Watermelon Leaf Disease Detection Using Transfer-Learning And EX-AI

Ibrahim Sakib
dept. CSE
Brac University
Dhaka, Bangladesh
ibrahim.sakib@g.bracu.ac.bd

M.A.B.Siddique Sanjan
dept. CSE
Brac University
Dhaka, Bangladesh
m.a.b.siddique.sanjan@g.bracu.ac.bd

Md Fuad Islam
dept. CSE
Brac University
Dhaka, Bangladesh
md.fuad.islam@g.bracu.ac.bd

Md Minhazul Islam Rimon
dept. CSE
Brac University
Dhaka, Bangladesh
md.minhazul.islam.rimon@g.bracu.ac.bd

Farah Binta Haque
dept. CSE
Brac University
Dhaka, Bangladesh
farah.binta.haque@g.bracu.ac.bd

Humaion Kabir Mehedi
dept. CSE
Brac University
Dhaka, Bangladesh
humaion.kabir.mehedi@g.bracu.ac.bd

Annajiat Alim Rasel
dept. CSE
Brac University
Dhaka, Bangladesh
annajiat@bracu.ac.bd

***Abstract*—During summer, one of the most demanded fruits in not only Bangladesh but all across the globe is watermelon. It not only has many nutritional benefits but is also very delicious for consumption. However, spoilage of watermelon due to unhealthy watermelon harvest can be seen every year. Thus it will be greatly beneficial for the farmers to be able to identify and treat the unhealthy watermelon crops as soon as possible. There is already some work being done in this area to automate the process of detecting disease among plants using machine learning, transfer learning, and deep learning approaches. Some common transfer learning models like Inception had 98.01% accuracy in detecting plant diseases. There are also other models like ResNet50 with 98.93% accuracy in some cases. Our goal is to use different transfer learning techniques and explainable AI models to detect watermelon leaf diseases. We will compare and contrast between different transfer learning models and will combine the best model with Grad-CAM, an EX-AI model in order to achieve the optimal way of detecting watermelon leaf diseases.**

***Index Terms*—ResNet, leaf, disease, detection, classifying**

I. Introduction

Watermelon greatly contributes to Bangladesh's farming. This fruit, beloved in the country, bolsters farmer's health, income, and fights climate change. Each year, Bangladesh yields over two million tons of watermelon. In the 2019-20 fiscal year, farmers allocated some 27,000 acres of land for watermelon cultivation, harvesting 188,000 tonnes of this refreshing summer fruit. An investigation into watermelon disease in a couple of specific districts in Bangladesh discovered different rates of downy mildew (a watermelon disease). The highest occurrence was 11.33%, and the worst-case severity reached 4%, specifically in South Kocchopiya in Noakhali. To be able to automate the process of detecting disease for leaf images will greatly help the farmer as treatment for said crop can begin fast and improve yield. There are already many works done in this area where different plant diseases were identified using machine learning and transfer learning approaches.Inception, ResNet50, ResNet34, VGG16, VGG19, MobileNet, SVM, CNN, KNN are some of the commonly known model used in this field for detecting plant diseases. In our time, technology regarding image processing and object detection from images have seen great strides in their field. In our work, we will compare different transfer learning models like Inception, ResNet50, MobileNet, DenseNet to find the best one which we will combine with Grad-CAM to further improve classifier accuracy to effectively detect watermelon leaf disease.

II. Literature review

Orchi et al.[1] did an extensive comparative study in his paper to find the difference between traditional machine learning approach and deep transfer learning approach for detecting crop disease. The researchers used PlantVillage dataset to train all their chosen models. For the machine learning approach, the researchers choose Support Vector Machine, k-Nearest Neighbors , Linear Discriminant Analysis, Classification and Regression Trees, NB, and RF models to test accuracy against deep transfer learning models like VGG19, VGG16, ResNet50, CNN and Inception. First and foremost, the researchers pre-processed their dataset images for accelerated computations.Than, features were extracted from the processed image which was later normalized for training. After the said models were trained using the extracted feature the end result was a classifier model that was used to verify accuracy of the model with test cases After all the experimentation, the researchers concluded that Inception V3 had the best accuracy of all models which was 98.01%. The other models namely RF, CART, CNN, ResNet50 had accuracy of 97.54%, 94.45%, 93.89% and 93.57%. As we are trying to find the comparatively best transfer learning model in order to create a hybrid model with EX-AI, we can also use Inception and Resnet for our study to detect leaf diseases.

