Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Plant leaf disease detection using deep learning

TECHNICAL ANSWERS FOR REAL WORLD PROBLEMS

ITE1901

TEAMATES:

KAVALI VENKATA SAI SANKAR KUMAR 19BIT0030

VARUN KUMAR 19BIT0058

SEJAL BHARATI 19BIT0286

AKASH GUPTA 19BIT0298

YASH KHANDELWAL 19BIT0345

SOMYA KHARTI 19BIT0365

Introduction

1. The Indian food industry has already expanded its commitment to the world food exchange because of its huge potential.
2. In India, approximately 15-25% of potential yield is lost due to various diseases in crops.
3. To tackle all these problems by the means of this project, we will ensure early detection of diseases in a crop so that we can provide the possible remedies to aid the farmer.

Abstract

In this paper we have done extensive literature survey and came up it several approaches to tackle this problem

1. First we explored **transfer learning** using resent, alexnet and vgg models, where alexnet performed the best with 92.02% accuracy.
2. Later we implemented **ensemble** which gave better performance than individual models with an accuracy of 94.52%.
3. We implemented **Custom EfficientB0 Architecture** which performed the best when compared to transfer learning and ensemble with an accuracy of 98.03%.
4. We implemented **Multipath CNN InceptionV3 variation** which gave the highest accuracy of 99.11%

HARDWARE/SOFTWARE USED

Parameter	Google Colab	Kaggle Kernel
GPU	Nvidia K80	Nvidia P100
GPU Memory	16GB	16GB
GPU Memory Clock	1.59GHz	1.32GHz
Performance	8.1 TFLOPS	9.3 TFLOPS
Support Mixed Precision	Yes	No
GPU Release Year	2018	2016
No. CPU Cores	2	2
Available RAM	26.75GB	12GB
Disk Space	358GB	5GB

FINDINGS FROM LITERATURE SURVEY

- In the literature survey we have analysed different approaches for the plant disease prediction. The machine learning Different deep learning models like VGG16, ResNet50, Alexnet, InceptionV3, InceptionResNet, DenseNet, RPN, Chan–Vese and machine learning algorithms like LSTM, SVM.
- The deep learning models achieved a higher accuracy than the machine learning algorithms.
- From the literature survey we also found that ensemble gives better accuracy compared to individual models.
- We also came across many custom models to achieve better accuracy or fix the issues with data discrepancy.

GAPS IN LITERATURE SURVEY

- The minor features aren't getting captured which is affecting the accuracy a lot.
- The time required for modeling with deep learning algorithms will be longer.
- Noisy data is effecting the accuracy
- Though models are having higher accuracy they aren't performing well on real-time environment due to lot of noise.

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

DATA DESCRIPTION

- Images were used from PlantVillage dataset from Kaggle
  https://www.kaggle.com/emmarex/plantdisease
- Contains 38 labels, 54306 images, 26 diseases, 14 crop species
- Chosen 5 classes of leaves containing 15110 images
  - Multiple Diseases
  - Healthy
  - Rust
  - Scab
  - Blackrot

DATA PREPROCESSING

To exhibit greater dataset size, several parameters such as rotation range, fill mode, batch size, rescale, width shift range, height shift range, shear range, and zoom range will be configured.

transforms.compose	It just clubs all the transforms provided to it. So, all the transforms in the transforms.Compose are applied to the input one by one.
transforms.RandomResizedCrop	Crop a random portion of the image and resize it to a given size.
transforms.Normalize	Normalizes the tensor image with mean and standard deviation.
transforms.ToTensor()	Convert a PIL Image to tensor
transforms.CenterCrop	Crop the center portion of the image and resize it to a given size
transforms.Resize	We Have Resized the input image to the given size.

DATA SPLITING

- The Training and testing data has been split in 80:20.
- And out of those 80 percent its again split in 75:25 for training and validation respectively.
- So overall its split into 60:20:20 for training, validation and testing.

