SCHOOL OF ELECTRICAL ENGINEERING &
COMPUTER SCIENCE (SEECS)

Fruit Disease classification using resource constrained devices

HASAN ALI KHATTAK

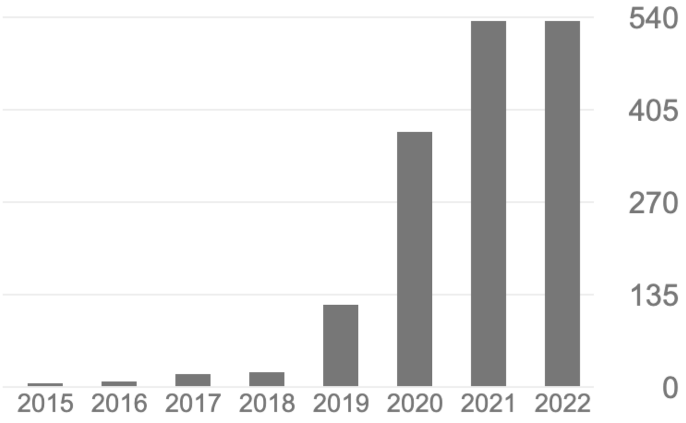
ESAW 2022 – OCTOBER 14, 2022

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY (NUST), ISLAMABAD.

about.me/hasanaliakhattak

- SisInf Lab - Politecnico di Bari 15’ - Italy
- Associate Professor - Department of Computing
- School of Electrical Engineering and Computer Science, NUST, Pakistan
- Internet of Things and Intelligent Systems Lab - SEECS
- IEEE Senior Member – ACM Professional Member
- Research Interests
  - Data Sciences - Internet of Things - Future Internet Architectures – Distributed Computing – World Wide Web

	All	Since 2017
Citations	1642	1623
h-index	22	22
i10-index	39	39



NUST SEECS

Agenda

- Introduction
- Motivation
- Related Work
- Problem Statement
- Methodology
- Implementation
- Results
- Future work
- References

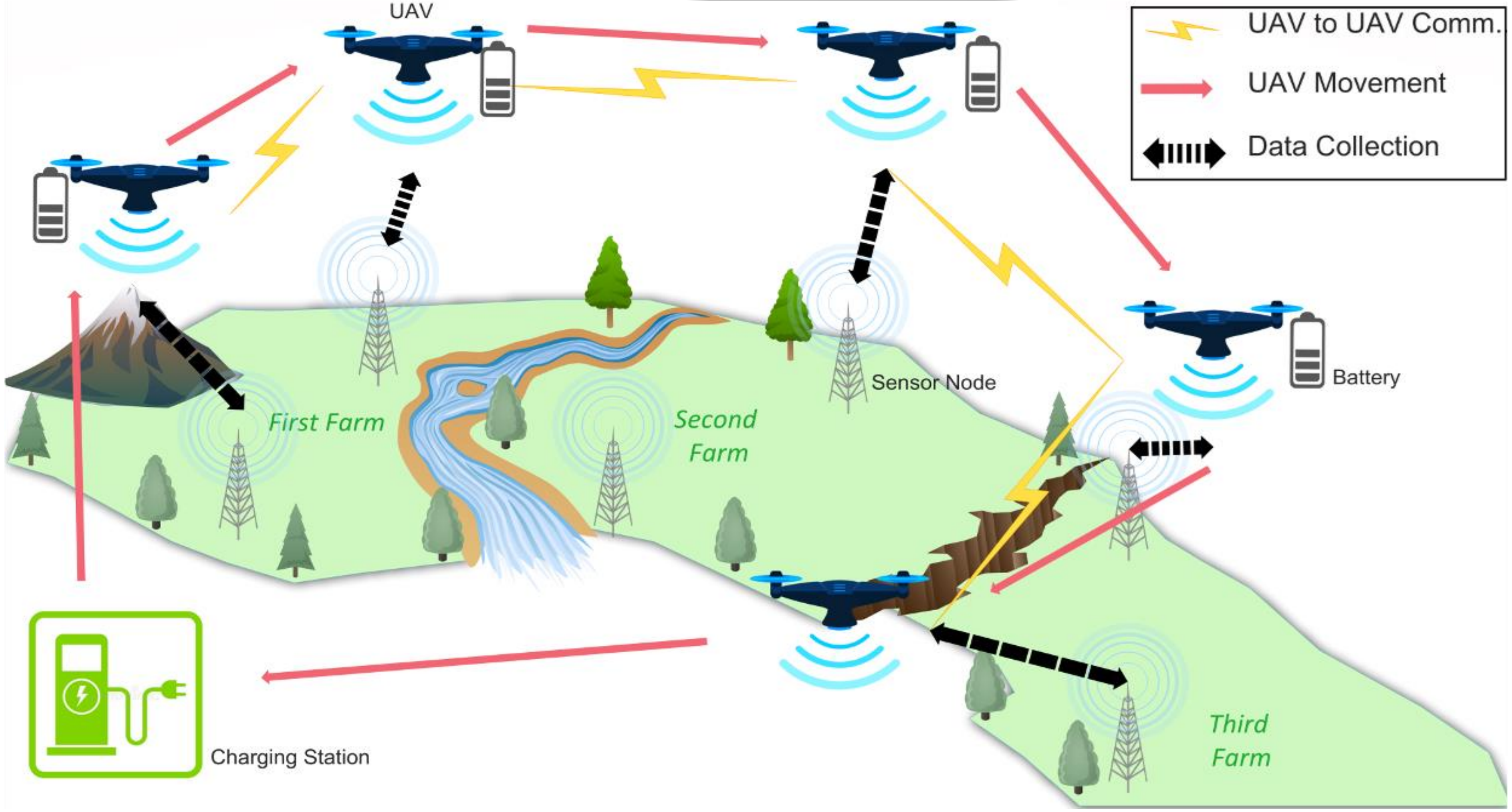
NUST SEECS

INTRODUCTION

- Machine Learning & computer vision have taken over the traditional computing methods, bringing a paradigm shift in all the sectors of technological development.
- On-device system benefits us by providing independence from remote resources.
- Fruit: Rich source of minerals and vitamins.

  Agriculture plays a vital role in economy of Pakistan.
- Contribution of apple market towards GDP is about 0.53% [1].

Motivation



AW. Malik (2022) Distributed Architectures for IoT Applications - ESAW2022 NUST Islamabad,

Motivation

- Lack of scientific means in agricultural development has placed Pakistan far behind in this competitive world.
- For identification of diseases, conventional approaches are used in Pakistan which are unreliable, inconsistent & time consuming
- Economic losses & reduced production in fruit sector.
- Mostly farmers are unable to procure expensive systems for fruits protection or its regular monitoring.
- We Need to equip farmers with a system which can identify/classify diseases at run time (Offline) via devices.

Related work

Authors	Title	Year	Dataset	Model	Accuracy	Gaps
Rabia Saleem , Jamal Hussain Shah, Muhammad Sharif	Mango Leaf Disease Recognition and Classification Using Novel Segmentation and Vein Pattern Technique	2021	Mango leaf	SVM	95.5%	Small leaf dataset, No deep learning models used, No edge device implementation & evaluation, No implementation in real time, minimization of identification time needed
Zhongxian Zhou a , Zhenzhen Song, et al	Real-time kiwifruit detection in orchard using deep learning on Android smartphones for yield estimation	2020	Kiwifruit	Mobile NetV2, InceptionV3	90.8%, 89.7%	Insufficient detection performance, detection speed and parameters does not meet requirements in real-time scenario, latest devices not used

