International Journal of Computer Applications Technology and Research
Volume 11–Issue 07, 247-254, 2022, ISSN:-2319–8656
DOI:10.7753/IJCATR1107.1001

A Comparative Study of Deep Learning and Transfer Learning in Detection of Diabetic Retinopathy

Jackson Kamiri
Murang'a University of Technology
Kenya

Geoffrey Mariga Wambugu
Murang'a University of Technology
Kenya

Aaron Mogeni Oirere
Murang'a University of Technology
Kenya

Abstract: Computer vision has gained momentum in medical imaging tasks. Deep learning and Transfer learning are some of the approaches used in computer vision. The aim of this research was to do a comparative study of deep learning and transfer learning in the detection of diabetic retinopathy. To achieve this objective, experiments were conducted that involved training four state-of-the-art neural network architectures namely; EfficientNetB0, DenseNet169, VGG16, and ResNet50. Deep learning involved training the architectures from scratch. Transfer learning involved using the architectures which are pre-trained using the ImageNet dataset and then fine-tuning them to solve the task at hand. The results show that transfer learning outperforms learning from scratch in all three models. VGG16 achieved the highest accuracy of 84.12% in transfer learning. Another notable finding is that transfer learning is able to not only achieve high accuracy with very few epochs but also starts higher than deep learning in the first epoch. This study has also demonstrated that in image processing tasks there are a lot of transferrable features since the ImageNet weights worked well in the Diabetic retinopathy detection task.

Keywords: Meta-Learning, Transfer learning, Deep learning, Medical Image processing, Diabetic Retinopathy.

1.0 INTRODUCTION

The evolution of machine learning has greatly contributed to solving some of the major problems in the world. Of particular interest is deep learning which has become a game-changer in computer vision due to its representation learning capabilities[1]. Under representational learning, a machine is fed with raw data and it develops its own representation needed to extract the data[1]. This is made possible by convolutional neural networks since they can extract features from an image using the convolutional layer and thus a separate feature extractor is not needed.

Deep learning has been applied in a variety of image processing tasks in various fields to solve image processing problems[2]. Deep learning has promising results in complex medical diagnostics. It helps physicians by providing a second opinion and flagging concerning areas in images[1].

Meta-Learning is also known as learning-to-learn makes it possible for deep learning models to do multitask learning and use transfer learning to enable them to solve a new but related task with just a few data samples also known as few-shot learning[3]. This helps in addressing the challenges of data shortage and increases the robustness of models developed using this approach.

Transfer learning involves training a deep learning architecture with huge amounts of data. This training involves feature extraction from the training dataset. Once the training is done the weights are then transferred and fine-tuned to a smaller dataset[4]. Thus, this makes it possible for the transfer learning model to leverage on previously acquired knowledge.

The main aim of this study was to do a comparative study of deep learning and transfer learning in the detection of diabetic retinopathy. Deep learning, involved training a model from scratch using the dataset. In transfer learning, a model is first pre-trained using the ImageNet dataset then the weights are transferred to the Diabetic retinopathy dataset. Further to this, the model is finetuned and the results of the two approaches are compared.

The other sections are organized as follows. 2.0 related works, 3.0 Methodology, 4.0 results, and discussion, 5.0 conclusion and future work.

2.0 RELATED WORKS

Diagnosis based on medical images has been very successful in using convolutional neural network-based methods. This is largely motivated by the fact that CNN has achieved human-level capabilities in tasks involving object classification[1]. CNN networks have also demonstrated strong performance in transfer learning in medical image-based diagnostics[1].

Diabetic retinopathy (DR) is an eye disease that is a result of diabetes. It is characterized by damaged blood vessels in the retina, swollen or leaking vessels, some close thus stopping blood from passing through them, and abnormal vessels can grow in the retina. These changes can eventually result in loss of vision[5].