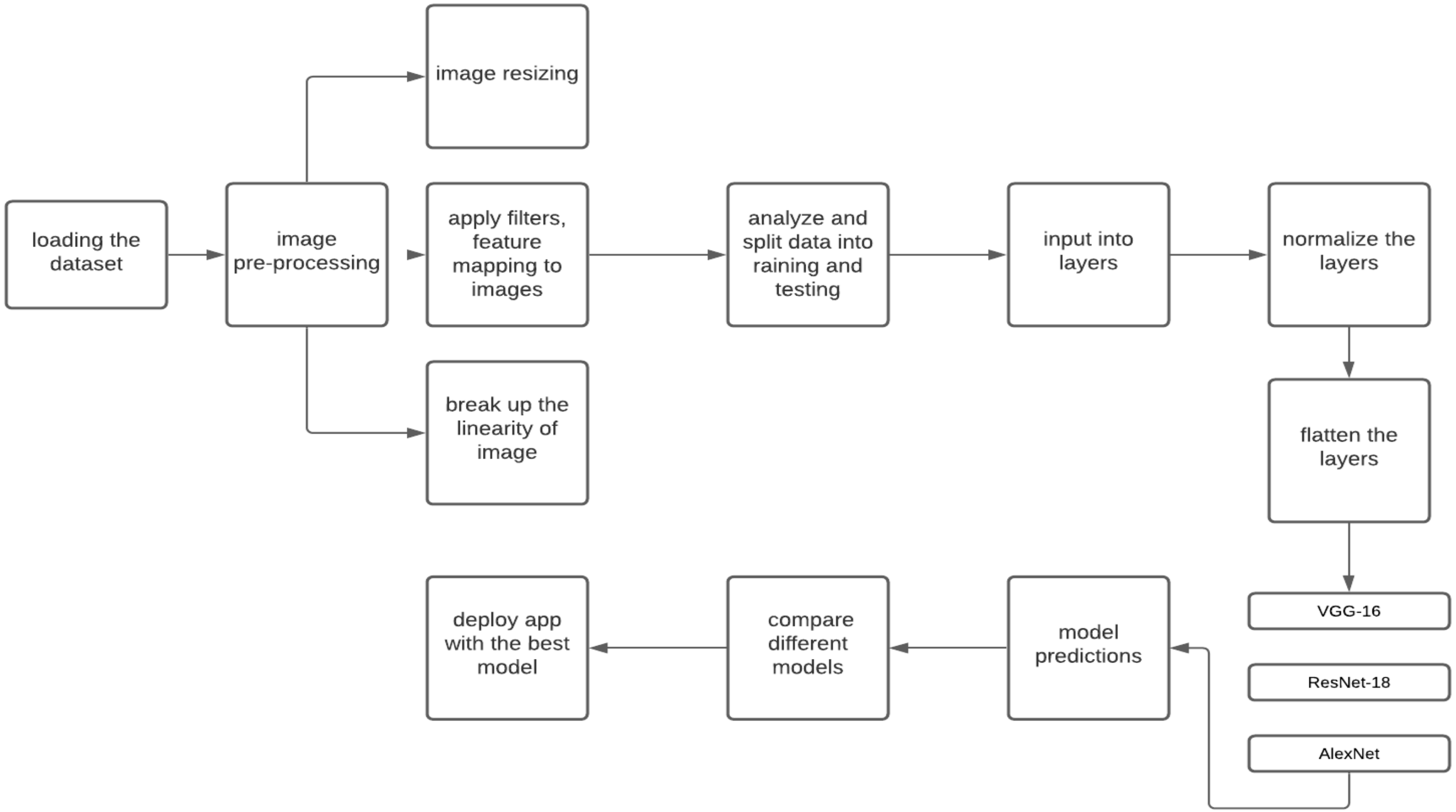
TRANSFER LEARNING

Approach#1

TRANSFER LEARNING

Transfer Learning is a well-known deep-learning technique that allows a model trained for one use case to be reused for a different job.