Uguz et al.[2] compared the accuracy of the VGG transfer learning models with their own proposed model according to the change of optimization algorithms. In their paper, they have extensively analyzed the VGG16 and VGG19 transfer models about what the characteristics of the images are and how many convolution layers were used. Then they suggested a model of their own based on CNN which processed images with 16 filters through 3 convolution layers where the filters gradually incremented to 256 filters at the last layer. Whereas the existing VGG models have 13-16 convolution layers and 64-512 filters. They have compared these three models with a few optimization algorithms which are Adam, AdaGrad, SGD and RMSpro. The goal of the paper is to find the difference in performance between their models with the effect of optimization models on olive leafs. There were 3 categories: healthy and two diseases. After running the models, there were complex results as some models performed better than usual with one optimization but performed worse in another. So, we can say that optimization algorithms perform differently in different models and we can apply that knowledge in our case.

AARIZOU et al.[3] in their study demonstrates the application of transfer learning to the identification of plant diseases on intricate photos through the use of CNN. In this paper they prepared their dataset by combining two public datasets PlantVillage and EdenLibrary where they labeled into two classes ‘Healthy’ and ‘Unhealthy’ by splitting a total of 54000 leaves images equally where PlantVillage have lab images and EdenLibrary dataset have real field images. Due to the EdenLibrary dataset’s smaller size than PlantVillage’s dataset, data augmentation was applied to all images to make it equal. Then three SOTA for image classifiers, DenseNet121, ResNet34 and AlexNet that have been previously trained on the dataset of ImageNet, were fine-tuned using this dataset. After training these models on the lab image dataset and testing on the same dataset they got accuracy of 99.85%, 99.91% and 99.63% respectively. Similarly for the real field images dataset they got 97.35%, 97.85% and 97.31% respectively. But when they trained on only laboratory images and tested on real field images accuracy dropped down to lower than 60%. But after they did combined training of both datasets their accuracy improved significantly. For DenseNet121, ResNet34 and AlexNet models combined dataset training they got 99.76%, 99.85% and 99.43% when testing for lab images respectively and 97.46%, 97.02% and 98.65% when testing for field images respectively. From these results they concluded that DenseNet121 gave the best accuracy for them. As in this research they have used only healthy and unhealthy classes, their models might not perform as good when it comes to detecting specific diseases. However, as DenseNet yielded the best result out of all the models mentioned here, this model may very well be suited for our tasks.

Arshad et al.[4] did a study where researchers measured the success of ResNet50 against VGG16 and MCNN (Multi-column Convolutional Neural Network). These models were made and trained independently for the purpose of identifying plant diseases. Tests and authentications were done on the PlantVillage dataset. Different measurements, such as accuracy, precision, recall, and the F1-score evaluated the models’ performance. When looking at the results, we find that ResNet50, VGG16, and MCNN were top performers for identifying plant diseases. ResNet50 led the pack with a staggering 98.93% accuracy while VGG16 was not too far behind at 98.3%, and MCNN came in third with 93.7% accuracy. This showcased the model’s effectiveness in identifying specific diseases. Certain detailed learning models like ResNet50, VGG16, and MCNN, can precisely pinpoint plant illness. According to the researchers, ResNet50 tops the list in terms of effectiveness. As we are also planning on using an effective transfer learning approach, we can use ResNet50 for our experiments.

III. DATASET

We are using the “Watermelon Disease Recognition Dataset” by Mohammad Imtiaz Nakib and Firoz Mridha from American International University Bangladesh. The dataset consist of 4 different categories of leaf images which are healthy, anthracnose, downy mildew, mosaic virus. So there are in total 3 disease categories provided in the dataset. Number of original images in the dataset is 1155, number of augmentation images is 5775. For our experiment, we will not include the anthracnose category and will use the augmented type images instead of original ones.