Diabetic retinopathy is usually classified into five main categories which are: No DR, Mild DR, Moderate DR, Severe DR, and Proliferative DR [6]. Several researchers have developed models aimed at classifying Diabetic retinopathy. The models are trained using public datasets such: as the EyePACs dataset, Indian Diabetic Retinopathy Dataset, APTOS 2019 blindness dataset, and Messidor Dataset[7][8] [9]. Researchers have used a combination of two or more of these datasets or just one dataset in developing their models.

Thiagarajan et al., (2020) used a Convolutional Neural Network and Grey-level co-occurrence matrix with the Indian Diabetic retinopathy dataset. The data preprocessing that was done involved horizontal flipping, scaling, zooming in, cropping, and translation. The model used the binary cross-entropy loss function to do binary classification.

Welling (2018) developed a model that pre-trained reptile with ImageNet dataset and then transferred the weights to the Diabetic retinopathy dataset. The researcher used the default reptile network architecture in pre-training and meta-learning. The batch normalization layer was removed during pre-training since it was found to be less transferrable. During transfer learning the weights apart from the last SoftMax layer were finetuned to the target dataset using the Adam optimizer and its default parameters (learning rate 0.001, $\beta1 = 0.9$, $\beta2 = 0.999$, and a batch size of 32) [10].

Ensembles of CNN models for transfer learning to create a meta-algorithm has also proved to be a good approach in the detection of Diabetic retinopathy. [11] proposed a model that ensembles state-of-the-art CNN networks which include: ResNet50, InceptionV3, Xception, Dense121, and Dense169. The networks were pre-trained using ImageNet and then fine-tuned to the Diabetic Retinopathy dataset. The ensembled model achieved 80.8%, 51.5%, 86.72%, 63.85%. and 53.74% in accuracy recall, specificity, precision, and F1-score respectively[11].

Previous works that tried to do a comparative study of transfer learning and other image processing approaches include; [4] which did a comparative study of deep transfer learning and shallow learning in accurate fingerprint detection. The deep transfer learning architectures considered were InceptionV3, NasNet, and ResNet50. While in shallow learning linear and non-linear Gaussian support vector machines were used together with the following image descriptors: Binarized statistical image features, weber local descriptor, and local phase Quantization. [4] did not compare Transfer learning against deep learning in the same environment set-up.

3.0 METHODOLOGY

In this section four neural network architectures which are EfficientNetB0, DenseNet169, VGG16, and ResNet50 have been used for both deep learning and transfer learning tasks.

DenseNet169

This architecture was proposed by Huang et al., [12]. It connects each layer to every other layer in the network in a feed-forward manner. Thus, for each layer, the feature maps for each preceding layer are used as inputs in the subsequent layers. DenseNets are advantageous in the following ways: they reduce the number of parameters, strengthen feature propagation, enhance feature reuse, and alleviate the vanishing gradient problem. DenseNets are easy to train since each layer has direct access to the gradient of the input layer from its loss function. This results to an implicit deep supervision[12], [13].

There exists the following variants of DenseNet: DenseNet121, DenseNet169, DenseNet201, DenseNet264 [12]. The numbers represent the depth of the networks[14]. DenseNet169 has the following features: 7*7 convolutional layers with 2 strides, 3*3 max pooling layer with 2 strides, a series of dense blocks, 7*7 classification layer, and 1000 D fully connected SoftMax [12],[15]. This architecture was chosen since literature has demonstrated it to be a high-performance architecture[15].

EfficientNetB0

EfficientNets were proposed by Tan & Le, (2019) as a new scaling method that uniformly scales in all dimensions of width, resolution, and height using a compound coefficient. By balancing the height, resolution, and width this architecture can achieve higher accuracy than the competitors. EfficientNets are a family that ranges from EfficientB0 to EfficientNetB7 which represents a combination of efficiency and accuracy on different scales. Compound scaling allows the EfficientNetB0 to avoid extensive grind search of hyperparameters thus surpassing models at every scale(Tan & Le, 2019).

EfficientNetB0 was chosen since it is the base model of the EfficientNet family and all the other architectures in the family are scaled from it by adding more layers [16],[17]. Thus, it provides a suitable baseline to compare deep learning and transfer learning. Also, the base model requires less computational power to achieve desirable performance[17].

