PATTERN RECOGNITION OF LETTUCE VARIETIES WITH MACHINE LEARNING

SERI 'AISYAH BINTI HASSIM

SUBMITTED TO THE

FACULTY OF ENGINERRING

UNIVERSITY OF MALAYA, IN PARTIAL

FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF ENGINEERING

(MECHATRONICS)

2019

UNIVERSITY MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: SERI 'AISYAH BTE HASSIM

Registration/Matric No: KQF170004

Name of Degree: MASTER OF ENGINEERING (MECHATRONICS)

Title of project Paper / Research Report: PATTERN RECOGNITION OF LETTUCE VARIETIES WITH MACHINE LEARNING

Field of Study: Mechatronics Engineering

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of work;

(2) This work is original;

(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any except or extract from, or reference to or reproduction of any copyright work and its authorship have been acknowledged in this work;

(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;

(5) I hereby assign all and every rights in the copyright to this work to the University Malaya("UM"), who henceforth shall be the owner of the copyright this work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;

(6) I am fully aware that if in the course of making this work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature: Date:

Subscribed and solemnly declared before,

Witness's Signature: Date:

Name:

Designation:

ABSTRACT

Determination of lettuce varieties through image processing is considered as part of precision farming. Automatic classification is becoming vital for precision farming practice as it is rapidly developing with emergence of many applications for agriculture. It is a hassling process to differentiate and identify the lettuce varieties through human capabilities as it is time consuming and prone to errors in identification process. Hence, there is a need to do this assisted by a machine capability which makes it faster with greater accuracy. Application of machine learning in agricultural is still not widely applied and many phases need to be improved. Differentiation of lettuce varieties with colour or shape similarity is quite challenging. This study focuses on designing the lettuce varieties recognition by using Convolution Neural Network (CNN) in MATLAB. The neural network model consists of layers such as Convolution Layer, Normalization Layer, ReLU Layer, Fully Connected Layer, Softmax Layer, and Classification Layer. The network needs to undergo training sessions before being able to recognize the lettuce varieties. A set of data are prepared for prediction after training. The accuracy for overall classifications is 94.4% while accuracy for specific lettuce varieties of Butterhead Lettuce, Celtucelove Lettuce, Italian Lettuce, Red Coral Lettuce, Red lettuce, Red Oakleaf Lettuce and Salad Grand Rapid Lettuce were 94.7%, 99.7%, 97%, 94%, 90.7%, 98%, 87% respectively.

ABSTRAK

Penentuan variati salad melalui pemprosesan imej dianggap sebagai sebahagian daripada pertanian tepat. Pengkelasan secara automatik menajadi penting untuk pertanian tepat kerana ianya berkembang pesat dengan kemunculan aplikasi dalam bidang pertanian. Ia merupakan proses yang rumit untuk membezakan dan mengenalpasti variati salad melalui keupayaan manusia kerana mengambil masa yang lama dan berkemungkinan terdapat kesilapan dalam proses pengkelasan tersebut. Oleh itu, ianya menjadi satu keperluan untuk melakukan proses ini dan dibantu oleh keupayaan mesin yang dapat menjadikan proses ini lebih cepat dan berketetapan yang lebih tinggi. Aplikasi yang menggunakan *machine learning* dalam bidang pertanian masih tidak digunakan secara meluas dan terdapat beberapa fasa yang perlu diperbaiki. Proses membezakan variati salad melalui bentuk mahupun warna adalah sangat mencabar. Dalam kajian ini, ianya akan lebih fokus untuk mereka bentuk pengecaman variati salad berdasarkan teknik Rangkaian Konvolusi Neural (CNN) menggunakan perisian MATLAB. Rangkaian ini terdiri daripada *Convolution Layer*, *Normalization Layer*, *ReLU Layer*, *Fully Connected Layer*, *Softmax Layer*, dan *Classification Layer.* Rangkaian ini akan melalui satu proses latihan sebelum ianya dapat mengenalpasti variati salad. Beberapa set data akan disediakan untuk rangkaian melakukan proses ramalan selepas sesi latihan. Secara keseluruhannya, ketepatan pengkelasan untuk semua variati salad adalah 94.4% manakala ketepatan untuk setiap satu variati salad *Butterhead*, *Celtucelove*, *Italian*, *Red Coral*, *Red*, *Red Oakleaf* dan *Salad Grand Rapid* masing-masing adalah 94.7%, 99.7%, 97%, 94%, 90.7%, 98%, 87%.

ACKNOWLEDGEMENT

First of all, I would like to express my gratitude to God that by His will and blessing, my project research was carried out smoothly. Not to forget special mention to my employer, Malaysian Agriculture and Research Development (MARDI) for sponsoring and providing me an opportunity to further my studies. I hope that my studies and research project with University of Malaya (UM) is able to provide small research contribution towards the field of Artificial Intelligence for modernizing our agricultural sector. I would also like to thank UM for providing me a great platform such as wonderful study environment, facilities and lecturer for sharing valuable knowledge, experience, information and ideas throughout my studies.

Over the course of this project, the most significant person whom I appreciated and thankful for is my supervisor, Ir. Dr. Chuah Joon Huang for kindness in sharing his knowledge about Artificial Intelligence (AI) and ideas relating to agricultural sector.

Not to forget, special thanks to all my friends who have been assisting me and sharing their knowledge and willing to sacrifice their precious time to help me and learn together during my study in UM. Thank you to my colleagues and support staff at MARDI who were willing to help me to cultivate the seven varieties of lettuce and to give an idea to improve my research project as well as applicable for further research. Most importantly to my parents and family, for praying and giving their blessing to me. Thanks for simply being there for me and their sacrifice and unlimited contribution towards me.

TABLE OF CONTENTS

ABSTRACT……………………………………………………………………………iii
ABSTRAK iv
ACKNOWLEDGEMENT v
TABLE OF CONTENTS vi
LIST OF FIGURES viii
LIST OF TABLES xi
LIST OF APPENDICIES..xiii
LIST OF ACRONYMS xii
CHAPTER 1: INTRODUCTION 1
1.1 Introduction 1
1.2 Research Background 1
1.3 Problem Statement 2
1.4 Objectives 2
1.5 Aim 2
1.6 Scope of Research 2
CHAPTER 2: LITERATURE REVIEW 3
2.1 Introduction 3
2.2 Artificial Intelligence Overview 3
2.3 Image Recognition in Agriculture sector: Overview 5
2.4 Convolutional Neural Network (CNN) 13
2.3 Summary 19
CHAPTER 3: METHODOLOGY 20
3.1 Introduction 20
3.2 Variables 20
3.2.1 Constant Variables 20
3.2.2 Measured Variable 21
3.3 Recognition Method 22
3.4 Image Acquisition 23
3.5 Image Pre-processing 26
3.6 Convolutional Neural Network Architecture 27
CHAPTER 4: RESULT AND DISCUSSION 32
4.1 Introduction 32
4.2 Results 32
4.2.1 Leaf Lettuce Recognition 32

4.2.2 Bouquet Lettuce Recognition..46
4.3 Summary 48
CHAPTER 5: CONCLUSION AND FUTURE WORK 49
REFERENCES..50

LIST OF FIGURES

Figure 2.1: Some definitions of artificial intelligence by (S. J. Russell and P. Norvig. 2010) 3

Figure 2.2: The classification of Artificial Intelligence by (A. Pannu, 2015) 4

Figure 2.3: The comparison of CNN architectures for different image size. (M. Dyrmann, 2017) 9

Figure 2.4: The structure of Convolutional Neural Network 14

Figure 2.5: Architecture of LeNet-5. (Yann LeCun et al., 1998) 14

Figure 2.6: Architecture of AlexNet. (Krizhevsky et al., 2017) 15

Figure 2.7: Architecture of VGGNet. 16

Figure 2.8: Comparison method of VGGNet with other method. (Simonyan and Andrew, 2015) 16

Figure 2.9: Architecture of GoogleNet. (Szegedy et al., 2015) 17

Figure 2.10: The architecture of ResNet. (Alex Krizhevsky, 2017) 17

Figure 2.11: Architecture of ResNeXt. 18

Figure 3.1: An example of confusion matrix chart for leaf lettuce recognition 21

Figure 3.2: Flow of the project 23

Figure 3.3: Lettuce images during growth 23

Figure 3.4: Training dataset folder for leaf lettuce 26

Figure 3.5: Sample images before and after resizing 27

Figure 3.6: Visualization of layer 28

Figure 3.7: An example of Pooling layer down samples the volume spatially 28

Figure 3.8: Softmax graph 29

Figure 3.9: Training option of the network 30

Figure 4.1: Training progress with mini batch size 80 and number of epoch 10............33

Figure 4.2: Confusion matrix for mini batch size 80 and number of epoch 10. 33

Figure 4.3: Random image from testing file after training the model 34

Figure 4.4: Training progress with mini batch size 80 and number of epoch 15............34