For the execution of this project, we will use pre-trained weights from the VGG-16, Alexnet, Resnet18 architectures.



TRANSFER LEARNING ARCHITECTURE

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

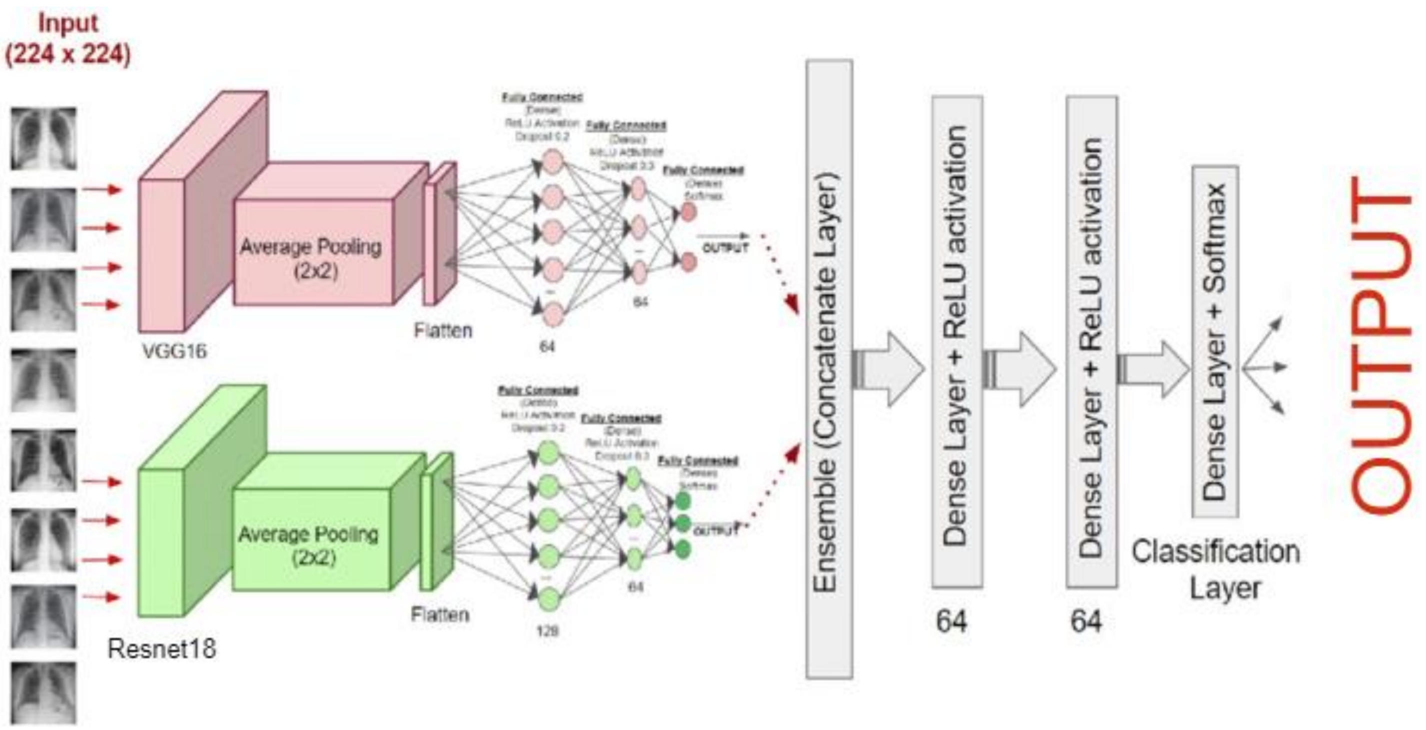
ENSEMBLE

Approach#2

ENSEMBLE

Ensemble modeling is a process where multiple diverse models are created to predict an outcome, either by using many different modeling algorithms or using different training data sets.