ResNet50

Residual networks were developed by He et al., [18] with aim of making it possible for neural networks which are deep to be trained with less complexity and achieve high performance. In ResNet, the layers are reformulated as learning residual functions with reference to the layer's input. ResNet50 refers to the 50-layer residual learning network, which also has 3.8 billion flops.

ResNet50 solves the saturation and degradation of accuracy problem[19]. Mukti & Biswas [19], further records that ResNet50 surpasses AlexNet, VCG16, and VCG19 in plant leaf disease classification using transfer learning. This justified the choice of ResNet50 for this task.

VGG16

VGG extended Alex Net by increasing the depth of the network using small 3*3 convolution filters. VGG networks demonstrated the capability to achieve high performance even when used in relatively simple pipelines[20]. The architecture of VGG involves the following: First, the input layer which takes 224*224 pixels colored images. Second, the convolution layers which use a very small receptive field. The convolution layers are accompanied by a 1*1 convolution filter and a Relu unit. Third, three fully connected layers in which the first two have 4096 channels while the last has 1000 channels. Fourth a series of hidden layers with all of them using the RELU activation function[20].

Simonyan and Zisserman [21] record that there are three variants of VGG Networks which are: First, VGG11 which supports 11 weight layers in the model (convolution layers). Second VGG13 which supports 13 weight layers. Third, VGG16 which supports 16 weight layers. Fourth VGG19 which supports 19 weight layers. VGG19 and VGG16 are the most commonly used architectures of the VGG model. Researchers such as Khan et al. [21] used VGG16 to classify diabetic retinopathy and achieved an average accuracy of 83.8%.

3.2 Experiment Setup

3.2.1 Dataset Used

This study used a combination of two datasets namely the Indian Diabetic Retinopathy dataset and the Aptos 2019 blindness detection dataset. The combined dataset resulted in 4125 labeled and colored fundus images. The images were labeled using numbers as follows: 0-No DR, 1-Mild DR, 2-Moderate DR, 3-Severe DR, and 4 Proliferative DR. The researchers then combined Moderate-DR and severe DR into a single class. This was informed by [22] who used the Messidor dataset which has the same distribution. Also,(*Messidor - ADCIS*, 2021.) records that Moderate DR and Severe DR overlap.

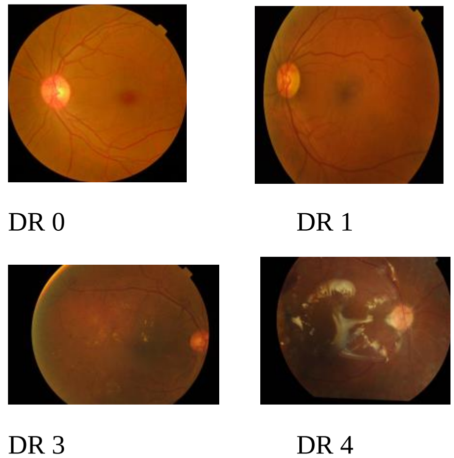

Figure 1: Sample Eye Fundus Images

Figure 1 shows sample Eye Fundus images and their respective labels.

3.2.2 Experiment

All the models were trained in Google Colaboratory with NVIDIA GPU, CUDA version 11.2, TensorFlow version 2.7.0, and Keras.

For the deep learning model algorithm, 3.1 was used in the experiment

3.1: Deep Learning Algorithm

Input →Fundus Images belonging to 4 classes (DR0, DR1, DR3, DR4)

Output →A model that classifies fundus images into the four classes

1) Load the dataset
2) Data pre-processing
   - Split 80% training and 20% Validation (X_{train} $Y_{validation}$)
   - Resizing images
   - Data Augmentation
     - Horizontal random flip
     - Random rotation
     - Random Contrast
     - Random translation (h-factor=0.1, W_factor=0.1)
3) Import the architectures without the weights= (EfficientNet B0, DenseNet169, ResNet50, VGG 16)
4) Set the model parameters (Optimizer, Loss function, Metrics)

While α=0.01

  For epochs= 1 to 50 do

    Update the model parameters for each batch of (X_{train} $Y_{validation}$)
    Assess the Accuracy and Loss

  End

End

The data was loaded into the model then data pre-processing which involved resizing all images to shape (224, 224, 3) was done. This was followed by data augmentation. Data augmentation aimed to have several variants of the same image so that the network can learn how to extract features from images from different viewpoints. It also prevents the overfitting of the model[24].