Figure 4.5: Confusion matrix for mini batch size 80 and number of epoch 15 35

Figure 4.6: Random image from testing file after training the model 35

Figure 4.7: Training progress with mini batch size 80 and number of epoch 20............36

Figure 4.8: Confusion matrix for mini batch size 80 and number of epoch 20 36

Figure 4.9: Random image from testing file after training the model 37

Figure 4.10: Training progress with mini batch size 64 and number of epoch 10..........37

Figure 4.11: Confusion matrix for mini batch size 64 and number of epoch 10 38

Figure 4.12: Random image from testing file after training the model 38

Figure 4.13: Training progress with mini batch size 64 and number of epoch 20..........39

Figure 4.14: Confusion matrix for mini batch size 64 and number of epoch 20 39

Figure 4.15: Random image from testing file after training the model 40

Figure 4.16: Training progress with mini batch size 40 and number of epoch 15..........40

Figure 4.17: Confusion matrix for mini batch size 40 and number of epoch 15 41

Figure 4.18: Random image from testing file after training the model 41

Figure 4.19: Training progress with mini batch size 40 and number of epoch 10..........42

Figure 4.20: Confusion matrix for mini batch size 40 and number of epoch 10 43

Figure 4.21: Random image from testing file after training the model 43

Figure 4.22: Training progress with mini batch size 32 and number of epoch 10..........44

Figure 4.23: Confusion matrix for mini batch size 32 and number of epoch 10 44

Figure 4.24: Random image from testing file after training the model 45

Figure 4.25: Result of leaf lettuce recognition..46

Figure 4.26: Training progress with mini batch size 20 and number of epoch 15..........46

Figure 4.27: Confusion matrix for mini batch size 20 and number of epoch 15 47

Figure 4.28: Random image from testing file after training the model 47

LIST OF TABLES

Table 2. 1: Applications of deep learning in agriculture. (Andreas Kamilaris1 and Francesc X., 2017) 9

Table 3. 1: Confusion matrix and common performance metrics calculated from it ….22

Table 3. 2: Number of images for leaf lettuce 24

Table 3. 3: Number of images for bouquet lettuce.. 25

Table 3. 4: Distribution of images for leaf lettuce dataset for training and test............. 25

Table 3. 5: Distribution of images for bouquet lettuce dataset for training and test....... 26

Table 4. 1: Summary of all parameter setting for the model……………………………45

LIST OF ACRONYMS

AI	Artificial Intelligence
ANN	Artificial Neural Network
CNN	Convolution Neural Network
DCNN	Deep Convolutional Neural Network
DL	Deep Learning
MARDI	Malaysia Agricultural Research and Development Institute
MLPN	Multi-Layer Perceptron Network
OLSA	Orthogonal least square algorithm
PCA	Principal Component Analysis
PNN	Probabilistic Neural Network
RBF	Radial Basis Function
RBPNN	Radial Basis Probabilistic Neural Network
ReLU	Rectified Linear Unit
RGB	Red, green, blue
SGDM	Stochastic Gradient Descent with Momentum
SVM	Support Vector Machine

LIST OF APPENDICIES

Appendix A……………………………………………………….……………………….53

Appendix B……………………………………………………..……………………….55

CHAPTER 1: INTRODUCTION

1.1 Introduction

This chapter describes the research background, problem statement and objectives to give ideas for the research work.

1.2 Research Background

Lettuce is categorized as healthier vegetables that has market value where some of the varieties only can be obtained from hypermarkets only. In order to maintain the product's qualities, proper plant monitoring is very important during the growth in order to get high yield. However, before developing an application for plant monitoring or get other information related to the lettuce varieties, recognition process with correct name is the essential part of the application. Designing the system that able to recognize lettuce varieties is necessary to facilitate fast classifying lettuce and allowing researchers or farmers to have proper system to manage them.

The intent of this project is to develop a lettuce recognition system using neural network technique in MATLAB simulation window. There are four basic type of lettuce and dozens of varieties within each type. Some of the varieties are Red Oakleaf, Butterhead, Red Coral, Italian, Salad Grand Rapid, Red Lettuce, Celtuce love, Flashy Butter Oak and etc. Basically, the different between them are based on shape, size, colour, surface and texture. Thus, it has difficulty to differentiate type of lettuce through manual identification and consume a lot of time to know or searching any others information. The fastest way to automatically identified lettuce varieties and others information is to use mobile application.

Machine learning technique are simple and fast in performing identification of lettuce varieties. Due to the computational power and memory, machine vision system can be used for identification of agricultural products. Image analysis based on the shape of lettuce varieties is sufficient to differentiate the varieties. This project proposed a Convolution Neural Network (CNN) for identification of lettuce varieties. A total of seven different lettuce will be considered in this project.

1.3 Problem Statement

In general, lettuce recognition is one of the automation processes in order to monitor plant growth in urban vertical farming under plant factor. There are many types of lettuce varieties that very difficult to differentiate it manually and its beneficial or others related information. Hence, the existence of automation recognition can help farmers to show plant growth with some related information to other people who visiting their plant factory. Thus, by adopting a neural network technique that is simple, fast and economic is the best way in order to automatically differentiate the type of lettuce.

1.4 Objectives

The objective of the research are:

1. To identify the varieties of lettuce by using artificial neural network
2. To achieve a classification accuracy of more than 90%.
3. To evaluate the efficiency of using machine learning technique for classification of lettuce varities

1.5 Aim

To propose a machine learning technique to differentiate the lettuce varieties.

1.6 Scope of Research

The project focusing on seven types of lettuce varieties such as Butterhead Lettuce, Celtucelove Lettuce, Salad Grand Rapid Lettuce, Red Coral Lettuce, Red Lettuce, Red Oakleaf Lettuce and Italian Lettuce.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

This chapter discusses about recognition process in agriculture sector such as rice grain recognition, plant recognition, leaf shape identification and etc. by using neural network. Some of the mobile application also has been developed for the sole purpose of plant classifying. At the end of this chapter, a summary for the whole chapter was discussed.

2.2 Artificial Intelligence Overview

Artificial Intelligence (AI) is part of universally field which is relevant to any intellectual task, ranging from general right to the specific application. There are a lot of definition about what is Artificial Intelligence (AI) is all about. According to (S. J. Russell and P. Norvig, 2010) some approaches of AI is from different people with different methods as shown in Figure 2.1 below. The top halves emphasized on thought processes and reasoning, whereas the behavioral are highlighted at bottom halves. The left-side defined success in terms of fidelity to human performance, whereas the right-side defined success against an ideal performance, called rationality.

Thinking Humanity	**Thinking Rationally**
"The exciting new effort to make computers think ...machines with minds, in the full and literal sense." (Haugeland, 1985) "[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning ..." (Bellman, 1978)	"The study of mental faculties through the use of computational models." (Charniak and McDermott, 1985) "The study of the computations that make it possible to perceive, reason, and act." (Winston, 1992)

Acting Humanly	Acting Rationally
"The art of creating machines that perform functions that require intelligence when performed by people." (Kurzweil, 1990) "The study of how to make computers do things at which, at the moment people are better." (Rich and Knight, 1991)	"Computational Intelligence is the study of the design of intelligent agents." (Poole et al., 1998) "AI ...is concerned with intelligent behavior in artifacts." (Nilsson, 1998)

Figure 2.1: Some definitions of artificial intelligence by (S. J. Russell and P. Norvig. 2010)

(Pannu, 2015) claimed that sectors or areas that adapting Artificial Intelligence resulted an incremental in quality and efficiency. The adoption of AI gives an impact on various fields as expert system widely use it in order to solved complex problem such as in science, engineering, business, medicine, weather forecasting. Artificial Intelligence has expanded into various fields as shown in Figure 2.2.

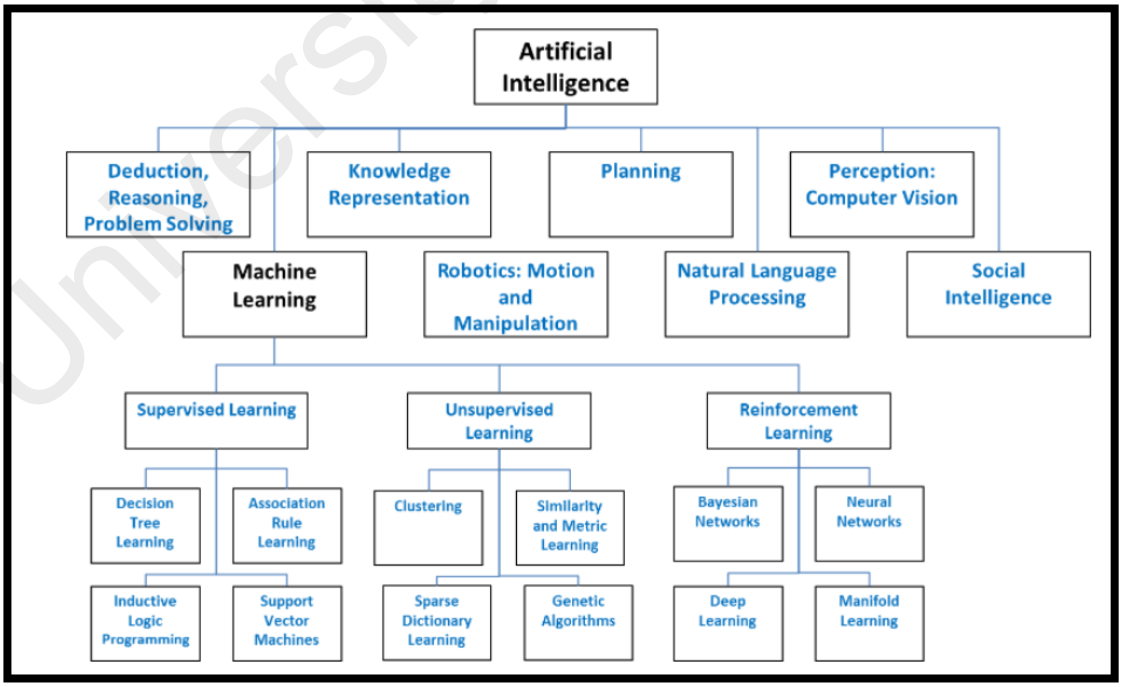


Figure 2.2: The classification of Artificial Intelligence by (A. Pannu, 2015)

