Industrial Internship Report on

”Crop Weed Prediction”

Prepared by

[Smita Kenjale]

Executive Summary
This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT). This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks’ time. My project was (Tell about ur Project) This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship.

TABLE OF CONTENTS

1 Preface

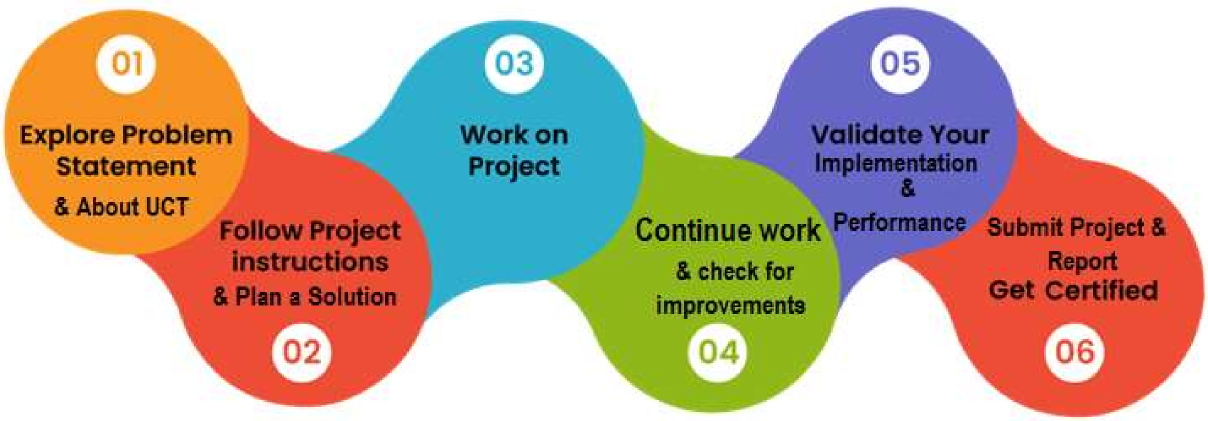
Summary of the whole 6 weeks' work.

About need of relevant Internship in career development.

Brief about Your project/problem statement.

Opportunity given by USC/UCT.

How Program was planned



Your Learnings and overall experience.

Thank to all (with names), who have helped you directly or indirectly.

Your message to your juniors and peers.

2 Introduction

2.1 About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.

For developing its products and solutions it is leveraging various **Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end** etc.

i. UCT IoT Platform (uct Insight)

UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA
- It supports both cloud and on-premises deployments.

It has features to

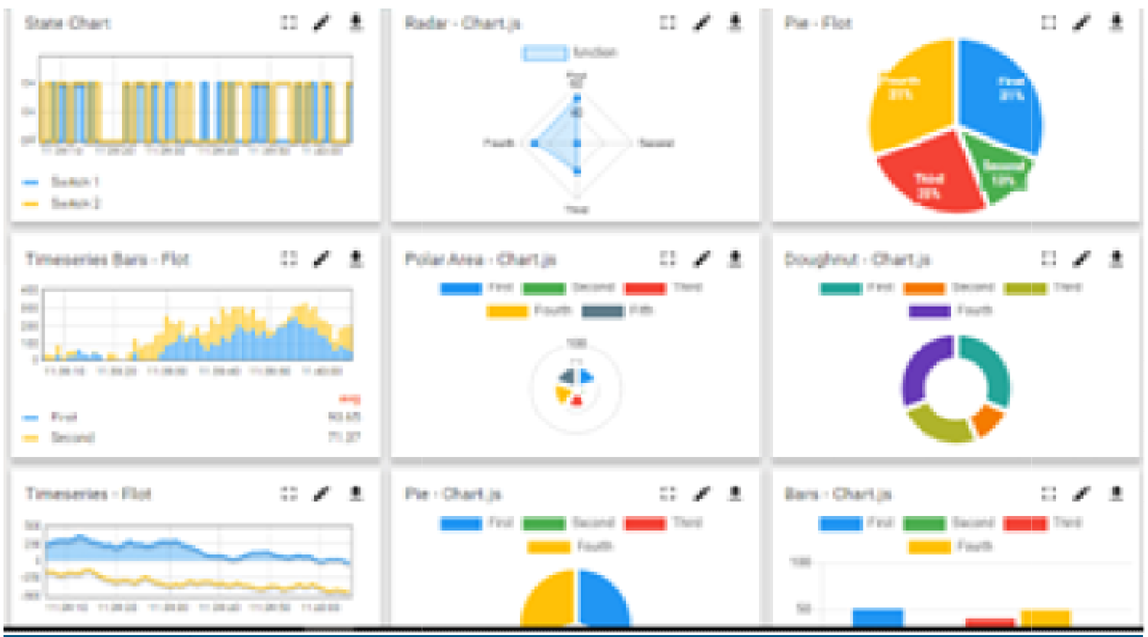
• Build Your own dashboard

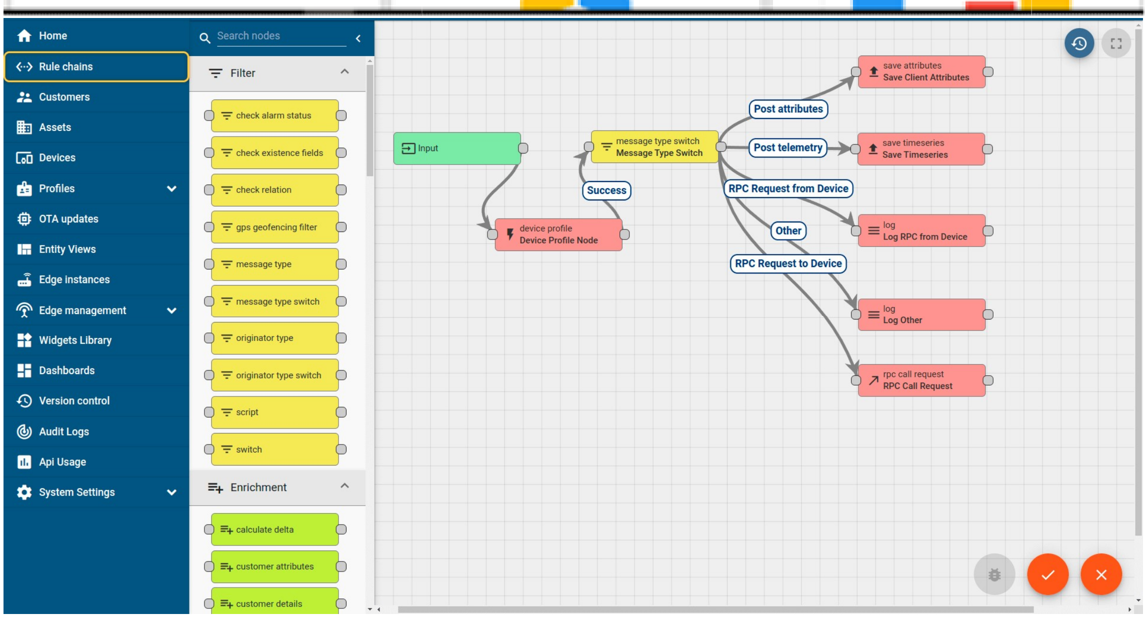
• Analytics and Reporting

• Alert and Notification

• Integration with third party application(Power BI, SAP, ERP)

• Rule Engine





ii. Smart Factory Platform (FACTORY WATCH)

Factory watch is a platform for smart factory needs.

It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.

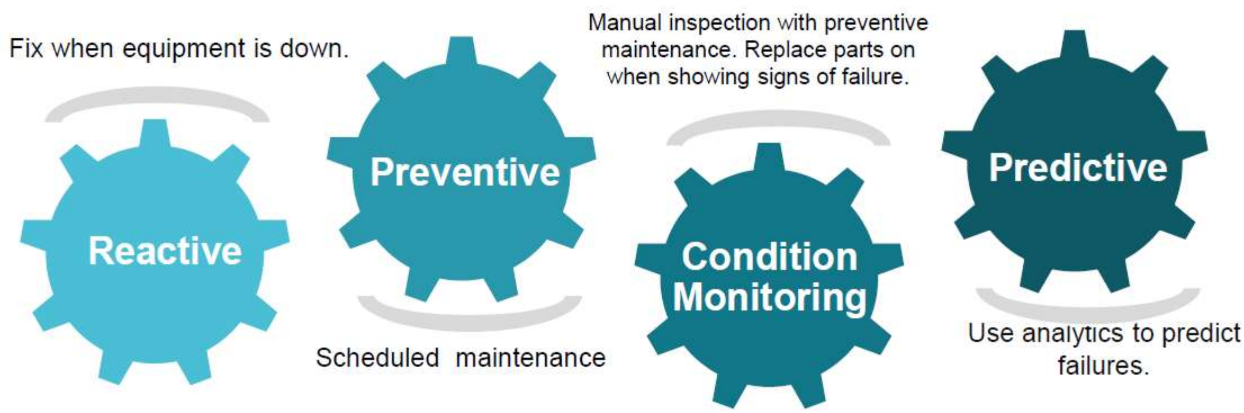
Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i

iii. LoRaWAN based Solution

UCT is one of the early adopters of LoRAWAN teschnology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

