Evaluation of the Efficiency of the Optimization Algorithms for Transfer Learning on the Rice Leaf Disease Dataset

Luyl-Da Quach, Khang Nguyen Quoc, Anh Nguyen Quynh, Hoang Tran Ngoc
FPT University, Cantho city, Vietnam

***Abstract*—To improve the model's efficiency, people use many different methods, including the Transfer Learning algorithm, to improve the efficiency of recognition and classification of image data. The study was carried out to combine optimization algorithms with the Transfer Learning model with MobileNet, MobileNetV2, InceptionV3, Xception, ResNet50V2, DenseNet201 models. Then, testing on rice disease data set with 13.186 images, background removed. The result obtained with high accuracy is the RMSprop algorithm, with an accuracy of 88% when combined with the Xception model, similar to the F1, Xception model, and ResNet50V2 score of 87% when combined with the Adam algorithm. This shows the effect of gradients on the Transition learning model. Research, evaluate and draw the optimal model to build a website to identify diseases on rice leaves, with the main functions including images and recording of disease identification points for better management purposes on diseased areas of rice.**

***Keywords*—Optimization algorithm; transfer learning; RMSprop; rice leaf disease; Adam**

I. Introduction

Transfer learning is a popular method used for machine learning recognition problems, with the goal of reusing as a starting point for a second task. Therefore, a Transfer learning model is used to learn on a certain source task and pre-trained this model, then it is used for another model so that the new model learns on the target tasks faster. With the use of pre-trained models, Transfer learning models take a big step up from previous models in improving the accuracy of recognition model and classification based on object features.

In practical applications, Transfer learning models are quite commonly used to improve the accuracy of deep learning models. Specifically in this regard, some studies can be mentioned such as: the problem of identifying patients with Parkinson's disease [1], predicting air quality at large time resolution [2], using VGG-16 classifies retinopathy caused by diabetes [3], improving the process of sleep organization method [4], improving ad accuracy by checking clicks [5], improving the accuracy in counting the number of wheat ears [6], improving the accuracy in classifying medicinal leaves [7], classifying diseases in poultry [8], etc. In general, when using a Transfer learning model with different data sets, the accuracy of the model is also significantly improved when the accuracy increases from 5-8%. This makes an important contribution in predicting more accurately in the recognition problem. However, the Transfer learning model also cannot get high accuracy on a number of different data sets. Therefore, the transfer learning model is interested in order to improve the accuracy of the model.

In improving, the accuracy of the Transfer learning model is also approached in various data problems. In which, there are some applied studies in improving the accuracy of Transfer learning models from input data, or combining algorithms. For the improvement of input data, these studies can be mentioned: A Stacked Denoising Autoencoder [9], Sparse Fingerprinting [10], converting High-Resource to Low-Resource Language [11], time series data augmentation [12], combine images [13], …. For the algorithm, there are studies such as Vector Segmentation [14], Kidney Segmentation [15], SURF features [16], etc. This shows that improving the accuracy of the Transfer learning model has been extremely attended in recent times, especially in the use of optimization models, in which there are some studies such as: using particle swarm. optimization (PSO)[17], heuristic optimization for gesture recognition [18], approaching adaptive fine-tuning for routing through custom or pre-trained layer – called SpotTune [19], using hyper-parameters optimization to identify diseases on maize leaves [20], using Bayesian optimization to create a mixed-signal analog circuit model [21], Adam Deep Learning optimization algorithm for recognizing flower images with background [22], ….

In general, in the past time, there have been many studies to improve the accuracy of Transfer learning models, but there has been no research to evaluate the accuracy by combining Transfer learning models with optimal algorithms. advantage on the rice leaf disease dataset. The main contribution of the study is the implementation of the tasks below:

- Combining optimization algorithms with Transfer learning models on rice leaf disease dataset has no image background.
- Evaluate the optimal model after testing; aim is checking the appropriateness of the Transfer learning model and the optimization algorithm.

Finally, the study uses the Transfer learning model and the optimal algorithm with the highest accuracy to build a system. This system is used to diagnose rice diseases by imaging.

II. Transfer Learning Models

In this section, the study will present six Transfer learning models used, through the introduction to get an overview of the model. In recent years, the MobileNet model has had good results for identifying and implementing embedded and mobile systems. There are different versions of this architecture. MobileNet [23], introduced in 2017, used depth integrals and started with two singles global hyper parameters that effectively balance latency and accuracy. MobieNetV2 [24] was introduced two years later with residuals opposite the number of linear bottlenecks between classes. This model takes tight low dimensional space (bottleneck) as input which is then expanded on high dimensional space. With filtered by deep light convolution, then projected back to low dimensional space with linear convolution. MobileNetV3 [25] is the higher generation of Mobilenets which is public at the ICCV conference in 2019, this model is a config for CPUs in mobile phones with the association of hardware-aware network architecture search (NAS), and the NetAdapt algorithm enhanced through novel architecture refinements. There are two sub-models in this model, MobileNetV3-Large and MobileNetV3-Small, with high and low resource-use matching, which helps the model to promote efficiency on hardware effectively. Another model to be introduced is the network model that goes deeper and deeper, which also makes the training process more difficult and degrades the accuracy of the training rapidly. To solve this problem, in 2016, Deep Residual Learning (ResNet) [26] implemented connection skipping and used batch normalization techniques, using Residual Block to remove the connection layer. This block allows each layer to connect its inputs to its outputs. Families in ResNet include ResNet50, ResNet101, and ResNet152 [27], corresponding to a network of 60 layers, 101 classes, and 152 classes trained on the ImageNet dataset. In 2018, similar to ResNet, the InceptionV3 model [28] is a network trained on the ImageNet dataset. It is introduced as a convolutional neural network model consisting of multiple convolution and maximal steps. DenseNet [29] further modified the model for ResNet, the purpose is that instead of aggregating the output feature maps from all previous layers, it will concatenate all feature maps sequentially instead of summarizing them. Merge output feature maps from all previous layers. DenseNet is used in three variants, DenseNet-121, DenseNet-169, and DenseNet-201. In 2020, EfficientNetB0 [30] was introduced with multiple optimizations when scaling each dimension using a fixed set of scaling factors. This approach surpasses other mod-ern models trained on the ImageNet dataset. In 2017, Xception [31] was introduced as an extension of the Inception architecture, which is intended to replace standard Inceptions with depth-separable aggregates. The architectural models used in this study is shown in Fig. 1.

