	Val acc
1	RestNet50V2_01	0.3571	0.8543	0.3029	0.8765
2	RestNet50V2_02	0.2677	0.8915	0.2661	0.8826
3	RestNet50V2_03	0.2300	0.9400	0.1689	0.9599
4	RestNet50V2_04	0.1117	0.9965	0.3351	0.9565
5	RestNet50V2_05	0.2119	0.9637	0.1891	0.9710
6	EfficientNetB0_01	0.2814	0.8841	0.4106	0.8173
7	EfficientNetB0_02	0.2577	0.8987	0.2552	0.9009
8	EfficientNetB0_03	0.1345	0.9663	0.1125	0.9731
9	EfficientNetB0_04	0.1254	0.9982	0.3076	0.9681
10	EfficientNetB0_05	0.1800	0.9806	0.2442	0.9721
11	InceptionV3_01	0.3630	0.8510	0.3183	0.8636
12	InceptionV3_02	0.3291	0.8582	0.3058	0.8670
13	InceptionV3_03	0.2283	0.9378	0.2661	0.9614
14	InceptionV3_04	0.1350	0.9965	0.2859	0.9628
15	InceptionV3_05	0.2119	0.9637	0.1891	0.9710

4.2 Prediction Result

The results of accuracy and f1-score using training data and validation data can be seen in table 3.
Based on the result in table 3, the models RestNetV2_05 and EfficientNetB0_03 have the best performance with prediction accuracy of 0.975 and F1-score of 0.97.

4.3 Discussion

The first experiment was conducted to test the performance of the initial transfer learning model. The result of the first experiment from the training model for train accuracy is 0.854, 0.88, and 0.851 (ResNet50V2_01, EfficientNetB0_01, and InceptionV3_01) and val accuracy is 0.876, 0.817, and 0.83. The second experiment was conducted to improve the performance of the model from the previous experiment. Added epochs, reduced learning rate values, and added size to the input image. The results of the second experiment showed an increase in train accuracy and validation accuracy for the models ResNet50V2_02, EfficientNetB0_02, and InceptionV3_02.

In the third experiment, epochs were added, fine-tuned and the learning rate optimizer was set to default. The results of the third experiment showed an increase in train accuracy and val accuracy, although there had been quite high fluctuations in the val loss of ResNet50V2_03 and InceptionV3_03. The fluctuation range is 0-50 for ResNet50V2_03 before the 10th epoch and InceptionV3_03 is 0-200 before the 10th epoch.

The fourth experiment was the same as the previous experiment but was carried out without augmentation and learning rate reduction. The results of the fourth experiment succeeded in resolving fluctuations in val loss, but there was no increase in val accuracy and the overall model experienced a little overfitting. In the fifth experiment the same as the previous experiment but added data augmentation. The results of the fifth experiment, the ResNet50V2_05

Table 3: Training Result

No	Experiments	Parasitized Accuracy	Uninfected Accuracy	Prediction Accuracy	F1-score
1	RestNet50V2_01	0.81	0.95	0.88	0.875
2	RestNet50V2_02	0.79	0.98	0.88	0.88
3	RestNet50V2_03	0.94	0.98	0.96	0.96
4	RestNet50V2_04	0.94	0.98	0.96	0.96
5	RestNet50V2_05	0.96	0.99	**0.975**	**0.97**
6	EfficientNetB0_01	0.67	0.98	0.83	0.82
7	EfficientNetB0_02	0.89	0.91	0.90	0.90
8	EfficientNetB0_03	0.96	0.99	**0.975**	**0.97**
9	EfficientNetB0_04	0.97	0.96	0.97	**0.97**
10	EfficientNetB0_05	0.96	0.98	0.97	**0.97**
11	InceptionV3_01	0.77	0.96	0.86	0.865
12	InceptionV3_02	0.77	0.96	0.87	0.865
13	InceptionV3_03	0.97	0.96	0.96	0.96
14	InceptionV3_04	0.96	0.96	0.97	**0.97**
15	InceptionV3_05	0.97	0.98	0.97	**0.97**

and EfficientNetB0_05 models have relatively high training accuracy and val accuracy results and have a close gap, but for the InceptionV3_05 model there is still a little overfitting.

5 CONCLUSION

Based on experiments conducted without fine tune, the accuracy ranges from 0.76 − 0.81 for ResNet50v2, 0.76 − 0.80 for EfficientNetB0, and 0.77 − 0.82 for InceptionV3. Meanwhile, in the experiment by fine-tuning the Resnet50V2_03 and InceptionV3_03 models, fluctuations occurred due to the possibility that the parameters used were not optimal. From all experiments conducted, the model with the best metrics is EfficientnetB0_03 and ResNet50v2_05 based on prediction accuracy and F1-score.

REFERENCES

[1] Rashmi Nayak S and Manohar N. 2021. Computer-Vision based Face Mask Detection using CNN. In *2021 6th International Conference on Communication and Electronics Systems (ICCES)*, IEEE, Tamil Nadu, 1780–1786. DOI:https://doi.org/10.1109/ICCES51350.2021.9489098

[2] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. 2018. A Survey on Deep Transfer Learning. In *The 27th International Conference on Artificial Neural Networks*, 270–279. DOI:https://doi.org/10.1007/978-3-030-01424-7_27

[3] Kevser Sahinbas and Ferhat Ozgur Catak. 2021. Transfer learning-based convolutional neural network for COVID-19 detection with X-ray images. In *Data Science for COVID-19*. Elsevier, 451–466. DOI:https://doi.org/10.1016/B978-0-12-824536-1.00003-4

[4] Aimon Rahman, Hasib Zunair, M Sohel Rahman, Jesia Quader Yuki, Sabyasachi Biswas, Md Ashraful Alam, Nabila Binte Alam, and M. R. C. Mahdy. 2019. Improving Malaria Parasite Detection from Red Blood Cell using Deep Convolutional Neural Networks. (July 2019).