iv. Predictive Maintenance

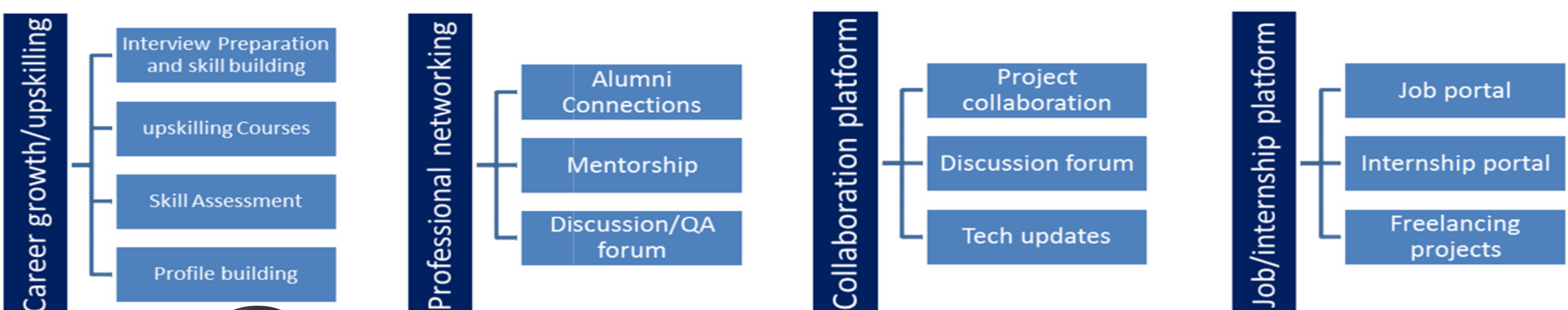
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



2.2 About upskill Campus (USC)

upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process.

USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.


Career growth/upskilling
Interview Preparation and skill building
upskilling Courses
Skill Assessment
Profile building
Professional networking
Alumni Connections
Mentorship
Discussion/QA forum
Collaboration platform
Project collaboration
Discussion forum
Tech updates
Job/internship platform
Job portal
Internship portal
Freelancing projects

2.3 The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

2.4 Objectives of this Internship program

The objective for this internship program was to

☛ get practical experience of working in the industry.

☛ to solve real world problems.

☛ to have improved job prospects.

☛ to have Improved understanding of our field and its applications.

☛ to have Personal growth like better communication and problem solving.

2.5 Reference

[1]

[2]

[3]

2.6 Glossary

Terms	Acronym

Chapter 1

Introduction

Invasive weeds are one of the main problems that every farmer encounters whilegrowing crops. In order to maximize the yield, timely removal of weeds is neces sary as the weeds kill or hinder the growth of crops by stealing water, nutrients,and sunlight. Farmers use herbicides to get rid of the weeds or manually removethem. However, the use of herbicides increases the cost of production and exposeshumans to dangerous chemicals. Moreover, herbicides can remain active in theenvironment for long periods of time, potentially causing soil and water contami nation, adversely effecting non-target organisms, and affecting the health of humanbeings [27].Manually removing the weeds is very labor intensive, is inefficient, andincreases the cost of production. Lastly, another main problem is predicting yieldas early as possible. There are many variables when considering yield prediction,including weather, temperature, pest, and many others. With finite land available,being able to predict crop yield allows us to understand food security, and predictif we will have enough food for the future. Yield prediction would allow farmers**to** manage crops base on yield prediction, which would allow them to maximizeprofit. With the advancement in machine learning techniques and availability ofmachine learning techniques, This work focus on leverage convolutional neural net works (CNNs) to make predictions on identification of weeds and predict yield of crop.

Chapter 2 Literature

2.1 Precision Agriculture

Precision agriculture (PA) is an information-based and production-based farming system that has the goal of optimum productivity and profitability, sustainability and protection of the land resource by minimizing the production costs [24]. The goal of precision agriculture can be simplify as maximizing output (i.e. crop yield) while minimizing input (i.e. fertilizer, pesticide, herbicide etc) in the processes of growing a crop. Advances in PA are important in order to develop sustainable agri culture practices. As the world population increases, the production of agriculture should increase while maintaining a low cost.

2.1.1 Yield Prediction

sions relating to storage, distribution, pricing, marketing, import-export decision4 to strength national food security. In small scale farmers benefit from predicting the yield to make informed management and financial decisions.[10] [15] In ancient times, crop yield forecasting was based on farmers early experience with the crop. Most of the time, yield mainly depended on the weather conditions. The dependence of weather condition was the major significant feature usedfor a long time. Weather affects crop differently during different stages of crop growth that's why farmers planted crops around the same time of the year to maintain the same weather patterns. However, the yield not only depended on the weather, butdistributiopattern of weather over the crop season. This increased the number ofvariables that needed to be considered to predict the yield.[10]In 1924, Fisher suggested a mathematical technique to estimate yield whichtook care of the distribution of weather over the crop season. Fisher assume thatweather patterns over successive weeks would not change erratically. Instead, thoseweather patterns over weeks follow some mathematical order. The mathematicalorder could be expressed with a polynomial function. As shown in Eq. 2.1, where nnumber is the number of weeks. The model was most effective in subsections of fiveweeks. For example, $i = 5$ was use to study the influence of rainfall on the yield ofwheat with accuracy. (Note Y denotes yield and Xi rainfall in the i-th week.) [6]$Y = A0 + A1X1 + A2X2 + ... + AnXn$Another approach to forecasting yield is based on plant characteristics. Thisapproached is computed per plant plot is more accurate. Volgel in 1985 introducedthe model based on plant characteristics. Some of the characteristic included biomass of the crop per plot, leaf area, number of tillers, and number of earheads(flower) etc. This approach could only be done with crops that had a period of5maturity of two to three months and the process needed a lot of man power tocollect the data. This new model added plant characteristics to the forecastingmodel. [2]Another significant research advancement in the field is by Jame and Cutforth,which showed the limitations of conventional experience that depended upon sta tistical analysis which has many limitations. The new approach called decisionsupport system for agro-technology transfer. according to them will improve theunderstanding of the process of yield forecasting. The idea is to match the biologi cal requirement of crop to physical characteristics of the land. In other words, the key to the future of crop forecasting is understanding the complexity of agricultural systems. They explore new factors such as plant genetic effects on soil function in context of climate change, selection of soil, micro-organisms in soil and plant soil feedback among others.[14]

2.2 Machine Learning

In computer science most problems can be solved by inputting a set of rules to the computer to execute and solves the problem. However, machine learning is a sub-field that focuses on finding the rules that solve the problem without implicitly or explicitly specifying the rules. These rules are learn from a set of data points to make predictions or perform decision making.[17] With recent increase in data availability and computer performance, machine learning can be found in everyday life. Machine learning is used in smart phones, self driving cars, consumer elec tronics, home appliances, manufacturing, and many other applications.[7] Within machine learning there exist two main types of algorithms: supervised learning and 6 Xunsupervised learning.

Supervised learning is much larger than unsupervised learning. The main idea behind supervised learning is to map and input to and output. The rules to map the input and output are learned. For example, given a set of input x and an output y, the pairs of data points would be use to learn rules; where D is the training set and N is the number of examples in the training set as shown in eq.2.2. Within supervised learning problems are divided into two categories; classification and regression. A classification problem has the goal to predict a discrete value that is mapped to a category. For example, given a image of a number 1 or number number 2 predict the number in the image. (Where $y \in \{1, 2\}$). A regression problem involves mapping an input to an output that is continuous in value. For example, given the value of a house x, predict the square footage y where $y \in \mathbb{R}$.

N

$T\ raining =$

$D(xi, yi)$ (2.2)

$i=1$

Unsupervised learning only considers the x input and the goal is to find the y output. The main goal is to find structure in data in order to help perform the desired task. The main two task are clustering and dimension reduction. Cluster ing is the task of clustering data points together based on features. For example, a dealership would like for people to be cluster together base on purchasing be haviors. Dimension reduction is the task of reducing the dimensions of data while maintaining information. For example, taking 3D data points and transforming them to a 2D data points while maintaining information on the data.

In machine learning problems are solved with a define objective function that is usually minimized. The object function also known as the cost function measures the error of the machine learning model's prediction from the ideal results. By7

X

1

|

–

|

measuring the error, the model updates the parameters to reduce the error of the cost function. The most basic objective/cost function in machine learning is the square mean error as shown in Eq. 2.3.