The architectures were imported from Keras. The model weights were excluded so that the model can be trained from scratch. The following model parameters were set, Adam optimizer with the default learning rate of 0.001, categorical cross-entropy loss function, and Accuracy metrics. The EfficientNetB0 model had a total of 4, 054, 695 parameters out of which 42, 023 were non-trainable parameters.

The DenseNet169 model had 12, 649, 540 total parameters out of which 158, 400 were non-trainable parameters. The ResNet50 model had a total of 23, 595, 908 out of which 53, 120 parameters were non-trainable. The VGG16 architecture had 14, 719, 301 total parameters out of which 3, 589 parameters were trainable. Each model was trained in its own notebook for 50 epochs. The performance in terms of accuracy was then observed and recorded for comparison.

For the Transfer learning, architecture data pre-processing was similar to that of deep learning since the study aimed at comparing the two within the same setup. Algorithm 3.2 is the Transfer learning algorithm that was used.

3.2: Transfer Learning Algorithm

Input →Fundus Images belonging to 4 classes (DR0, DR1, DR3, DR4)

Output →A model that classifies fundus images into the four classes

1) Load the dataset
2) Data pre-processing
   - Split 80% training and 20% Validation (X_{train} $Y_{validation}$)
   - Resizing images
   - Data Augmentation
     - Horizontal random flip
     - Random rotation
     - Random Contrast
     - Random translation (h-factor=0.1, W_factor=0.1)
3) Import pre-trained architectures with the weights= (EfficientNet B0, DenseNet169, ResNet50, VGG16)
   - Rebuild the top layer
   - Unfreeze the base model
4) Set the model parameters (Optimizer, Loss function, Metrics)

```
While model.trainable=true
    While α=1^e-5(low learning rate)
        For epochs= 1 to 50 do
            Update the model parameters for each batch of (X_train Y_validation)
            Assess the Accuracy and Loss
        End
    End
End
```

The imported architectures were pre-trained using the ImageNet dataset. The architectures were then restructured to fit the task. The top layer was rebuilt as follows: A Global Average pooling 2D layer, a Batch Normalization layer, a dropout layer with a dropout rate of 0.2, and a dense layer as the output with SoftMax activation function, were added. The trainable function of the model was then set to true.

The following model parameters were used for all the models: Adam optimizer with a learning rate of ($\alpha=1^{e}$-5), categorical cross-entropy loss function, and accuracy metrics. The models were trained for 50 epochs and the performance was recorded. The EfficientNetB0 model had a total of 4, 059, 815 parameters out of which 44, 583 were non-trainable parameters. The DenseNet169 had a total of 12, 656, 196 parameters out of which only 161, 728 were non-trainable parameters. ResNet50 had a total of 23, 604, 100 out of which 57, 216 parameters were non-trainable. VGG16 had a total of 14, 719, 301 out of which 14, 718, 277 were trainable and 1, 024 were non-trainable.

4.0 RESULTS AND DISCUSSION

Table 1 shows a summary of the performance of the four models in both deep learning and transfer learning.

Table 1: Comparison of Accuracy achieved by the models

Model	Deep Learning	Transfer Learning
EfficientNetB0	79.03%	**80.12%**
DenseNet169	75.39%	**83.15%**
ResNet50	77.70%	**81.33%**
VGG16	73.78%	**84.12%**

The results demonstrate that all the models achieved higher accuracy in transfer learning compared to learning from scratch. Also, VGG16 achieved the highest accuracy of 84.12%. This surpasses what is recorded in literature by [11] who achieved an average accuracy of 80.8%, and [21] who achieved 83.8%.

