IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Plant Disease Classification Using Deep Learning Resnet Algorithm

A. Ishwariya [1], B. Sujitha [2], K. Suvetha [3], J. Vidhubala [4]

[1] Assistant Professor, [2, 3, 4] B.E. Student

Department of Computer Science and Engineering, P.S.R. Engineering College, Sivakasi, TamilNadu, India.

Abstract

Plant diseases that harm the leaves of the plants halt the growth of the plants themselves. Plant illnesses that are detected earlyand accurately may lessen the chance that the plant will sustain additional damage. The fascinating strategy required more effort, focus, and expertise. Plant leaf diseases are identified using images of the leaves. Deep learning (DL) research seems to offer a lot of promise for increasing accuracy. The significant progress andexpansions in deep learning have made it possible to enhance the accuracy and coordination of the system for recognizing and valuing plant leaf diseases. It is a cutting-edge deep learning method for classifying diseases. An investigation was conducted into the efficacyof diagnosing diseases in plant leaves. The ResNet classifier is used to remove color, texture,and plant leaf arrangement geometry from the given photos. A few efficacy metrics demonstrate that the suggested strategy outperforms current methods with an accuracy rate concert measurements are employed to carryout these procedures. These metrics are used for analysis and to propose a suggested way. The phases of disease detection involve the following steps: image separation, categorization, noise removal, and picture collection.

Key words: Residual Network, Biodiversity, Plant Species, New Plant Disease dataset

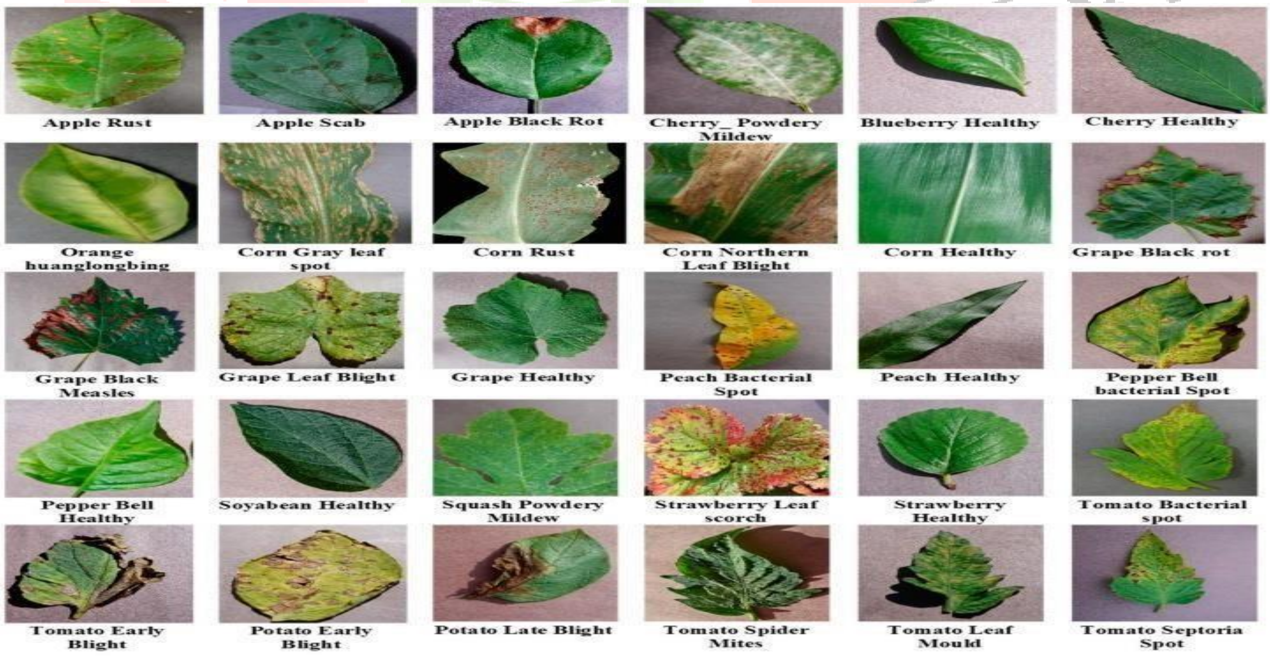
1. Introduction

Farmers and agronomists frequently deal with plant diseases, which are a decline in a plant'sinitial state that disturbs and modifies its essentialfunctions. Plant diseases can be caused by a number of factors, including the surrounding crops, the presence of pathogens, and the environmental circumstances. The physical characteristics of plants, such as changes in color,size, or shape, can be used to identify symptoms during the early stages of the majority of diseases.