2.3 Image Recognition in Agriculture sector: Overview

There are a lot of research works being conducted for image classification in the agricultural sector. This technology development caters towards high-tech agriculture to help increase productive output. This type of research and development able to create interest among younger generation to involve in agricultural sector especially in Malaysia. Furthermore, application of image classification also can aide the farmers in term of efficient farm management, produce more quality products and increase yield to cater local and also export markets.

Some of the research have been done is to classified the plant species based on leaf and use a different method to recognize them. (Stephen Gang Wul et al., 2007) used a Probabilistic Neural Network (PNN) approach for automated leaf recognition for plant classification. The writer used feature extraction in order to allow the computer to obtain feature values automatically. The feature extraction involved five basic geometry features that can be define as digital morphological features for leaf recognition. The Principal Component Analysis (PCA) was used to represent the information of original data as linear combination of certain linear irrelevant variables. The author also mentioned that PNN was used in this research because of its simple structure and the training part was easy and instantaneous. PNN is derived from Radial Basis Function (RBF) which scales the variable non-linearly. 1800 leaves were trained and has an accuracy percentage greater than 90 percent to classified 32 type of plants.

(Sue Han Lee et al., 2015) studied CNN for 44 different plant species to learn unsupervised feature. Author compared the performance of MK leaf dataset with different classifier. The result obtained one of the important feature to identify plant species is the venation structure that has the accuracy 99.5 percent. Author also justified that for a better representation images for leaf, it is better for learning features through CNN compared to hand-crafted features. (ArunPriya et al., 2012) applied the Support Vector Machine

(SVM) for plant classification. Three important phases involved in this approach which is pre-processing, feature extraction and classification. 12 features obtained are extracted and processed by PCA to formed the input vector of SVM. From the result obtained, author stated that the proposed algorithm produces better accuracy and required less time for execution compared to k-NN method.

(Jyotismita and Ranjan, 2011) uses Neural Network classifiers and shape-based features for plant leaf recognition. Three different plant types are analysed using the Moments-Invariant (M-I) model and CentroidRadii (C-R) model. Between of both models, C-R method get better accuracy compared to M-I model where C-R performed 100 percent accuracy.

(Jixiang et al. 2005) proposed Radial Basis Probabilistic Neural Network (RBPNN) in order to recognize shape. Orthogonal least square algorithm (OLSA) is use to trained RBPNN and recursive OLSA is use to optimize the structure of RBPNN. Author also compare the RBPNN classifier with multi-Layer perceptron network (MLPN). 20 species from different plants are use as leaf image dataset where 40 leaves images for each species. From the result obtain, the percentage of recognition rate for both method are 96.2 and 94.4 for RBPNN and MLPN respectively. However, the training time for RBPNN is less than MLPN which only took 48 seconds for RBPNN and 272 seconds for MLPN.

(Jiazhi Pan and Yong He, 2008) proposed recognition of plants using leaves digital image and neural network. The data were divided into two parts, one is for training and the other is for validation. The author took images of soybean, goose grass and alligator alternanthera at the fields. Two types of detection were applied which are border segmentation and area segmentation. Author choose Radial Basis Network (RBN) as it has a strong classification power. The layer consist a hidden radial basis layer and output linear layer. Dataset use for this experiment is about 145 blocks which 100 blocks were used as training dataset and 45 blocks were used to check validation of the model. The result for this model, correctly achieved classification by more than 80 percent.

(Vijay et al., 2013) compared the classification of leaf recognition for plant identification using Artificial Neural Network (ANN) and Euclidean (KNN) classifier. The proposed approach consists of pre-processing, feature extraction and classification. The extraction phase features are based on colour and shape of leaf images. The accuracy for both classifier ANN and KNN are 93.3 percent and 85.9 percent respectively.

Another research in agricultural sector is classification type of rice. (Chathurika Sewwandi Silva and Upul Sonnadara, 2013) using MLP for classification of rice grains. The model was developed for feature set individually and combined. Combined feature model gets an overall accuracy of 92 percent while individual feature gets the overall accuracy 51 percent, 63 percent and 34 percent for morphological model, texture model and colour model respectively.

(Vaibhav Amit Patela and Manjunath V. Joshi, 2017) also do a research of rice type classification by using CNN with transfer learning. 2-class model trained 1600 images in order to classify a broken and normal rice whereas 5-class model trained 4000 images in order to classify the rice types. With and without of transfer learning of classification, the overall accuracy achieved by model with 5-class is 86.8 percent and 94.32 percent respectively, while 2-class model gets an overall accuracy of 99.3 percent.

Another research using CNN is to detect the plant disease identification. (Ferentinos, 2018) identify plant diseases using healthy leaves images and plant diseases. Author compared the result with the different CNN model architecture such as AlexNet, AlexNetOWTBn, GoogLeNet. Overfeet and VGG. All of the architecture achieved the success rate more than 97 percent and VGG get the highest accuracy of 99.53 percent in classification. Author stated that the model can be used as an early warning notification or as a support to an integrated plant disease identification system.

(Juncheng et al., 2018) recognize cucumber diseases based on Deep Convolutional Neural Network (DCNN) using leaf symptom images under field condition. Data augmentation methods is utilize in order to decrease of overfitting. Author conduct an experimentation using DCNN, AlexNet, Random Forest and Support Vector Machines. From the result obtain, the accuracy for DCNN and AlexNet achieved 93.4 and 94 percent whereas RF and SVM achieved 81.9 and 84.8 percent respectively.

There are some researches done for image classification at field. (M. Dyrmann, 2017) doing an automatic detection and classification of weed seedlings under natural light conditions. The experiment is able to handle weed detection and classification in natural environments. Thus, the methods applied able to help and reduce operational cost involved which in turn lead to higher potential rate of adoption compared with existing precision techniques of weed control that lead to potential saving in regards to consumption of herbicide. Based on the results of accuracy percentage, VGG 19 architecture achieved the highest accuracy of 87.3 percent with a total of 2967 plants spread over 17 weed species. Figure 2.3 below showed the comparison of CNN architectures for different image size.

Method	Image size	Top-1 acc. (%)	95% Accuracy confidence interval(%)	Filter and feature map size (MiB)	Execution time (ms)
CaffeNet by Krizhevsky et al. (2012)	256 × 256	84.4	83.0 — 85.7	276.17	2.09
	224 × 224	83.7	82.2 — 85.0	222.49	1.80
	192 × 192	83.1	81.7 — 84.4	177.05	1.60
	128 × 128	82.2	80.7 — 83.5	110.85	1.23
GoogLeNet by Szegedy et al. (2014)	256 × 256	80.6	79.1 — 81.9	75.67	6.99
	224 × 224	78.5	77.0 — 80.0	63.13	5.95
MDNet by Dyrmann et al. (2016b)	256 × 256	77.2	75.6 — 78.6	68.91	1.85
	224 × 224	76.5	74.9 — 77.9	52.61	1.16
	192 × 192	77.3	75.7 — 78.7	38.51	1.12
	128 × 128	77.2	75.6 — 78.7	16.94	0.93
ResNet152 by He et al. (2015)	256 × 256	73.8	72.2 — 75.4	532.23	92.54
	224 × 224	71.7	70.0 — 73.2	459.39	39.13
SqueezeNet1.1 by Iandola et al. (2016)	256 × 256	76.3	74.8 — 77.8	29.24	1.93
	224 × 224	77.4	75.9 — 78.9	23.35	1.88
	192 × 192	82.7	81.3 — 84.0	18.27	1.79
	128 × 128	75.0	73.4 — 76.5	10.52	1.80
VGG19 by Simonyan and Zisserman (2014a)	256 × 256	84.0	82.6 — 85.2	735.09	10.20
	224 × 224	85.6	84.5 — 87.0	595.78	8.15
	192 × 192	87.3	86.0 — 88.4	475.05	7.24
	128 × 128	83.5	82.0 — 84.7	289.31	4.87

Figure 2. 3: The comparison of CNN architectures for different image size. (M. Dyrmann, 2017)

A survey on the usage of Deep Learning (DL) in agriculture was done by (Andreas Kamilaris1 and Francesc X., 2017). From their findings, image processing techniques offers less in terms of performance and unable to compete with deep learning techniques. Signs are very encouraging for deep learning techniques to go even further for smarter, sustainable farming and secure food production. Table 2.1 below show some of the application of deep learning in agriculture based on the author findings.