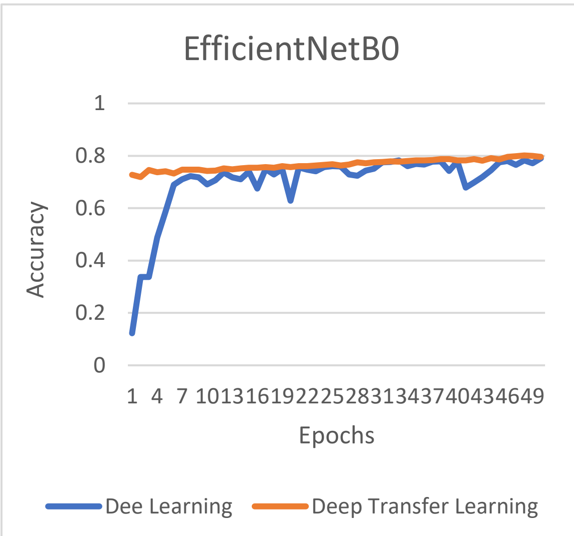

Figure 2: EfficientNetB0

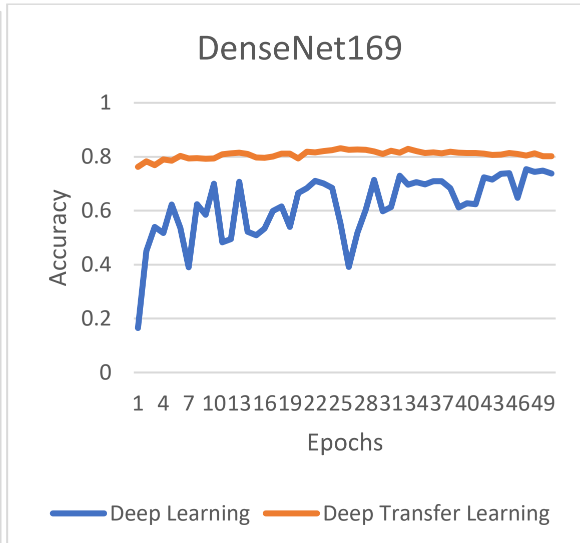

Figure 3: DenseNet169

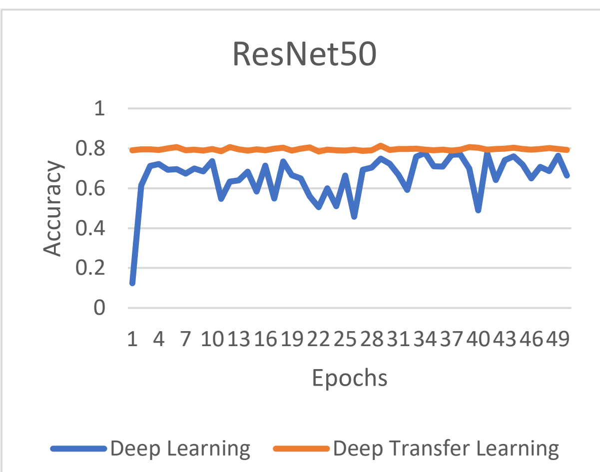

Figure 4: ResNet50

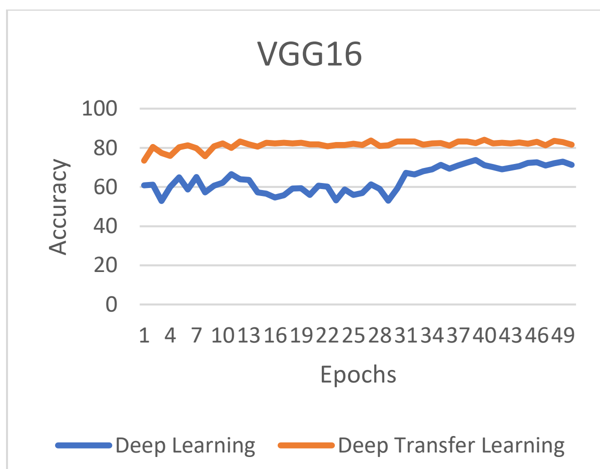

Figure 5: VGG16

Figures 2,3, 4, and 5 show a comparison of accuracy curves for both deep learning (learning from scratch) and transfer learning for 50 epochs. The results demonstrate that with transfer learning the models are easily able to extract features from the images by leveraging on their previous knowledge. While as in deep learning the models start by learning from scratch that's why they achieve very low accuracies in the first epoch.

