Model	Accuracy
Efficientnet_b4 - for 4 classes(normal, rot, bloatch, scab)	C> 100% 8/8 [00:15<00:00, 1.93s/ft] Test Loss: 0.6401390954852104 Test Acc: 0.7578125
epochs =20 Ir = 0.001	
Efficientnet_b4 - for 4 classes(normal, rot, bloatch, scab)	100% 8/8 [00:01<00:00, 5.41it/s] Test Loss: 0.5483963638544083 Test Acc: 0.7734375
epochs =200 Ir = 0.00001	
Efficientnet_b4 - for 2 classes(normal, defected)	<pre>v [105] model.load_state_dict(torch.load('/content/ColabPneumoniaModel.pt')) model.eval() avg_test_loss, avg_test_acc = trainer.valid_batch_loop(model,testloader)</pre>
epochs =250 Ir = 0.00001	<pre>print("Test Loss : ()".format(avg_test_loss)) print("Mest Acc : {)".format(avg_test_acc))  100%  Test Loss : 0.3484538018703461 Test Acc : 0.8395832777023315</pre>
Efficientnet_b4 - for 2 classes(normal, defected) - More data - merged data (2 classes + 4 classes)	C* 100% 17/17 [00:02<00:00, 7.71it/s]  Test Loss: 0.25881360076806126  Test Acc: 0.8918067216873169
epochs =200 lr = 0.00001	
Efficientnet_b4 - for 2 classes(normal, defected) - More data - merged data (2 classes + 4 classes)	Average accuracy:  100% 9/9 [00:02<00:00, 4.23it/s]  Test Loss: 0.26860874477359986  Test Acc: 0.8993055820465088
Increased Batch size to 32	Best Accuracy at epoch 167:
epochs =200 Ir = 0.000005	Epoch : 167 Train Loss : 0.221962 Train Acc : 0.925123 Epoch : 167 Valid Loss : 0.280662 Valid Acc : 0.906250
Efficientnet_b4 - for 2 classes(normal, defected) - More data - merged data (2 classes + 4 classes)	Average Accuracy:  D- 100% 5/5 [00:01<00:0]  Test Loss: 0.28407591581344604  Test Acc: 0.8888392448425293
Increased Batch size to 64	Best Accuracy at epoch 169:
epochs =200 Ir = 0.000005	Epoch : 169 Train Loss : 0.250199 Train Acc : 0.906420 Epoch : 169 Valid Loss : 0.264553 Valid Acc : 0.918750

Efficientnet\_b4 - for 2 classes(normal, defected) - More data - merged data (2 classes + 4 classes)

#### Increased Batch size to 64

epochs =400 lr = 0.000005

## Average accuracy:

100% 5/5 [00:01<00:00, 3.03it/s]
Test Loss: 0.2485705479979515
Test Acc: 0.909375011920929

#### Best Accuracy at epoch 336:

Epoch: 136 Train Loss: 0.191113 Train Acc: 0.927891
Epoch: 136 Valid Loss: 0.246882 Valid Acc: 0.912500

# \*\*\*\* Efficientnet\_b4

Batch\_size = 32 Epochs = 500 Learning\_rate = 0.000005

#### Dataset ->

apple\_dataset\_normal\_and\_defecte d\_with\_extra\_data\_for\_resnet\_only \_validation\_data

#### Dataset link →

https://drive.google.com/drive/folder s/1NlmW5\_Vja29hoDbwE0AxnfaR M4T2wQG-?usp=sharing

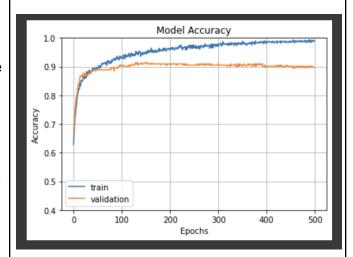
# Trained Efficientnet\_b4 model link →

https://drive.google.com/drive/folder s/1-BEEEKPcD8uDyZK QWg85A0 -lePAzbEg?usp=sharing

## Loading this model:-

"/content/drive/MyDrive/CMPE-295-A/my EfficientNetB4 model 1"

## Epoch 500/500



Training Accuracy: 98.97

Validation Accuracy: 89.71

	precision	recall	f1-score	support
DEFECTED	0.94	0.90	0.92	157
NORMAL	0.83	0.90	0.86	86
accuracy			0.90	243
macro avg	0.88	0.90	0.89	243
weighted avg	0.90	0.90	0.90	243

Using this instead of restnet-50 as this gives prediction results under 30ms.