Table 2.1: Applications of deep learning in agriculture. (Andreas Kamilaris1 and Francesc X., 2017)

No.	Agriculture Area	Problem Description	Classes and Labels	DL Model Used	Value of Metric Used
1	Leaf classification	Classify leaves of different plant specie	32 classes: 32 Different plant species	CNN + RF classifier	97.3% ±0.6%
2	Crop type classification	Classification of crops wheat, maize, soybeans sunflower and sugar beet	11 classes: water, forest, grassland, bare land, wheat, maize, rapeseed, cereals, sugar	CNN	94.60%

			beet, sunflowers and soybeans.		
3		Classification of crops oil radish, barley, seeded grass, weed and stump	7 classes: oil radish, barley, weed, stump, soil, equipment and unknown (pixel of the image)	Adapted version of VGG16 CNN	79% (CA), 0.66 (IoU)
4	Plant recognition	Recognize 7 views of different plants: entire plant, branch, flower, fruit, leaf, stem and scan	1,000 classes: Species that include trees, herbs, and ferns, among others.	AlexNet CNN	48.60%
5		Recognize 44 different plant species	44 classes: Species such as acutissima, macranthera, rubra, robur f. purpurascens etc	AlexNet CNN	99.60%
6		Identify plants from leaf vein patterns of white, soya and red beans	3 classes: Legume species white bean, red bean and soybean	CNN	96.90%

7	Fruit counting	Predict number of tomatoes in the images	Estimated number of tomato fruits (scalar value	Modified Inception Res Net CNN	91% (RFC) 1.16 (RMSE) on real images, 93% (RFC) 2.52 (RMSE) on synthetic images
8		Map from input images of apples and oranges to total fruit counts	Number of orange or apple fruits (scalar value)	CNN (blob detection and counting) + Linear Regression	0.968 (RFC), 13.8 (L2) for oranges 0.913 (RFC), 10.5 (L2) for apples
9		Fruit detection in orchards, including mangoes, almonds and apples	Sections of apples, almonds and mangoes at the image (bounding box	Faster Region based CNN with VGG16 mode	0.904 (apples) 0.908 (mango) 0.775 (almonds)
10	Identification of weeds	Classify weed from crop species based on 22 different species in total. Dataset	22 classes: Different species of weeds and crops at early growth stages e.g. chamomile, knotweed, cranesbill, chickweed and veronica	Variation of VGG16	86.2%

11		Automating weed detection in color images despite heavy leaf occlusion	Detect single weed instances in images of cereal fields (bounding box). A coverage map is produced	Based on Detect Net CNN (which is based on GoogLeN et CNN)	0.64 (IoU), 86.6% (P- IoU), 46.3% (R-IoU)
12	Leaf disease detection	13 different types of plant diseases out of healthy leaves	15 classes: Plant diseases (13), healthy leaves (1) and background images (1)	CaffeNet CNN	96.30%
13	Plant disease detection	Identify 14 crop species and 26 diseases	38 class labels as crop- disease pairs	AlexNet, GoogleN et CNNs	0.9935
14		Classify banana leaves' diseases	3 classes: healthy, black sigatoka and black speckle	LeNet CNN	96+% (CA), 0.968 (F1)

15	Land cover classification	Identify 21 land-use classes containing a variety of spatial patterns	21 land-use classes: Agricultural, airplane, sports, beach, buildings, residential, forest, freeway, harbor, parking lot, river etc	CNN + Multiview model averaging	93.48%
16		Extract information about cultivated land	2 classes: Cultivated vs. non-cultivated	CNN	88-91%
17		Land cover classification considering time series	11 classes (dataset 1), 9 classes (dataset 2). Land cover classes such as trees, crops, forests, water, soils, urban areas, grasslands, etc. (Image object or pixel)	One-unit LSTM + RFF, One-unit LSTM + SVM	First Dataset: 75.34% (CA), 0.7463 (F1) Second Dataset: 84.61% (CA), 0.8441 (F1) RF

2.4 Convolutional Neural Network (CNN)

CNN is a type of Deep Neural Networks (DNN) that consists of many layers such as the Convolution layers, Pooling layer, and Fully-connected layer. It is mainly used for image classification purposes.

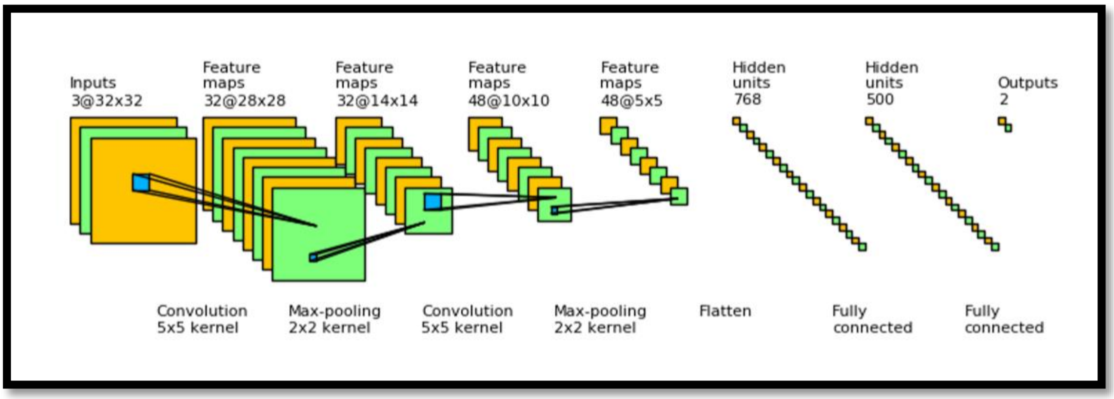


Figure 2. 4: The structure of Convolutional Neural Network

Basically, CNN architecture successively applying layers of convolutional onto input which follows the same design principles, the spatial dimensions in down sampling while the number of feature map is increasing. CNN is divided into two categories; classic and modern network architectures. LeNet-5, AlexNet and VGG 16 are part of classic network architectures while present architectures involve GoogleNet or Inception, ResNet, ResNeXt, and DenseNet.

Handwritten digits for zip code recognition in postal services adopted the LeNet-5 model as an identification. It was developed in 1998 by Yan LeCun which decrease computation and symmetry in the network was break by force, subset from the previous layer used by the convolutional layers. According to (Yann LeCun et al., 1998) LeNet-5 consists of seven layers, three convolutional layers, two sub-sampling (pooling) layers and one fully connected layer as illustrated in Figure 2.5 below.

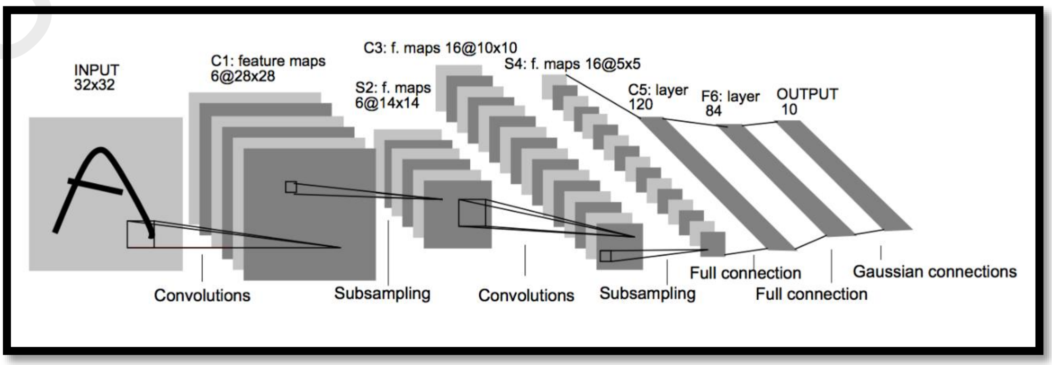


Figure 2. 5: Architecture of LeNet-5. (Yann LeCun et al., 1998)

In 2012, AlexNet was develop by Krizhevsky et al. Basically, general architecture is similar to LeNet-5. Many of computer vision community were convinced to have a look into deep learning for computer vision tasks. From the paper published, (Krizhevsky et al., 2017) was classify 1.2 million high-resolution images into the 1000 different classes and achieved top-5 test error rate 15.3 percent in the ILSVRC-2012. AlexNet consists of five convolutional layers, max-pooling layers, and three fully connected layers with a 1000-way softmax. In order to reduce overfitting in the fully connected layers, regularization method is applied. Figure 2.6 below show the architecture of AlexNet.