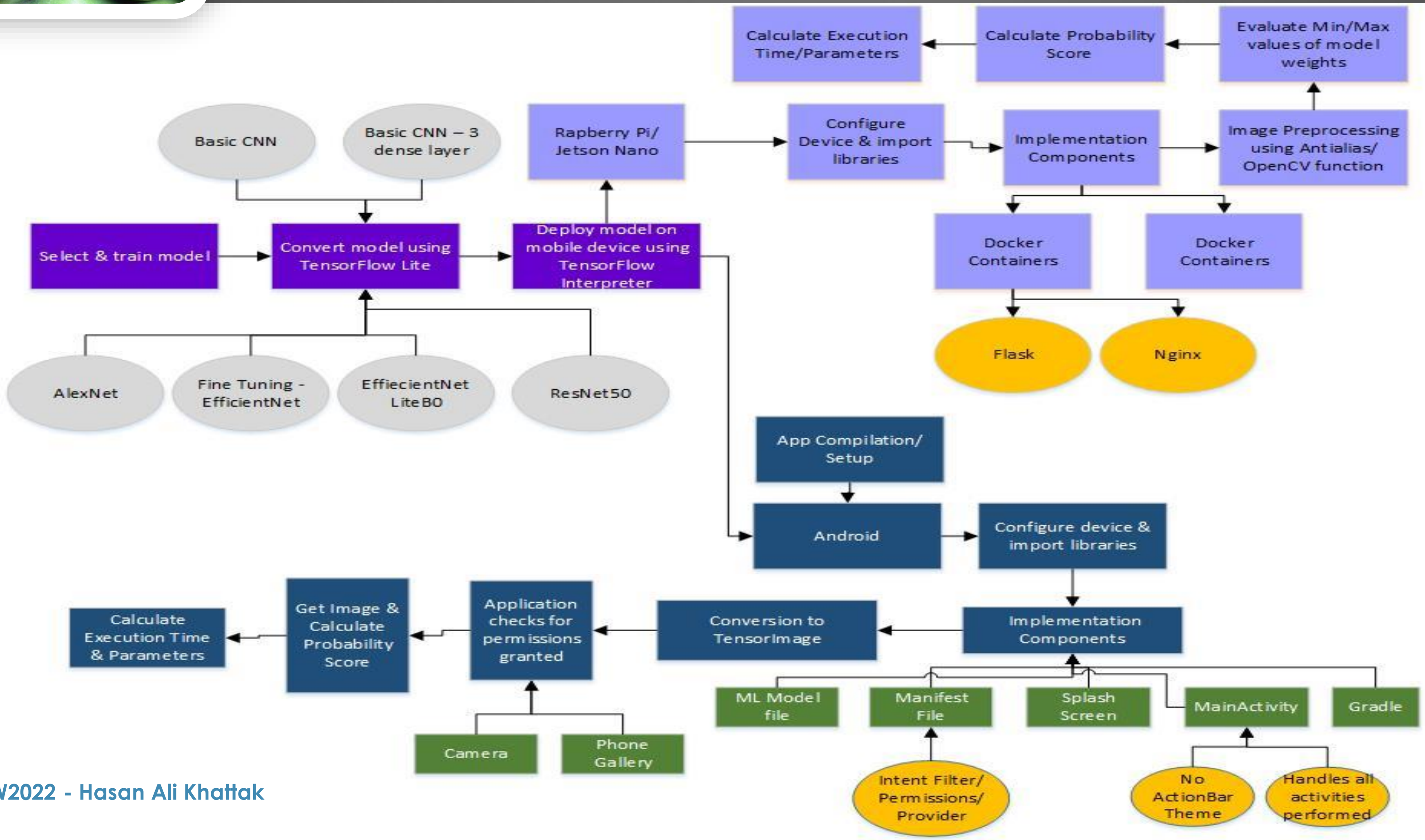
Related work

Authors	Title	Year	Dataset	Model	Accuracy	Gaps
Md. Tarek Habib,Md. Jueal Mia b , Mohammad Shorif	An In-depth Exploration of Automated Jack Fruit Disease Recognition	2020	Jack Fruit	Random Forest	89.52%	No implementation on edge devices, need to work on local fruits and deep learning models
Md. TarekHabib,Anup Majumder,A.Z.M .Jakaria	Machine vision based papaya disease recognition	2020	Papaya	K-means, SVM	90%	User sends image to system online. it is sent to the back-end server, where experts share results, Time consuming. No deployment on edge devices. Not applicable in rural areas

PROBLEM STATEMENT

Conventional methods of fruit disease identification are error prone, time consuming, expensive & sometimes biased. Farmers are not equipped with an offline system which can help them to identify/classify various kind of fruit diseases at run time with accuracy through resource constraint devices. Cloud-based machine learning has issues such as latency & greater computational effort.

Methodology



wsgi

DATASET

Apple Fruit Disease Dataset - Kaggle

Total No of images: 504

3 diseased classes & 1 healthy class

- **Testing Data Files**

Blotch_Apple	Normal_Apple	Rot_Apple	Scab_Apple
30 files	24 files	38 files	28 files

- **Training Data Files**

Blotch_Apple	Normal_Apple	Rot_Apple	Scab_Apple
116 files	67 files	114 files	85 files

Devices

Android

Samsung Galaxy A03s, Android version 11

Raspberry Pi

Raspberry Pi 4 Desktop Starter Kit - SC0400US MD-01721 (8GB Memory)

Arm32v7/Raspbian
python:3.7

Jetson Nano

Nvidia Jetson Nano BO1 4GB. Quad core ARM Cortex-A57 Processor

arm64v8/ubuntu:18.04
Python 3.6.9

NUST SEECS

Implementation

1. Model implementation on dataset
2. Classify fruit diseases with the help of various machine learning models on resource constraint devices (Raspberry Pi, Jetson Nano & Android).
3. Attain useful classification results in remote areas where internet or cloud services are not feasible.
4. Compare each model via measures like accuracy & confusion matrix etc.
5. Evaluate system's performance for checking its effectiveness and impact of different models & devices in order to utilize it in real time applications where fast execution time is of prime importance.

Implementation

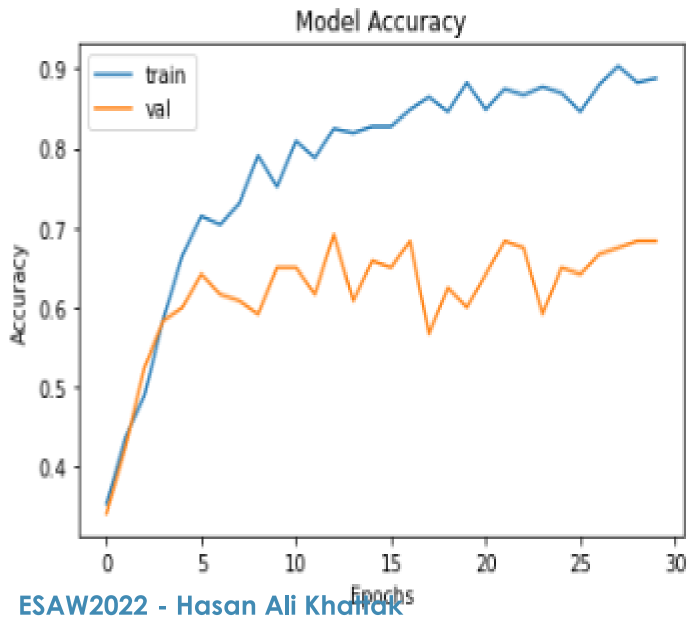
- **Languages**
- Python
- Kotlin

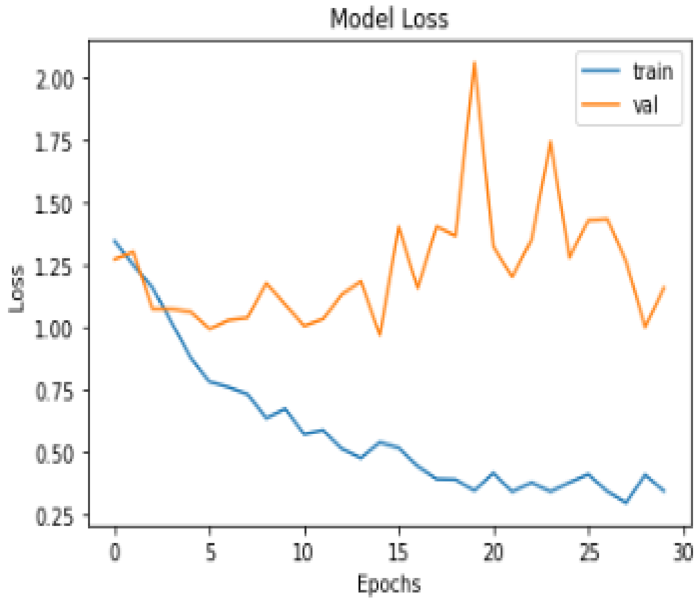
- **Tools**
- Android Studio
- Jupyter Notebook
- Visual Code

