Recursive Journal of Informatics
Vol. 2, No. 1, March 2024
p-ISSN 2963-5551 https://journal.unnes.ac.id/journals/index.php/rji e-ISSN 2986-6588

Optimization of the Convolutional Neural Network Method Using Fine-Tuning for Image Classification of Eye Disease

Vivi Wulandari[1], Anggyi Trisnawan Putra[2]
[1,2]Computer Science Department, Faculty of Mathematics and Natural Sciences,
Universitas Negeri Semarang, Indonesia

Abstract. The eye is the most important organ of the human body which functions as the sense of sight. Most people wish they had healthy eyes so they could see clearly about life around them. However, some people experience eye health problems. There are many types of eye diseases ranging from mild to severe. With advances in technology, artificial intelligence can be used to classify eye diseases accurately, one of which is deep learning. Therefore, this study uses the Convolutional Neural Network (CNN) algorithm to classify eye diseases using the VGG16 architecture as a base model and will be combined using a fine-tuning model as an optimization to improve accuracy.
Purpose: To find out the accuracy results obtained in the fine-tuning optimization model on Convolutional Neural Network (CNN) method in classifying images in eye disease.
Methods/Study design/approach: Combining the Convolutional Neural Network (CNN) method with fine-tuning optimization models for image classification in eye disease. The two methods will be compared to determine the best result.
Result/Findings: The accuracy results obtained from testing the Convolutional Neural Network method with the VGG16 architecture were 82.63% while the accuracy results from testing the fine-tuning model were 94.13%.
Novelty/Originality/Value: The test results on the fine-tuning model have better accuracy than the testing of the Convolutional Neural Network method. This can be seen in the fine-tuning model which has an increase in accuracy of 11.5%.

Keywords: Eye Disease, Convolutional Neural Network, Fine-Tuning, VGG16, Deep Learning
Received August 22, 2023 / **Revised** November 29, 2024 / **Accepted** March 22, 2024

INTRODUCTION

The five senses that are very important and sensitive are the eyes. The eye is a very important organ in human life where most of the visual information can be absorbed and used in various forms of activity because of the role of the eye. The part of the eye, one of which is the cornea, which is at the front of the eye which is transparent and convex in shape, is the fact that the eye is not a perfect ball, but two main parts that are fused into one [1]. The cornea is directly connected to the larger part of the eye, the sclera, the white part of the eye. These two parts are connected by a circle of tissue called the limbus.

Along with the many uses of the eye in humans, many diseases attack the sense of sight. If the eye is disturbed, of course it hinders human activity. Some examples are cataract, glaucoma, and diabetic retinopathy. The disease is most often one of the causes of blindness in the eye. Cataract is an eye condition that occurs due to changes in the clear and transparent lens, so it becomes dark. Cataract blindness occurs when the cataract is so cloudy that the lens cannot emit any light at all [2]. Although cataracts can be removed, however, many countries still have barriers for patients to perform surgery. As a result, cataracts are the leading cause of blindness in the world. Not only cataracts, but glaucoma has also increased rapidly over the last few decades. In 2010, as many as 60.5 million individuals became glaucoma sufferers. In most cases, glaucoma is the second leading cause, after cataract. Glaucoma generally has no clear symptoms. Globally, glaucoma sufferers are estimated to reach 76 million in 2020 and 111.8 million in 2040. This disease can impair visual abilities and can even lead to blindness [3].

[1]*Corresponding author.
Email addresses: vivicomel17@students.unnes.ac.id (Wulandari)
DOI: 10.15294/rji.v2i1.73625

Diabetic Retinopathy (DR) is a disease of the retinal micro blood vessels that grows gradually which can threaten eye health and is associated with prolonged hyperglycemia and several other conditions associated with diabetes mellitus (DM) such as hypertension. Diabetic Retinopathy is an eye disorder that often occurs in almost all patients with long-standing diabetes mellitus [4]. In cases where the eye is endangered, intervention should be carried out via laser photocoagulation, injection of intravitreal drugs or vitreoretinal surgery [5].