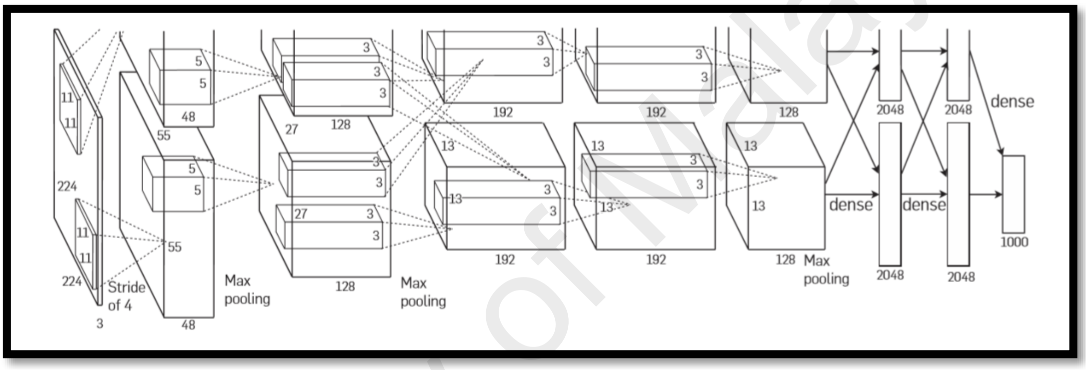


Figure 2. 6: Architecture of AlexNet. (Krizhevsky et al., 2017)

Next is VGGNet network that the simpler variant of the convolutional layer. It was introduced in 2014. The convolution layer involve convolution layer, ReLu layer, Max-pooling layer, Fully-connected layer and softmax. Figure 2.7 below is the architecture of VGGNet. Whereas Figure 2.8 show the validation error percentage of VGGNet obtained without outside training data compared to others (Simonyan and Andrew, 2015).

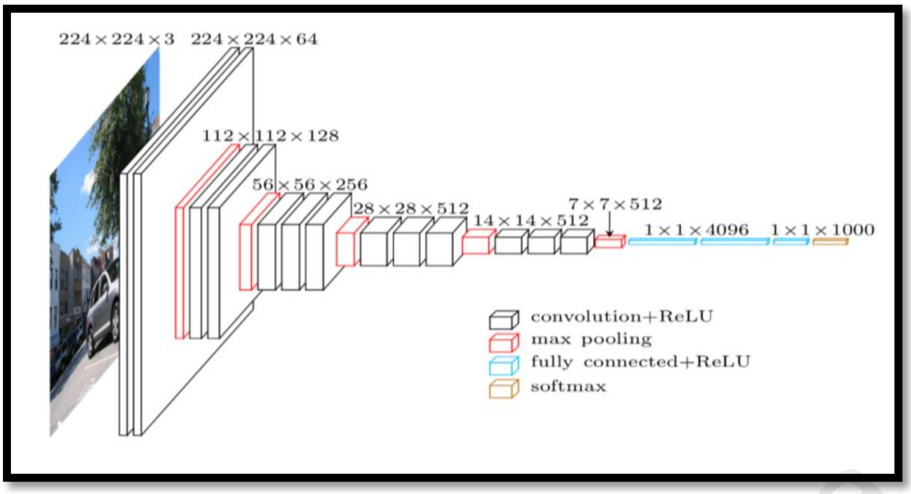


Figure 2. 7: Architecture of VGGNet.

Method	top-1 val. error (%)	top-5 val. error (%)	top-5 test error (%)
VGG (2 nets, multi-crop & dense eval.)	**23.7**	**6.8**	**6.8**
VGG (1 net, multi-crop & dense eval.)	24.4	7.1	7.0
VGG (ILSVRC submission, 7 nets, dense eval.)	24.7	7.5	7.3
GoogLeNet (Szegedy et al., 2014) (1 net)	-	7.9	
GoogLeNet (Szegedy et al., 2014) (7 nets)	-	**6.7**	
MSRA (He et al., 2014) (11 nets)	-	-	8.1
MSRA (He et al., 2014) (1 net)	27.9	9.1	9.1
Clarifai (Russakovsky et al., 2014) (multiple nets)	-	-	11.7
Clarifai (Russakovsky et al., 2014) (1 net)	-	-	12.5
Zeiler & Fergus (Zeiler & Fergus, 2013) (6 nets)	36.0	14.7	14.8
Zeiler & Fergus (Zeiler & Fergus, 2013) (1 net)	37.5	16.0	16.1
OverFeat (Sermanet et al., 2014) (7 nets)	34.0	13.2	13.6
OverFeat (Sermanet et al., 2014) (1 net)	35.7	14.2	-
Krizhevsky et al. (Krizhevsky et al., 2012) (5 nets)	38.1	16.4	16.4
Krizhevsky et al. (Krizhevsky et al., 2012) (1 net)	40.7	18.2	-

Figure 2. 8: Comparison method of VGGNet with other method. (Simonyan and Andrew, 2015).

The inception network or GoogleNet was develop by researcher at Google in 2014. The network consists of 22 layer deep CNN. An overall network performance can be improved by adding two auxiliary outputs. The addition of auxiliary output will give a benefit at the end of the performance model. The depth and width of the network is increase while keeping the computational budget constant. Figure 2.9 below is the architecture of GoogleNet. The architecture is to improve utilization of the computing resources inside the network. According to (Szegedy et al., 2015), their method produces solid proof that by moving to sparser architecture in general is feasible and very useful idea.

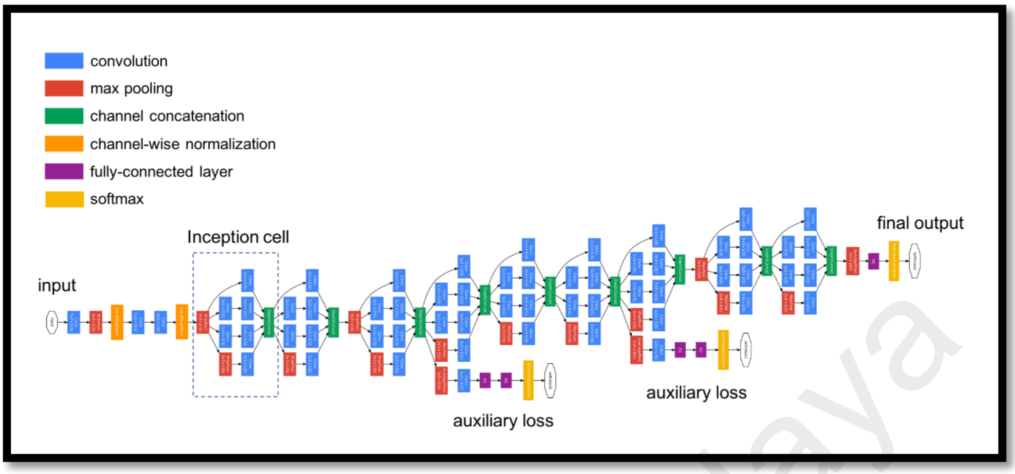


Figure 2. 9: Architecture of GoogleNet. (Szegedy et al., 2015)

In 2015, ResNet is developed by Kaiming He et al. The number of layers for ResNet are based on the name which are ResNet-34, ResNet-50, ResNet-100 and ResNet-150. Figure 2.10 below is the architecture of ResNet.

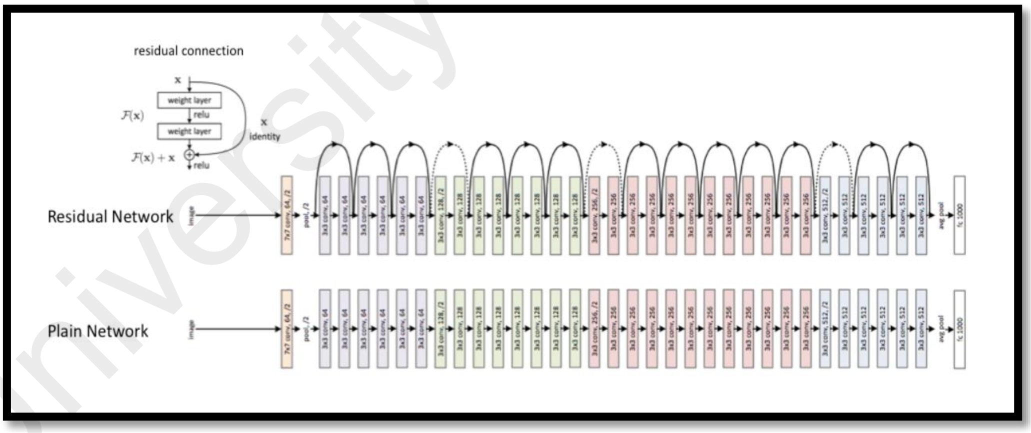


Figure 2. 10: The architecture of ResNet. (Alex Krizhevsky, 2017)

The deep residual network is an extension of ResNeXt architecture is an extension of the deep residual network which replaces standard residual block used in the Inception models. Before merging the results, the block's input was projected into a channel dimensional representations of which separately by applying a few convolutional filters.