$MAE =$

n

y

$\hat{y}$

(2.3)

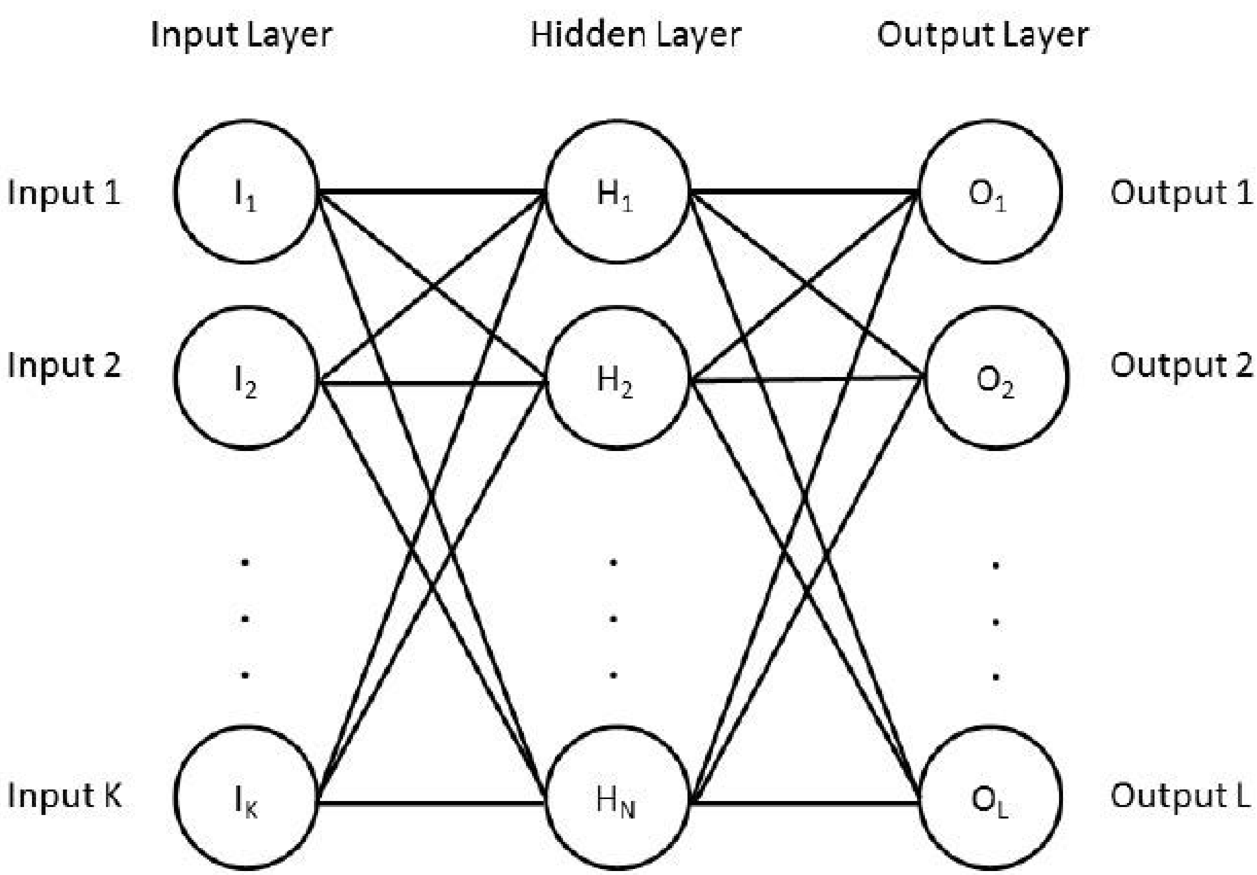
n

$n=1$

Most machine learning models update parameters by computing the gradient vector of objective/cost function with respect to the models parameters. The gra dient vector is the direction with the highest increase in the cost function, however since we are trying to minimize the vector in reverse by multiplying with a negative one. The new gradient is the direction in which the parameters are changed to have the greatest decrease of error of the objective/cost function. The negative gradient is then multiplied by a λ with a value between $0 < \lambda \leq$, which is the size of the step in that direction. This process describes a common optimization method in machine learning called Gradient Descent.[23]

2.2.1 Deep Learning

Deep learning is the most commonly used sub-field of machine learning in recent years. The raise of deep learning can be seen in many domains such as image recognition, speech recognition, classification, and many different forms of machine modeling. In contrast with classical machine learning, deep learning can perform many task with minimal domain expertise to design a data-set or select features. This allows for machine learning engineer applied models to many fields such as the medical field. The model itself would choose the most useful features. Machine Learning engineers can easily transform raw data into a machine learning model that can make decision and predict values. The deep learning model are8 compose of many layers that represent different learning layers. with each deeper layer, the model is able to represent the data into a more abstract level. [7] For example as shown in g 2.1, a deep learning base neural



2.2.2 Feedforward Neural Networks

For example as shown in g 2.1, Feed-forward Neural Networks which is deep learning base. The input in the left hand side is the features into the model. As the data propagates to the right, the model is able to abstract more abstract features out of the data features. Typically, the layers after the input are called the hidden layers, with the exception of theoutput layer. with each layer, we applied a none-linear function. Finally, in the end we have the output layer with a size of L. The size L9 can be the number of things that the model is trying to predict. [7]

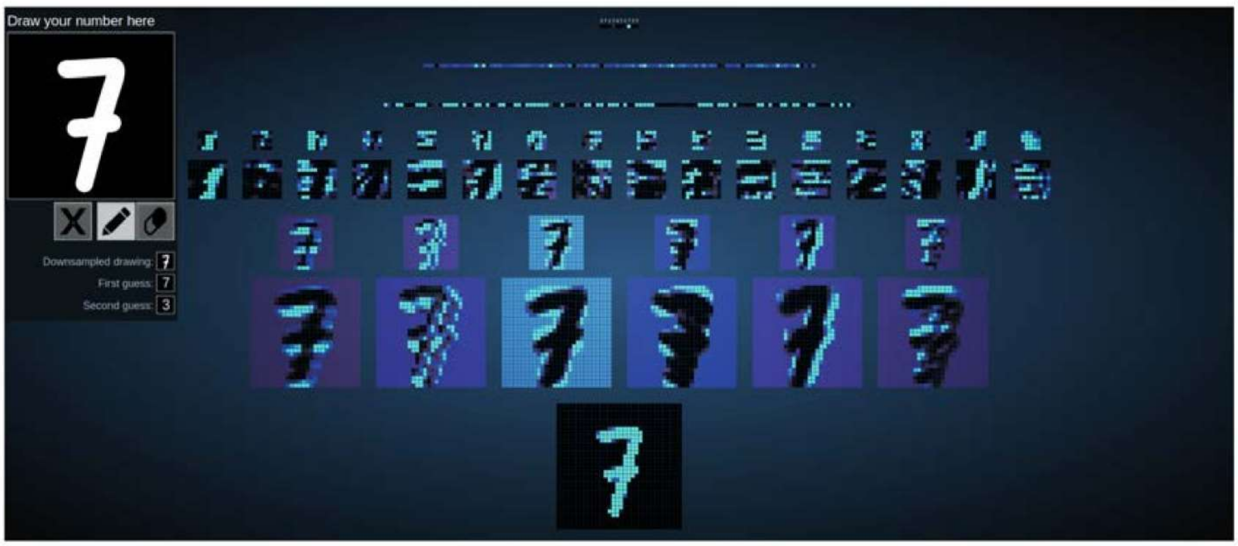


Figure 2.2: Convolutional Neural Network visualization of digit classification [8]

2.2.3 Convolutional Neural Networks

Convolutional neural networks (CNN) are inspired by biological process in all living things. CNN mimics the way the visual cortex process light from eyes and how is processed in thebrain. CNN became really popular in machine learning after the popular ImageNet competition of 2012. During that years competition, a CNN called alexnet was able to outperformed all other models by a large margin [5]. The competition consisted of training millions of images and predict 1,000 different classes out of these images. With the success of alexnet in the following years, all state of the art models for image recognition or detection task were CNN base. Today, some of these model have approached human performance in some task. The most basictaskdonewith CNNs is the classification of digits. Given an10 input number in a form of an image, identify the number in the image. as shown in fig, 2.2 you can see the visualization at the different levels of the CNN model. As you go deeper into the model it starts abstracting larger features of the image.[8]

2.2.4 Adam

Adam is an adaptive learning rate optimization algorithm that's been designed specifically for training deep neural networks[16]. The algorithm can be looked at

as a combination of RMSprop and Stochastic Gradient Descent with momentum.
Adam optimizer significantly speeds up the training time in deep learning models.

2.2.5 Batch Normalization

Batch normalization reduces the amount of variance in the hidden unit values shift around (co-variance shift). This is done by normalizing the subset of the data set values that are input into the neural network. This method can decrease the training time and increase the rate at which neural networks learn[12].