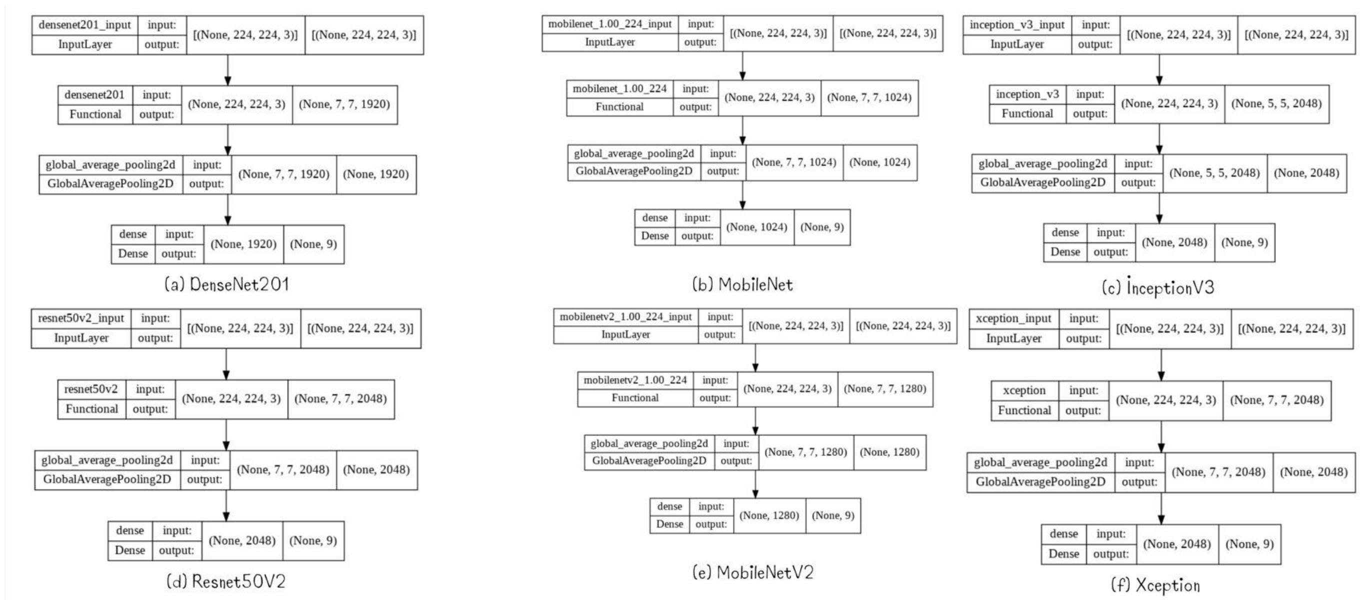

Fig. 1. The Architecture of the Models is used in the Study to Evaluate the Effectiveness of the Optimization Algorithm on Transfer Learning Models.

III. Optimization Algorithm

In this section, the study will theoretically present some optimization algorithms used in the research. The main purpose helps to correct the learning coefficient to accelerate the training, increase the accuracy with fast convergence on the Transfer learning model.

A. Adaptive Moment Estimation (Adam)

Adam algorithm [32] adjusts the learning rate for each weight of the neural network by calculating the first and second intervals of the slope. This makes the algorithm optimal in using the learning rate selection method based on the particular situation. Adam algorithm uses parameters v_t, m_t to estimate of the first moment (the mean) and the second moment (the uncentered variance) of the gradients respectively; β_1, β_2 is speed of movement, g_t is gradient at time t in the formula (1-4) and (5) is the Adam update rule, it is the same as in the RMSprop optimization algorithm.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{1}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) {g_t}^2 \tag{2}$$

$$\widehat{m}_t = \frac{m_t}{1 - {\beta_1}^t} \tag{3}$$

$$\hat{v}_t = \frac{v_t}{1 - {\beta_2}^t} \tag{4}$$

$$W_{t+1} = W_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} \cdot \hat{m}_t \quad (5)$$

B. Adaptive Gradient Algorithm (Adagrad)

Adagrad algorithm [33] implements varying the learning rate to reduce gradients in machine learning techniques. The use of learning weights based on previous inputs to automatically adjust the learning rate to the best divisor instead of a single learning rate, which makes Adagrad better than other algorithms. In the formula (6), η is learning rate, ϵ is guaranteed parameter, g_t is gradient at time t, G_t is the diagonal matrix containing the square of the derivative of the parameter vector at t .

$$W_{t+1} = W_t - \frac{\eta}{\sqrt{G_t + \epsilon}} \cdot g_t \quad (6)$$

C. Root Mean Squared Propagation (RMSprop)

For gradient normalization, RMSprop [34] uses the mean square of the gradient. This helps balance the step size – reduce the step for large gradients to avert Exploding Gradient and increase the step for small gradients to sidestep Vanishing Gradient. Formula (7) is used to calculate the running time parameter $E[g^2]_t$, and form aa rules for formula (8).

$$E[g^2]_t = 0.9E[g^2]_{t-1} + 0.1{g^2}_t \quad (7)$$

$$W_{t+1} = W_t - \frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} \cdot g_t \quad (8)$$

D. Adaptive Delta (Adadelta)

Adadelta algorithm [35] allows to reduce the learning rate on each dimension of SGD. Adadelta restricts the previously accumulated gradients to a predefined size w, which makes the difference in the algorithm by not having to sum the previously squared gradients together. The sum of the gradients is defined recursively as the descending average of all previous gradients. The only variables that affect the running average E[t] at time step t are the previous average and the current gradient (9). By substituting the descending mean on the previous squared gradients for the diagonal matrix (G[t]), in terms of parameter update vector $\Delta\theta_t$ (10).

$$E[g^2]_t = \gamma E[g^2]_{t-1} + (1-\gamma)g_t^2 \quad (9)$$

$$\Delta\theta_t = -\frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} g_t \quad (10)$$

E. Stochastic Gradient Descent (SGD)

Stochastic is a variant of Gradient Descent [36]. Instead of after each epoch we will update the weight (Weight) once, in each epoch with N data points we will update the weight N times. Therefore, it is usually much faster and can also be used to learn online. SGD performs frequent updates with high variance causing the objective function to fluctuate a lot with the corresponding formula with x(i) and y(i) presented for each training instance.

$$\theta = \theta - \eta \cdot \nabla_\theta J(\theta; x^{(i)}; y^{(i)}) \quad (11)$$

IV. Methodology

A. Data and Data Preparation

With the background dataset provided by the Mekong Delta Rice Institute taken by phone (Samsung SM-N770F) with image size (1816x4032). We combed and collected additional leaf disease datasets from multiple sources including Kaggle, GitHub, and from previous related studies in Google Scholar[1]. This dataset contains fully captured rice leaf images from leaf roll to leaf apex, which increases the accuracy of the model and is accompanied by a more complete assessment (Fig. 2). By collecting progress, our dataset consists of 9839 images of diseased leaves of rice consisting of the eight most common diseases namely Bacteria blight, Brown spot, Hispa, Leaf scald, Leaf blast, Leaf smut, Narrow brown spot, and Tungro with the characteristics shown in Fig. 3. There are 3347 images of Healthy leaves. After gathering data, the original size of images in the dataset has various sizes. Hence, we resize it to a large size (1024 x 1024 pixels) for keeping the quality of the image and having a standard size. Besides, we split this dataset into 3 parts: training, validation, and testing with a ratio is 60:30:10 is shown in the Table I.