The idea is quite similar proposed in the AlexNet which shares the convolution computation across two GPUs. Figure 2.11 below is the architecture of ResNeXt.

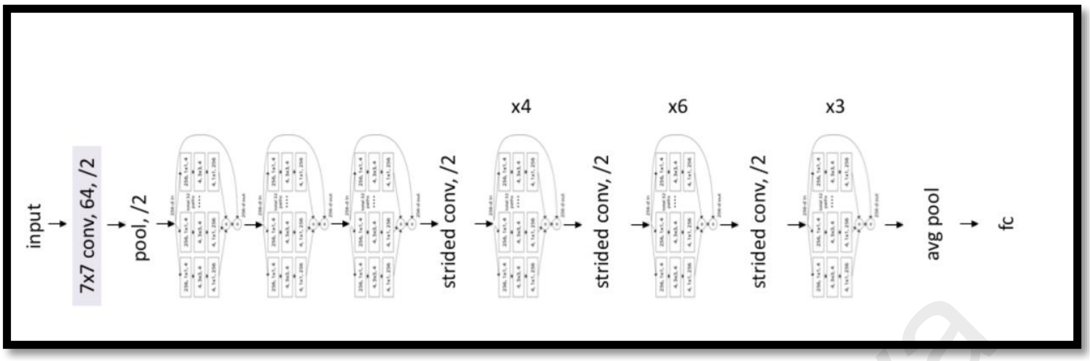


Figure 2. 11: Architecture of ResNeXt.

Reference feature maps from earlier in the network uses DenseNet which is very useful. Each layers of feature map is concatenated with input from every successive layer within a dense block. DenseNet achieved better performance with less complexity compared to ResNet model. The parameter of DenseNet are as follow:

- 0.8 million (DenseNet-100, k=12)
- 15.3 million (DenseNet-250, k=24)
- 40 million (DenseNet-190, k=40)

There are many ways to improve the performance of CNN. Among of them are tune the parameters, image data augmentation, deeper network topology and handle overfitting and underfitting problem. The number of epochs is one of the parameter to change in order to improve the performance. According to the reduction in training loss, the number of epochs can be decided. Next is image data augmentation that generally used to increase the data sample count. The image of dataset can be added by using image augmentation such as rotation, zoom, shear and so on. (W. Shi et al., 2016) has proposed Min-Max objective into the training procedure in order to improve the performance of CNN. The proposed improvement is universally applied to different CNN models. For different dataset including CIFAR-10, CIFAR-100, SVHN and MNIST are used to trained with the Min-Max objective and its achieved remarkable performance improvements.

2.3 Summary

From the past review papers, researchers choose AI in order to achieve the high accuracy in classification. There are many uses of AI in agricultural research such as leaf classification, plant recognition, leaf disease detection, plant disease detection and so on. All of this research can be applied and give an impact to agriculture sector in terms of quality of the products, proper arrangement of plant, reduce the cost of labor and so on. This project will focus on leaf lettuce classification and compare the result with the bouquet lettuce classification. The performance for both classifications will be evaluate using confusion matrix.

CHAPTER 3: METHODOLOGY

3.1 Introduction

This chapter discusses about the implementation method to recognize the lettuce varieties by using MATLAB software. Convolution Neural Network (CNN) was use in this project in order to recognize them. Two experiments were conducted for this project which are recognition of the leaf lettuce and bouquet lettuce. Both experiments were run to compare which process able to produce higher accuracy percentage to differentiate types of lettuce. Data training was divided into seven different varieties of lettuce for leaf lettuce recognition while three different varieties for bouquet lettuce. The cultivation of seven varieties lettuce were planted inside of plant factory and greenhouse that available at Malaysia Agricultural Research and Development Institute (MARDI).

3.2 Variables

Constant and measured variable is very important in order to build CNN model. The constant variable in this project involve number of images sample, image pixel, number of epoch and batch size whereas the accuracy percentage of prediction is the measured variable.

3.2.1 Constant Variables

The total images involved in leaf lettuce recognition is 7000 which is 70 percent and 30 percent from the total image used for training and testing dataset respectively. The size of image was set to 32 by 32 as training input. Number of training sample to work through before internal parameter were updated was controlled by Batch Size which is a hyperparameter. The best batch size for leaf lettuce recognition is 32 whereas bouquet lettuce is 20. Number of epoch is the number of complete passes through the training dataset. The number of epoch is dependent on the total and diversities of the dataset obtained. The number of epoch for leaf lettuce recognition and bouquet lettuce recognition are set to 32 and 20 respectively in order to get the better prediction accurateness.

3.2.2 Measured Variable

The accuracy percentage of prediction for leaf lettuce and bouquet lettuce is a measured variable in this project. The prediction is based on the testing data where the leaf recognition have 300 images for each class whereas 40, 80 and 80 images for varieties Butterhead, Red Coral and Red Oakleaf respectively for bouquet lettuce. Figure 3.1 below show the confusion matrix that represent the accuracy of prediction for each class and the average accuracy for the total leaf lettuce recognition.

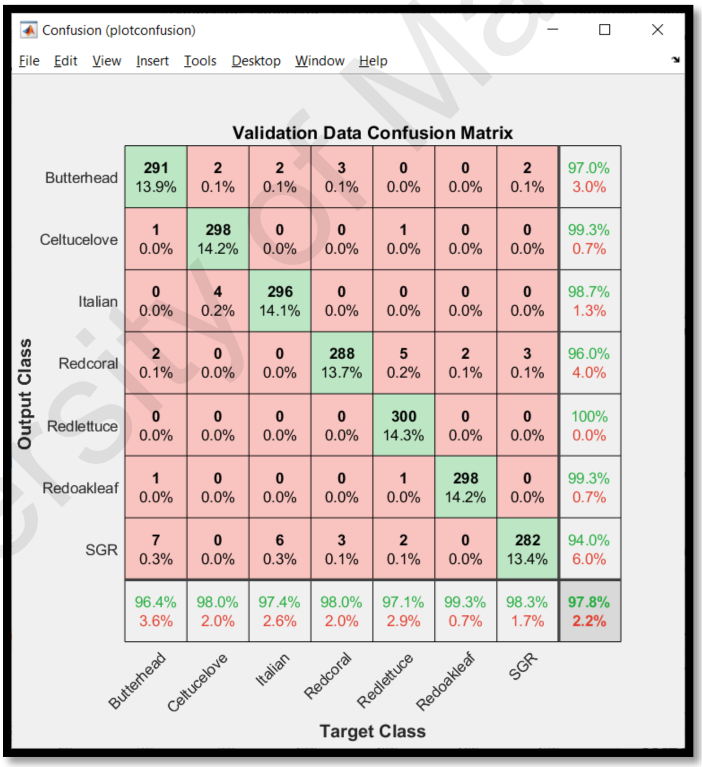


Figure 3. 1: An example of confusion matrix chart for leaf lettuce recognition.

The confusion matrix chart above displays the total number of observation in each class. The row and column of the confusion matrix correspond to the true class and predicted class respectively.

Table 3. 1: Confusion matrix and common performance metrics calculated from it

		True class	
		p	n
Prediction condition	Y	True Positives	False Positives
	N	False Negatives	True Negatives
	Column totals:	P	N

Where; $$fp\ rate = \frac{FP}{N} \qquad tp\ rate = \frac{TP}{P} \tag{3.1}$$

$$precision = \frac{TP}{TP+FP} \quad recall = \frac{TP}{P} \tag{3.2}$$

$$accuracy = \frac{TP+TN}{P+N} \tag{3.3}$$

(Fawcett, 2005) use the label {Y,N} to distinguish between the actual class and the predicted class for class predictions produced by a model as Table 3.1 above. A given classifier and an instance come with four possible outcomes. A true positive is counted if both the instance and classified is positive. False negative counted when only the classified is negative. True negative counted when both instance and classified is negative whereas false positive counted when only the classified is positive.

3.3 Recognition Method

Figure 3.2 below is the flow of the Lettuce recognition process that involves three main phase which are image acquisition, image pre-processing and image recognition.

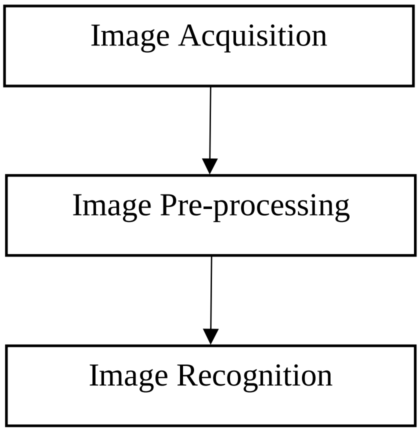


Figure 3. 2: Flow of the project

3.4 Image Acquisition

The cultivation of seven varieties lettuce are planted inside of a plant factory and greenhouse that is available at Malaysia Agricultural Research and Development Institute (MARDI). Figure 3.3 below show the lettuce during growth.