2.2.6 Faster RCNN

Faster R-CNN is an object detection algorithm from the family of the R-CNN algorithms. The Faster R-CNN is broken into two modules. The first module is a deep fully convolution network that proposes regions. The second module uses the proposed regions. The first module is called region Proposal Networks, which takes in any image of any size through a pre-trained convolutions network such as (VGG-16 model or inception)[25, 13]. The output is a set of rectangular object proposals. For example, 11 in an RPN of dimensions W × H each location can be the center for K possible anchors. By default, the model consider 3 scales and 3 aspect ratios, which gives K =9

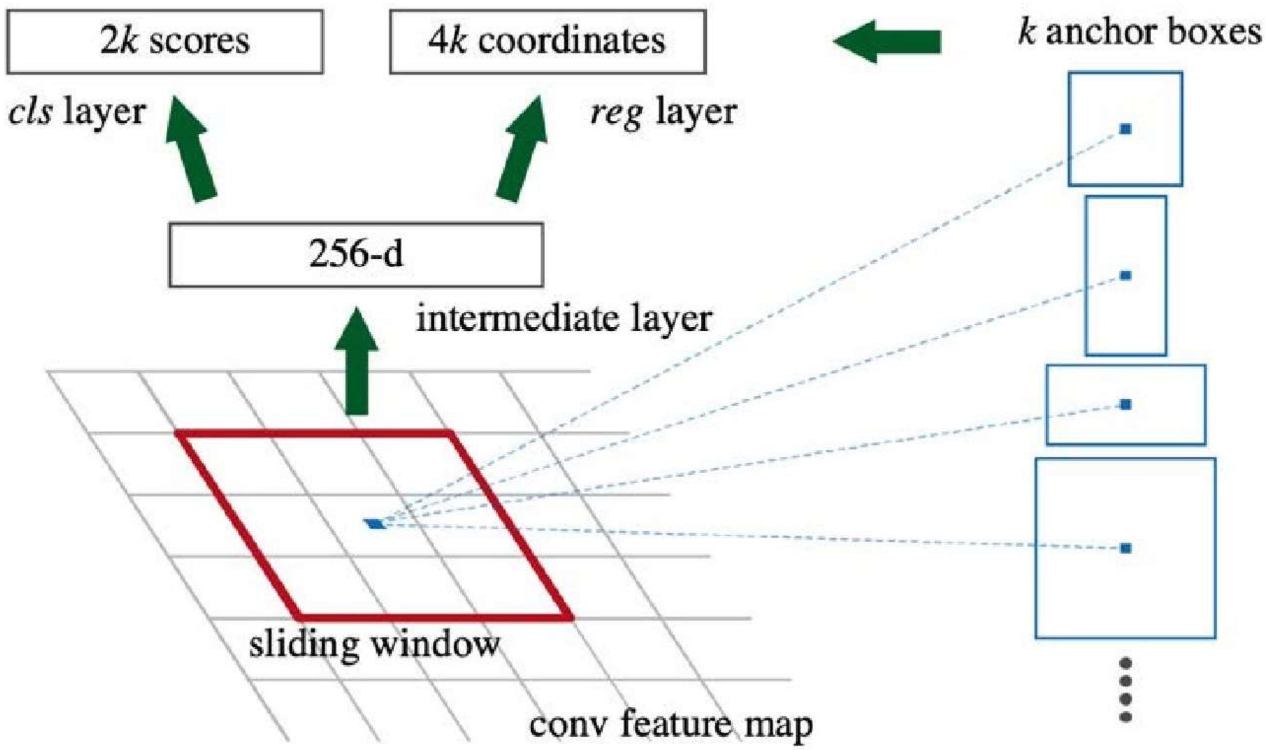


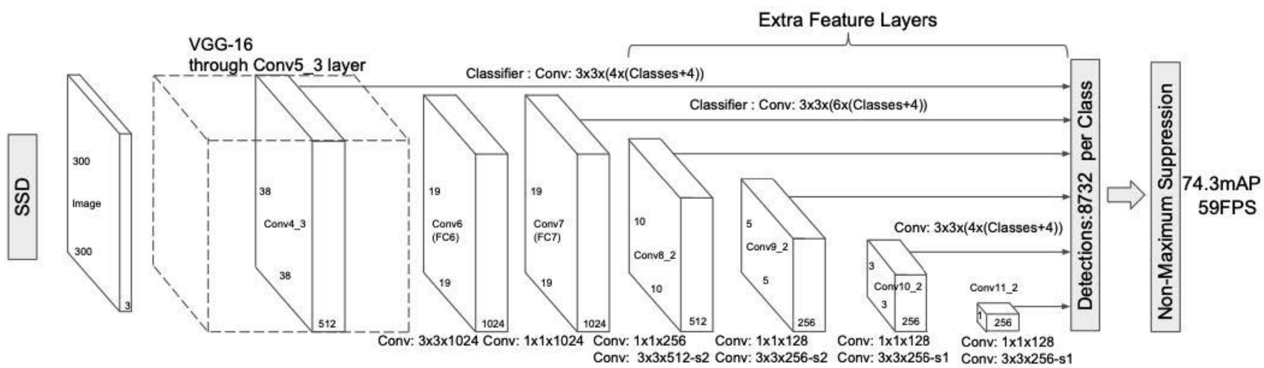
Figure 2.3: Region Proposal Network (RPN) [22]

A small convolution network window of dimensions n × n is slide over the RPN.

Each sliding network is then mapped to a lower dimensional vector(VGG-d 512 and

256-d for ZF). This features are fed into two sibling fully connected layers. One for an abjectness score and another for box regression. Each location can be the center for K possible anchors.

After applying it to the whole RPN, I choose 256 mini batch of anchors out of the RPN W × H × K . We try to maintain a balance between foregrounds(pos) and background(neg). The foregrounds are selected if the IoU is above 0.7. The backgrounds are selected if the IoU is lower than 0.3. If the regions is neither, then it is not consider for training purposes. During back propagation, the gradient is12



Problem Statement

In the assigned problem statement Figure 2.4: Single Shot MultiBox Detector architecture [28]

that measures the objectness of the computed bounding box. *Lconf* uses cross entropy as a loss function over the soft-max. The second term is location loss which measures the loss from the suggested bounding box to the truth. The total loss can be computed combining both terms as show in Eqs. 2.5.

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)) \quad (2.5)$$

During training, I determined which default bounding boxes correspond to truth bounding boxes. The default bounding boxes with IoU greater than 0.5 are a predicted bounding box. I also mine negative bounding boxes while trying to keep a ratio between negatives and positives of at most 3:1. The gradient is only computed through the selected default bounding box area.

2.2.8 You Only Look Once (YOLO)

Similarly to SSD, the You Only Look Once (YOLO) model applies a single neural network to the full image as shown in Fig. 2.5. YOLO model is significantly faster and is able to make object locality and classification in real time.

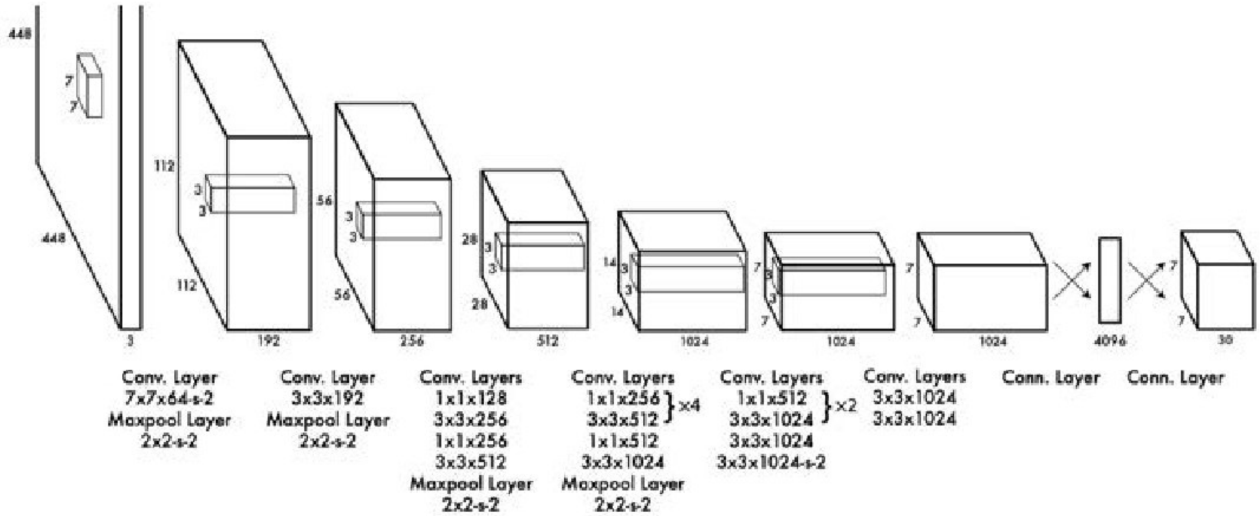


Figure 2.5: The YOLO architecture of 24 convolutional layers follow by two fully

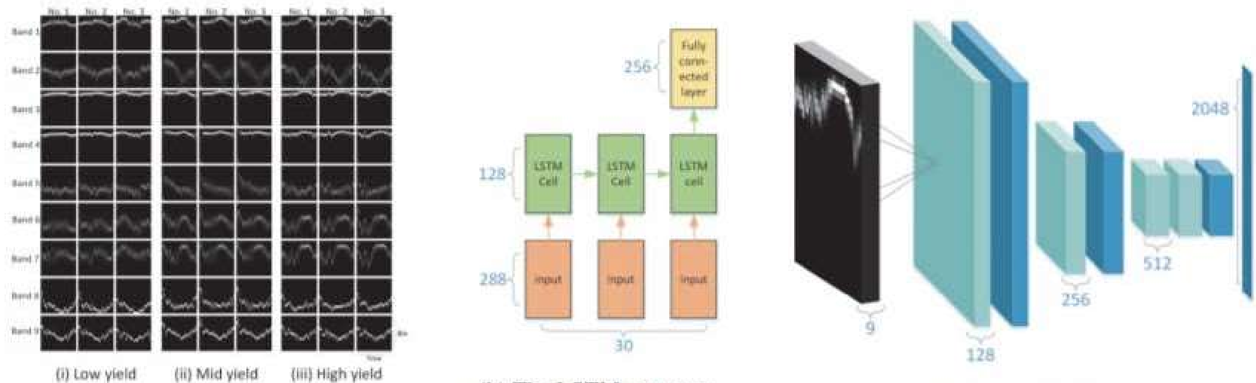
connected layers. [21]

number of boxes, confidence of those boxes and C class probability,which can be broken down to $S \times S \times (B \times 5 + C)$ tensors predicted per image. More specifically, each bounding box predicts: x,y,w,h and confidence. The (x,y) represent the center of the box relative to the bound of the grid. In contrast, (h,w) is predicted relative the the size of the image. Finally, the confidence prediction represents the IoU between predicted box and a truth box. Lastly, the grid predicts a classification of class.