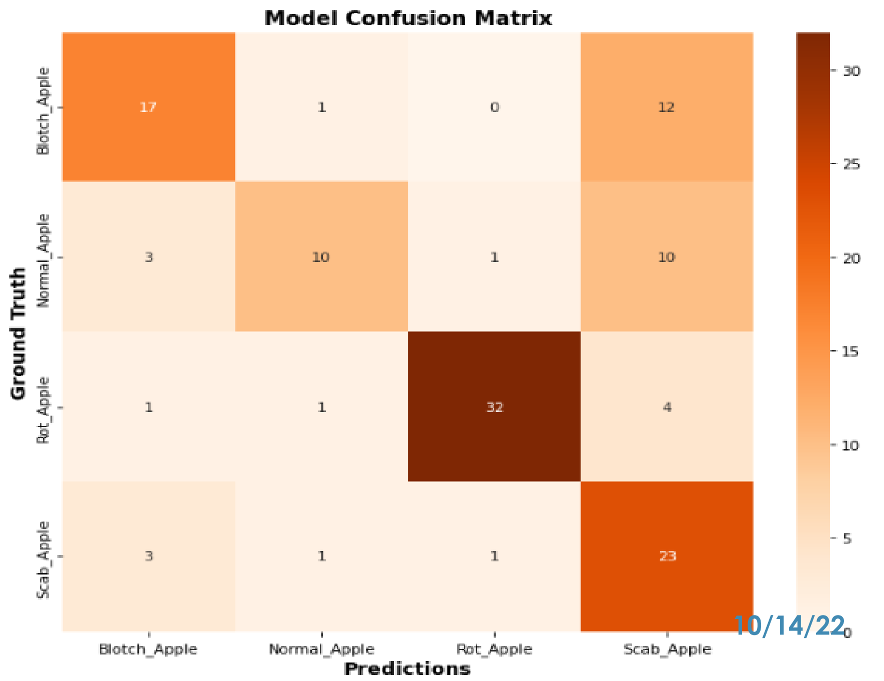
Basic CNN model

Layer	Input	Input Size	Activation	Nodes	Output Size
Convolutional2D 1	Input Image	64x64x3	ReLU	-	62x62x32
MaxPooling2D 1	Convolutional2D 1	62x62x32	-	-	31x31x32
Convolutional2D 2	Maxpooling2D 1	31x31x32	ReLU	-	29x29x64
MaxPooling 2	Convolutional2D 2	29x29x64	-	-	14x14x64
Flatten	MaxPooling2D 2	14x14x64	-	-	12544
Dense 1	Flatten	12544	ReLU	32	32
Dense 2	Dense 1	32	ReLU	64	64
Dense 3	Dense 2	64	ReLU	128	128
Dense 4	Dense 3	128	ReLU	256	256
Dense 5	Dense 4	256	ReLU	256	256
Dense 6	Dense 5	256	SoftMax	4	4

	precision	recall	f1-score	support
Blotch_Apple	0.71	0.57	0.63	30
Normal_Apple	0.77	0.42	0.54	24
Rot_Apple	0.94	0.84	0.89	38
Scab_Apple	0.47	0.82	0.60	28
micro avg	0.68	0.68	0.68	120
macro avg	0.72	0.66	0.66	120
weighted avg	0.74	0.68	0.69	120
samples avg	0.68	0.68	0.68	120



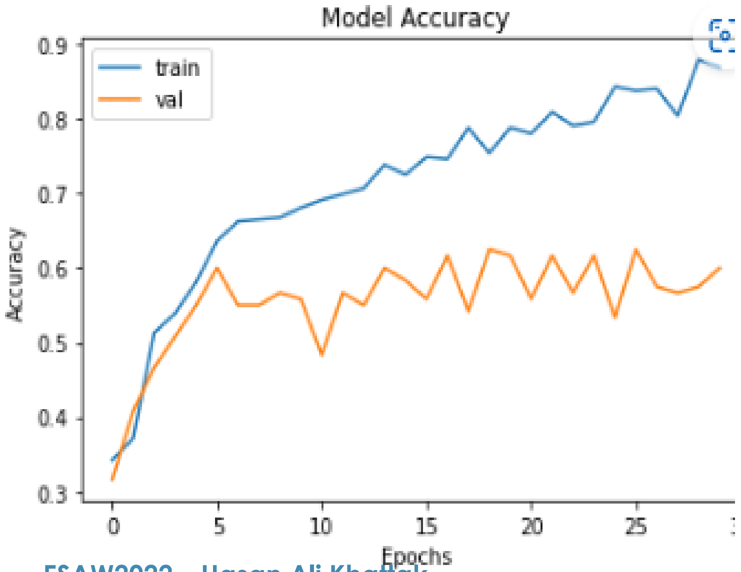


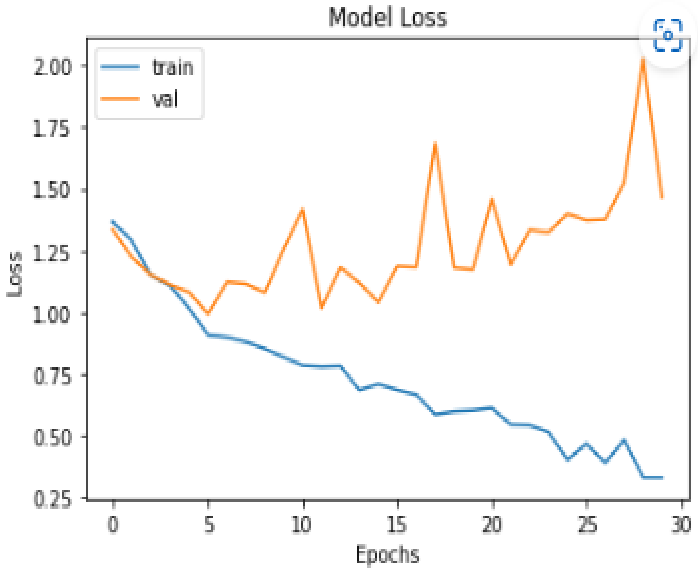


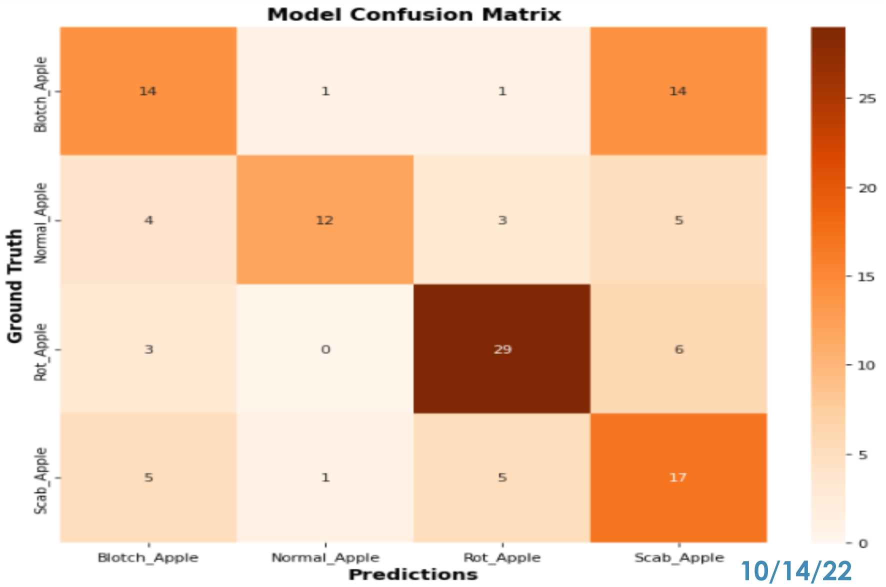
Basic CNN model – 3 Dense layers

Layer	Input	Input Size	Activation	Nodes	Output Size
Convolutional2D 1	Input Image	64x64x3	ReLU	-	58x58x32
MaxPooling2D 1	Convolutional2D 1	58x58x32	-	-	29x29x32
Dropout	MaxPooling2D 1	29x29x32	-	-	29x29x32
Convolutional2D 2	Dropout	29x29x32	ReLU	-	23x23x32
MaxPooling2D 2	Convolutional2D 2	23x23x32	-	-	11x11x32
Flatten	MaxPooling2D 2	11x11x32	-	-	3872
Dense 1	Flatten	3872	ReLU	64	64
Dense 2	Dense 1	64	ReLU	128	128
Dense 3	Dense 2	128	SoftMax	4	4

	precision	recall	f1-score	support
Blotch_Apple	0.54	0.47	0.50	30
Normal_Apple	0.86	0.50	0.63	24
Rot_Apple	0.76	0.76	0.76	38
Scab_Apple	0.40	0.61	0.49	28
micro avg	0.60	0.60	0.60	120
macro avg	0.64	0.58	0.60	120
weighted avg	0.64	0.60	0.61	120
samples avg	0.60	0.60	0.60	120