Figure 3. 3: Lettuce images during growth

The leaf lettuce and bouquet lettuce were taken using a smartphone dual camera with a resolution of 12 megapixels and optical image stabilization. The image taken is in RGB (Red, Green, blue) image. (Wang, 2008) stated that the background image needs to be clean either in white color or any color that contrast with the sample. The background chosen for the image sample is white since it has reasonable color contrast with the sample. The height of the camera from the sample was set at 1.5 feet in order to get clear visibility image of the leaf lettuce and bouquet lettuce. The name of the lettuce varieties and number of image used is illustrated in Table 3.2 below for leaf lettuce recognition whereas Table 3.3 is for the bouquet lettuce.

Table 3. 2: Number of images for leaf lettuce

No	Images	Scientific Name	Number of Images
1		Butterhead	1000
2		Celtucelove	1000
3		Itallian	1000
4		Red Coral	1000
5		Red Lettuce	1000
6		Red Oakleaf	1000

7		Salad Grand Rapid (SGR)	1000

Table 3. 3: Number of images for bouquet lettuce

No	Images	Scientific Name	Number of Images
1		Butterhead	200
2		Red Coral	400
3		Red Oakleaf	400

The dataset of leaf lettuce was divided into 2 parts, 70 percent from each class was used as training and 30 percent used for testing while for whole lettuce 80 percent from each class was used as training and 20 percent for testing. The number of images for each class is as listed in Table 3.4 and Table 3.5 respectively.

Table 3. 4: Distribution of images for leaf lettuce dataset for training and test.

No	Names	Training	Testing	Total
1	Butterhead	700	300	1000
2	Celtucelove	700	300	1000
3	Itallian	700	300	1000
4	Red Coral	700	300	1000

5	Red Lettuce	700	300	1000
6	Red Oakleaf	700	300	1000
7	Salad Grand Rapid (SGR)	700	300	1000

Table 3. 5: Distribution of images for bouquet lettuce dataset for training and test.

No	Names	Training	Testing	Total
1	Butterhead	160	40	200
2	Red Coral	320	80	400
3	Red Oakleaf	320	80	400

All data set are stored under two main folders training and testing. Each folder consists of sub-folders which filled with 7 varieties of leaf lettuce and 3 varieties of bouquet lettuce images respectively. Figure 3.4 below is an example of training folder for leaf lettuce.

Figure 3. 4: Training dataset folder for leaf lettuce

3.5 Image Pre-processing

Some of the image dataset has different sizes due to camera orientation selected during the photoshoot. The original image size is 3024 by 3024 pixel and 3024 by 4024 pixel. Hence, the images needed to be resized in order for all images has same size. The image size was changed to 32 by 32 pixel because to reduce time taken for training whereas training time will be prolong if maintained the original size of the images. Figure 3.5 below shows the image of the sample before and after resizing.

Figure 3. 5: Sample images before and after resizing

3.6 Convolutional Neural Network Architecture

The CNN model of this project consists of Convolutional Layer, Rectified Linear Unit (ReLU) Layer, Max-Pooling Layer, Fully-Connected Layer, Softmax Layer and Classification Layer. Convolutional Layer, Pooling Layer and Fully-Connected Layer are the three main type of layers to build the architectures.

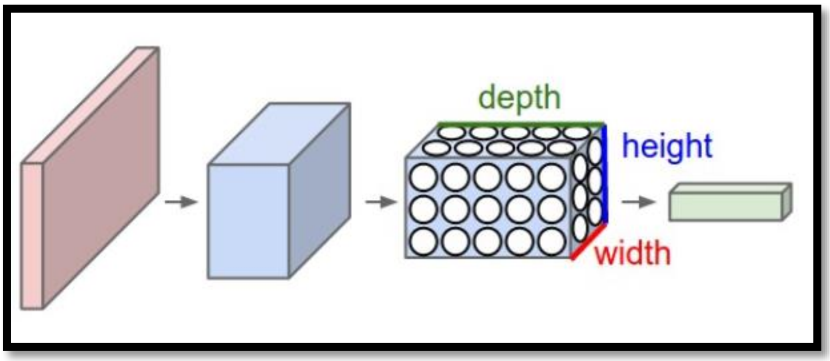


Figure 3.6: Visualization of layer

Neurons in CNN was arranged in three dimensions which are width, height and depth. As shown in Figure 3.6 above, the input layer will hold the image, then its width and height are referred to the dimensions of the image (32 x 32 pixel) and the depth is 3 refer to the Red, Green, Blue (RGB) channel. The output of neurons that are connected to local regions in the input will be compute by the Convolutional Layer with each computing a dot product between its weights and small region connected in the input volume. For this project, the convolutional Layer creates 32 filters of size [5 5]. Then, each of the Convolutional layer will proceed with the Pooling and ReLU layer.

The function of Pooling layer is to control overfitting, reduce the number of parameters and computation in the network. Figure 3.7 show an example of Pooling layer down samples the volume spatially.

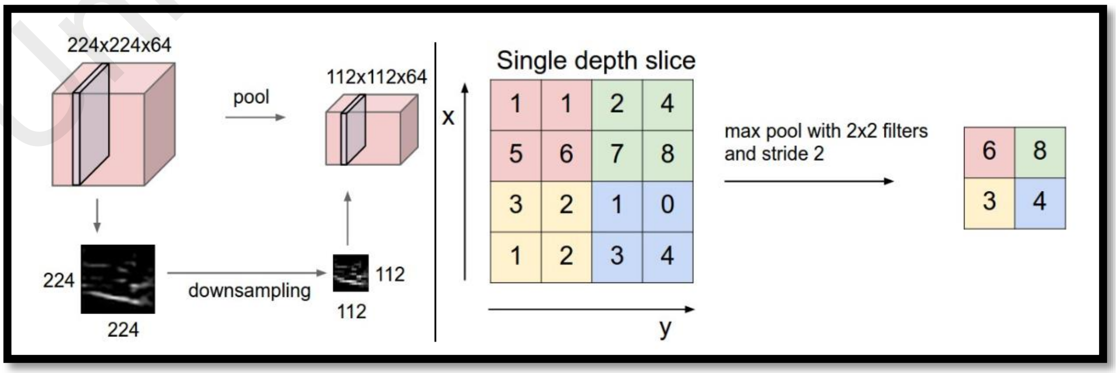


Figure 3.7: An example of Pooling layer down samples the volume spatially

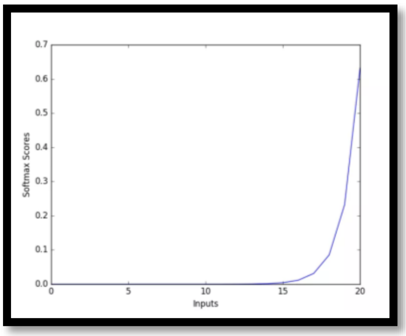
ReLU layer is applied after the Pooling layer. A ReLU layer will performs a threshold operation where when the input is greater than zero, then the output is equal to the input value whereas when the input is less than zero, the output is equal to zero. The operation of ReLU layer is equivalent as equation below:

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x \leq 0 \end{cases} \tag{3.4}$$

The network then proceed with the Fully-Connected layer that will compute the class scores, in resulting in volume of size [1x1x7], where seven is corresponds to the categories of the leaf lettuce recognition. Then the Softmax layer is applied since this project have more than two categories for both leaf recognition and bouquet lettuce recognition. The Softmax layer is basically the normalized exponential probability of class observations represented as neuron activations. Categorical probability distribution is equivalent to the output of softmax function. Softmax function mathematically equation is shown below.

$$\sigma(z)_j = \frac{e^{Zj}}{\sum_{k=1}^{K} e^{Zk}} \tag{3.5}$$

The exponential (e-power) of the given input value and sum of exponential values of all the values in the inputs are calculated by the formula. The output of the softmax function is the ration of the exponential of the input value and sum of exponential values. Figure 3.8 below show the softmax graph where the high value will have the high probability



.

Figure 3.8: Softmax graph

Lastly, the Classification Layer was applied in order to computes the cross-entropy loss for multi-class classification problems with mutually exclusive classes. Bishop, 2006 mentioned the trained network would assigned each input to one of the K mutually exclusive classes using the cross-entropy function for a 1-of-K coding scheme based on the value from Softmax function.

$$loss = -\sum_{i=1}^{N}\sum_{j=1}^{K} t_{ij} ln y_{ij} \quad (3.6)$$

Where: N = number of sample

K = number of classes

t_{ij} = indicator $_i$ sample is belongs to the $_j$ class

y_{ij} = output for sample $_i$ *for class* $_j$

The network is trained with the training option as the following command shown in Figure 3.9 below.

```
opts = trainingOptions('sgdm', ...
    'InitialLearnRate', 0.001, ...
    'LearnRateSchedule', 'piecewise', ...
    'LearnRateDropFactor', 0.1, ...
    'LearnRateDropPeriod', 8, ...
    'L2Regularization', 0.004, ...
    'MaxEpochs', 10, ...
    'MiniBatchSize', 32, ...
    'Verbose', true, ...
    'Plots','training-progress');
```

Figure 3.9: Training option of the network

Optimizers in this project is from Stochastic Gradient Descent with Momentum (SGDM). SGD with momentum (Ning Qian, 1998) is an approach to assist acceleration SGD in the relevant direction and dampens oscillations. This optimizer was chosen because it can increase the speed of learning and used to make updates from the stored velocity of all parameters. Adam et al. (2018) had conducted test run on the methods used in datasets CIFAR10 and CIFAR100. From the test run, SGD with momentum produces converges to a solution with lower test error better than Adaptive Moment Estimation (Adam).