During training we only want one bounding box predicted to be responsible for one object, since each grid can only predict one class. The loss function consist of three terms. The first term is the loss of predicting a given class. The second term is the loss of the predicted bounding box with respect to true bounding box. The third term is the confidence loss or the objectness of the box.

The model used includes a custom YOLOv3 architecture target towards detect ing smaller objects. Some of the changes in YOLOv3 included fitting anchors to our training data, reducing stride and adjusting which anchors size to focus in the15

YOLO model divides the image into an $S \times S$ grid. Each grid cell predicts fix



(a) 3-D histogram visualization
(b) The LSTM structure
(c) The CNN structure

Figure 2.6: architecture of the deep learning model to predict yield [29]

remote-sensing base methods. The architecture use several deep learning neural networks as shown in the fig. 2.6 The CNN helped abstract information from the image and feed it to a forward neural network which yield the ouput. The metric use to measure the accuary of the model is RMSE.
In the 2018 Syngenta Crop Challenge, Syngenta released several large datasets that recorded the genotype and yield performances of 2,267 maize hybrids planted in 2,247 locations between 2008 and 2016 and asked participants to predict the yield performance in 2017 [15]. The models used in the challenge were mainly deep learning approaches. In particular, the metrics used to measure accuracy were RMSE and correlation coefficient as shown in Figure 2.8.
While work has been done in large scales such as satellites imagery with imagery, very little work has been done in a smaller scale. In this paper I hope to combine some of the classical methods of collecting data by hand in conjunction with deep learning models to predict the yield of a crop. In hopes that in the future farmers could pick up pre-trained models and have an idea of their yield

Model	Response variable	Training RMSE	Training correlation coefficient (%)	Validation RMSE	Validation correlation coefficient (%)
DNN	Yield	10.55	88.3	12.79	81.91
	Check yield	8.21	91.00	11.38	85.46
	Yield difference	11.79	45.87	12.40	29.28
Lasso	Yield	20.28	36.68	21.40	27.56
	Check yield	18.85	28.49	19.87	23.00
	Yield difference	15.32	19.78	13.11	6.84
SNN	Yield	12.96	80.21	18.04	60.11
	Check yield	10.24	71.18	15.18	60.48
	Yield difference	9.92	58.74	15.19	11.39
RT	Yield	14.31	76.7	15.03	73.8
	Check yield	14.55	82.00	14.87	69.95
	Yield difference	17.62	21.12	15.92	5.1

The average ± standard deviation for yield, check yield, and yield difference are, respectively, 116.51 ± 27.7, 128.27 ± 25.34, and −11.76 ± 14.27. DNN, Lasso, SNN, and RT stand for deep neural networks, least absolute shrinkage and selection operator, shallow neural network, and regression tree, respectively.

Figure 2.7: Results form the Syngenta Crop Challenge by [15]

2.3.2 Weed Identification

Weed identification is the process of identifying weeds apart from crops. This has many application such as removal with an unmanned ground vehicle (UGV). In crop detection by machine vision for weed management (Pulido-Rojas et al) [20], the authors build an algorithm to detect weeds base on a threshold on the color green and size base feature extraction. This approach includes classical computer vision techniques that focus on color matching and geometric of leafs. The algorithm is based on classifying weeds based on the size of leafs. The algorithm performs well, however it lacks real run time analysis on weed detection and algorithm was developed to detect vegetables crops. The metrics used to measure performance were sensitivity and specificity.

Deep learning models started being used in precision agriculture and more specifically in identifying weeds. In this particular paper, the authors use a deep learning model to identify the various weed species in their natural environment. This was the first work that contributed to a large public, mutliclass image data set of weed species from the Australian range

lands.

[Explain your problem statement]

Figure 2.8: Results using the weed identification by [20]
The data set consisted of over 1,000 images collected for each class. There were a total of eight weed species in the data set from various locations as shown in Fig 2.9 . In order to balance any scene bias, an even split of positive and negative examples were collected from each location. The models used were inception and resnet. The performance across weed species was in the range of 80% to 90%. The reason why some species performed worse is due to the species having less unique visible features to train on. [18]

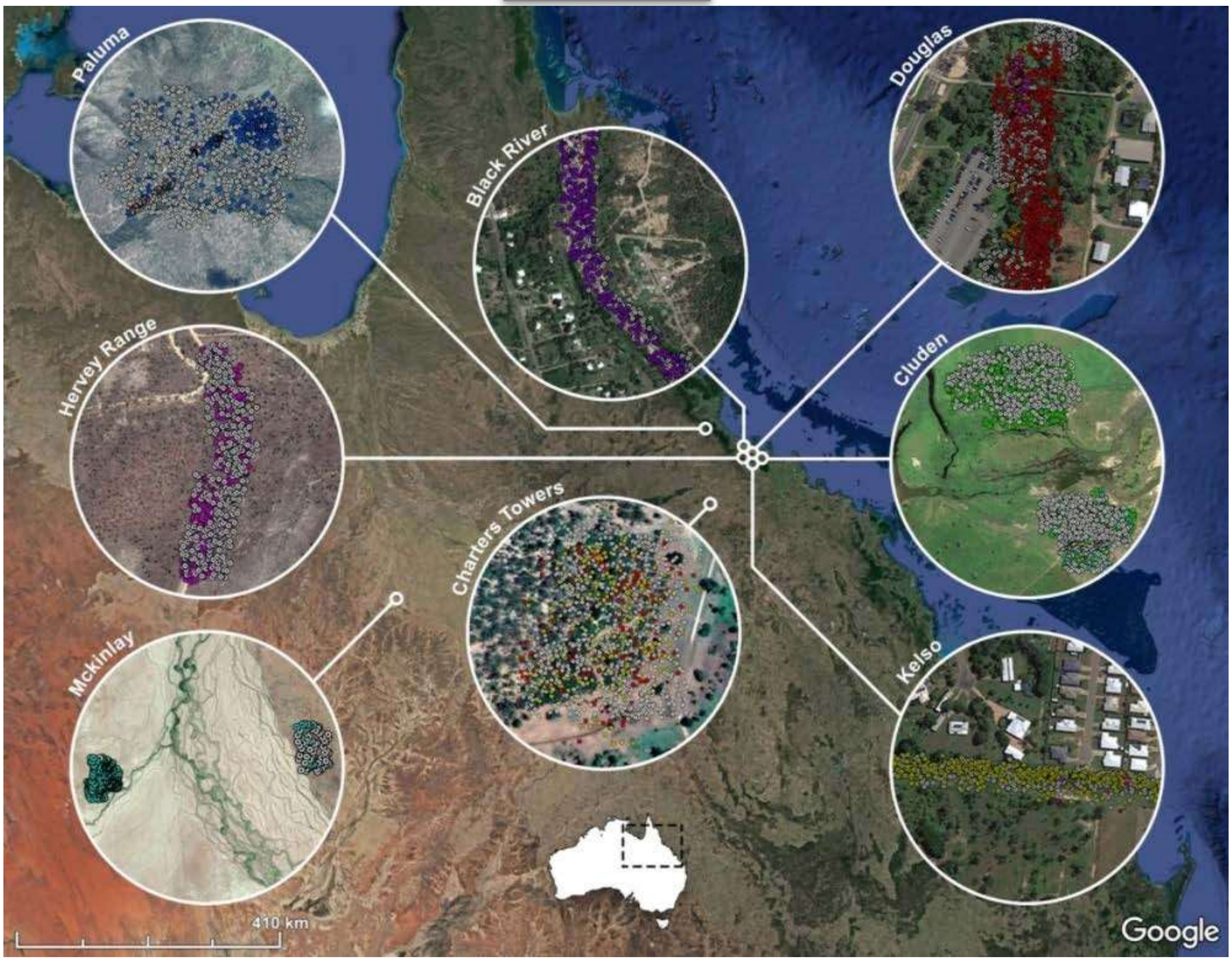


Figure 2.9: The varies weeds are color coded and from different locations

2.4 Metrics

2.4.1 Intersection Over Union (IoU)

Intersection Over Union (IoU) a metric to evaluate the accuracy of object detection. IoU measures how close the true bounding box is to a predicted bounding box generated by an algorithm. Let A be the true bounding box and B the predicted bounding box bu using Eqs.(2.7) IoU can be computed. If IoU is greater than 0.5 the bounding box predicted is consider to be a positive example.20

$IoU = |A \cap B|$

$|A \cup B|$

(2.7)

Recall and Precision

Recall is a metric used in modern object detection to evaluate the testing data set. The metric measures how many bounding boxes detected are recall from the true bounding boxes. The metric uses true positive (TP) bounding boxes and false negative (FN) bounding boxes. Recall can be computed by using Eqs. 2.8.