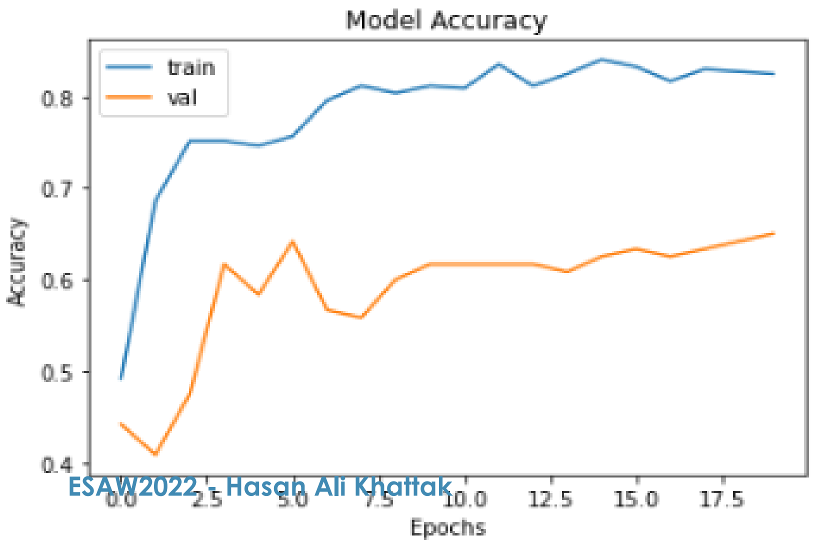


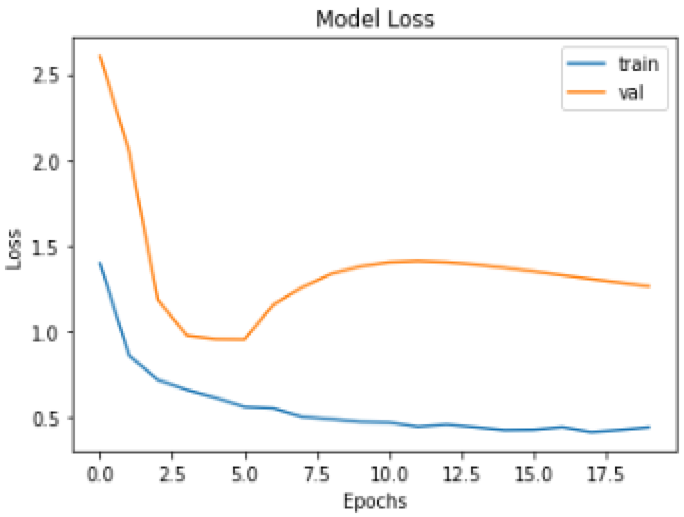


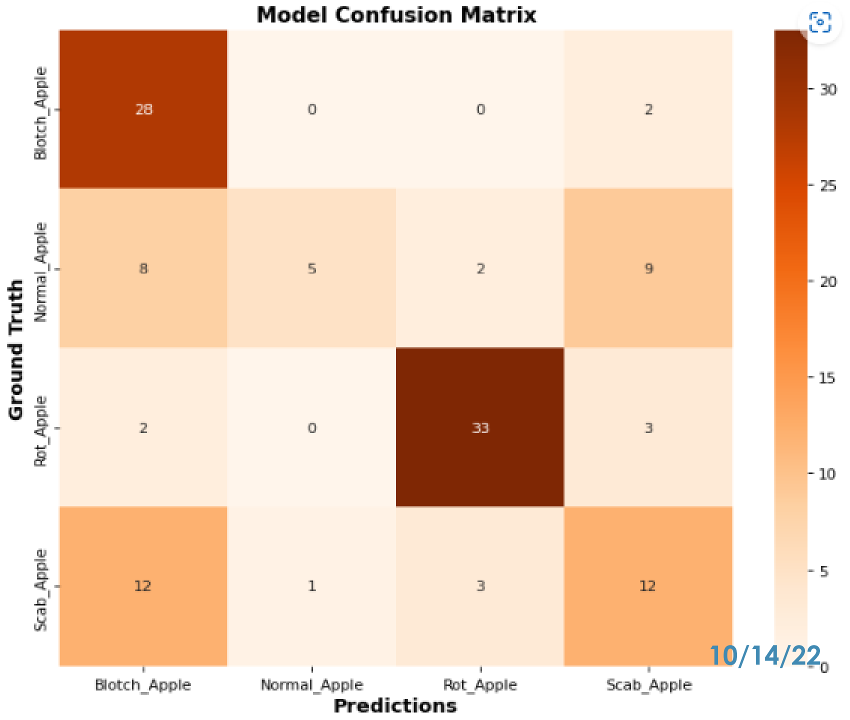
AlexNet Model

Layer	Input	Input Size	Activation	Nodes	Output Size
Input	Input Image	227x227x3	-	-	227x227x3
Conv2D 1	Input	227x227x3	-	-	55x55x96
BatchNormalization 1	Conv2D 1	55x55x96	-	-	55x55x96
Activation 1	BatchNormalization 1	55x55x96	ReLU	-	55x55x96
MaxPooling 1	Activation 1	55x55x96	-	-	27x27x96
Conv2D 2	MaxPooling 1	27x27x96	-	-	27x27x256
BatchNormalization 2	Conv2D 2	27,27,256	-	-	27x27x256
Activation 2	BatchNormalization 2	27x27x256	ReLU	-	27x27x256
MaxPooling 2	Activation 2	27x27x256	-	-	13x13x256
Conv2D 3	MaxPooling 2	13x13x256	-	-	13x13x384
BatchNormalization 3	Conv2D 3	13x13x384	-	-	13x13x384
Activation 3	BatchNormalization 3	13x13x384	ReLU	-	13x13x384
Conv2D 4	Activation 3	13x13x384	-	-	13x13x384
BatchNormalization 4	Conv2D 4	13x13x384	-	-	13x13x384
Activation 4	BatchNormalization 4	13x13x384	ReLU	-	13x13x384
Conv2D 5	Activation 4	13x13x384	-	-	13x13x256
BatchNormalization 5	Conv2D 5	13x13x256	-	-	13x13x256
Activation 5	BatchNormalization 5	13x13x256	ReLU	-	13x13x256
MaxPooling 3	Activation 5	13x13x256	-	-	6x6x256
Flatten	MaxPooling 3	6x6x256	-	-	9216
Dense 1	Flatten	9216	ReLU	4096	4096
Dense 2	Dense 1	4096	ReLU	4096	4096
Dense 3	Dense 2	4096	SoftMax	4	4

	precision	recall	f1-score	support
Blotch_Apple	0.56	0.93	0.70	30
Normal_Apple	0.83	0.21	0.33	24
Rot_Apple	0.87	0.87	0.87	38
Scab_Apple	0.46	0.43	0.44	28
micro avg	0.65	0.65	0.65	120
macro avg	0.68	0.61	0.59	120
weighted avg	0.69	0.65	0.62	120
samples avg	0.65	0.65	0.65	120







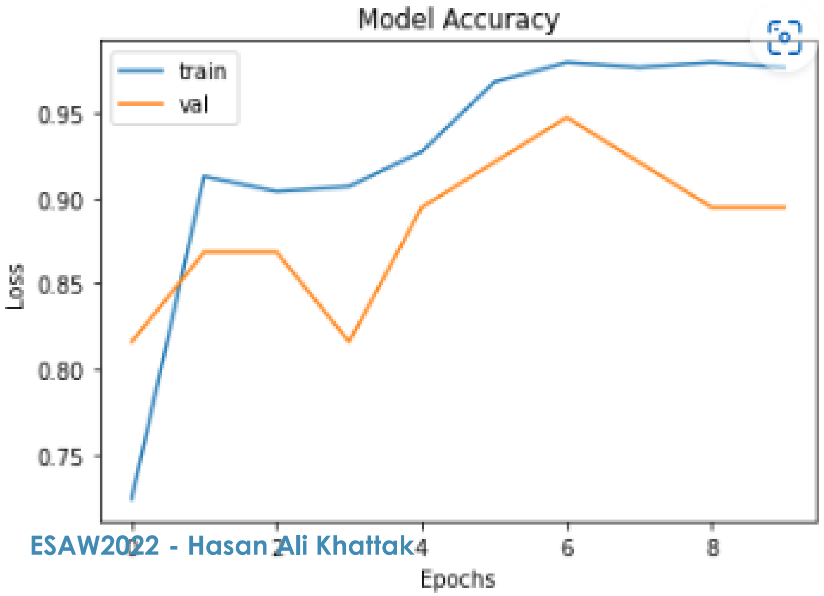
EfficientNet Lite Model – Fine Tuning

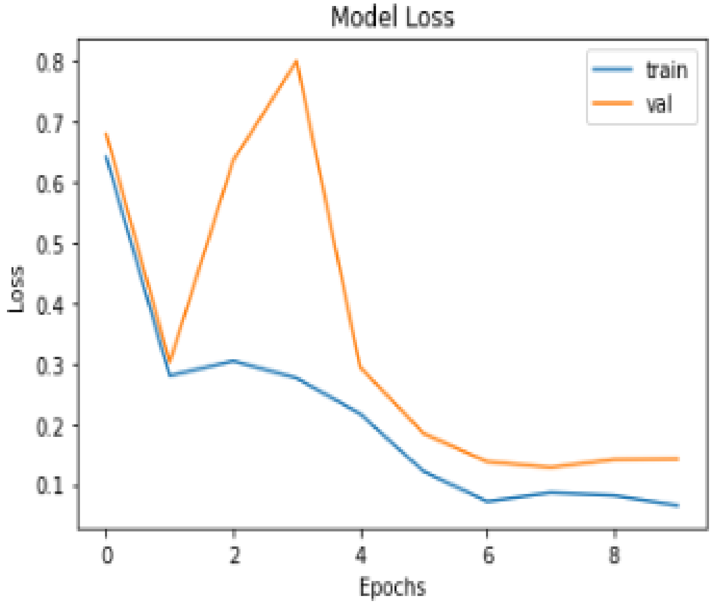
EfficientNet Stages Architecture Overview