Ensembling of VGG16 and Resnet18 has been done, which showed us better results than individual ones.


Input
(224 x 224)
Average Pooling
(2x2)
VGG16
Flatten
64
Average Pooling
(2x2)
Resnet18
Flatten
128
Ensemble (Concatenate Layer)
Dense Layer + ReLU activation
64
Dense Layer + ReLU activation
64
Dense Layer + Softmax
Classification
Layer
OUTPUT

CUSTOM EFFICIENT B0

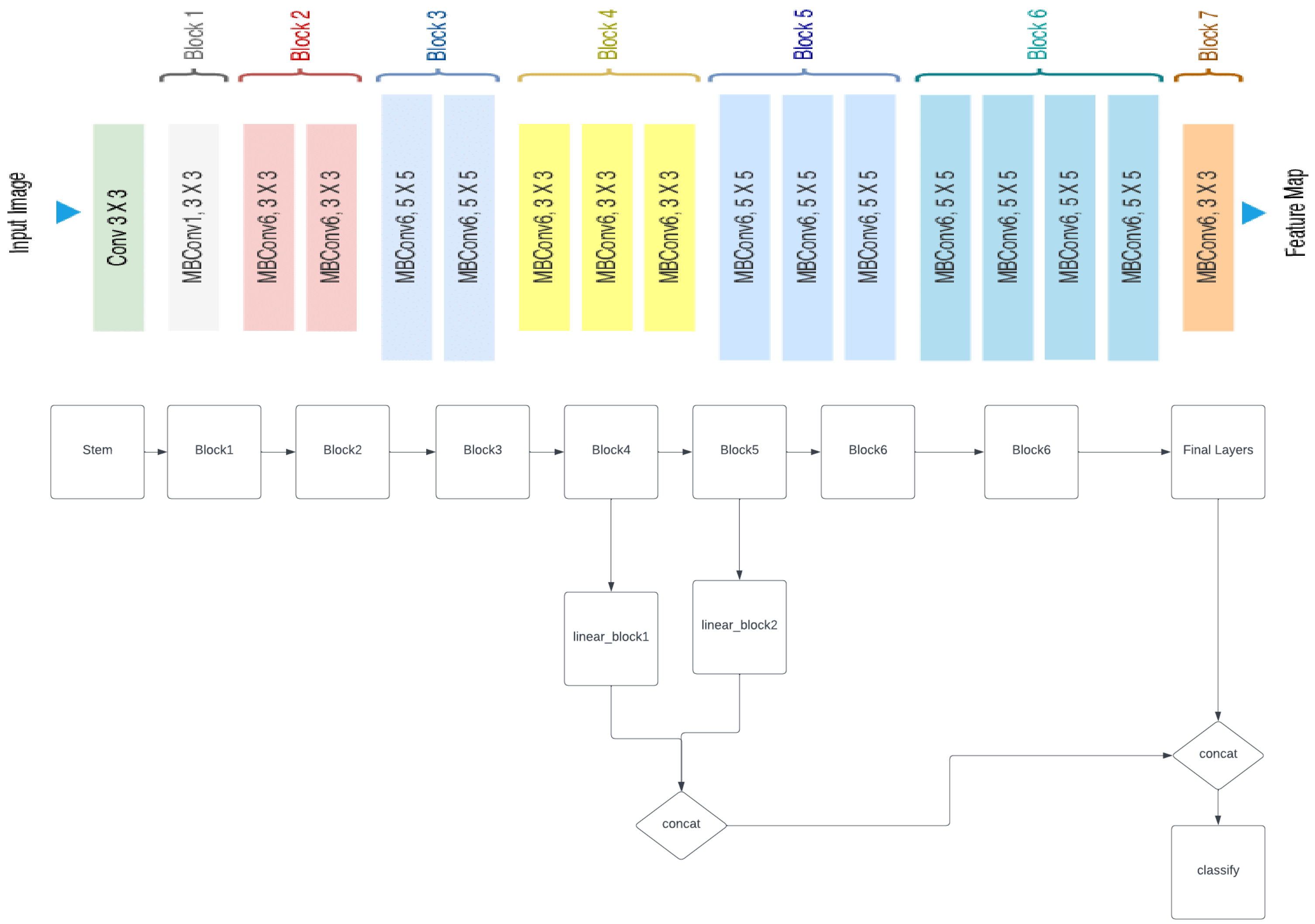
Approach#3

CUSTOM EFFICIENT B0

The EfficientNet-B0 architecture wasn't developed by engineers but by the neural network itself.

Taking B0 as a baseline model, the authors developed a full family of EfficientNets from B1 to B7.