TP

Recall = (2.8)
|*TP* ∪ *FN* |
Precision is a metric that measures the ability of predicting bounding boxes that are relevant bounding boxes. The ratio of relevant bounding boxes is predicted as show in Eqs. 2.9 using true positives (TP) and false positives (FP).
TP
P recision =
(2.9)
|*TP* ∪ *FP* |21

Chapter 3

Research Goals

3.1 Goals

The goal of this work is to train algorithms to identify weeds and predict the yield of strawberry plants from imagery. The imagery is RGB images. The various models train for weed identify will be compare in term of accuracy and real time capabilities. I will perform an anaylsys on the data collected for predict plants' yield from imagery.

The Research goals are as follows:
1. Collect data set of RGB images of weeds in a plot of lettuce plants.
2. Collect data set of RGB images of strawberry plants and count number of strawberries per plant.
3. Train convolutional neural network to predict weeds from imagery and com pare different models.
4. Train convolutional neural network to predict yield of strawberry plants.22

Chapter 4 Methodology

4.1 Dataset Creation for Weed

4.1.1 Image Collection

The images were collected from the lettuce subplots with dimensions of 4' × 12' from a experimental plot [19] as shown in Fig. 4.1. The images collected from the third row of the subplots are selected as the testing set. The images collected from the remaining rows of the subplot were used as the training set. A total of four thousand images were collected for the training set and one thousand images for the testing set over a period of two weeks. The images were collected from two distinct angles. One of the angles is from the top of the plant pointing down. The second angle faces the lettuce plants sideways. The angles were chosen from possible angles that a robotic arm would be able to take images. The images that did not contain any weeds were discarded.

4.1.2 Image Resizing

The images were resized to 765× 1020 dimensions by using the resize function in openCV [4] . Bi-linear interpolation was applied to the images to conserve as much information when resizing the image. The size was chosen based on the input size appropriate for the Faster R-CNN. The same dimensions were used to train the models using SSD and YOLO. However, the inpu

Figure 4.2: Two processed images that are ready to be used for training.

4.1.3 Annotating Images

For annotating the images, LabelImg [26] was used. It is an open source tool widely use in the deep learning community to label bounding boxes on images. This tool allowed us to label bounding boxes around the weeds. Any visible weeds were labeled then resized the images. Most deep learning models struggle with bounding boxes smaller than dimensions of 32 × 32. Images with bounding boxes lower than 32 pixels were discarded as they were too small for learning and accurate prediction. This significantly increased the accuracy of the models. However, the weeds detected have to contain at least an area greater than 32 × 32 pixels, and25 bounding boxes with a smaller area can decrease the models performance.

Result: Processed image ready to be trained
Images = opencsv(images.txt);
for *image in Images* **do**
img = open(image,color) ;
oldSize=img.size img = resize(img,(765,1020)) ;

```
annotation = image.split(".")+"txt";
boundingBoxes = open(annotation);
NewBoundingBoxes = "";
for boundingbox in boundingBoxes do
xmin, ymin, xmax, ymax=boundingbox.strip()
xmin = xmin×(765 ÷ oldsize[0])
xmax = xmax(765 ÷ oldsize[0])
ymin = ymin(1020 ÷ oldsize[1])
ymax = ymax(1020 ÷ oldsize[1])
if dist(xmin, xmax)≥ 32 and dist(ymin, ymax)≥32) then
NewBoundingBoxes.append(boundingbox);
end
end
save("result/"+image,img);
save("result/"+annotation,NewBoundingBoxes);
end
```

Algorithm 1: The processing pipeline for images.26

4.1.4 Processing Images

The images were processed using all the method mentioned above using the pipeline as shown in Algorithm 1. The algorithm also resized the bounding boxes according to the images resized. Most object detection algorithms are only able to detect bounding boxes that are greater than 32× 32 pixels in each dimension. The end result is the images and labels that were the input to the training for the object detection models.

4.2 Weed Identification training

The models take in images and notations that were previously processed. SSD and Faster R-CNN are open and available through the Tensorflows object detection library. [1] The models are available with multiple pre-trained weights and different architectures. However, the more parameters the model contains the slower the detections are able to processed. These models support GPU training. YOLO is available through the YOLO library.[21] There are several YOLO architectures available. In this paper, YOLOv3 is used to train on the images collected. YOLO also supports GPU training which is essential for our purpose.

The models trained on Google colab. Google colab is a linux based GPU envi ronment that is available free of cost. Colab supports the installation of libraries for YOLOv3, SSD, and faster R-CNN. These models can be run in a jupyter note books, widely available for research purposes and project purposes. as shown in Fig. 4.3 some of the architecture available in the YOLO library

Model	Train	Test	mAP	FLOPS	FPS	Cfg	Weights
SSD300	COCO trainval	test-dev	41.2	-	46		link
SSD500	COCO trainval	test-dev	46.5	-	19		link
YOLOv2 608x608	COCO trainval	test-dev	48.1	62.94 Bn	40	cfg	weights
Tiny YOLO	COCO trainval	test-dev	23.7	5.41 Bn	244	cfg	weights
SSD321	COCO trainval	test-dev	45.4	-	16		link
DSSD321	COCO trainval	test-dev	46.1	-	12		link
R-FCN	COCO trainval	test-dev	51.9	-	12		link
SSD513	COCO trainval	test-dev	50.4	-	8		link
DSSD513	COCO trainval	test-dev	53.3	-	6		link
FPN FRCN	COCO trainval	test-dev	59.1	-	6		link
Retinanet-50-500	COCO trainval	test-dev	50.9	-	14		link
Retinanet-101-500	COCO trainval	test-dev	53.1	-	11		link
Retinanet-101-800	COCO trainval	test-dev	57.5	-	5		link
YOLOv3-320	COCO trainval	test-dev	51.5	38.97 Bn	45	cfg	weights
YOLOv3-416	COCO trainval	test-dev	55.3	65.86 Bn	35	cfg	weights
YOLOv3-608	COCO trainval	test-dev	57.9	140.69 Bn	20	cfg	weights
YOLOv3-tiny	COCO trainval	test-dev	33.1	5.56 Bn	220	cfg	weights
YOLOv3-spp	COCO trainval	test-dev	60.6	141.45 Bn	20	cfg	weights

Model	Train	Test	mAP	FLOPS	FPS	Cfg	Weights
SSD300	COCO trainval	test-dev	41.2	-	46		link
SSD500	COCO trainval	test-dev	46.5	-	19		link
YOLOv2 608x608	COCO trainval	test-dev	48.1	62.94 Bn	40	cfg	weights
Tiny YOLO	COCO trainval	test-dev	23.7	5.41 Bn	244	cfg	weights
SSD321	COCO trainval	test-dev	45.4	-	16		link
DSSD321	COCO trainval	test-dev	46.1	-	12		link
R-FCN	COCO trainval	test-dev	51.9	-	12		link
SSD513	COCO trainval	test-dev	50.4	-	8		link
DSSD513	COCO trainval	test-dev	53.3	-	6		link
FPN FRCN	COCO trainval	test-dev	59.1	-	6		link
Retinanet-50-500	COCO trainval	test-dev	50.9	-	14		link
Retinanet-101-500	COCO trainval	test-dev	53.1	-	11		link
Retinanet-101-800	COCO trainval	test-dev	57.5	-	5		link
YOLOv3-320	COCO trainval	test-dev	51.5	38.97 Bn	45	cfg	weights
YOLOv3-416	COCO trainval	test-dev	55.3	65.86 Bn	35	cfg	weights
YOLOv3-608	COCO trainval	test-dev	57.9	140.69 Bn	20	cfg	weights
YOLOv3-tiny	COCO trainval	test-dev	33.1	5.56 Bn	220	cfg	weights
YOLOv3-spp	COCO trainval	test-dev	60.6	141.45 Bn	20	cfg	weights

Model	Train	Test	mAP	FLOPS	FPS	Cfg	Weights
SSD300	COCO trainval	test-dev	41.2	-	46		link
SSD500	COCO trainval	test-dev	46.5	-	19		link
YOLOv2 608x608	COCO trainval	test-dev	48.1	62.94 Bn	40	cfg	weights
Tiny YOLO	COCO trainval	test-dev	23.7	5.41 Bn	244	cfg	weights
SSD321	COCO trainval	test-dev	45.4	-	16		link
DSSD321	COCO trainval	test-dev	46.1	-	12		link
R-FCN	COCO trainval	test-dev	51.9	-	12		link
SSD513	COCO trainval	test-dev	50.4	-	8		link
DSSD513	COCO trainval	test-dev	53.3	-	6		link
FPN FRCN	COCO trainval	test-dev	59.1	-	6		link
Retinanet-50-500	COCO trainval	test-dev	50.9	-	14		link
Retinanet-101-500	COCO trainval	test-dev	53.1	-	11		link
Retinanet-101-800	COCO trainval	test-dev	57.5	-	5		link
YOLOv3-320	COCO trainval	test-dev	51.5	38.97 Bn	45	cfg	weights
YOLOv3-416	COCO trainval	test-dev	55.3	65.86 Bn	35	cfg	weights
YOLOv3-608	COCO trainval	test-dev	57.9	140.69 Bn	20	cfg	weights
YOLOv3-tiny	COCO trainval	test-dev	33.1	5.56 Bn	220	cfg	weights
YOLOv3-spp	COCO trainval	test-dev	60.6	141.45 Bn	20	cfg	weights

4.3 Data-set Creation for Strawberry’s Yield

4.3.1 Image Collection

Images of strawberries were collected from top angle every three days from different plants. The number of strawberries per plan was collected during the collection of the images. This will be used to assess the model performance with the ground truth. The ground data was stored in a spread sheet that was then mapped to a list of images that map one to one.