According to a report by The International Agency for the Prevention of Blindness [6], around 1.1 billion people worldwide experience vision loss, with 90 million of them being children and adolescents, of which 2.1 million people are blind. This can be prevented if it is detected sooner and gets early treatment for the symptoms that appear. Detection of eye disease is usually done by manual analysis by a specialist on fundus images. Because of this, the results of the diagnosis between one doctor and another doctor may differ [7]. There are many factors that influence the success or failure of eye disease treatment in the community, including patient awareness to consult a doctor, as well as the doctor's ability to diagnose and treat eye disease quickly [8]. Along with the rapid development of technology. The development of this technology is characterized by the emergence of new systems to assist humans in classifying digital images, including in the medical field. Utilization of this technology is by utilizing artificial intelligence. One of them is deep learning which is part of machine learning which can handle large and increasing cases of data. By using machine learning, the process of classifying diseases in the patient's retina can be carried out in a shorter time, thereby speeding up the patient treatment process. Some studies that carry out classification with deep learning generally use image data more [9].

The application of the field of deep learning with CNN is one of the most popular classification methods. The network process starts from input to output which runs in one direction and an architecture that can mimic the function of the human brain is an advantage [10]. In the case of image classification, images are like food for CNN, the more images that are trained, the maturity of the model will be achieved to solve problems with optimal performance. CNN is used to solve digital classification problems with a high amount of data because it can work to achieve optimal performance, but not for a low amount of data. The CNN model works by trying to imitate an image recognition system in the human visual cortex [11]. CNN is usually used on image data to detect and recognize objects in an image.

Transfer learning is a method in which a model that has been trained by a previous dataset is reused on a model with a different dataset. Therefore, the transfer learning method is suitable for models with small datasets [12]. Classification for low data usually uses a transfer learning approach from pre-trained CNN architectures that are already available, such as ResNet introduced the concept of feature residues [13]. VGG as an improvement from AlexNet [14]. MobileNet with claims to provide a simpler architecture but reliable in performance [15]. However, the use of transfer learning is not all suitable for use in problems. Therefore, this study applies a fine-tuning optimization model. Fine-tuning is a transfer learning technique by applying retraining to the end of the convolution block and letting the other convolution layers not be retrained. Retraining on the final layer aims to keep the weight value on the layer unchanged.

Several studies that have been carried out using artificial intelligence include the use of CNN to detect eye disease in fundus images using the transfer learning method to obtain an accuracy of 72%. CNN optimization then uses a different object from this study by adding several hyperparameters to the CNN architecture to obtain an accuracy of 91.2%. Whereas if you don't use the hyperparameter, the accuracy is only 67.6%. There are three models to increase the accuracy value, namely dropout, padding, and stride [16]. However, from these studies there have not been many fine-tuning CNN development studies to classify eye diseases caused by various diseases with fine-tuning optimization. So that is the background for the author to conduct research that will focus on optimizing the CNN method using fine-tuning for image classification in eye disease.

METHODS

This research was conducted to obtain accurate results from the optimization of the fine-tuning model on the CNN algorithm. Several stages of this research began with literature study, data collection, dataset testing, implementation, and ended with analysis and report generation. The stages carried out in this study are presented in the research flow in Figure 1.

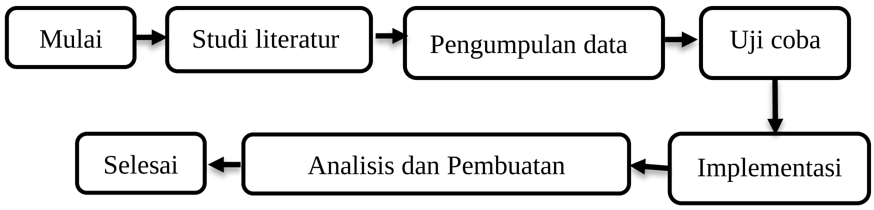


Figure 1. Research Flow

Dataset

The dataset in this study begins with conducting a literature study, then collecting a dataset from Kaggle with the name eye_diseases_classification which can be downloaded at the URL address https://www.kaggle.com/datasets/gunavenkatdoddi/eye-diseases-classification. This dataset has 4 disease classes, namely cataract with 1038 image data, glaucoma 1007 with image data, diabetic retinopathy with 1098 image data, and normal with 1074 image data. The data collected will be divided into 3 data sets, namely training data, validation data, and testing data with a ratio of 80:10:10. The results of data division are shown in Table 1.