Plant diseases are known to seriously impede a plant's growth and to be one of the variablesinfluencing the security of food supplies. Therefore, it is crucial to identify quickly and accurately in order to stop the spread of disease. Nevertheless, the process's intricacy hinders the identification process. The multiplicity of symptoms associated with some plant diseases makes it difficult for even seasoned agronomists and plant biologists to distinguish between them. where an incorrect diagnosis results in insufficient or ineffective therapy. Farmers and agronomists will have less work to do thanks to this level of automation. Machine learning algorithms can now run smoothly on servers thanks to the exponentialincrease in processing power over the last severalyears, especially in the graphics processing unit (GPU), which allows processes to be computed inparallel. Furthermore, developers now have a newplatform to speed compute-intensive applications thanks to the introduction of GPU platforms like Nvidia's Compute Unified Device Architecture(CUDA). This is achieved by making optimal useof the GPU's processing capability for the execution of intricate algorithms like gradient descent, back-propagation, and two gradient descent. Technological developments have made deep learning techniques generally applicable in a widerange of fields and applications. Neural network architectures are combined to generate the deep learning model. A deep learning model may havehundreds of hidden layers, as opposed to the few hidden layers found in a residual neural network architecture. This is among the factors that allow deep learning to generally outperform traditional techniques in terms of accuracy.

One of the deep learning models is the Residual Neural Network (ResNet34), which is distinguished by its ability to extract features forpicture recognition and classification. The ResNet34 model can perform a wide range of operations on images depending on its design. Inaddition to deep learning, scientists have been working on various machine learning algorithmsto find ways to identify plant illnesses. We think that the concept of edge computing—in which the microprocessor functions as an AI acceleratorto execute ResNet34 for image classification tasks on the edge—can be enabled by the emergence of microprocessor classes like theTensor Processing Unit (TPU) and the Vision Processing Unit (VPU). The VPUs and TPUs typically come in the shape of USB sticks or drives that work with a wide range of devices and systems. Programming libraries or specialized development kits can be used to integrate and implement deep learning models to the edge withease on VPUs.

We chose Google Colab for the project, which gives us the ability to leverage the model optimizer function to apply the best-performing models on peripheral devices. The created model can be modified using the model optimizer function to run as efficiently as possible on Intel-based hardware, including CPU, GPU, FPGA, andVPU. Next, the model was assessed via the Google Colab platform. The well-known New Plant Diseases dataset, which is frequently used toassess the effectiveness of multi-plant disease detection, was used to prepare and assess these four models. By assessing the model's performance across various hardware andsoftware configurations, we may determine whether executing the model on the edge is feasible. The performance of the top-performingmodel under various setups is compared and reported.



Sample images from New Plant Disease dataset form 38 types of leaf diseases.

2. Related Work

The authors presented a deep learning model called EfficientNet to be used in the classification of various plant diseases. The Plant Village dataset, which has 55,448 photos, and its expanded version, which contains 61,486 images, were used to train the model. The effectiveness of the suggested EfficientNet model is contrasted with that of various cutting- edge CNN models, including VGG, AlexNet and Inception. Every layer in transfer learning was configured to be trainable. The Efficient NetB5 and B6 models produced the best results out of all the models. These days, there are a lot of methods that are helpful for phishing attempts.

1. To categorize various plant leaf diseases, M. Akila and P. Deepan employed Single Shot Multibox Detector (SSD), Faster Region-based CNN (Faster R- CNN), and Region-based Fully Convolutional Network (R-FCN). Images of plant leaf diseases on commercial crops such bananas, sugarcane, cotton, potatoes, brinjal, carrots, chilly, rice, wheat, and guava are included in thedataset. Both personally taken photos and photographs from the internet were gathered. To increase the dataset and stop the model from overfitting, image augmentation techniques such rotations, affine and perspective transformations, and image intensity modifications were used.
2. To categorize plant diseases, the authors employed the Caffe deep learning system with Imagenet weights. Eight learning layers, five convolution layers, and five fully connected layers make up the model. To prepare the dataset, pictures were downloaded from the internet. To expand the dataset and avoid overfitting during the training phase, image augmentation was done. The accuracy of the suggested Caffe model was 96%. By utilizing image processing techniques to target the disease-affected areas, plant diseases can be detected.
3. To divide the leaf and lesion areas, respectively, Sanjay and Shrikant employed triangle thresholding techniques and basic threshold approaches. Bycalculating the area of the leaf and the lesion, five diseases are identified. The accuracy of the suggested system was 98.60%.
4. With an overall accuracy of 95%, Revathi and Hemalatha classified several cotton leaf diseases using a Cross Information Gain Deep Forward Neural Network and particle swarm optimization (PSO) for feature extraction. Particles warm optimization is used to extract the color, shape, and texture information in order to identify various diseases. The feature extraction technique improves the overall accuracy of the model and aids in locating the diseased leaf patches.
5. Kulkarni and Patil presented a system that makes use of artificial neural networks (ANN) and image processing techniques to detect illnesses in plant leaves. The Gabor filter was used to filter and segment the images. When the ANN model was trained using the collected characteristics, it was able to categorize the healthy and diseased samples with a 91% accuracy rate.