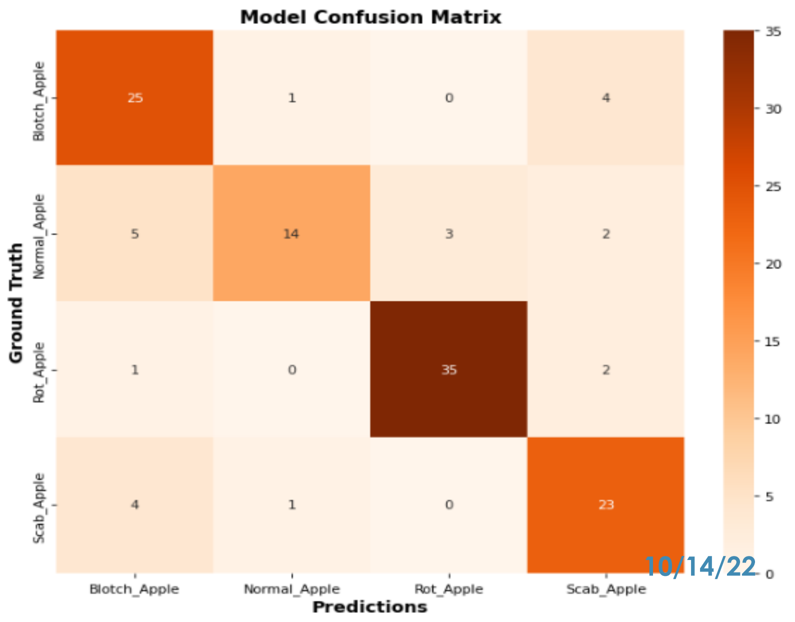
Stage	Operator Layer	Resolution	Number of Channels	Number of Layers
1	Conv, 3 × 3	224 × 224	32	1
2	MBConv1, k 3 × 3	112 × 112	16	1
3	MBConv6, k 3 × 3	112 × 112	24	2
4	MBConv6, k 5 × 5	56 × 56	40	2
5	MBConv6, k 3 × 3	28 × 28	80	3
6	MBConv6, k 5 × 5	14 × 14	112	3
7	MBConv6, k 5 × 5	14 × 14	192	4
8	MBConv6, k 3 × 3	7 × 7	320	1
9	Conv 1 × 1 & Pooling & FC	7 × 7	1280	1

Layer	Input	Input Size	K	S	Activation	Nodes	Output Size
Efficient Net lite 0	Input Image	224x224x3	.	.	ReLU	1280	1280
Dropout 1	Efficient Net lite 0	1280	.	.	.	.	1280
Dense 1	Dropout 1	1280	.	.	.	4	4

	precision	recall	f1-score	support
Blotch_Apple	0.71	0.83	0.77	30
Normal_Apple	0.88	0.58	0.70	24
Rot_Apple	0.92	0.92	0.92	38
Scab_Apple	0.74	0.82	0.78	28
micro avg	0.81	0.81	0.81	120
macro avg	0.81	0.79	0.79	120
weighted avg	0.82	0.81	0.81	120
samples avg	0.81	0.81	0.81	120







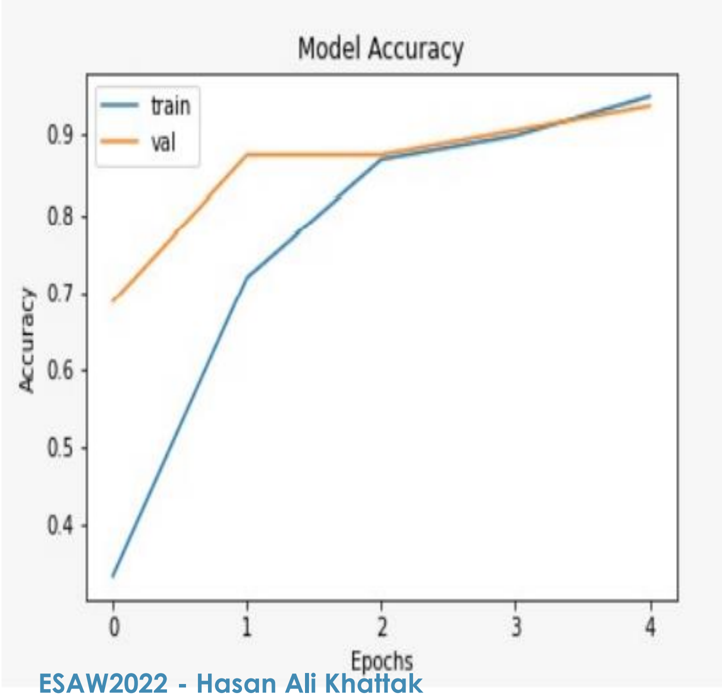
Efficient LiteB0 Model – Transfer Learning

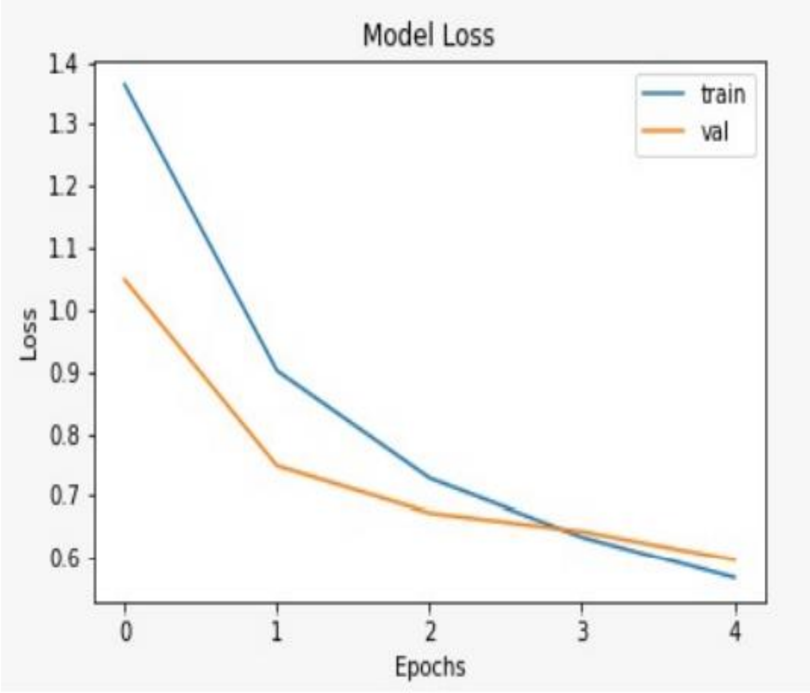
EfficientNet Stages Architecture Overview

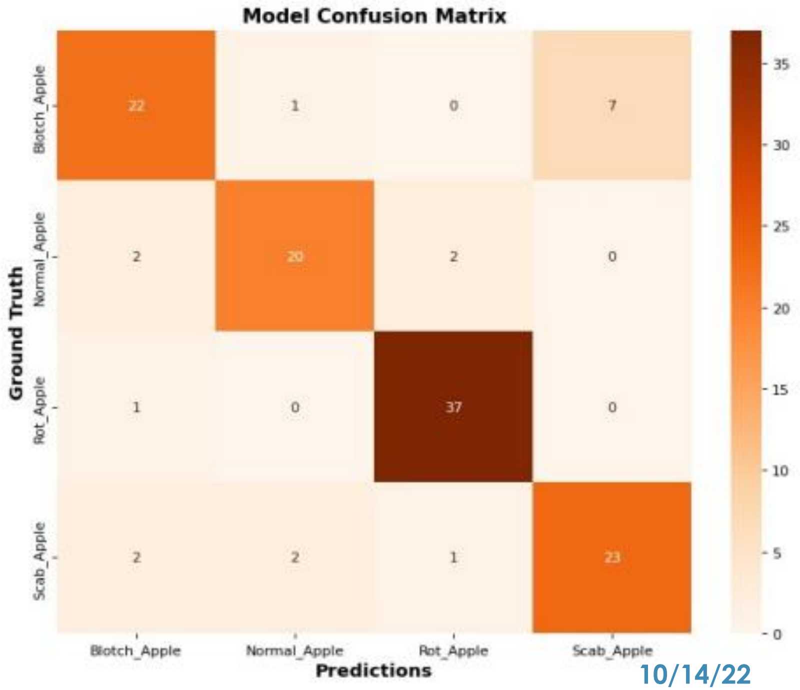
Stage	Operator Layer	Resolution	Number of Channels	Number of Layers
1	Conv, 3 × 3	224 × 224	32	1
2	MBConv1, k 3 × 3	112 × 112	16	1
3	MBConv6, k 3 × 3	112 × 112	24	2
4	MBConv6, k 5 × 5	56 × 56	40	2
5	MBConv6, k 3 × 3	28 × 28	80	3
6	MBConv6, k 5 × 5	14 × 14	112	3
7	MBConv6, k 5 × 5	14 × 14	192	4
8	MBConv6, k 3 × 3	7 × 7	320	1
9	Conv 1 × 1 & Pooling & FC	7 × 7	1280	1