Table 1. Results of Data Divison

Class	Training	Validation	Testing
Cataract	830	103	108
Glaucoma	805	100	102
Diabetic Retinopathy	878	109	111
Normal	859	107	105
Total		**4217**	

Data Augmentation

The dataset used will be processed using data augmentation techniques. Data augmentation is the process of modifying images and is only done on training data so that the machine can recognize different images. Some of the augmentation stages used in this study include the following.

1. Rescale the image is the value used to multiply the data before the next processing. The original image data consists of RGB coefficients at 0-255, but these values are too high for the model to process. Then using rescale=1./255 will convert pixels in the range [0.255] to the range [0.1].
2. Image rotation is rotating the image randomly through degrees between 0 - 360. This rotation will move outward and leave an empty area that needs to be filled in with the nearest pixel value. So that the maximum rotation in the study will only be used by 20 degrees.
3. Image shifts (random shifts) are performed for image data that has objects not in the center of the image. Then you can shift the image by shifting the image pixels both horizontally and vertically. Image shift in the study uses a value of 0.2.
4. Zoom of the image in this study was used at 0.1.
5. Shear is used so that the image is distorted along the axis. This aims to improve the angle of image perception so that it makes it easier for the computer to read it. However, it is better to use a value that is not too large because it can make the image become flat. For this study used a value of 0.2.
6. Flipping is used to reverse the position of the image horizontally or vertically. The flip value used is 0.2.

Method Design

The first stage imports the transfer learning model using the CNN architecture, namely VGG16. When importing pre-trained VGG16, the top model layer is not included in the base model because the main task of the top model is classification. However, the base model without the top model will be frozen to avoid changes during the training process. Next add a new classifier layer above the frozen base model layer. During training, this new classification can change the old features so that it can classify images for eye disease. This process is done by adding several fully connected layers to the CNN model. Some of the layers include flatten, dense as a hidden layer, dropout, and ends with a dense layer as the output layer.

The second stage is to combine the CNN method with the fine-tuning model. The trained model will be added to the fine-tuning optimization model to improve performance and accuracy. Several stages in fine-tuning are as follows.

1. The base model on the previous VGG16 pre-trained model will be unfreezed first.
2. The initial layer on the convolution block will be frozen with the aim that the weight value does not change.
3. The retraining process is carried out using the final layer on the convolution block.

Training Model

The training model will use the keras library by setting some hyperparameters. The hyperparameters used can be seen in Table 2.

Table 2. Hyperparameters

Hyperparameters	Value
Image Size	$224 \times 224 \times 3$
Batch Size	64
Optimization	Adam
Loss Function	Categorical Cross Entropy
Metrics	Accuracy

Model Evaluation

At this stage an evaluation is carried out from the previous stage, namely the testing stage. At this stage, the authors use the Confusion Matrix method which consists of Accuracy, Precision, and Recall for 4 classes of eye disease. Testing using the confusion matrix can be seen in Table 3.

Table 3. Confusion Matrix

Actual	Predicted	
	Positive	Negative
Positive	TP	FP
Negative	FN	TN

Accuracy is used to describe how accurate the model is in correctly classifying it which will be calculated through Equation 1. Precision is used to calculate the comparison of the level of the amount of data that is correctly predicted to the overall data predicted by the system which will be calculated through Equation 2. Recall is used to evaluate the comparison of the number of correctly predicted data with the total actual data which will be calculated through Equation 3. F1-Score is used to calculate the average of the comparison of precision and recall which will be calculated through Equation 4.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

$$\text{FI-score} = 2\,\frac{P\,.\,R}{P + R} \quad (4)$$

RESULT AND DISCUSSION

Data Augmentation Results

In this study the authors used the library from Keras to perform data augmentation. With the techniques in data augmentation, several examples of augmentation results will be obtained illustrated in Figure 2.