3. Data collection: To collect data for a deep learning plant disease classification project, you can follow these steps:

a) Identify the Plant Species: Select the plant species you wish to concentrate on when classifying diseases. Indicate your aim because different plants may have different illnesses.

b) Find Datasets: Look for datasets that already exist that include pictures of both healthyand sick plants. Academic institutions, plant research organizations, and online repositories like GitHub or Kaggle are a fewhelpful sources. Make sure the disease categories are labeled on the datasets.

c) Data Augmentation: You might need to add to the current dataset if you are unable to locate enough data. To create more varied photos, data augmentation techniques including rotation, scaling, cropping, and flipping are used.

d) Collect Your Own Data: If appropriate datasets are hard to come by, think about gathering your own. Take pictures of plants in different environments and describe them appropriately. Assure consistency and high image quality.

e) Annotate Data: Put the appropriate ailment or state of health next to each image. While it can takesome time, this is a necessary step in the training of any deep learning model.

4. **Data preprocessing:** Your photosmust match the network's input size in order for it to be trained and trained on new data to produce predictions. You can rescale or trim your data to the necessary size if you need to change the size of your photographs to fit the network. By implementing randomized augmentation on your data, we can efficiently expand the quantity of training data.

You can also train networks to be invariant to distortions in picture data by using augmentation. To make a network invariant to the presence of rotation ininput images, for instance, you can add randomized rotations to the input images. A convenient technique to apply a small set of augmentations to 2-D images for classification issues is through an augmented Image Datastore. You can begin with a built-in datastore for more complex preprocessing procedures, to preprocess photos for regression problems, or to preprocess 3-D volumetric images. With the help of the combine and transform functions, you can also preprocess photos in accordance with your own pipeline.

5. **Feature Extraction:** The dimensionality reduction method, which divides and reduces an initial collection of raw data into more manageable groups, includes feature extraction. It will therefore be simpler to process when you want to.These huge datasets abundance of variables is by far their most significant feature. Processing these variables takes a lot of computer power.

6. **Data splitting:** Models are trained and developed using the training data set, which is a two-part data split. Training sets are frequently used to compare the performance of several models or estimate various parameters. After training is complete, the testing data set is used. The final model's functionality is verified by comparing the test and training sets of data.

7. **Training the model**: Learning RateScheduling: A learning rate scheduler will be used in place of a set learning rate, and it will adjust the learning rate following each training batch. The "One Cycle Learning Rate Policy" is one of several strategies available for adjusting the learning rate during training. It entails beginning with a low learning rate, progressively raising it batch by batch toa high learning rate for roughly 30% of the epochs, and then gradually lowering it to a very low value for the remaining epochs. Weight Decay: Another regularization strategy we employ is weight decay, which adds a term to the loss function to keep the weights from growing too large. Gradient Clipping: Inaddition to layer weights and outputs, limiting gradient values to a narrow range might assist avoid unintended parameter changes brought on by high gradient values. We refer to this easy-to-use yet powerful method as gradient clipping.

8. **Model selection:** Rather of learning unreferenced functions, Residual Networks, orResNets, learn residual functions with reference to thelayer inputs. Relative nets allow these stacked layers to match a residual mapping rather than assuming that each one of them directly fits a specified underlying mapping. To create a network, they build residual blocks on top of one another. For instance, a ResNet-34 uses fifty layers by stacking these blocks. Formally, we allow the stacked nonlinearlayers match another mapping of F(x): = H(x)−x, where H(x) is the desired underlying mapping. We recast the original mapping into F(x)+x. Empirical data suggests that these networks may be optimized more easily and can achieve significantly higher accuracy levels with more depth.