Since the data is enough, research used various preprocessing steps like image resizing to 224 x 224 and using the processing function of each Transfer learning model to preprocess images to train. With preprocessing dataset, the image in the dataset transforms into an array of RGB colors. The value of each element in array that was scaled to [0; 1], [-1; 1], or [0; 255] depending on each Transfer learning model is shown in Fig. 4.

B. Evaluation Methods

Accuracy is an instance that is precisely determined from the total number of data sets. In other words, it is the correct predictions made by the model relative to the total predictions. It is given by equation (12)

$$Accuracy = \frac{TP + TN}{Total\ sample} \quad (12)$$

In which, TP (True Positive): Total number of Positive matches. TN (True Negative): Total number of Negative matches.

TABLE I. STATISTICS OF TRAINING, VALIDATION AND TESTING DATASETS FOR EACH TYPE OF RICE LEAF DISEASE

Class	Training	Validation	Test	Total
Bacteria blight	496	248	83	827
Brown spot	1,738	869	290	2,897
Hispa	1,189	594	198	1,981
Leaf blast	1,803	901	301	3,005
Leaf scald	214	107	37	358
Leaf smut	203	102	34	339
Narrow brown spot	211	105	36	352
Tungro	48	24	8	80
Healthy	2,008	1004	335	3,347
Total	7,910	3,954	1,322	13,186

[1] The aggregated dataset is stored at https://github.com/raianrido/diseases-on-rice-leaf

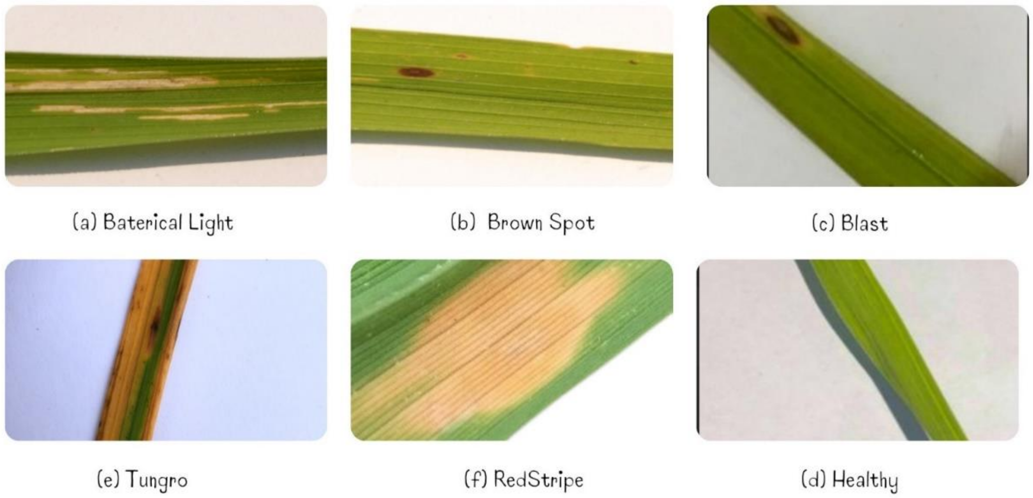

Fig. 2. Images are Collected from the Rice Institute in the Mekong Delta.

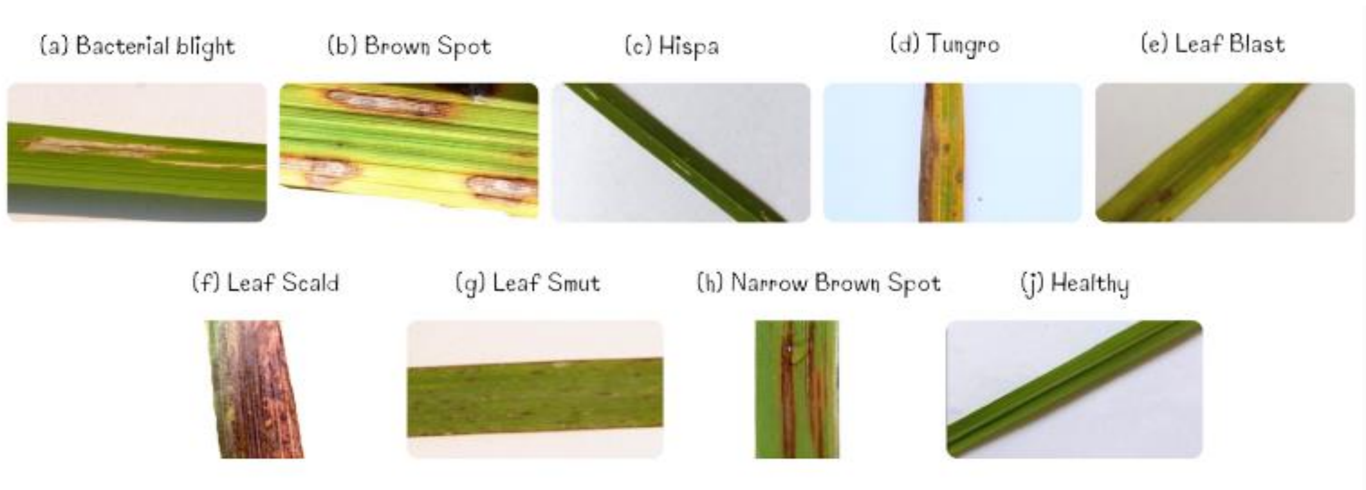

Fig. 3. Illustrated Image of Disease Data on Rice Leaf.