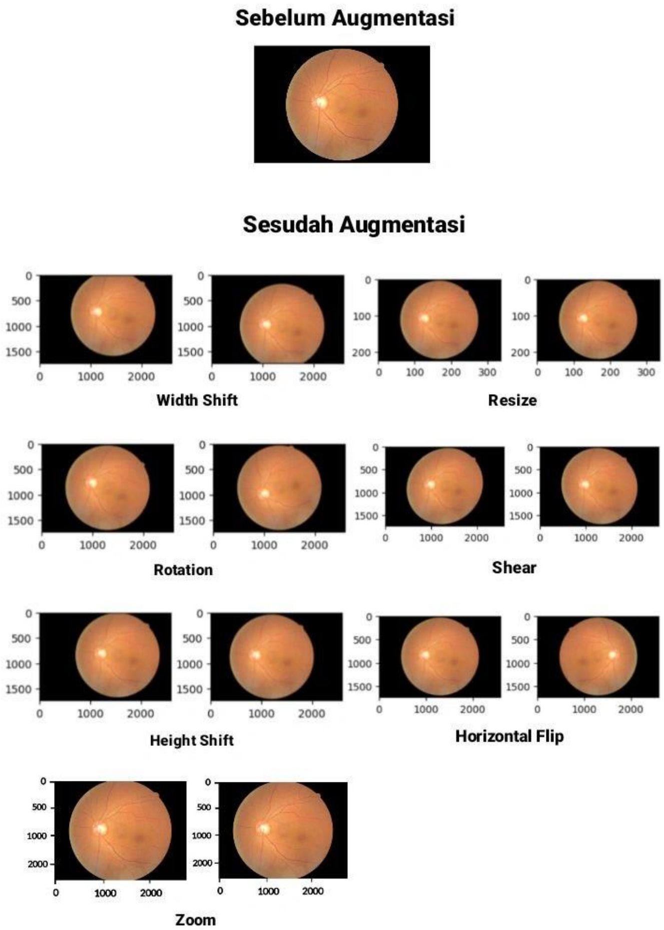


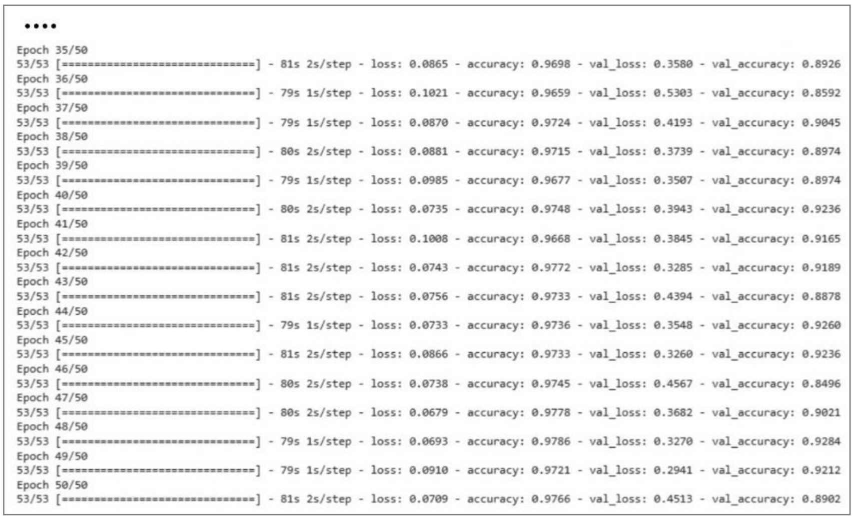
Figure 2. Data Augmentation

After data augmentation, data training will be carried out. In this study, 2 scenarios were carried out, namely testing the CNN without fine-tuning and using fine-tuning. Following are the results of the training process from the CNN method with the VGG16 architecture using 50 epochs with 53 steps per epoch, which can be seen in Figure 3. Meanwhile, the results of the training process using the fine-tuning model can be seen in Figure 4.

```
....

Epoch 35/50
53/53 [==============================] - 78s 1s/step - loss: 0.4208 - accuracy: 0.8301 - val_loss: 0.4578 - val_accuracy: 0.8329
Epoch 36/50
53/53 [==============================] - 86s 2s/step - loss: 0.4398 - accuracy: 0.8307 - val_loss: 0.4206 - val_accuracy: 0.8473
Epoch 37/50
53/53 [==============================] - 78s 1s/step - loss: 0.4150 - accuracy: 0.8336 - val_loss: 0.4415 - val_accuracy: 0.8282
Epoch 38/50
53/53 [==============================] - 78s 1s/step - loss: 0.4231 - accuracy: 0.8310 - val_loss: 0.5544 - val_accuracy: 0.7876
Epoch 39/50
53/53 [==============================] - 78s 1s/step - loss: 0.4166 - accuracy: 0.8333 - val_loss: 0.4880 - val_accuracy: 0.8234
Epoch 40/50
53/53 [==============================] - 80s 2s/step - loss: 0.4090 - accuracy: 0.8360 - val_loss: 0.4802 - val_accuracy: 0.8186
Epoch 41/50
53/53 [==============================] - 77s 1s/step - loss: 0.4083 - accuracy: 0.8357 - val_loss: 0.4728 - val_accuracy: 0.8329
Epoch 42/50
53/53 [==============================] - 79s 1s/step - loss: 0.4163 - accuracy: 0.8375 - val_loss: 0.4874 - val_accuracy: 0.8258
Epoch 43/50
53/53 [==============================] - 78s 1s/step - loss: 0.4188 - accuracy: 0.8363 - val_loss: 0.4579 - val_accuracy: 0.8258
Epoch 44/50
53/53 [==============================] - 79s 1s/step - loss: 0.4200 - accuracy: 0.8321 - val_loss: 0.4309 - val_accuracy: 0.8353
Epoch 45/50
53/53 [==============================] - 76s 1s/step - loss: 0.4014 - accuracy: 0.8384 - val_loss: 0.5779 - val_accuracy: 0.7685
Epoch 46/50
53/53 [==============================] - 78s 1s/step - loss: 0.4237 - accuracy: 0.8298 - val_loss: 0.3645 - val_accuracy: 0.8663
Epoch 47/50
53/53 [==============================] - 77s 1s/step - loss: 0.4291 - accuracy: 0.8265 - val_loss: 0.4597 - val_accuracy: 0.8305
Epoch 48/50
53/53 [==============================] - 78s 1s/step - loss: 0.4040 - accuracy: 0.8396 - val_loss: 0.4586 - val_accuracy: 0.8234
Epoch 49/50