Layer	Input	Input Size	K	S	Activation	Nodes	Output Size
EfficientNet-Lite	Input Image	224x224x3	-	-	ReLU	1280	1280
Dropout 1	Efficient Net lite 0	1280	-	-	-	-	1280
Dense 1	Dropout 1	1280	-	-	-	4	4

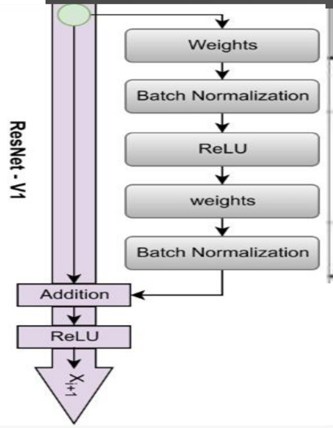
Classes	Precision	Recall	F1-Score
Blotch_Apple	0.82	0.74	0.77
Normal_Apple	0.87	0.84	0.85
Rot_Apple	0.93	0.97	0.95
Scab_Apple	0.77	0.82	0.79





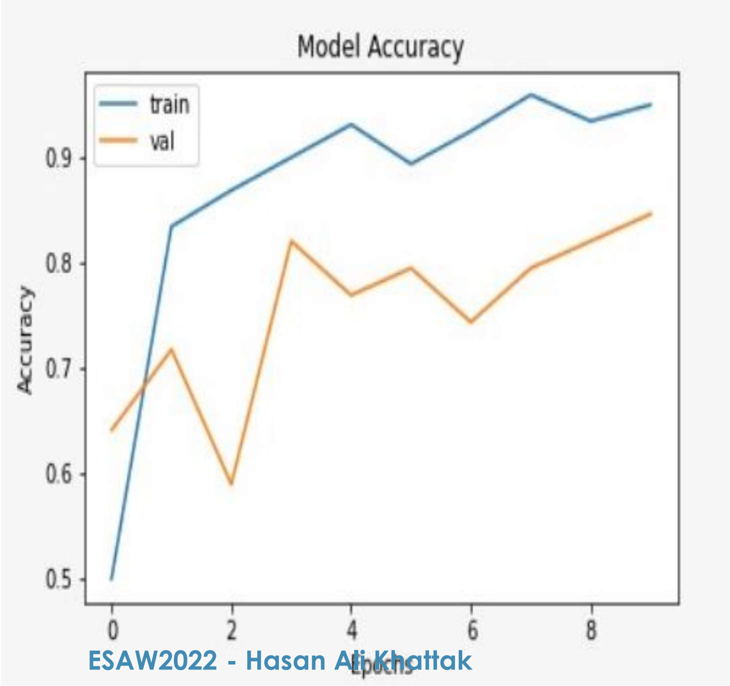


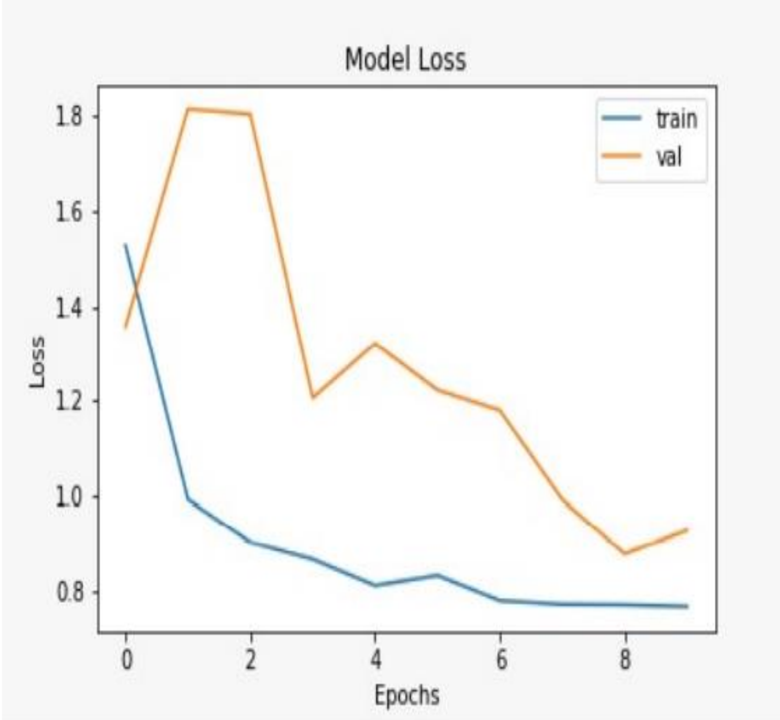
Resnet50 Model – Transfer Learning

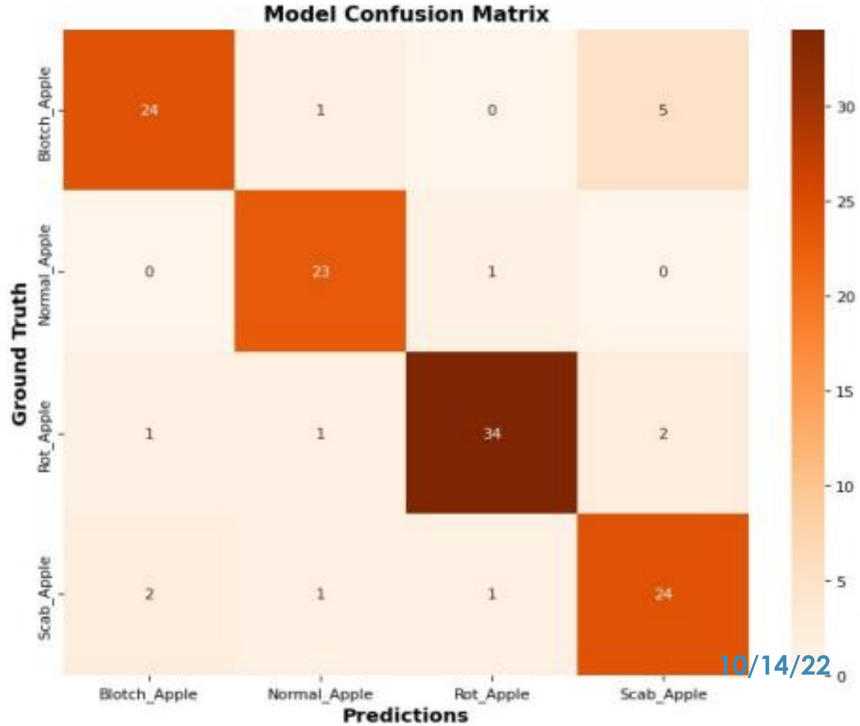


Layer	Input	Input Size	Activation	Nodes	Output Size
ResNet50 baseModel	Input Layer	224x224x3	ReLU	-	2048
Dropout	ResNet50 baseModel	2048	-	2048	2048
Dense	Dropout	4	SoftMax	4	4

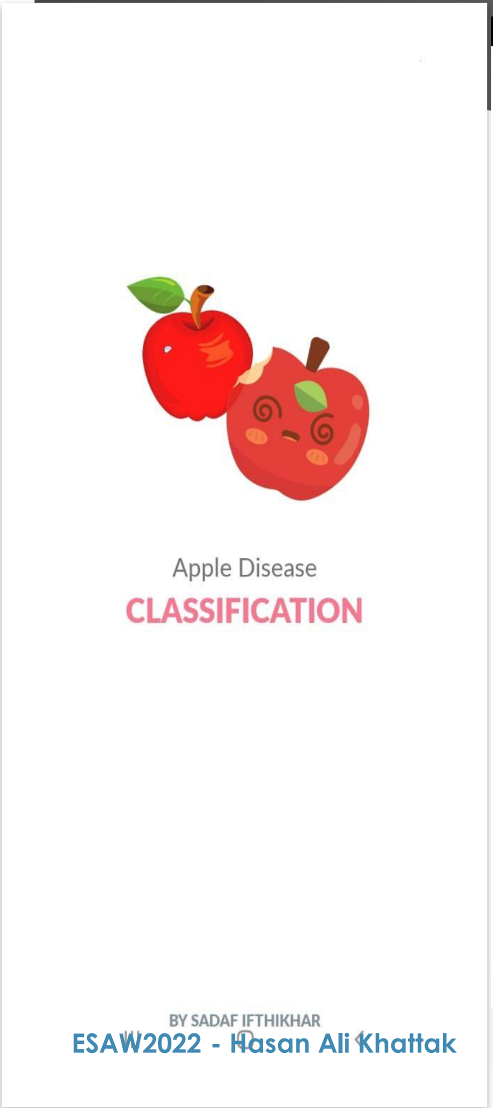
Classes	Precision	Recall	F1-Score
Blotch_Apple	0.89	0.80	0.84
Normal_Apple	0.88	0.96	0.92
Rot_Apple	0.94	0.89	0.92
Scab_Apple	0.77	0.86	0.81