In the same way we too developed a custom efficient architecture from efficient b0.



CUSTOM EFFICIENT B0

VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

COMPARISION

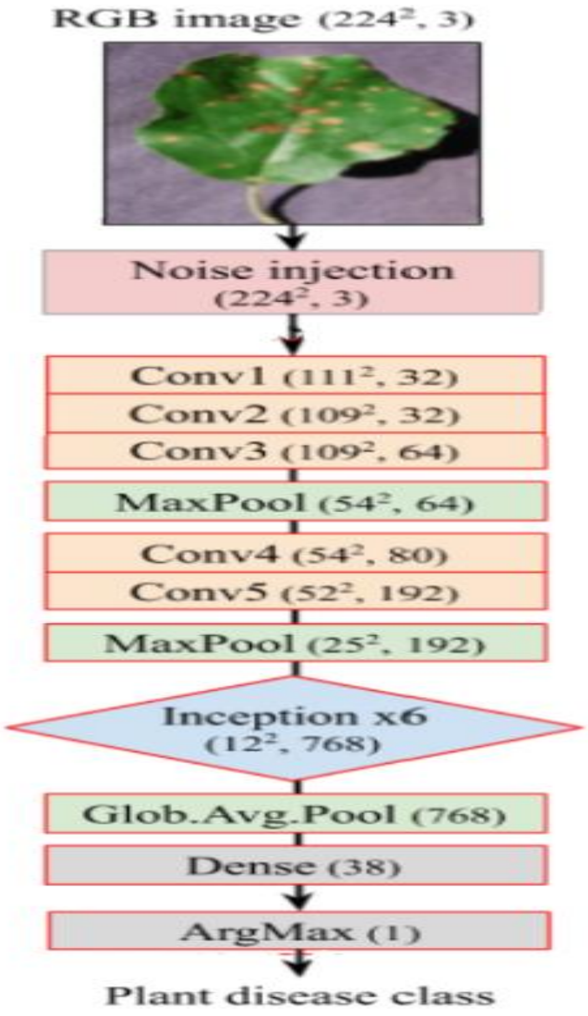
Model	Training		Validation	
	Accuracy %	Loss	Accuracy %	Loss
Alexnet	92.02	0.3228	96.08	0.1280
Resnet18	86.63	0.3831	96.47	0.1147
VGG16	89.70	0.2947	97.25	0.0873
ENSEMBLE	94.52	0.1603	98.04	0.0566
Custom Efficient B0	98.03	0.7165	97.72	0.0797

MULTIPATH CNN INCEPTION VARIATION

Approach#4

Baseline Model

- Toda Y, Okura F. How Convolutional Neural Networks Diagnose Plant Disease.
- Inception v3 architecture
- Plant Village Dataset
- 5 convolutional layers
- 95.06% accuracy