53/53 [==============================] - 76s 1s/step - loss: 0.3916 - accuracy: 0.8455 - val_loss: 0.4316 - val_accuracy: 0.8377
Epoch 50/50
53/53 [==============================] - 78s 1s/step - loss: 0.4009 - accuracy: 0.8372 - val_loss: 0.5410 - val_accuracy: 0.7852
```

Figure 3. Training Using CNN VGG16



```
....
Epoch 35/50
53/53 [==============================] - 81s 2s/step - loss: 0.0865 - accuracy: 0.9698 - val_loss: 0.3580 - val_accuracy: 0.8926
Epoch 36/50
53/53 [==============================] - 79s 1s/step - loss: 0.1021 - accuracy: 0.9659 - val_loss: 0.5303 - val_accuracy: 0.8592
Epoch 37/50
53/53 [==============================] - 79s 1s/step - loss: 0.0870 - accuracy: 0.9724 - val_loss: 0.4193 - val_accuracy: 0.9045
Epoch 38/50
53/53 [==============================] - 80s 2s/step - loss: 0.0881 - accuracy: 0.9715 - val_loss: 0.3739 - val_accuracy: 0.8974
Epoch 39/50
53/53 [==============================] - 79s 1s/step - loss: 0.0985 - accuracy: 0.9677 - val_loss: 0.3507 - val_accuracy: 0.8974
Epoch 40/50
53/53 [==============================] - 80s 2s/step - loss: 0.0735 - accuracy: 0.9748 - val_loss: 0.3943 - val_accuracy: 0.9236
Epoch 41/50
53/53 [==============================] - 81s 2s/step - loss: 0.1008 - accuracy: 0.9668 - val_loss: 0.3845 - val_accuracy: 0.9165
Epoch 42/50
53/53 [==============================] - 81s 2s/step - loss: 0.0743 - accuracy: 0.9772 - val_loss: 0.3285 - val_accuracy: 0.9189
Epoch 43/50
53/53 [==============================] - 81s 2s/step - loss: 0.0756 - accuracy: 0.9733 - val_loss: 0.4394 - val_accuracy: 0.8878
Epoch 44/50
53/53 [==============================] - 79s 1s/step - loss: 0.0733 - accuracy: 0.9736 - val_loss: 0.3548 - val_accuracy: 0.9260
Epoch 45/50
53/53 [==============================] - 81s 2s/step - loss: 0.0866 - accuracy: 0.9733 - val_loss: 0.3260 - val_accuracy: 0.9236
Epoch 46/50
53/53 [==============================] - 80s 2s/step - loss: 0.0738 - accuracy: 0.9745 - val_loss: 0.4567 - val_accuracy: 0.8496
Epoch 47/50
53/53 [==============================] - 80s 2s/step - loss: 0.0679 - accuracy: 0.9778 - val_loss: 0.3682 - val_accuracy: 0.9021
Epoch 48/50
53/53 [==============================] - 79s 1s/step - loss: 0.0693 - accuracy: 0.9786 - val_loss: 0.3270 - val_accuracy: 0.9284
Epoch 49/50
53/53 [==============================] - 79s 1s/step - loss: 0.0910 - accuracy: 0.9721 - val_loss: 0.2941 - val_accuracy: 0.9212
Epoch 50/50
53/53 [==============================] - 81s 2s/step - loss: 0.0709 - accuracy: 0.9766 - val_loss: 0.4513 - val_accuracy: 0.8902
```