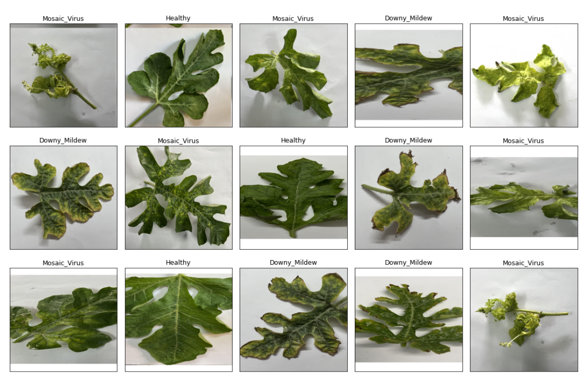

Figure 1: *Example images from the dataset.*

IV. Methodology

A. Data analysis and preprocess techniques

The dataset is structured in a manner where all the categories are separated in different folders. For example, in the folder “Healthy”, only healthy leaf images are stored. Thus, we decided to first label each image in a tabular format. After labeling, we created three distinct directory train, val and test and divided the labeled images in these categories. 75% of the dataset was used for training, 20% for validation and 5% for testing purposes. Further pre-processing of the image data for neural networks training was done differently for every model selected.

For every model, certain image data generation functions were used. However, certain elements of it were the same in all functions. For example, in almost every function rotation_range=20 was set in order for the model to be able to handle variation in orientation. Again there were unique parameters also. In DenseNet, the images pixel values were rescaled to a range of [0,1] by dividing each pixel with 255. The image data generator function was optimized in this way so that it could handle different orientations, zooms, width and height, horizontal flips. Now, when training images were randomly zoomed/rotated some new pixels can be found, in order to handle this new pixel a method was used where these pixels were assigned data according to the nearest one. The brightness of the image was also altered, color was slightly altered as well using a constant parameter in the generator function. Also, the image was sheared as well for training some models.

B. Model Selection

We choose DenseNet201, Inception V3, ResNet50, EfficientNet and lastly MobileNet for first phase training purposes. Our aim is to find the model with the best light model with great performance among all the mentioned ones. And for the second phase of training, we will combine the best model found in the first phase with Grad-CAM and compare the resulting accuracy with the rest of the models.

C. Training

While training each model, some parameters were kept constant. For instant, image size was always 124*124 in all models, batch size was 20 and epoch used was 10. Weights were first assigned from the ImageNet dataset. This is beneficial as it becomes a good starting point for further training models

Firstly for DensNet201, in the base model some custom layers were added.Global average pooling layer was the first of them. Then there was a fully connected dense layer with 3 units and ReLU activation as well as a dropout layer to prevent overfitting. The dropout rate was set 0.2. Lastly there was another dense layer where softmax activation was used. Thus, all the layers were customized for the task in hand and the model was ready for training with the dataset.

Then for Inception V3, similarly some custom layers were added. Here after the global average pooling layer (GAP), flattening was done. Then with ReLU activation a fully connected dense layer was added. A dropout layer with 0.3 dropout rate was added. Then a second fully connected dense layer was added with ReLU activation. Another but same dropout layer was added. Lastly, a dense layer with softmax activation was added.

In the case of ResNet50, the number of custom layers added was much less compared to other ones. After the GAP layer, two dense layers were added with ReLU activation and lastly a dropout layer with 0.3 dropout rate was added.

For EfficientNet, first the batch normalization layer was added. Then similar to the last few models a dense layer and a dropout layer was added. Lastly for MobileNet, in addition to the base layer two additional dense layers with ReLU activation were added.

For optimizers, adam optimizer was used in every model. After training all the models, running validation and testing through them we found that MobileNet was the best choice for our second phase training. The results will be thoroughly discussed in detail in upcoming chapters. Now, for second phase training we combined MobileNetV2 and Grad-CAM which stands for Gradient-weighted Class Activation Mapping. By combining them it helps the model to visualize and understand the regions that need to be focused while predicting using the pre-trained weight on our dataset.