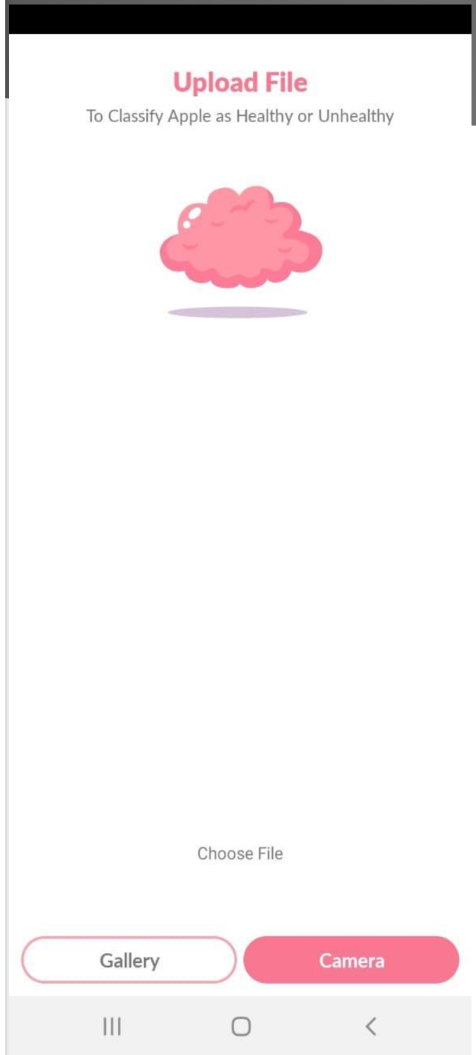


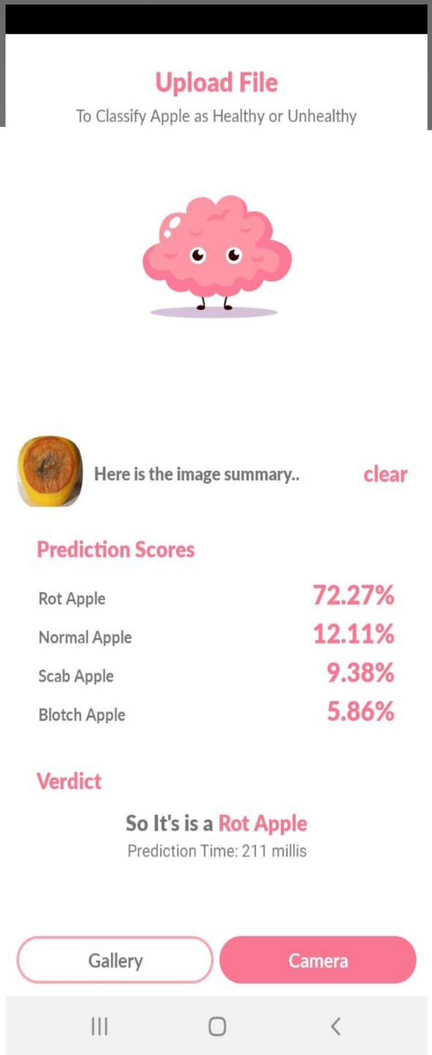


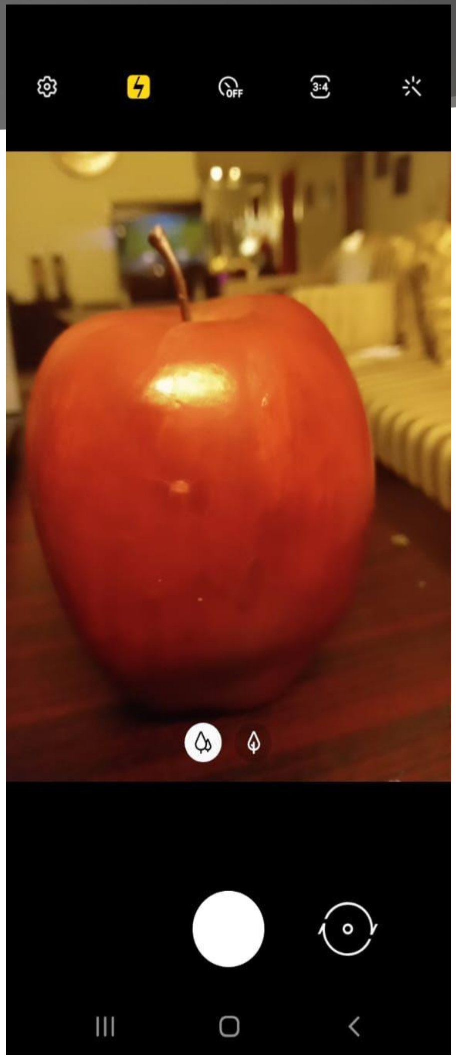


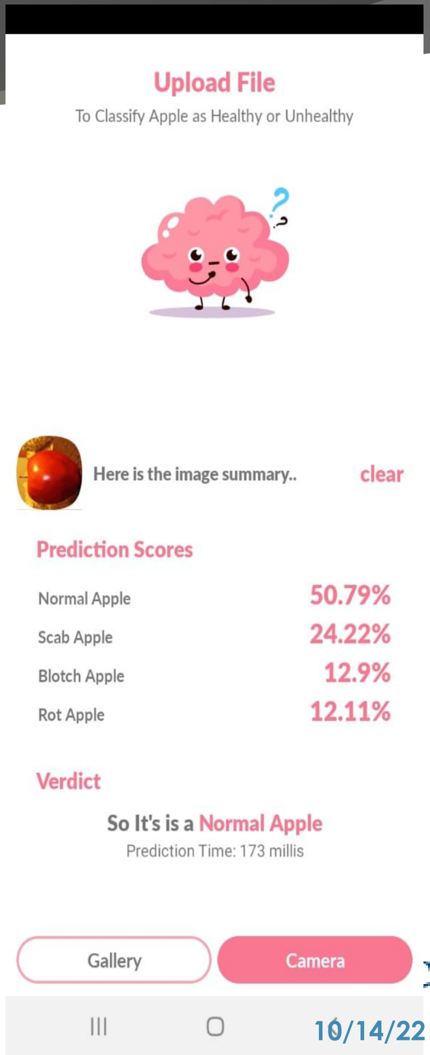
Android Application



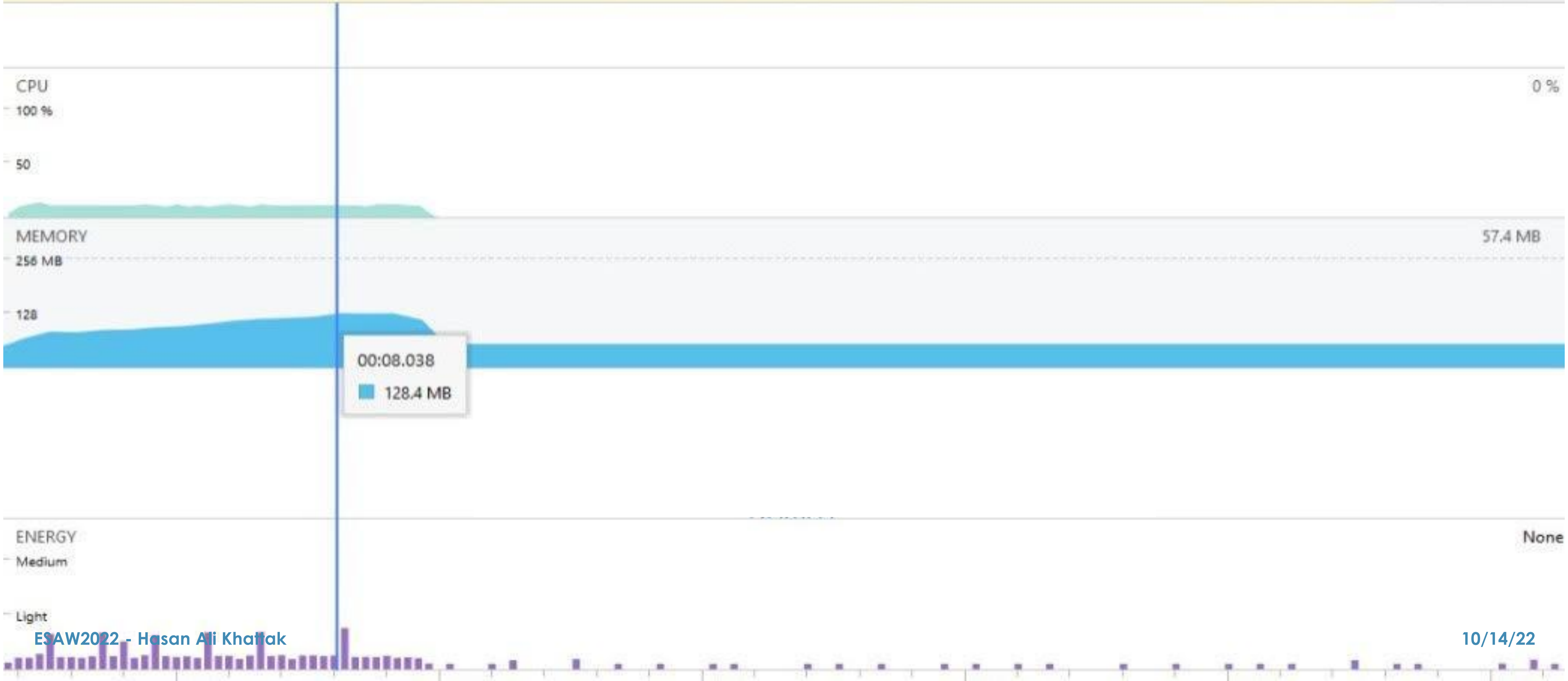




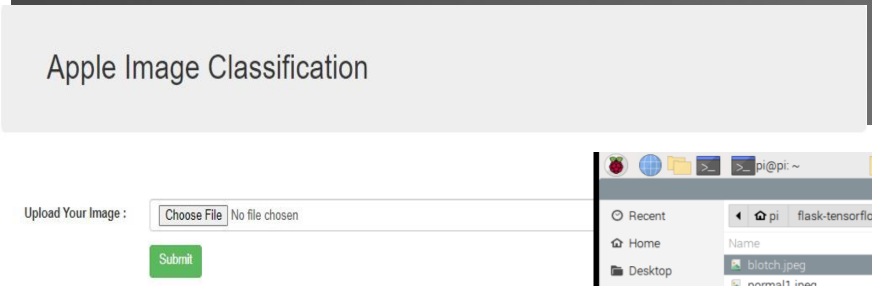


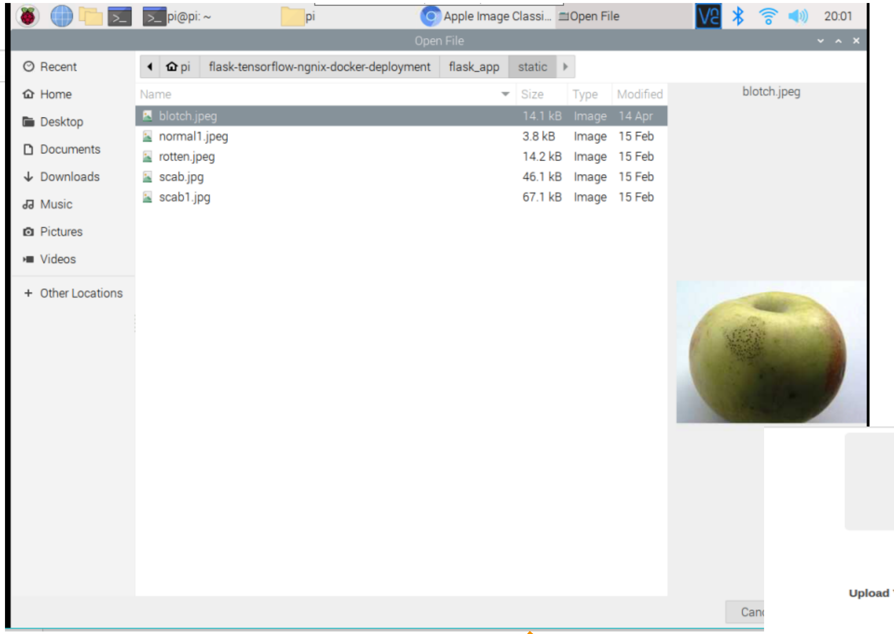


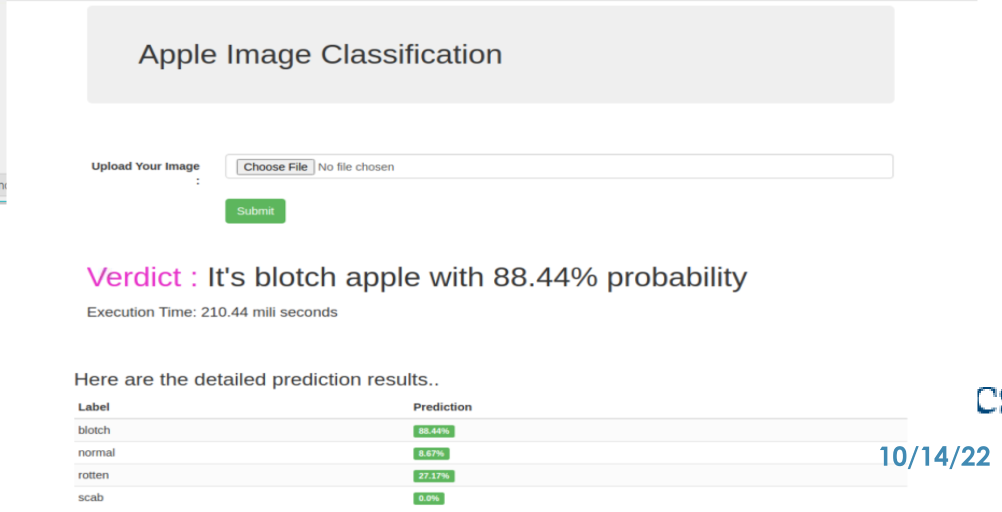
Android Application Utilization



Raspberry Pi/Jetson Nano Application







Device Comparison Parameters

Comparison Parameters	Android	Raspberry Pi	Jetson Nano
Classification Time	192 millis	210 Millis	109 Millis
Energy/Power	Light Energy	5V/2A	5V/581 mW
CPU Load	10% (initially) drops to 0%	1.30%	0.89%
Memory	128.4MB (initially) drop to 57mb	4.0%	0.10%

Jetson Nano	CPU%	Memory%
NginX	0.00%	0.09%
Flask	0.02%	5.86%
Raspberry Pi	**CPU%**	**Memory%**
NginX	0.00%	0.03%
Flask	0.03%	1.59%

NUST SEECS

Future work

A more refined data set with huge quantity and good quality of images can further strengthen the work.