Figure 4. Training Using Fine-Tuning

From the training results above, the best model will be taken, and testing will be carried out using data testing to test the accuracy of the model. Comparison of the results of testing the two models can be seen in Table 4.

Table 4. Accuracy Comparison

No	Method	Accuracy
1	CNN dengan VGG16	82,63
2	Fine-Tuning	94,13

The accuracy obtained by the testing process will produce a confusion matrix. The accuracy obtained in the testing process will produce a confusion matrix. The results of the CNN confusion matrix model with the VGG16 architecture can be seen in Figure 5. Meanwhile, the fine-tuning confusion matrix model can be seen in Figure 6.

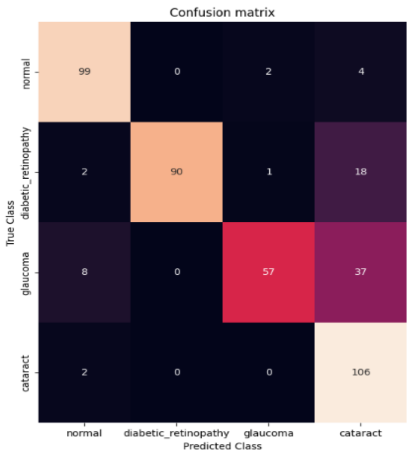


Figure 5. Confusion Matrix CNN VGG16

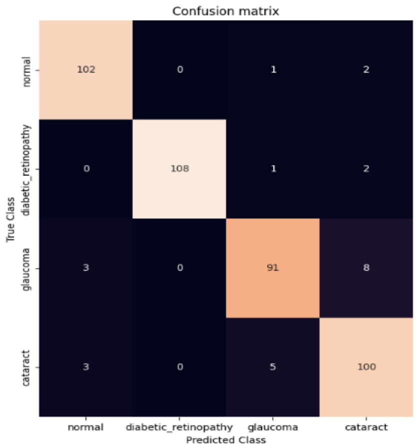


Figure 6. Confusion Matrix Fine-Tuning

Test analysis using the confusion matrix was carried out by manual calculation in each class of eye disease which is the result of the CNN method with the VGG16 architecture can be seen in Table 5.

Table 5. Calculation of Confusion Matrix with CNN VGG16

Class	Accuracy	Precision	Recall	F1-Score
Normal	95%	89%	94%	92%
Diabetic Retinopathy	95%	100%	81%	90%
Glaucoma	89%	95%	56%	70%
Cataract	86%	64%	98%	78%

Furthermore, the results of the confusion matrix analysis with manual calculations in each class of eye disease using the fine-tuning model can be seen in Table 6.