EfficientNetB0 represented in Figure 2 achieved 12.24% accuracy for deep learning and 72.73% for transfer learning in the first epoch. Within the first 15 epochs, deep learning had achieved an accuracy of 73.58% while transfer learning had achieved an accuracy of 75.15%.

DenseNet169 represented in Figure 3achieved 16.48% for deep learning and 76.24% for transfer learning in the first epoch. Also, within the first 15 epochs, DenseNet169 had already achieved an accuracy of 81.45% for transfer learning while deep learning had only attained an accuracy 0f 70.6%.

ResNet50 represented in Figure 4 attained an accuracy of 79.03% in the first epoch for transfer learning while deep learning attained an accuracy of 12.36%. Within the first 15 epochs, deep learning got an accuracy of 73.58% while transfer learning got an accuracy of 80.61%.

VGG16 represented in Figure 5 attained an accuracy of 60.85% and 73.5% in the first epoch of deep learning and transfer learning respectively. Within the first 15 epochs, deep learning had achieved an accuracy of 66.55% while transfer learning achieved an accuracy of 83.15% in the first 15 epochs.

4.1 Discussion

The results show that VGG16 surpassed all the others in transfer learning. This demonstrates that the CNN and the filters in the VGG16 architecture play a very critical role in feature extraction. While the replaceable fully connected layer enables the model to have some level of domain shift generalizability.

The accuracy curves show that transfer learning enables a model to easily avoid learning a new task from scratch and thus the model is able to achieve a high performance faster and with fewer computing resource needs. This is a great phenomenon since it makes it possible for developers to develop models with just a few shots of data.

The results from the study also demonstrate that weights obtained by pre-training a model on the ImageNet dataset can easily be transferred to solving diabetic retinopathy classification tasks. In this study, we have been able to compare deep learning (learning from scratch) with transfer learning across three state-of-the-art neural network architectures. We have also been able to attain a high accuracy with transfer learning compared to Qummar et al [11] and Pratt et al[25].

CONCLUSION

Meta-learning and transfer learning are some of the best developments in Neural Networks due to their capabilities. They have made it possible for researchers to explore medical imaging research even when big volumes of labeled data are a limitation. Thus, researchers can now be able to create models that automate the classification of diabetic retinopathy.

This study mainly focused on comparing the performance of deep learning and transfer learning in performing the task of classifying diabetic retinopathy. The findings of this study can act as a reference point for researchers who wish to explore computer vision research using neural networks.

DECLARATIONS

Authors declare that they have no conflict of interest in this research paper.

REFERENCES

[1] A. Esteva *et al.*, "A guide to deep learning in healthcare," *Nat. Med.*, vol. 25, no. 1, pp. 24–29, 2019, doi: 10.1038/s41591-018-0316-z.

[2] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, *Artificial Neural Networks and Machine Learning – ICANN 2018*, vol. 11141, no. November. Springer International Publishing, 2018.

[3] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," *34th Int. Conf. Mach. Learn. ICML 2017*, vol. 3, pp. 1856–1868, 2017.

[4] D. Impedovo, V. Dentamaro, G. Abbattista, V. Gattulli, and G. Pirlo, "A comparative study of shallow learning and deep transfer learning techniques for accurate fingerprints vitality detection," *Pattern Recognit. Lett.*, vol. 151, pp. 11–18, 2021, doi: 10.1016/j.patrec.2021.07.025.