For future work, respective edge devices could be assembled on unmanned aerial vehicles (UAVs) technologies which can be utilized to increase agricultural productivity while lowering labor costs, inspection times, and crop management expenses.

More powerful GPU device utilization can be performed.

REFERENCES

[1] [internet] https://www.tridge.com/intelligences/apple/PK/export

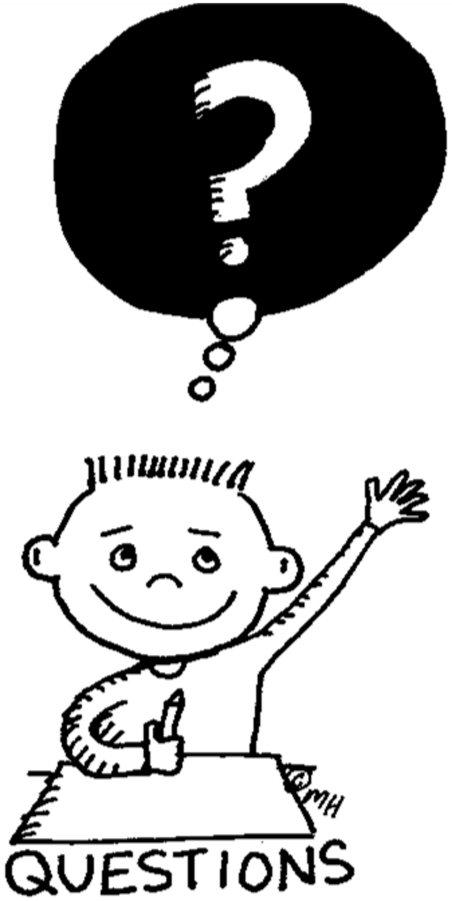
[2]"Mango Leaf Disease Recognition and Classification Using Novel Segmentation and Vein Pattern Technique", Saleem, Rabia and Shah, Jamal Hussain and Sharif, Muhammad and Yasmin, Mussarat and Yong, Hwan-Seung and Cha, Jaehyuk, 2021

[3] "Real-time kiwifruit detection in orchard using deep learning on Android™ smartphones for yield estimation", ZhongxianZhou,ZhenzhenSong,LongshengFu, 2020

[4] "An in-depth exploration of automated jackfruit disease recognition", Md. TarekHabibab Md. JuealMia, Mohammad ShorifUddin", 2020

[5] "Machine vision based papaya disease recognition", Md. TarekHabib, AnupMajumder", 2020

THANK YOU



You can contact me by just searching me
https://www.google.com/search?q=hasan+ali+khattak