Table 6. Calculation of Confusion Matrix with Fine-Tuning

Class	Accuracy	Precision	Recall	F1-Score
Normal	98%	94%	97%	96%
Diabetic Retinopathy	99%	100%	97%	99%
Glaucoma	96%	93%	89%	91%
Cataract	95%	89%	93%	91%

In manual calculations in the analysis of the confusion matrix obtained accuracy, precision, recall, and f1-score with different results. The first is that the accuracy resulting from manual calculations for each disease class is quite high. Furthermore, precision and recall are different things. Where in this study it can be likened to that precision is when the model predicts a person has eye disease when in fact he is not affected by eye disease. Meanwhile, recall is when the model predicts that a person will not get eye disease when in fact he has eye disease. When the recall is very high, the precision will be very low and vice versa. Furthermore, the f1-score is used to align precision and recall values. In conclusion, if the f1-score produces a good value then the model used has good precision and recall.

CONCLUSION

Using the fine-tuning model produces better results than using the CNN method with the pre-trained VGG16 model. This is because the CNN method with the pre-trained VGG16 model has a weakness, namely the neural network weights used during training are not necessarily suitable for all problems so that they can produce low accuracy.

REFERENCES

[1] D. Dameria, G. Andayani, K. Rahman, and S. Soedarman, *Pedoman nasional pelayanan kedokteran retinopati diabetika*. 2018. [Online]. Available: https://perdami.or.id/wp-content/uploads/2022/03/Panduan-Nasional-Pelayanan-Kedokteran-Retinopati-Diabetik.pdf

[2] J. . Hutauruk and S. R. Siregar, *Katarak: 101 jawaban atas pertanyaan anda*, 2nd ed. PT.Gramedia Pustaka Utama, 2017.

[3] I. Sidarta and S. R. Yulianti, *Ilmu penyakit mata*, 5th ed. Jakarta: Fakultas Kedokteran Universitas Indonesia, 2011.

[4] F. Ghanchi, “Diabetic retinopathy guidelines,” *R. Coll. Ophthalmol.*, no. December, p. 147, 2012, [Online]. Available: www.rcophth.ac.uk

[5] A. A. of Ophthalmology, “Retina and vitreous,” *Basic Clin. Sci. Course*, vol. 12, pp. 1–395, 2014.

[6] IAPB, “The international agency for the prevention of blindness (IAPB),” *Vision Atlas*, 2020. https://www.iapb.org/learn/vision-atlas/magnitude-and-projections/

[7] R. Indraswari, W. Herulambang, and Others, “Deteksi penyakit mata pada Citra fundus menggunakan convolutional neural network (CNN),” *Techno.Com*, vol. 21, no. 2, pp. 378–389, 2022, doi: 10.33633/tc.v21i2.6162.

[8] A. K. Khurana, *Comprehensive ophthalmology*. New Delhi, India: New Age International, 2007.

[9] A. Peryanto, A. Yudhana, and R. Umar, “Rancang bangun klasifikasi citra dengan teknologi deep learning berbasis metode convolutional neural network,” *J. Ilm. Tek. Inform.*, vol. 8, no. 2, p. 138, 2020, doi: 10.22441/format.2019.v8.i2.007.

[10] W. Rawat and Z. Wang, “Deep convolutional neural networks for image classification: A comprehensive review,” *J. Neural Comput.*, vol. 29, no. 9, pp. 2352–2449, 2017.

[11] F. Cahya, N. Hardi, and Others, “Klasifikasi penyakit mata menggunakan convolutional neural network (CNN),” *J. Sist.*, vol. 10, no. 3, pp. 618–626, 2021, [Online]. Available: http://sistemasi.ftik.unisi.ac.id

[12] R. Rismiyati and A. Luthfiarta, “Transfer learning with xception architecture for snakefruit quality classification,” *J. Appl. Intell. Syst.*, vol. 7, no. 2, pp. 162–171, 2022, doi: 10.33633/jais.v7i2.6797.

[13] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 1–8, 2015, doi: 10.1109/CVPR.2016.90.

[14] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.

[15] G. Howard, M. Zhu, and Others, “MobileNets: Efficient convolutional neural networks for mobile

vision applications," 2017, [Online]. Available: http://arxiv.org/abs/1704.04861
[16] R. Ramadhani, A. Thohari, and Others, "Optimasi akurasi metode convolutional neural network untuk identifikasi jenis sampah," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 2, pp. 312–318, 2021, doi: 10.29207/resti.v5i2.2754.