V. Result and Analysis

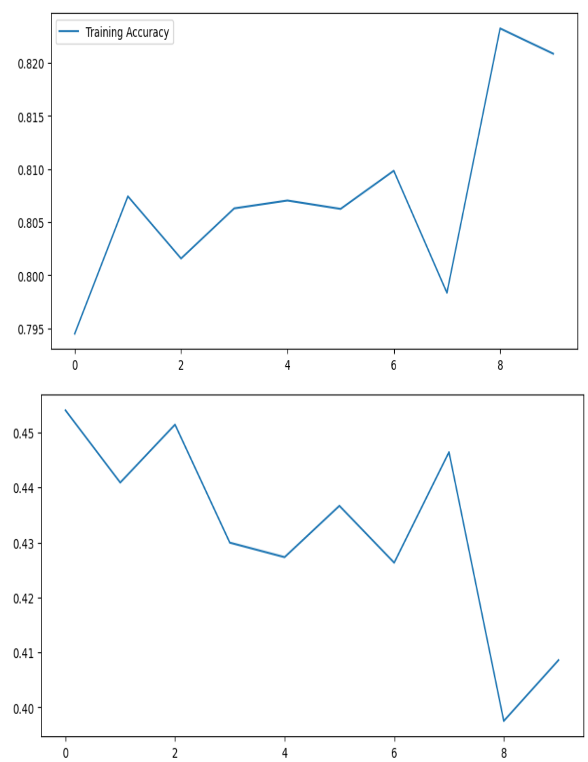

Figure 2: *DenseNet201 accuracy and loss.*

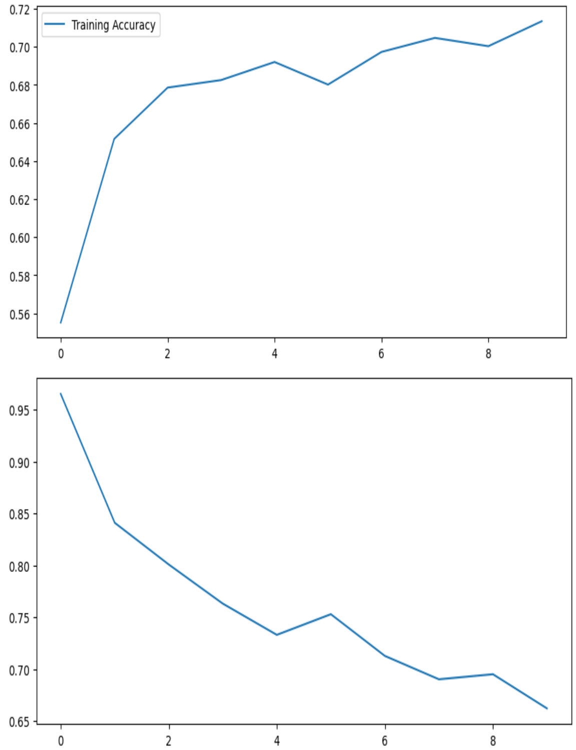

Figure 3: *InceptionV3 accuracy and loss*

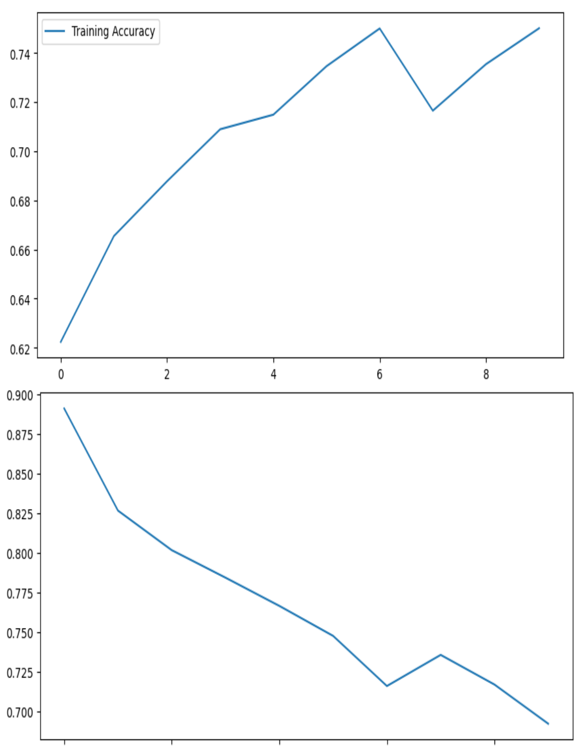

Figure 4: *ResNet50 accuracy and loss*

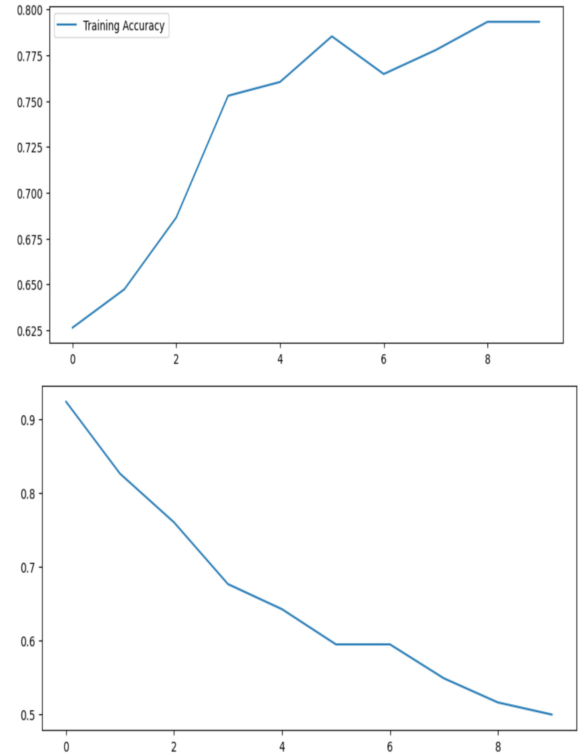

Figure 5: *EfficientNetB3 accuracy and loss*

Model Name	Accuracy
DenseNet201	95.62%
Inception V3	92.83%
ResNet50	78.49%
EfficientNetB3	98.80%
MobileNetV2	99.46%
MobileNet+Grad-CAM	99.46%

Figure 6: *Classifying accuracy for all tested models.*

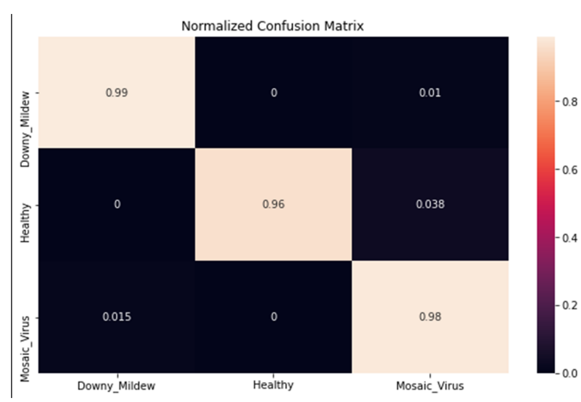

Figure 7: *MobileNetV2 Confusion Matrix*

In the above figure of MobileNetV2 confusion matrix we can see that our trained model can predict Down_Mildew 99%, Healthy 96%, and Mosaic_Virus 98% of the time correctly.