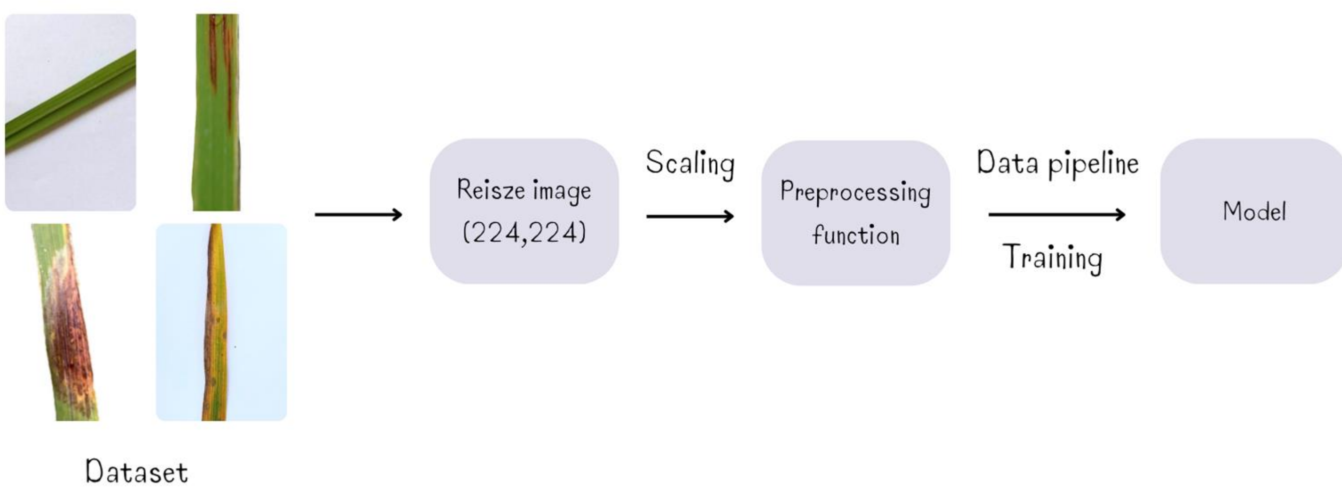

Fig. 4. Data Preparation on Rice Leaf Disease Dataset.

Loss function evaluates the effectiveness of the learning algorithm for the model on the used data set. It produces a non-negative real number representing the distinction between two quantities: a, the predicted label, and y, the correct label. This is a reward-punishment mechanism, the model will have to pay a penalty each time the prediction is wrong and the penalty is proportional to the size of the error. In any supervised math problem, the goal always includes reducing the total penalty payable. In the ideal case a = y, the loss function will return a minimum value of 0. Cross-entropy (CE) is considered as a measure of classifier performance (13).

$$CE = -\sum_{i}^{c} a_i log(p_i) \tag{13}$$

In which, c is the number of classes, a_i the actual value, p_i is the predicted value.

The Confusion matrix gives a better view of how the data points are classified as true/false. It is the summary used to evaluate the classification model performance. The number of true and false predictions is summarized by the count values and broken down by class. When trying to increase the accuracy of the model, the recall will decrease and vice versa. The parameter F1-score reconciles the two values above.

$$F_1 - Score = \frac{2}{\frac{1}{Recall}+\frac{1}{Precision}} \tag{14}$$

$$Prescision = \frac{TP}{Total\ predicted\ positive} = \frac{TP}{TP+FP} \tag{15}$$

$$Recall = \frac{TP}{Total\ actual\ positive} = \frac{TP}{TP+FP} \tag{16}$$

In which, FP (False Positive) is the total number of cases that predict the observations of the label Negative to Positive, FN (False Negative) is the total number of cases that predict the observations of the label Positive to Negative.

F1-Score is a harmonized average of Precision and Recall. It can have the lowest score of 0 (perfect precision and recall) and the highest of 1. The quality of the model in machine learning or deep learning depends on the value of the F1-Score, F1 is high which demonstrates the model is reasonable. Combining Transfer learning model and optimization algorithm.

In this research, Transfer learning will help resolve the issue by utilizing a previously trained in ImageNet dataset as the base for a new model. Transfer learning allows us to customize an already-trained model for a specific problem. Therefore, we are fine-tuning the last layer of six Transfer learning models for the classification of leaf disease consisting of MobileNet, MobileNetV2, InceptionV3, Xception, Resnest50V2, and DenseNet201. By the base layer of model architecture, whose weights were pre-trained and imagined to be frozen, the default input is 224x224 pixels for each image. Then we add the last layers global_average_pooling2d (None, 512) and Dense (0,9) with the activation 'SoftMax'. Before training, each model compiles one in five optimization algorithms. The proposed work offers deep learning, which causes an increase in diseases on rice leaf classification efficiency. The research plotted the history of training processing with accuracy in the training dataset and validation dataset to evaluate the difference between each model with optimization. All optimization algorithms have a default value of a parameter such as a learning rate is 0.001. After training progress, we plotted the confusion matrix and classification report on the testing dataset to have a different view of accuracy in each class and model (Fig. 5).

C. Results

The study is tested on the system CPU Intel Xeon X5675 3.07 GHz, RAM 24 Gbytes and graphics devices NVIDIA Quadro K2200. Experimental results on five optimization algorithms combined with six Transfer learning models on the rice leaf disease data set are presented in Table II and Fig. 6. The study evaluates the accuracy with parameters F1-Score and Accuracy. The results obtained are as follows:

- For the F1-Score evaluation, the association of the ResNet50V2 model, Xception, and Adam algorithm at 87% was the model that reached the highest results, and the lowest was 50% when combining the MobileNet model and Adadelta algorithm.
- For accuracy evaluation, the results show that the discrepancy when the highest accuracy is RMSprop algorithm applying to the Xception model is 88%, and the lowest is 49% when incorporating MobileNet and Adadelta.

D. Application System

A website system is built based on the Xception trained model which compiles RMSprop algorithm with 88% accuracy on the test data set (Fig. 7), with main functions including:

Image recognition: When the user uploads an image at the browse button under the title Detect now and clicks submit, the image will be recognized to detect the disease and return the disease name, symptoms, treatment, and prevention measures in both English and Vietnamese.

- Report false identification and contribute to the system: This is a report function button, helping users to contribute images related to the disease when the system recognizes it wrong. Nevertheless, it will be put into the retraining system when passing the check processing by the admin. At that time, the Report screen will be divided into 3 states: pending (waiting for approval), reject (report has been processed and user information reported is not valid (false)), accept (system error, needed repair and optimization).
- Disease area map: When the user uses the system to identify the disease, the system will use the user's location (Google API) to get information and create a map of the disease. Each color of the dot corresponds to a different disease.