## EfficientNetV2L-transfer-learning

#### Dataset:-

apple\_dataset\_normal\_and\_de
fected with extra data for

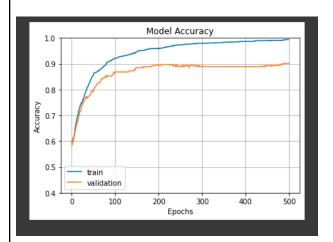
Epoch 350/350

resnet only validation data Val accuracy = 88.48% Batch size: - 32 epochs = 350Ir = 0.000002 EfficientNetV2L-transfer-learning Epoch 500/500 - 10s 322ms/step - loss: 0.1043 -Dataset:accuracy: 0.9784 - val loss: 0.2767 apple dataset normal and de val accuracy: 0.8848 fected with extra data for resnet only validation data Val accuracy = 88.48% Batch size: - 32 Model Accuracy epochs = 5001.0 Ir = 0.000003 0.9 0.8 0.7 0.6 0.5 - train validation 100 500 300 400 200 Epochs Resnet-50 Val accuracy = 0.8807 Batch size = 32Model Accuracy 1.0 Epochs = 150validation Learning rate = 0.000001 0.9 Dataset -> Accuracy 0.7 apple\_dataset\_normal\_and\_defecte 0.6 d with extra data for resnet only 0.5 validation data 140 20 80 100 120 60 Epochs Resnet-50 Epoch 500/500 Batch size = 324s 118ms/step - loss: 0.0598 -Epochs = 500accuracy: 0.9938 - val loss: 0.2697Learning rate = 0.000001 val accuracy: 0.9012 Dataset ->

Training Accuracy: 99.38

apple\_dataset\_normal\_and\_defecte d\_with\_extra\_data\_for\_resnet\_only validation\_data

## Validation Accuracy: 90.12



## \*\* Resnet-50

Batch\_size = 32 Epochs = 500 Learning\_rate = 0.000001

#### Dataset ->

apple\_dataset\_normal\_and\_defecte d\_with\_extra\_data\_for\_resnet\_only \_validation\_data

## Dataset link $\rightarrow$

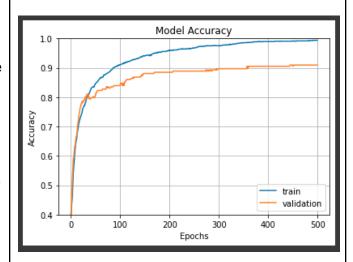
https://drive.google.com/drive/folder s/1NlmW5\_Vja29hoDbwE0AxnfaR M4T2wQG-?usp=sharing

## Trained RestNet-50 model link →

https://drive.google.com/drive/folders/193lKkplTwwEWv\_5RanUPVnuzYvR83qCL?usp=sharing

## Loading this model:-

"/content/drive/MyDrive/CMPE-295-A/my\_resnet\_model"



Training Accuracy: 99.38

Validation Accuracy: 90.95

[116] print(classification_report(labels, predictions, target_names=classnames))							
	precision	recall	f1-score	support			
DEFECTED NORMAL	0.95 0.85	0.91 0.91	0.93 0.88	157 86			
accuracy macro avg weighted avg	0.90 0.91	0.91 0.91	0.91 0.90 0.91	243 243 243			

## Resnet-50

Batch\_size = 32 Epochs = 500 Learning\_rate = 0.000001

Dataset ->

/content/drive/MyDrive/CMPE
-295-A/apple\_dataset\_normal
\_and\_defected\_with\_extra\_da
ta\_for\_resnet\_only\_validati
on\_data\_more\_data\_added

Training Accuracy: 100%

Validation Accuracy: 97%

The model was trained wrong because dataset has repeated images and hence images were repeating training set and validation set.

Can be improved but will look into it after 14th.

\*\* Work on dataset