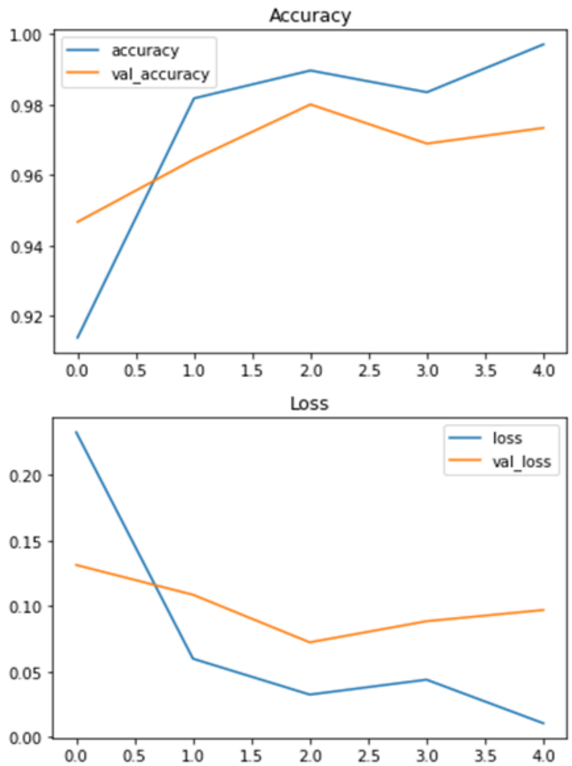

Figure 8: *MobileNetV2 Accuracy and Loss.*

From the MobileNetV2 accuracy and loss graphs we can see that our MobileNetV2 model's accuracy and val_accuracy finally set to 99.46% and 96.55% respectively, and 0.0180 and 0.1023 for loss and val_loss respectively .

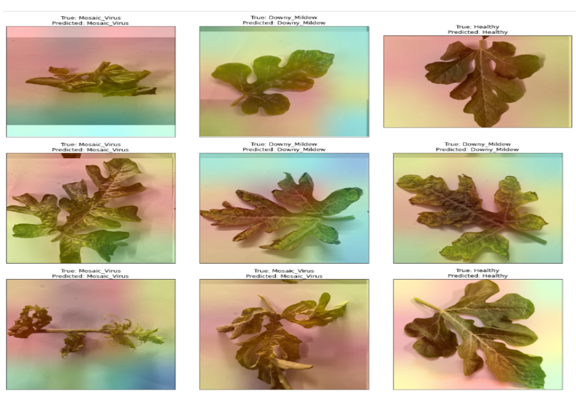

Figure 9: *MobileNetV2+GradCam.*

Here from the above figures we can see that using GradCam with our trained MobileNetV2 model we can understand where our model is actually focusing while predicting classes from testing images.

VI. Limitation

The dataset that we used had a limited amount of original data, even though we used few augmentation methods to increase the amount so that we can properly train our models, it would've been better if we had more original data. Additionally, the dataset only has three leaf categories, more categories would allow us to experiment with the models better and test the transfer learning capabilities.

VII. Future Work

In our work, we have only added one optimization algorithm as the optimization algorithms work differently according to the models and we did not dive further into those categories and compare them. That is a huge amount of work as a low performing model can give better results according to the optimization algorithm. And finally, we can also try to add different leaf dataset and try to implement transfer learning on them which has pre-learned memories from this experiment.

VIII. Conclusion

In our study, we did compare different transfer learning models for detecting watermelon leaf disease. Our chosen models consist of both light and heavy resource demanding ones too. We also proposed a combination of MobileNet and Grad-CAM to better improve the efficiency of the task in hand. From our findings we can conclude that MobileNet being a light model can be run on various categories of devices and its accuracy is also above par and if run in combination with Grad-CAM the effectiveness increases as well. This technology can be easily modified to not only detect watermelon leaf disease but any and all kinds of plant disease in the future. This in return will greatly improve our agriculture fields output and help thousands of farmers across the globe to better treat their crops. For future work, this technology can be embedded into mobile applications for ease of access so anybody can capture an image to test and find if their crops are unhealthy and what sort of remedy to use. We believe that automation of this technology will surely help society.

References

[1] Orchi, H., Sadik, M., Khaldoun, M., & Sabir, E. (2023). Automation of Crop Disease Detection through Conventional Machine Learning and Deep Transfer Learning Approaches. Agriculture, 13(2), 352. https://doi.org/10.3390/agriculture13020352

[2] Uğuz, S., & Uysal, N. (2020). Classification of olive leaf diseases using deep convolutional neural networks. Neural Computing and Applications, 33(9), 4133–4149. https://doi.org/10.1007/s00521-020-05235-5

[3] Aarizou, A., & Merah, M. (2022). Transfer learning for plant disease detection on complex images. 2022 7th International Conference on Image and Signal Processing and Their Applications (ISPA). https://doi.org/10.1109/ispa54004.2022.9786306

[4] Arshad, M. S., Rehman, U. A., & Fraz, M. M. (2021). Plant disease identification using transfer learning. 2021 International Conference on Digital Futures and Transformative Technologies (ICoDT2). https://doi.org/10.1109/icodt252288.2021.9441512