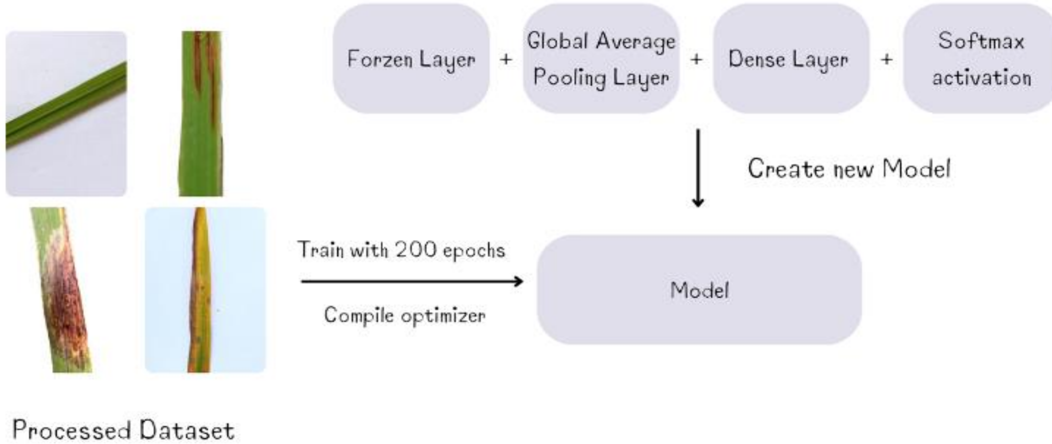

Fig. 5. Illustrating the Combining Transfer Learning Model and Optimization Algorithm.

TABLE II. RESULT TABLE OF COMBINING THE OPTIMAL ALGORITHM WITH THE TRANSFER LEARNING MODEL

Optimization Algorithms	**SGD**		**Adam**		**Adagrad**		**Adadelta**		**RMSprop**	
Model	F1-Score	Accuracy	F1-Score	Accuracy	F1-Score	Accuracy	F1-Score	Accuracy	F1-Score	Accuracy
MobileNet	84%	75%	86%	76%	77%	69%	50%	49%	85%	76%
MobileNetV2	84%	74%	83%	73%	79%	70%	53%	55%	83%	73%
InceptionV3	81%	76%	82%	76%	77%	70%	55%	58%	81%	74%
Xception	84%	76%	**87%**	80%	76%	67%	52%	55%	81%	**88%**
ResNet50V2	85%	77%	**87%**	78%	81%	73%	54%	57%	84%	76%
DenseNet201	80%	71%	80%	69%	77%	69%	85%	77%	80%	69%

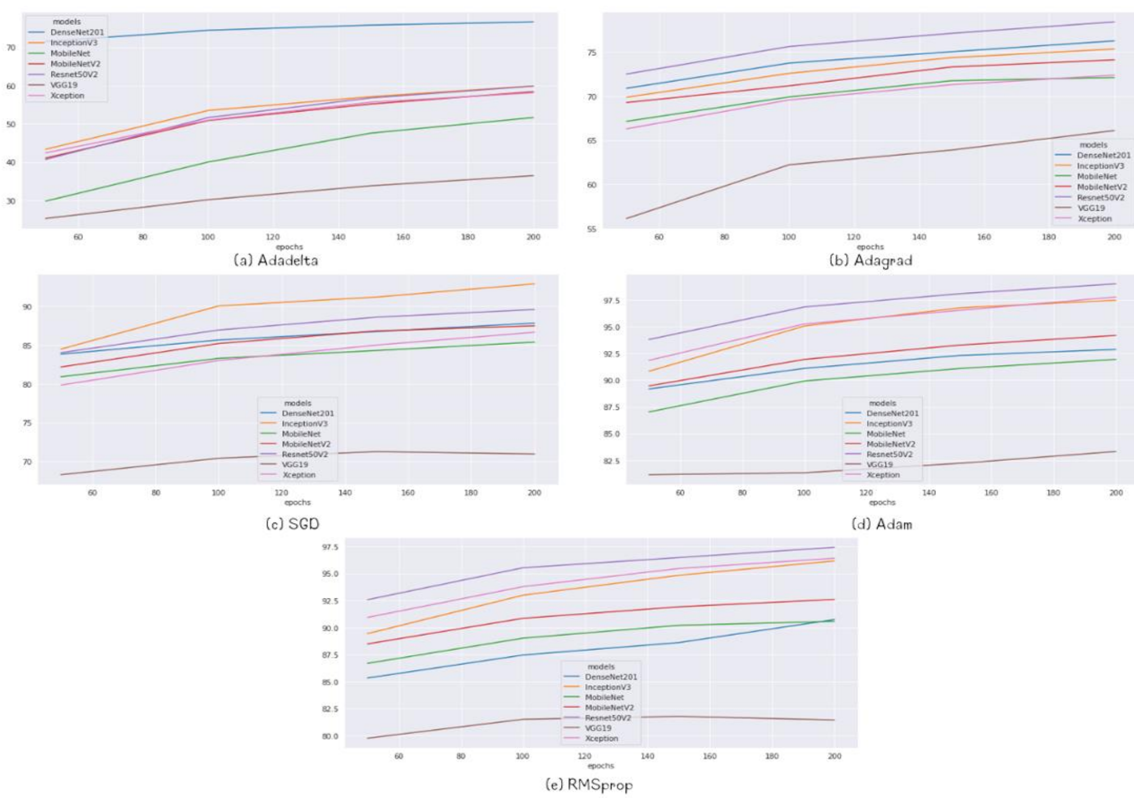

Fig. 6. Compare the Accuracy of Each Algorithm when Combined with Each Transfer Learning Model.