BASELINE MODEL

Drawbacks in baseline model

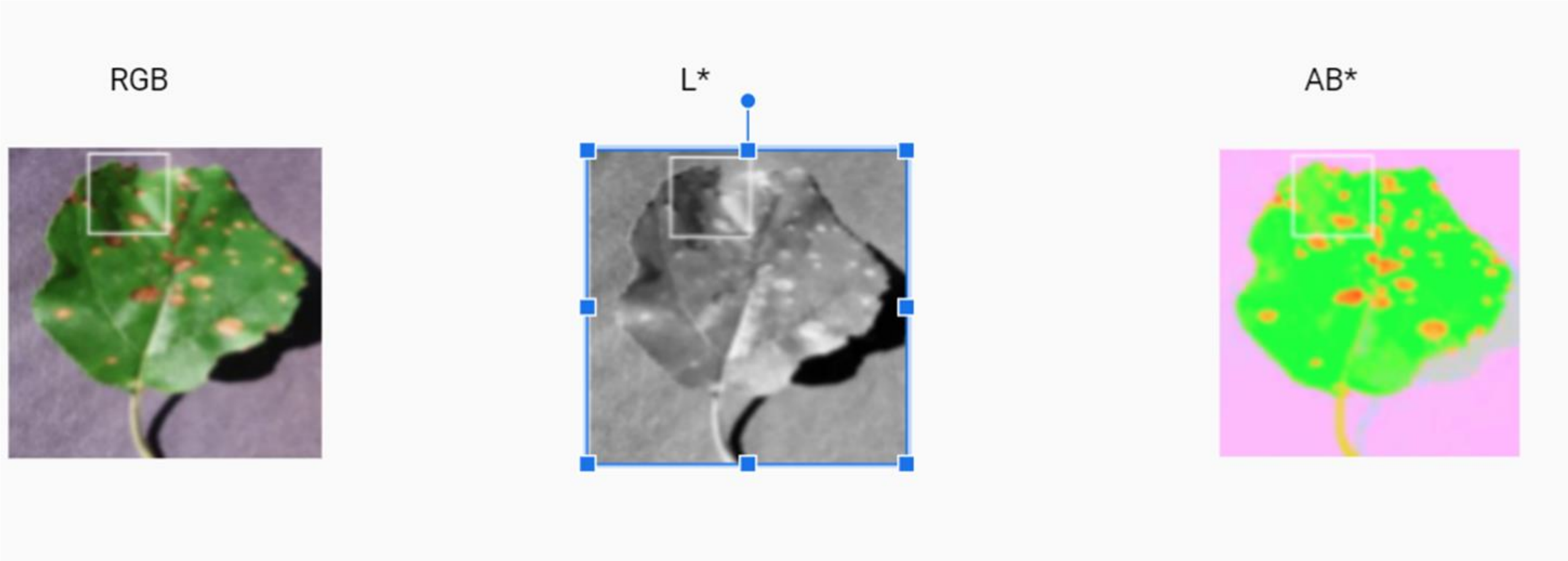
- RGB image is used.
- No separation of the lightness and the color components.
- Poor performance when leaf is physically damaged.
- Cannot handle images containing blur, motion blur, occlusion and other defects.

Proposed Methodology

- Inception V3 variation
- Multipath CNN with 2 branches
- Input RGB image converted to L* and AB* components
- 1 branch is fed with achromatic L-channel
- Other branch is fed with AB-channel
- Provides resistance to adverse noise effects on the data