4.3.2 Processing

The images collected contain plants of strawberries and weeds around the strawber ries. The images were cut according to the bounding boxes, using the same images annotated with labelimage, in order to have the strawberry plant make up most of the image. The images were resized to 64 × 64 pixels using bi-linear interpolation in order to be used in the input of the Convolutional neural network. The resulting image is the input to the different models that we will be using.

4.4 Yield Prediction Training

4.4.1 Baseline

The architecture shown in Fig. 4.4 would be reconstructed using tensor-flow back end (Keras) [1]. The model performs regression on the data collected from the ground for each plant image and its corresponding yield. The model will be using the mean absolute error as the loss function. The model batch size is eight and

Table 5.2: Metric results based on testing set predictions by SSD,YOLO and Faster R-CNN.
Model
Feature Extractor AP@.5oU Recall@0.5IoU Spped (FPS)
Faster R-CNN Inception V2 [13] **65.5**
80
7
SSD MobileNet [11] 38.3 62
59
YOLOv3
VGG16 [25] 49.5
86
45

5.2.1 Faster RCNN Analysis

Figure 5.3 shows the labeled bounding boxes and the predicted bounding boxes of Faster R-CNN model of an image with various sizes of bounding boxes. The predictions closely resemble the true bounding boxes due to a high AP. The model also predicted two unlabeled weed leaves as shown in Figure 5.3(a) on the top right and right above the second lettuce from left. The model struggles predicting smaller weeds with a high percentage of background in the bounding box. The figure 5.4 is use to assets the performance with weeds with some distance form the camera. The performance is good only missing the smallest weed. Figure 5.5 was use to assets the performance of when a lot of weeds overran the plant. The performance good with medium weeds but performs poorly with large weeds and small weeds. Finally figure 5.6 show the side angle with small weeds and the model performed better with smaller weeds from this angel. These are some of the images that show the models performance under different conditions. The model performed well in all conditions the images were taken. However, Faster R-CNN is not capable of real-time predictions.35

5.2.2 SSD Analysis

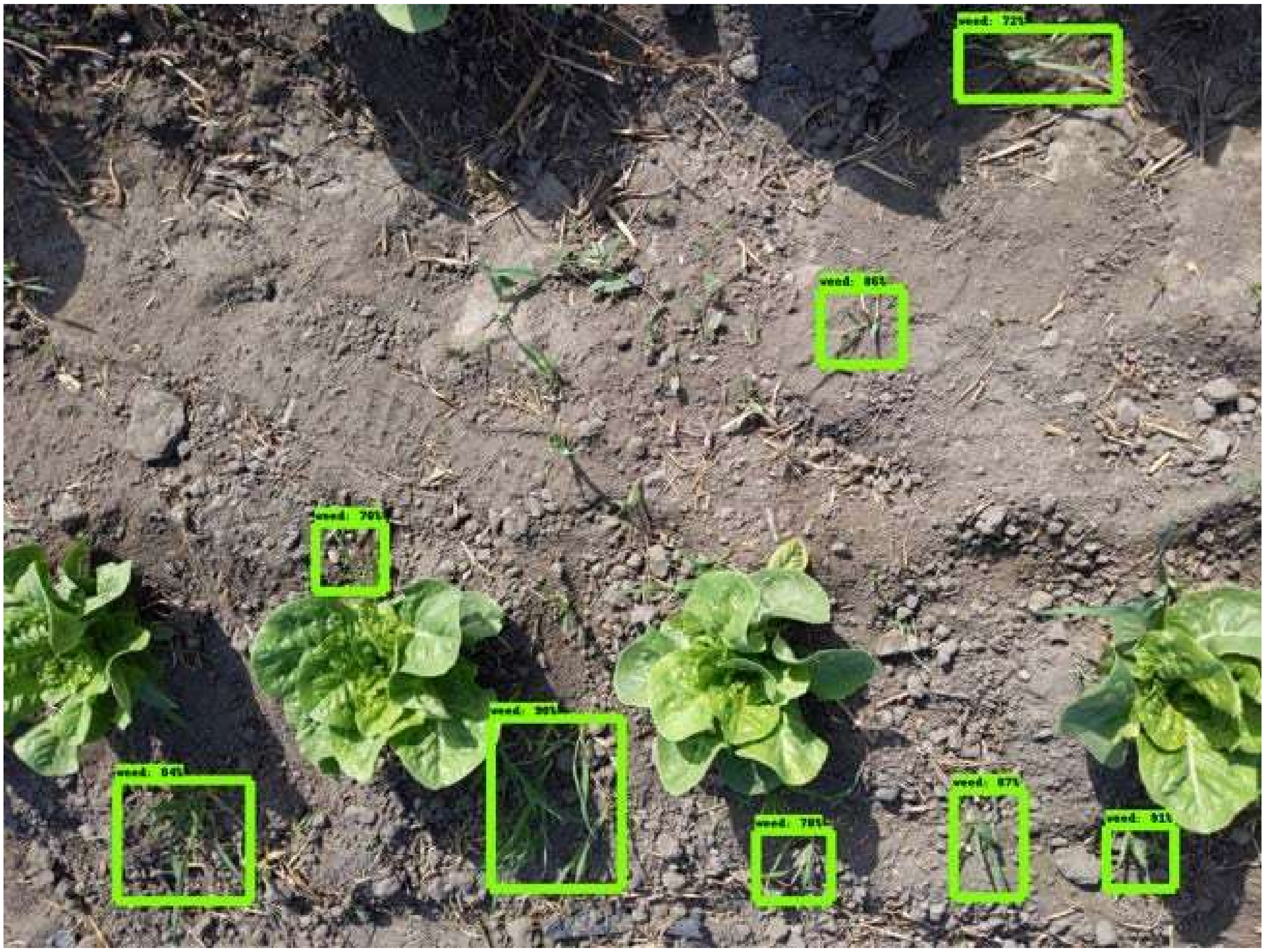
Figure 5.7 shows the labeled bounding boxes and the predicted bounding boxes of SSD model of an image with various sizes of bounding boxes. One observation that can be made is that the predictions made were only of weeds that were dense with leaves. Moreover, the Figure 5.8 is use to assets the performance with weeds with some distance form the camera. The model has a very similar performance to the Faster RCNN, while being capable of detecting in real time. The Figure 5.9 shows weeds starting to overran the lettuce. Surprisingly, the SSD perform the best out of all the models under this condition and I believe is due to the models architecture. Finally figure and 5.10 show the side angle with small weeds. The model detected the same weeds as the Baseline model. Overall the model performed poorly in bounding boxes that had a low ratio of weed to background. The model performed better than expected considering the trade off on accuracy vs speed.