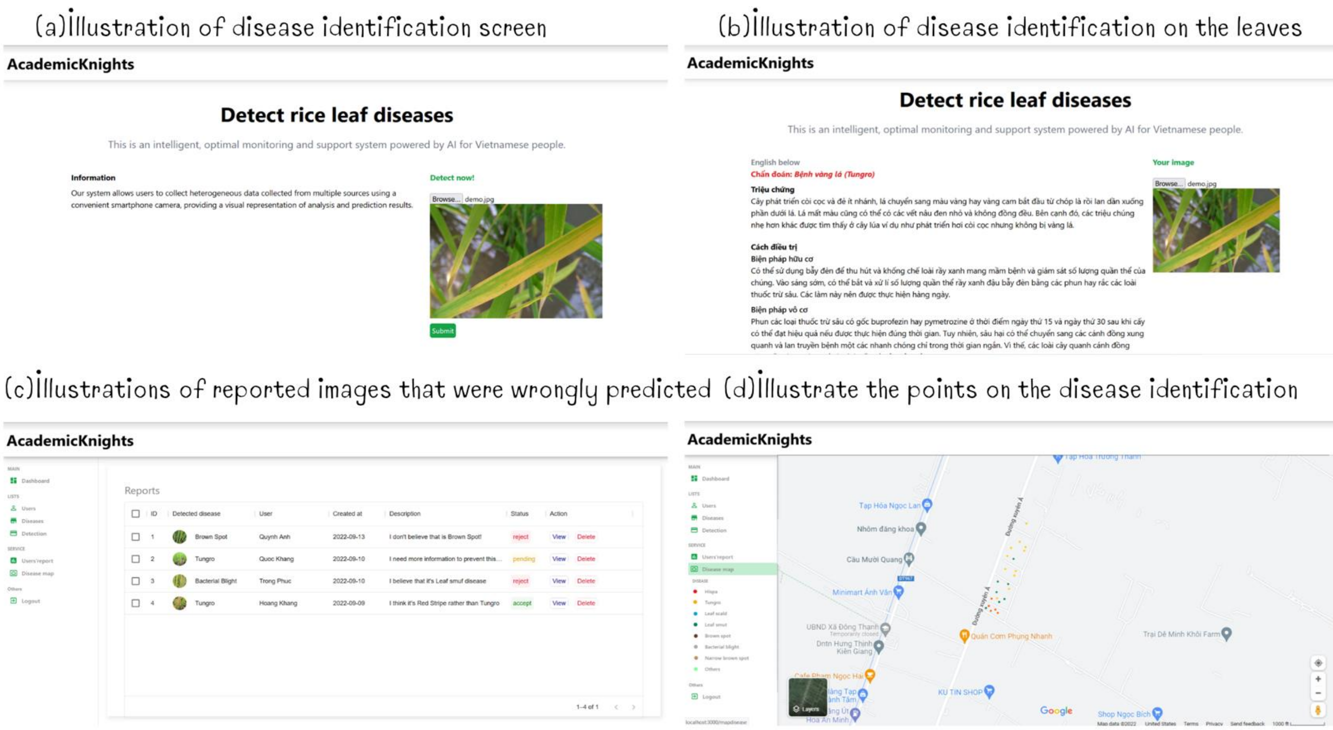

Fig. 7. Illustrate the Website System to Detect Diseases on Rice Leaves.

V. Discussion

Based on test results at Table II and Fig. 6, ResNet50V2 and Xception models combined with Adam algorithm show the efficiency of the algorithm. Analysis of the results shows that Adam algorithm performs better than other algorithms when the accuracy is always high at 87% (F1-Score) and 80% (accuracy). This is because the Adam algorithm further extends stochastic gradient descent to update the network weights during training. The Adam algorithm updates the learning rate for each individual network weight when using a random gradient descent algorithm. Therefore, the optimizers of the Adam algorithm inherit the features of the Adagrad and RMSprop algorithms. In Adam algorithm, instead of adjusting the learning rate based on the first moment (average) but in RMSprop it also uses the second moment of the gradient. In which, the cause comes from the representations of the decay rate β_1 and β_2 of the average of the gradients (17). Regardless, the model that compiles the Adam algorithm is still flawed by incorrectly receiving many images of healthy rice leaves, which figure shows this error.

$$m_t = \beta_1 m_t + (1-\beta_1)\left[\frac{\delta L}{\delta w_t}\right] v_t = \beta_2 v_{t-1} + (1-\beta_2)\left[\frac{\delta L}{\delta w_t}\right]^2 \quad (17)$$

However, when evaluating the accuracy of the algorithm, the optimal model RMSprop shows high efficiency when applied on six models tested on research with parameters of accuracy and F1-Score. This is because RMSprop maintains a moving average of the quadratic gradient, while adjusting the updates to the weights by this magnitude.

In the opposite direction, Adadelta shows no efficiency when combined with Transfer learning model. As proof for this, we can see the results of 5/6 models such as MobileNet, MobileNetV2, InceptionV3, Xception and ResNet50V2. But it works well on the DenseNet201 model (Fig. 8 shows diagnostic discrepancies on some data sets). This result is because the Adadelta algorithm does not use a learning rate parameter. Instead, the algorithm uses its own rate of change as a parameter to adjust the learning rate. Adadelta needs to use two state variables to store the quadratic moments of the gradient and of the change in parameters. In addition, Adadelta uses a leak average to store dynamic estimates of the required statistics. As a result, the learning rate has to be changed manually and the learning rate will decline and become smaller at some point. Therefore, after a certain number of iterations, the model will no longer be able to learn new knowledge.