[5] W. Wang and A. C. Y. Lo, "Diabetic retinopathy: Pathophysiology and treatments," *Int. J. Mol. Sci.*, vol. 19, no. 6, 2018, doi: 10.3390/ijms19061816.

[6] S. Kusuhara, Y. Fukushima, S. Ogura, N. Inoue, and A. Uemura, "Pathophysiology of diabetic retinopathy: The old and the new," *Diabetes Metab. J.*, vol. 42, no. 5, pp. 364–376, 2018, doi: 10.4093/dmj.2018.0182.

[7] M. T. Hagos and S. Kant, "Transfer Learning based Detection of Diabetic Retinopathy from Small Dataset," 2019.

[8] B. Tymchenko, P. Marchenko, and D. Spodarets, "Deep learning approach to diabetic retinopathy detection," *ICPRAM 2020 - Proc. 9th Int. Conf. Pattern Recognit. Appl. Methods*, pp. 501–509, 2020, doi: 10.5220/0008970805010509.

[9] A. S. Thiagarajan, J. Adikesavan, S. Balachandran, and B. G. Ramamoorthy, "Diabetic retinopathy detection using deep learning techniques," *J. Comput. Sci.*, vol. 16, no. 3, pp. 305–313, 2020, doi: 10.3844/JCSSP.2020.305.313.

[10] M. Welling, "Meta-Learning for Medical Image

Classification," no. Midl, pp. 7–9, 2018.

[11] S. Qummar *et al.*, "A Deep Learning Ensemble Approach for Diabetic Retinopathy Detection," *IEEE Access*, vol. 7, pp. 150530–150539, 2019, doi: 10.1109/ACCESS.2019.2947484.

[12] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 2261–2269, 2017, doi: 10.1109/CVPR.2017.243.

[13] N. Hasan, Y. Bao, A. Shawon, and Y. Huang, "DenseNet Convolutional Neural Networks Application for Predicting COVID-19 Using CT Image," *SN Comput. Sci.*, vol. 2, no. 5, pp. 1–11, 2021, doi: 10.1007/s42979-021-00782-7.

[14] J. Islam and Y. Zhang, "An Ensemble of Deep Convolutional Neural Networks for Alzheimer's Disease Detection and Classification," no. c, pp. 1–5, 2017.

[15] A. Nandy, "A densenet based robust face detection framework," *Proc. - 2019 Int. Conf. Comput. Vis. Work. ICCVW 2019*, pp. 1840–1847, 2019, doi: 10.1109/ICCVW.2019.00229.

[16] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.

[17] L. T. Duong, P. T. Nguyen, C. Di Sipio, and D. Di Ruscio, "Automated fruit recognition using EfficientNet and MixNet," *Comput. Electron. Agric.*, vol. 171, no. January, p. 105326, 2020, doi: 10.1016/j.compag.2020.105326.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[19] I. Z. Mukti and D. Biswas, "Transfer Learning Based Plant Diseases Detection Using ResNet50," *2019 4th Int. Conf. Electr. Inf. Commun. Technol. EICT 2019*, no. December, pp. 20–22, 2019, doi: 10.1109/EICT48899.2019.9068805.

[20] 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.

[21] Z. Khan *et al.*, "Diabetic Retinopathy Detection Using VGG-NIN a Deep Learning Architecture," *IEEE Access*, vol. 9, pp. 61408–61416, 2021, doi: 10.1109/ACCESS.2021.3074422.

[22] C. Lam, D. Yi, M. Guo, and T. Lindsey, "Automated Detection of Diabetic Retinopathy using Deep Learning," pp. 147–155.

[23] "Messidor - ADCIS." https://www.adcis.net/en/third-party/messidor/ (accessed Jan. 08, 2022).

[24] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *J. Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0197-0.

[25] H. Pratt, F. Coenen, D. M. Broadbent, S. P. Harding, and Y. Zheng, "Convolutional Neural Networks for Diabetic Retinopathy," *Procedia Comput. Sci.*, vol. 90, no. July, pp. 200–205, 2016, doi: 10.1016/j.procs.2016.07.014.