9. **Performance evaluation:** Evaluationcriteria including accuracy, precision, recall, F1-score, and confusion matrix can be used to evaluate the model's performance. A more reliable estimate of the model's performance can be obtained by cross- validation.

Accuracy: It is a metric for determining the percentage of correctly classified results out of all the cases.

$$\text{Accuracy (in Percentage)} = (TN + TN/TP + TN+ FP +FN) \times 100$$

Precision: This calculates the percentage of accurately anticipated positive rates among all expected positive rates. A high precision classifier is one with a Precision rating of 1.

Precision = TP/ TP + FP

Recall: The recall rate is actually positive. A recall of 1 is considered to be a competent classifier.

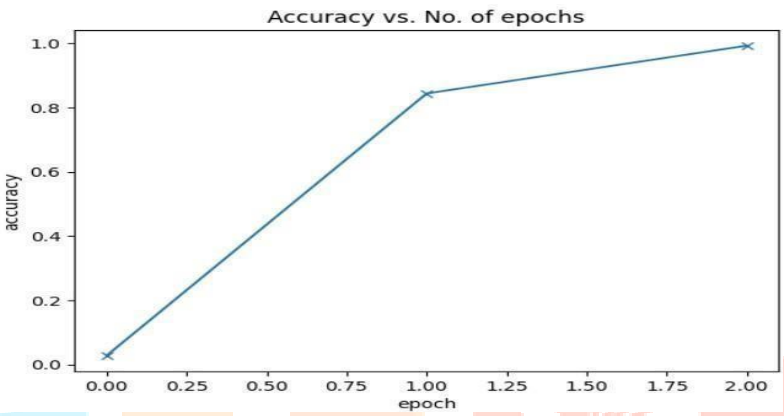
Recall(inPercentage)=TP/TP+FN

F1 Score: It is a metric that takes the dimensions of Precision and Recall into account. Only when both measures, such as recall and precision, are 1, does the F1 score increase to 1.

F1Score (inPercentage) =2*Precision*Recall /Recall + Precision

The conversions of the True Positive (TP),False Positive (FP), True Negative (TN), and False Negative (FN) metrics are the most widely used metrics.

10. Validation accuracy



11. Epochs accuracy

```
epochs = 2
max_lr = 0.01
grad_clip = 0.1
weight_decay = 1e-4
opt_func = torch.optim.Adam
```

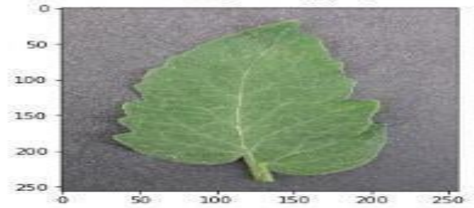
```
%%time
history += fit_OneCycle(epochs, max_lr, model, train_dl, valid_dl,
                        grad_clip=grad_clip,
                        weight_decay=1e-4,
                        opt_func=opt_func)
```

```
Epoch [0], last_lr: 0.00812, train_loss: 0.7466, val_loss: 0.5865, val_acc: 0.8319
Epoch [1], last_lr: 0.00000, train_loss: 0.1248, val_loss: 0.0269, val_acc: 0.9923
CPU times: user 11min 16s, sys: 7min 13s, total: 18min 30s
Wall time: 19min 53s
```

12. Training the images

```
show_image(*train[0])
Label :Apple___Apple_scab(0)
```

```
show_image(*train[70000])
Label :Tomato___healthy(37)
```



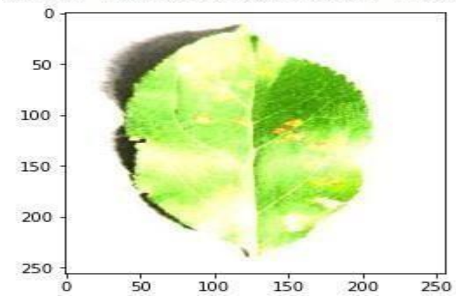
13. Simulation results

In order to diagnose plant illnesses, one section of the study used cutting edge deep learning models that applied the transfer learning methodology. The pretrained Resnet 34 networks, which had previously been trained with the Image dataset, were further trained using the publically accessible New Plant Diseases Dataset. Three samples were taken from the dataset: training, test, and validation. Eighty percent of the samples in the New Plant Diseases Dataset were utilized to train the Inception ResNetmodels before they were fully trained. Every model was ran for two epochs, and it was discovered that our model began to converge with good accuracyafter one epoch.