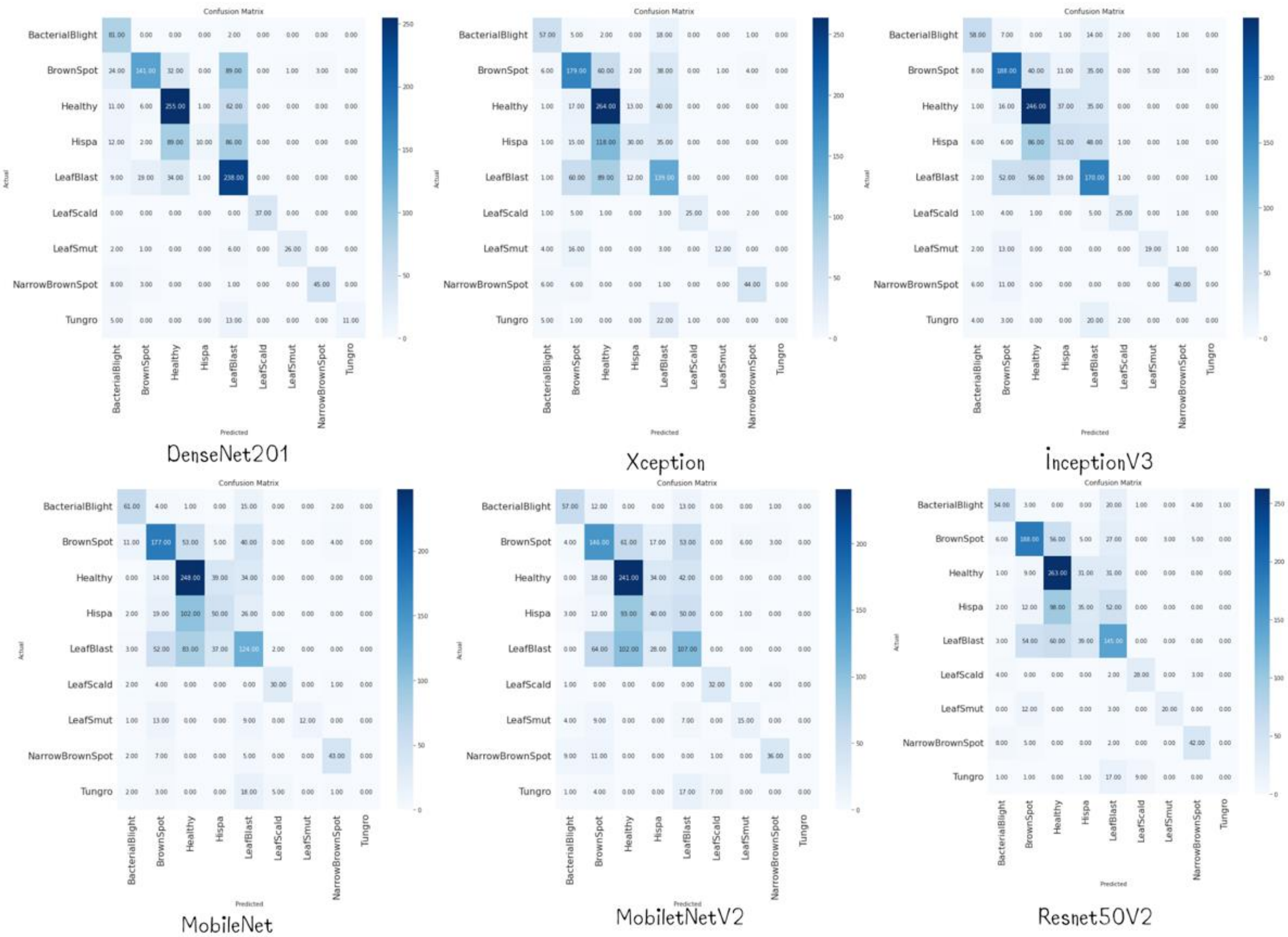

Fig. 8. Confusion Matrix of Adadelta Algorithm with Transfer Learning Models.

VI. CONCLUSIONS

The study proposes that Adam's method is effective when combined with Transfer learning models through gradient expansion to update network parameters and reduce random gradients. This also works for the SGD and RMSprop models. Besides, the low accuracy may come from the influence of input parameters. Although the used image has eliminated the background influence of the image, the resolution parameters cannot be controlled because it is taken from many different sources, in which the data source is generated from study with higher accuracy.

VII. FUTURE WORKS

Based on the research results, it can be seen how effective the optimization algorithms are on different Transfer learning models. However, in this study, we have not gone into detailed evaluation of the combination of models with the optimal algorithm when adjusting the learning rate. Therefore, the study opens up the research and evaluation of the changes of the parameters on the model with the optimization algorithm. Accordingly, the proposal of follow-up studies can more closely evaluate the relationship between the parameters in the optimal model, the influence of the input data set on experiment, and so on. In further research, Transfer Learning models will be fine turning with a more complex output structure and compile implementation with more optimization algorithms, increasing the efficiency of the model.

REFERENCES

[1] J. C. Vásquez-Correa et al., "Transfer learning helps to improve the accuracy to classify patients with different speech disorders in different languages," Pattern Recognition Letters, vol. 150, pp. 272–279, Oct. 2021, doi: 10.1016/j.patrec.2021.04.011.

[2] J. Ma, J. C. P. Cheng, C. Lin, Y. Tan, and J. Zhang, "Improving air quality prediction accuracy at larger temporal resolutions using deep learning and Transfer learning techniques," Atmospheric Environment, vol. 214, p. 116885, Oct. 2019, doi: 10.1016/j.atmosenv.2019.116885.

[3] N. B. Thota and D. Umma Reddy, "Improving the Accuracy of Diabetic Retinopathy Severity Classification with Transfer learning," in 2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS), Springfield, MA, USA, Aug. 2020, pp. 1003–1006. doi: 10.1109/MWSCAS48704.2020.9184473.

[4] H. Phan et al., "Towards More Accurate Automatic Sleep Staging via Deep Transfer learning," IEEE Trans. Biomed. Eng., vol. 68, no. 6, pp. 1787–1798, Jun. 2021, doi: 10.1109/TBME.2020.3020381.

[5] Y. Su et al., "Improving click-through rate prediction accuracy in online advertising by Transfer learning," in Proceedings of the International Conference on Web Intelligence, Leipzig Germany, Aug. 2017, pp. 1018–1025. doi: 10.1145/3106426.3109037.

[6] J. Ma, Y. Li, H. Liu, Y. Wu, and L. Zhang, "Towards improved accuracy of UAV-based wheat ears counting: A Transfer learning method of the ground-based fully convolutional network," Expert

Systems with Applications, vol. 191, p. 116226, Apr. 2022, doi: 10.1016/j.eswa.2021.116226.

[7] N. Duong-Trung, L.-D. Quach, and C.-N. Nguyen, "Learning Deep Transferability for Several Agricultural Classification Problems," ijacsa, vol. 10, no. 1, 2019, doi: 10.14569/IJACSA.2019.0100107.

[8] L.-D. Quach, N. Pham-Quoc, D. C. Tran, and Mohd. Fadzil Hassan, "Identification of Chicken Diseases Using VGGNet and ResNet Models," in Industrial Networks and Intelligent Systems, vol. 334, N.-S. Vo and V.-P. Hoang, Eds. Cham: Springer International Publishing, 2020, pp. 259–269. doi: 10.1007/978-3-030-63083-6_20.

[9] C. Kandaswamy, L. M. Silva, L. A. Alexandre, R. Sousa, J. M. Santos, and J. M. de Sa, "Improving Transfer learning accuracy by reusing Stacked Denoising Autoencoders," in 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC), San Diego, CA, USA, Oct. 2014, pp. 1380–1387. doi: 10.1109/SMC.2014.6974107.

[10] K. Adamkiewicz, P. Koch, B. Morawska, P. Lipiński, K. Lichy, and M. Leplawy, "Improving UWB Indoor Localization Accuracy Using Sparse Fingerprinting and Transfer learning," in Computational Science – ICCS 2021, vol. 12747, M. Paszynski, D. Kranzlmüller, V. V. Krzhizhanovskaya, J. J. Dongarra, and P. M. A. Sloot, Eds. Cham: Springer International Publishing, 2021, pp. 291–302. doi: 10.1007/978-3-030-77980-1_23.