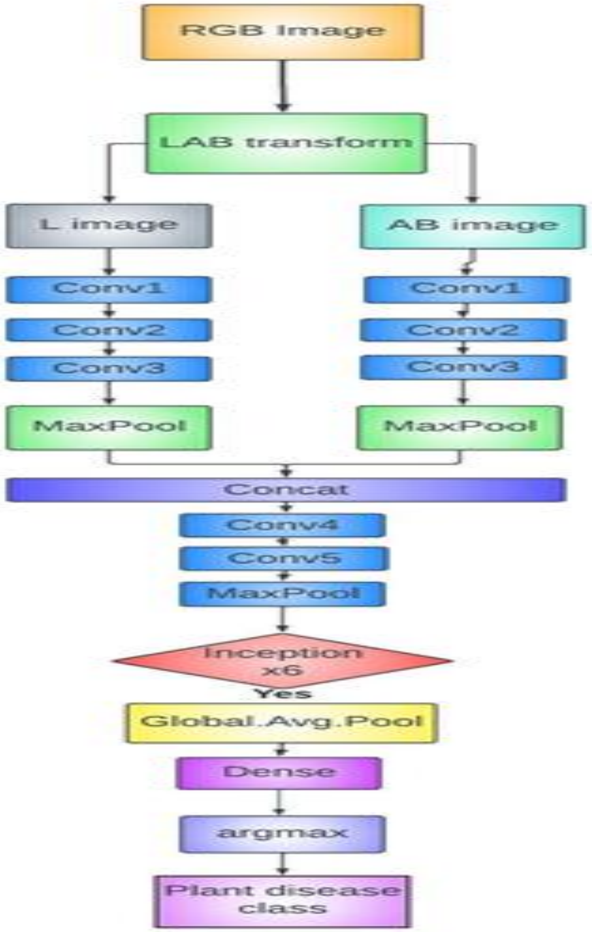
RGB v/s LAB

- RGB and LAB are 2 different color spaces
- RGB operates on 3 channels, red, green and blue
- LAB is a conversion of the same RGB image into a lightness component L* and a color component AB*
- So we are able to separate the lightness component from the color component.



RGB VS LAB

Proposed Architecture



Advantages of Proposed Methodology

- Effective isolation of lightness and color components
- Improved accuracy
- Resistance to noisy data
- Resistance to physically damaged leaves

Experimental Setup

- Our code was coded with Keras/Tensorflow
- Each convolutional layer is composed of a 2D convolution, a batch normalization and a ReLU activation function
- convolutional filters from Conv1 to Conv5 are of the size 3 × 3
- Conv4 filter is of size 1 × 1
- optimization method is stochastic gradient descent

- loss function is weighted categorical cross entropy
- batch size is 32
- Model is trained for 30 epochs
- 3 variations of 2 path CNN have been tried
    - 20%L-80%AB
    - 50%L-50%AB
    - 80%L-20%AB

Results

- Accuracy for 20%L-80%AB is 99.48%
- Accuracy for 50%L-50%AB is 99.11%
- Accuracy for 80%L-20%AB is 99.08%
- Proposed model outperforms baseline model
- 20%L-80%AB shows the best performance
- Model shows similar accuracy for noisy data
- Model is resistant to physically damaged leaves

Web application

Using Flask API

WEB APPLICATION

- Two paths inception model has been exported and used in the web application since it works good on noisy data too.
- Flask api has been used to create endpoints for uploading image and displaying the result.
- The result contains the disease (class label) of the leaf and remedies to mitigate the disease.
- HTML, CSS and flask template engine has been in the frontend.

CONCLUSION

1. Our project's major goal is to recognize and determine whether a leaf is sick or healthy.
2. We used pre-trained models such as VGG-16, AlexNet and ResNet18 to improve the model. The ImageNet dataset's pre-trained weights will assist us adjust for the restricted amount of photographs per class as well as the computing requirements (Transfer Learning). This gave us fair accuracy.
3. To improve it more we also tried ensemble of resnet and vgg16 which increased the accuracy from 92 to 94%
4. We further developed the model so it does not suffer from considerably high variance or excessive bias using custom efficient b0 which showed a great performance with an accuracy of 98.04%.
5. To tackle with the noisy environment in the real world we implemented multi path CNN inception variation which gave same amount of accuracy in both noisy as well as normal data i.e., 99.48%

FUTURE WORKS

1. Use distributed system like Apache spark, Hadoop to decrease the processing time as well as training time.
2. Develop an IOS/Android app along with the existing web application.
3. Along with making the models perform better on noisy data, we should also try to remove the noise from data before processing.

Thank you

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)