High yields in agricultural production depend onthe early detection of crop disease. The most recent technological advancements should be used in the early detection of plant disease in order to maintain a high output rate. The literature review revealed that transfer learning-based models are effective in removing training complexity and large dataset requirements, while deep learning models are effective in classifying images. To ascertain which of the four pre-trained models ResNet-34 was most effective in diagnosing different plant diseases, weassessed them in this work.

Testing the images

```
# predicting first image
img, label = test[0]
plt.imshow(img.permute(1, 2, 0))
print('Label:', test_images[0], ', Predicted:', predict_image(img, model))
Label: AppleCedarRust1.JPG , Predicted: Apple___Cedar_apple_rust
```



6. Conclusion

A transfer learning approach was explored to develop an edge computing solution for plant disease classification. Four pre-trained models supported by Google Colab were studied as the feature extraction tools. Image augmentation and down sampling techniques were applied to address the imbalance data problem. These four models were experimented with a dataset having a total of 52,837 images with 35 classes of healthy and diseases plant leaves. From the experiment, it was shown that the ResNet 34 was the best performing model out of the four models with the highest validation accuracy of 99.2% with is comparable with other reported methods. The model was then converted to different model formats using the Intel Model Optimizer. The new formats were required to run the inference on different Intel hardware (CPU, GPU and VPU). Results shows that there were no significant differences on the macro averaged metrics, even though the model is operated different configurations. The developed model was able to maintain relatively high recall values, precision and F1 scores. Conversely, we can observe a significant difference in inference time when time when the model on CPU but under two different environments. The shortest inference time was observed when the model operates on GPU mode. Lastly, result of operating the model on Intel NCS2 (VPU) with Google Colab go beyond the result of operating the model on CPU mode in the Google Colab environment in terms of inference time and accuracies. This indicates the compact capabilities of the model and the possibility to implement it at the edge.

7. Future Enhancement

Train the system on a larger dataset of plant images. This would improve the accuracy of the system, especially for species that are not well- represented in the current dataset. Adapt the system to classify other types of plants, such as flowers, herbs, and trees. This would make the system more versatile and useful for a wider range of applications. Develop a mobile app for the system. This would make it easier for people to use the system to identify plants in the field. Integrate the system with other systems, such as agricultural databases. This would allow the system to be used to develop new applications, such as systems for predicting crop yields or identifying pests and diseases.

References

[1] Mrunalini R. et al., An application of Kmeans clustering and artificial intelligence in pattern recognition for crop diseases ,2011.

[2] S.Raj Kumar , S.Sowrirajan," Automatic Leaf Disease Detection and Classification usingHybrid Features and Supervised Classifier", International Journal of Advanced Research inElectrical, Electronics and InstrumentationEngineering, vol.5, Issue 6,2016.

[3] Tatem, D. J. Rogers, and S. I. Hay, "Global transport networks and infectious disease spread," Advances in Parasitology, vol. 62, pp. 293– 343,2006. View at Publisher · View at Google Scholar · View at Scopus.

[4] J. R. Rohr, T. R. Raffel, J. M. Romansic, H. McCallum, and P. J. Hudson, "Evaluating thelinks between climate, disease spread, andamphibian declines," Proceedings of theNational Academy of Sciences of the United States of America, vol. 105, no. 45, pp.17436– 17441, 2008. View at Publisher ·View at Google Scholar · View at Scopus.

[5] L.C. Ngugi, M. Abelwahab, M. Abo-Zahhad,Recent advances in image processing techniquesfor automated leaf pest and diseaserecognition – a review, Inform. Process. Agric. 8 (1) (2021) 27–51, https://doi.org/10.1016/j.inpa.2020.04.004.

[6] T.F. Gonzalez, Handbook of approximationalgorithms and metaheuristics, HandbApproximat Algorithm Metaheuristics (2007)1–1432, https://doi.org/10.1201/9781420010749.

[7] Srdjan Sladojevic, Marko Arsenovic, Andras Anderla, Dubravko Culibrk, and DarkoStefanovic, "Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification," Hindawi Publishing Corporation Computational Intelligence and Neuroscience, vol. 2016, ArticleID 3289801,2016

[8] S. B. Patil and S. K. Bodhe, "Leaf disease severity measurement using image processing," International Journal of Engineering and Technology, vol.3, no.5, pp.297–301, 2011.

[9] P. Revathi and M. Hemalatha, "Identification of cotton diseases based on cross information gain

deep forward neural network classifier with PSO feature selection," International Journal of Engineering and Technology, vol.no.5, 6, pp.4637–4642, 2014.