[11] S. Durrani and U. Arshad, "Transfer learning from High-Resource to Low-Resource Language Improves Speech Affect Recognition Classification Accuracy." arXiv, Mar. 04, 2021. Accessed: Sep. 16, 2022. [Online]. Available: http://arxiv.org/abs/2103.11764.

[12] K. Bandara, H. Hewamalage, Y.-H. Liu, Y. Kang, and C. Bergmeir, "Improving the accuracy of global forecasting models using time series data augmentation," Pattern Recognition, vol. 120, p. 108148, Dec. 2021, doi: 10.1016/j.patcog.2021.108148.

[13] T. K. Yoo, J. Y. Choi, J. G. Seo, B. Ramasubramanian, S. Selvaperumal, and D. W. Kim, "The possibility of the combination of OCT and fundus images for improving the diagnostic accuracy of deep learning for age-related macular degeneration: a preliminary experiment," Med Biol Eng Comput, vol. 57, no. 3, pp. 677–687, Mar. 2019, doi: 10.1007/s11517-018-1915-z.

[14] V. Gripon, G. B. Hacene, M. Lowe, and F. Vermet, "Improving Accuracy of Nonparametric Transfer learning Via Vector Segmentation," in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Apr. 2018, pp. 2966–2970. doi: 10.1109/ICASSP.2018.8462273.

[15] J. B. Graham-Knight et al., "Accurate Kidney Segmentation in CT Scans Using Deep Transfer learning," in Smart Multimedia, vol. 12015, T. McDaniel, S. Berretti, I. D. D. Curcio, and A. Basu, Eds. Cham: Springer International Publishing, 2020, pp. 147–157. doi: 10.1007/978-3-030-54407-2_13.

[16] L.-D. Quach, N. P. Quoc, N. H. Thi, D. C. Tran, and M. F. Hassan, "Using SURF to Improve ResNet-50 Model for Poultry Disease Recognition Algorithm," in 2020 International Conference on Computational Intelligence (ICCI), Bandar Seri Iskandar, Malaysia, Oct. 2020, pp. 317–321. doi: 10.1109/ICCI51257.2020.9247698.

[17] B. H. Nguyen, B. Xue, and P. Andreae, "A particle swarm optimization based feature selection approach to Transfer learning in classification," in Proceedings of the Genetic and Evolutionary Computation Conference, Kyoto Japan, Jul. 2018, pp. 37–44. doi: 10.1145/3205455.3205540.

[18] T. Ozcan and A. Basturk, "Transfer learning-based convolutional neural networks with heuristic optimization for hand gesture recognition," Neural Comput & Applic, vol. 31, no. 12, pp. 8955–8970, Dec. 2019, doi: 10.1007/s00521-019-04427-y.

[19] Y. Guo, H. Shi, A. Kumar, K. Grauman, T. Rosing, and R. Feris, "SpotTune: Transfer learning Through Adaptive Fine-Tuning," p. 10.

[20] M. Subramanian, K. Shanmugavadivel, and P. S. Nandhini, "On fine-tuning deep learning models using Transfer learning and hyper-parameters optimization for disease identification in maize leaves," Neural Comput & Applic, vol. 34, no. 16, pp. 13951–13968, Aug. 2022, doi: 10.1007/s00521-022-07246-w.

[21] J. Liu, M. Hassanpourghadi, Q. Zhang, S. Su, and M. S.-W. Chen, "Transfer learning with bayesian optimization-aided sampling for efficient AMS circuit modeling," in Proceedings of the 39th International Conference on Computer-Aided Design, Virtual Event USA, Nov. 2020, pp. 1–9. doi: 10.1145/3400302.3415687.

[22] J. Feng, Z. Wang, M. Zha, and X. Cao, "Flower Recognition Based on Transfer learning and Adam Deep Learning Optimization Algorithm," in Proceedings of the 2019 International Conference on Robotics, Intelligent Control and Artificial Intelligence - RICAI 2019, Shanghai, China, 2019, pp. 598–604. doi: 10.1145/3366194.3366301.

[23] A. G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications." arXiv, Apr. 16, 2017. Accessed: Sep. 17, 2022. [Online]. Available: http://arxiv.org/abs/1704.04861.

[24] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks." arXiv, Mar. 21, 2019. Accessed: Jun. 27, 2022. [Online]. Available: http://arxiv.org/abs/1801.04381

[25] A. Howard et al., "Searching for MobileNetV3," in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), Oct. 2019, pp. 1314–1324. doi: 10.1109/ICCV.2019.00140.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[27] O. Russakovsky et al., "ImageNet Large Scale Visual Recognition Challenge," Int J Comput Vis, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/s11263-015-0816-y.

[28] J. M. Ahn, S. Kim, K.-S. Ahn, S.-H. Cho, K. B. Lee, and U. S. Kim, "A deep learning model for the detection of both advanced and early glaucoma using fundus photography," PLoS ONE, vol. 13, no. 11, p. e0207982, Nov. 2018, doi: 10.1371/journal.pone.0207982.

[29] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks." arXiv, Jan. 28, 2018. Accessed: Jun. 27, 2022. [Online]. Available: http://arxiv.org/abs/1608.06993.

[30] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks." arXiv, Sep. 11, 2020. Accessed: Jun. 27, 2022. [Online]. Available: http://arxiv.org/abs/1905.11946.

[31] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, Jul. 2017, pp. 1800–1807. doi: 10.1109/CVPR.2017.195.

[32] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization." arXiv, Jan. 29, 2017. Accessed: Jul. 05, 2022. [Online]. Available: http://arxiv.org/abs/1412.6980.

[33] J. Duchi, E. Hazan, and Y. Singer, "Adaptive Subgradient Methods for Online Learning and Stochastic Optimization," p. 39.

[34] F. Zou, L. Shen, Z. Jie, W. Zhang, and W. Liu, "A Sufficient Condition for Convergences of Adam and RMSProp," in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, Jun. 2019, pp. 11119–11127. doi: 10.1109/CVPR.2019.01138.

[35] M. D. Zeiler, "ADADELTA: An Adaptive Learning Rate Method," 2012, doi: 10.48550/ARXIV.1212.5701.

[36] S. Ruder, "An overview of gradient descent optimization algorithms." arXiv, Jun. 15, 2017. Accessed: Sep. 17, 2022. [Online]. Available: http://arxiv.org/abs/1609.04747.