[5] A. Sai Bharadwaj Reddy and D. Sujitha Juliet. 2019. Transfer Learning with ResNet-50 for Malaria Cell-Image Classification. In *2019 International Conference on Communication and Signal Processing (ICCSP)*, IEEE, 0945–0949. DOI:https://doi.org/10.1109/ICCSP.2019.8697909

[6] Tayyaba Jameela, Kavitha Athotha, Ninni Singh, Vinit Kumar Gunjan, and Sayan Kahali. 2022. Deep Learning and Transfer Learning for Malaria Detection. *Comput Intell Neurosci* 2022, (2022), 2221728. DOI:https://doi.org/10.1155/2022/2221728

[7] P Kim. 2017. Convolutional neural networks. In *Mathlab Deep Learning*. CA: Apress, Berkeley, 121–147.

[8] Samir S. Yadav and Shivajirao M. Jadhav. 2019. Deep convolutional neural network based medical image classification for disease diagnosis. *J Big Data* 6, 1 (December 2019), 113. DOI:https://doi.org/10.1186/s40537-019-0276-2

[9] Kshitij Bantupalli and Ying Xie. 2018. American Sign Language Recognition using Deep Learning and Computer Vision. In *2018 IEEE International Conference on Big Data (Big Data)*, IEEE, 4896–4899. DOI:https://doi.org/10.1109/BigData.2018.8622141

[10] Ruoyu Yang, Shubhendu Kumar Singh, Mostafa Tavakkoli, Nikta Amiri, Yongchao Yang, M. Amin Karami, and Rahul Rai. 2020. CNN-LSTM deep learning architecture for computer vision-based modal frequency detection. *Mech Syst Signal Process* 144, (October 2020), 106885. DOI:https://doi.org/10.1016/j.ymssp.2020.106885

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 770–778. DOI:https://doi.org/10.1109/CVPR.2016.90

[12] Ahsan Aziz, Muhammad Attique, Usman Tariq, Yunyoung Nam, Muhammad Nazir, Chang-Won Jeong, Reham R. Mostafa, and Rasha H. Sakr. 2021. An Ensemble of Optimal Deep Learning Features for Brain Tumor Classification. *Computers, Materials & Continua* 69, 2 (2021), 2653–2670. DOI:https://doi.org/10.32604/cmc.2021.018606

[13] Mingxing Tan and Quoc v. Le. 2019. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *International Conference on Machine Learning* (May 2019).

[14] Rashidul Hasan Hridoy, Fatema Akter, and Aniruddha Rakshit. 2021. Computer Vision Based Skin Disorder Recognition using EfficientNet: A Transfer Learning Approach. In *2021 International Conference on Information Technology (ICIT)*, IEEE, 482–487. DOI:https://doi.org/10.1109/ICIT52682.2021.9491776

[15] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, 4510–4520. DOI:https://doi.org/10.1109/CVPR.2018.00474

[16] Ananta Kusuma Pangkasidhi, Henry Novianus Palit, and Alvin Nathaniel Tjondrowiguno. 2021. Aplikasi Pendukung Diagnosis COVID-19 yang Menganalisis Hasil X-Ray Paru-paru dengan Model EfficientNet . *INFRA* 9, 2 (2021)

[17] Gusrifaris Yuda Alhafis, Jasril Jasril, Suwanto Sanjaya, Fadhilah Syafria, and Elvia Budianita. 2022. Klasifikasi Citra Daging Sapi dan Daging Babi Menggunakan Ekstraksi Ciri dan Convolutional Neural Network. *JURIKOM: Jurnal Riset Komputer* 9, 3 (2022)

[18] Ahmad Mustafid, Muhammad Murah Pamuji, and Siti Helmiyah. 2020. A Comparative Study of Transfer Learning and Fine-Tuning Method on Deep Learning Models for Wayang Dataset Classification. *IJID (International Journal on Informatics for Development)* 9, 2 (December 2020), 100–110. DOI:https://doi.org/10.14421/ijid.2020.09207

[19] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the Inception Architecture for Computer Vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2818–2826. DOI:https://doi.org/10.1109/CVPR.2016.308

[20] Krit Sriporn, Cheng-Fa Tsai, Chia-En Tsai, and Paohsi Wang. 2020. Analyzing Malaria Disease Using Effective Deep Learning Approach. *Diagnostics* 10, 10 (September 2020), 744. DOI:https://doi.org/10.3390/diagnostics10100744

[21] Cheng Wang, Delei Chen, Lin Hao, Xuebo Liu, Yu Zeng, Jianwei Chen, and Guokai Zhang. 2019. Pulmonary Image Classification Based on Inception-v3 Transfer Learning Model. *IEEE Access* 7, (2019), 146533–146541. DOI:https://doi.org/10.1109/ACCESS.2019.2946000

[22] Sivaramakrishnan Rajaraman, Sameer K. Antani, Mahdieh Poostchi, Kamolrat Silamut, Md. A. Hossain, Richard J. Maude, Stefan Jaeger, and George R. Thoma. 2018. Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images. *PeerJ* 6, (April 2018), e4568. DOI:https://doi.org/10.7717/peerj.4568