5.2.3 YOLO Analysis

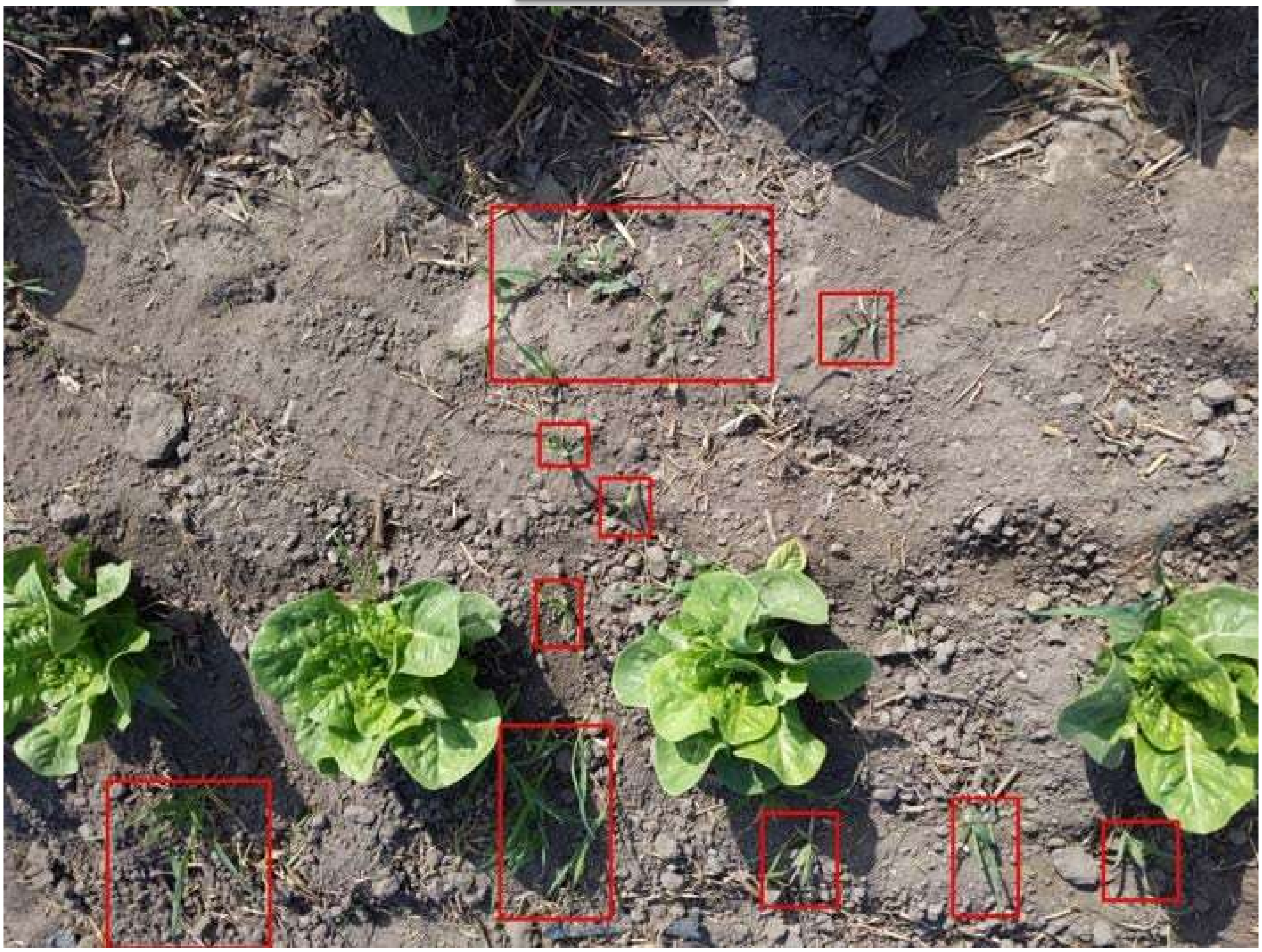
Figure 5.11 shows the labeled bounding boxes and the predicted bounding boxes of the YOLO model of an image with various sizes of bounding boxes and the performance is similar to the Faster RCNN. The custom architecture performs well with smaller objects than the regular YOLO architecture. However, Figure 5.12 shows image of weeds at a distance and performed poorly in this specific image. The model is not consistent in detecting smaller weeds across all images. Figure 5.13 shows weeds starting to overran the lettuce, and the performance is good similar to Faster RCNN. In this figure, it can be noted that the model does struggle with very small detection that even with the naked eye is hard to detect. The model struggle with really small plants due to its direct connected network. lastly, 5.14 shows the36 side angle with small weeds and the model performed the best with medium size weeds. The Recall was higher than the Faster R-CNN model probably due to high accuracy with medium/smallish size weeds, which is an impressive task considering YOLO is capable of real-time predictions.

5.2.4 Comparative Analysis

It is difficult to analyze models based on pure images or metrics. However, there seems to be a trend that Yolo and SSD struggle with detection of really small weeds. Moreover, SSD performed well when there were a large number of bounding boxes in the image. This is possible due to the way SSD mines positive samples and negative samples from the image. In contrast, Yolo is fully connected from image to the neural network. YOLO performed the best with semi small to medium weeds, which compress most of the weeds in the images. As a result, YOLO had a higher recall percentage than the other models. to s The Faster RCNN performed the best, but lacks the speed that we are looking



(a) Fater R-CNN weed predictions on testing set image.



(b) True bounding boxes on unseen data.

5.3 Strawberry Yield Prediction

Table 5.3: The results for the baseline model and the deep learning model for yield prediction. Model Mean Absolute Error
Linear Regression (Baseline) 0.185664
Deep Neural Network 0.15852
In Table 5.3, we see the results of the linear regression and the deep neural network on predicting yield per plant on the testing set. Both of the models had a low error. The low error does not mean high accuracy. the accuracy is high due to the data having low variance as shown in Table 5.1 due to crop being overrun by weeds. However, the baseline results shown in Fig. 5.15 and the neural network in Fig. 5.16. The data points predicted by deep learning model are closer to the grown truth shown in the graphs as the green line. deep learning model has a lower error than the linear regression. We can conclude that deep learning model is able to learn more

features out of the convolutions network as we compare the
Both weren't able to predict plants with high yield, probably due to a combination of strawberries under leafs and low number of plants with high yield in the data-set. Another observation in the graph is that plants with low number of strawberries were predicted with a much higher number of strawberries in the deep learning model. In terms of visualization of the input image of size 64 × 64 pixels, plants with two or more strawberries often sprout next to each other, which often look as one strawberry. Since there were more plants with two or more strawberries due to low variance, the model learned a bias toward higher predictions. However, due to the low variance in the data, we can't concluded that the models are able to

(a) reliably predict yield per plant.

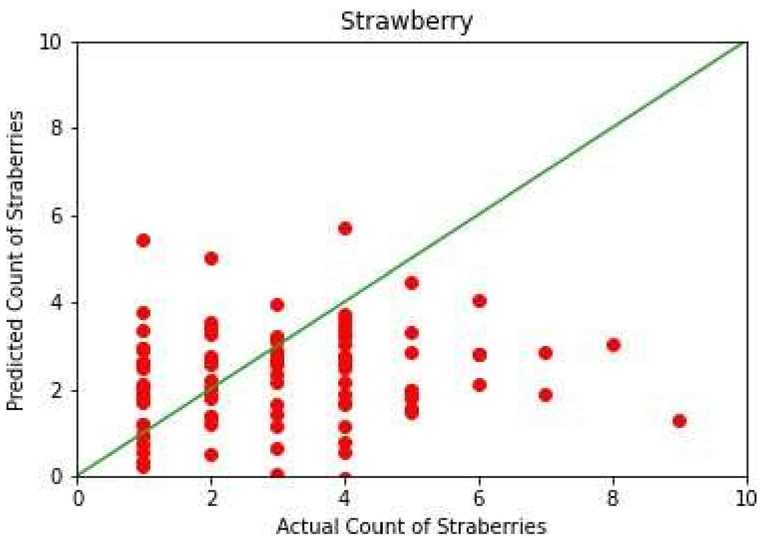


Figure 5.16: The results on the testing set for the deep learning model.

Chapter

Conclusion

In this work, we developed methodology for detecting invasive weeds in lettuce plants from images and video footage. There were multiple models tested and it was found that modern objected detection models are capable of detecting weeds with high accuracy and high recall. We were able to learn some of the key weaknesses such as not being able to detect small weeds. Given these results, we are able to concluded that real time object detection models are capable of detecting weeds with the similar accuracy as models that do not run in real time. Next, we found that yield prediction is more accurately predicted from deep learning base models. he model was able to predict yield of data with low variance and very early in their growing stage. However, the prediction from imagery struggled to predict high yield plants. In such case, these models have the potential of replacing classical metting .

Chapter 6
Future Work

There is still much more work to be done to improve the models and develop models that can be use daily. First, we now understand the type of angles that objected detection needs in order to detect weeds accuracy. Second, we learned that small weeds are hard to detect so we can improve the models by taking high resolution images and breaking them into smaller images. Another solution is to take images closer for small weeds so they appear larger in the image. We tried newer versions of SSD and YOLOv3 such as the YOLOv4 announce in 2020 and SSD512. The model can potentially be used with imagery from drones. This would allowed to also find thelocation of the weed. In our yield prediction results, we can automate the bounding box labeling with a object detection model similar to our weed detection. This would allowed for faster processing. Moreover, imagery from drone can be used to collect images to predict the yield. Drone imagery could contain multiple plants per image, this way we can predict yield by square feet on multiple plants. Moreover, This can potentially be done in real-time as we can have a real time object detection model detecting rows of plants then passing that part of the image 52into a neural network to detect the yield if done using dronne

Bibliography

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Cor rado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Man´e, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Vi´egas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous sys tems, 2015. Software available from tensorflow.org.

[2] R. Agrawal. Forecasting techniques in crops. *Indian Agricultural Statistics Research Institute*, 2006.

[3] J. Alsakran, A. Rodan, N. Alhindawi, and H. Faris. Visualization analysis of feed forward neural network input contribution. *Scientific Research and Essays*, 9:645–651, 07 2014.

[4] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
54[6] R. A. Fisher. The influence of rainfall on the yield of wheat at rothamsted. *Journal of Biological Science*, 213:402–410, Jan 1925.
[7] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. The MIT Press, 2016.
[8] A. W. Harley. An interactive node-link visualization of convolutional neural networks. In *ISVC*, pages 867–877, 2015.

[9] P. Henderson and V. Ferrari. End-to-end training of object class detectors for mean average precision. *CoRR*, abs/1607.03476, 2016.

[10] T. Horie, M. Yajima, and H. Nakagawa. Yield forecasting. *Agricultural Sys tems*, 40(1):211 – 236, 1992. Systems approaches for agricultural.

[11] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural net works for mobile vision applications, 2017. cite arxiv:1704.04861.

[12] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

[13] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

[14] Y. Jame and H. Cutforth. Crop growth models for decision support systems. *Canadian Journal of Plant Science*, 76, 01 1996.

[15] S. Khaki and L. Wang. Crop yield prediction using deep neural networks. *Frontiers in Plant Science*, 10:621, 2019

2.7 Code submission (Github link)

2.8 Report submission (Github link) : first ma