CHAPTER 4: RESULT AND DISCUSSION

4.1 Introduction

This chapter is about the results achieved for lettuce recognition by using CNN. The recognition process is divided into two parts; leaf lettuce and bouquet lettuce. Seven different varieties of lettuce were used for the leaf recognition process whereas three different varieties of lettuce for bouquet recognition.

The total of sample images for leaf lettuce and bouquet lettuce recognition are 7000 and 1000 of images respectively. From the total images of leaf lettuce recognition, 70% was used for training and 30% was used for testing whereas for bouquet lettuce recognition, 80% was used for training and 20% was used for testing.

4.2 Results

4.2.1 Leaf Lettuce Recognition

Figure 4.1 shows the training progress of the model. The system is in learning process from the epoch 0 until 10. From the training options, number of epoch and mini batch size were tuned to get the high accuracy. The learning rate for this model is 0.001. The model was tested with mini batch size 80 and number of epoch 10. From the figure below, training accuracy percentage is below than 90%.

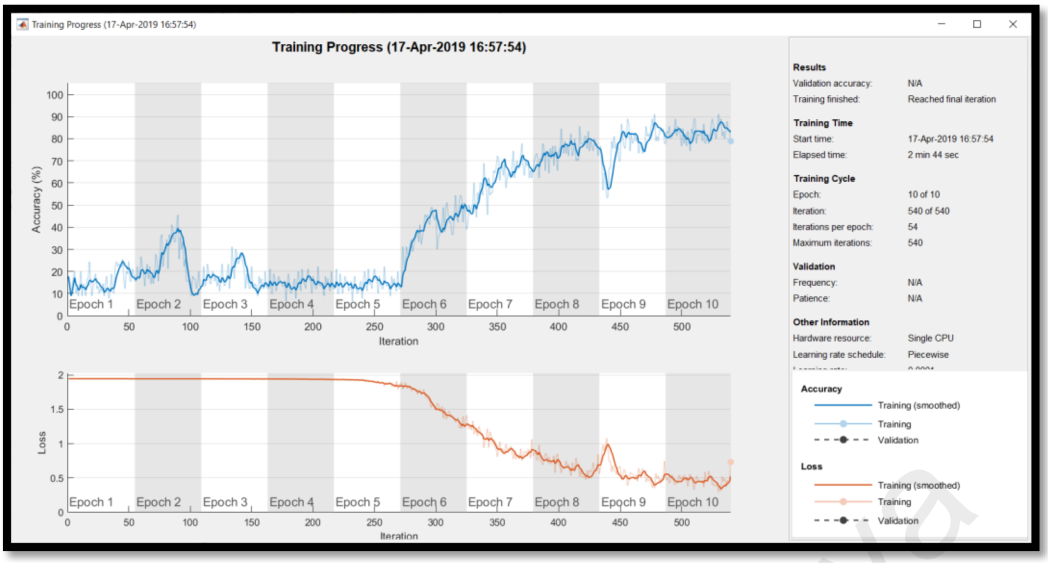


Figure 4. 1: Training progress with mini batch size 80 and number of epoch 10.

Figure 4.2 is the confusion matrix for this mini batch size where mostly resultant lower accuracy for several varieties of lettuce such as Butterhead, Red Coral, Red lettuce and SGR. The average accuracy for leaf lettuce recognition with mini batch size 80 is 79.8%. Thus, number of epochs and mini batch need to change until reached a better accuracy percentage. Figure 4.3 show the model randomly pick one image from the testing file.

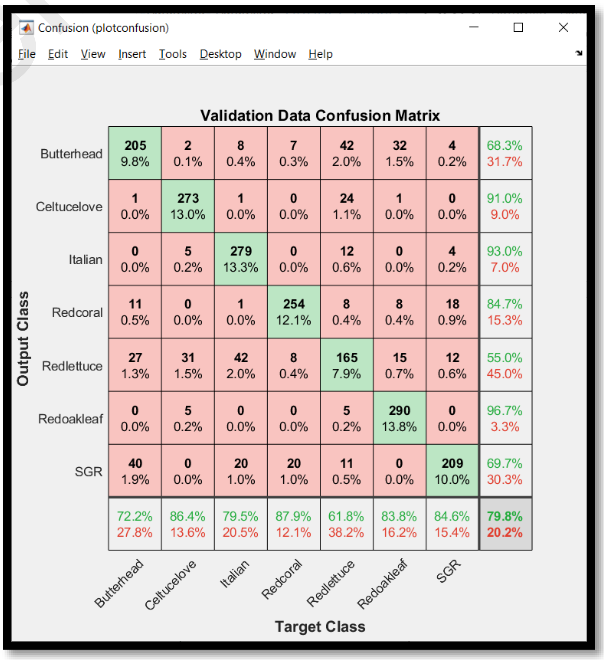


Figure 4. 2: Confusion matrix for mini batch size 80 and number of epoch 10.

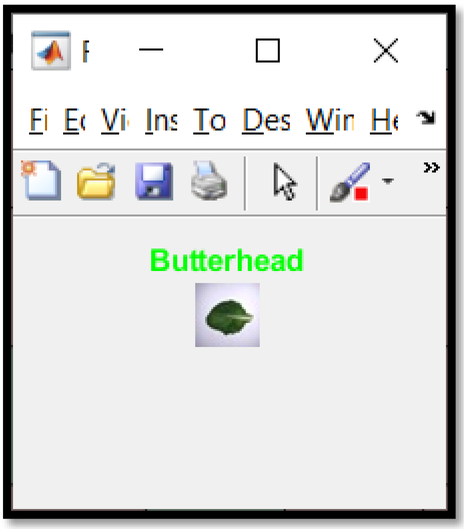


Figure 4. 3: Random image from testing file after training the model

To observe if there is an improvement towards training accuracy, number of epochs was increased to 15 but mini batch was maintained at 80. However, from the Figure 4.4, training accuracy percentage is still below 90% and time of training was increased to 6 minutes 20 seconds.

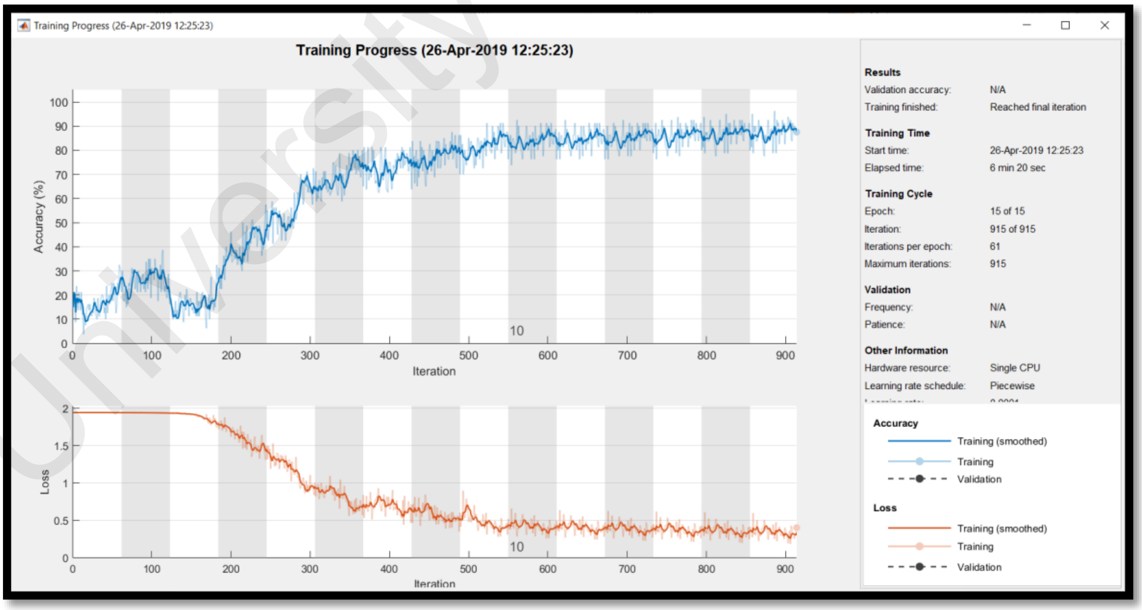


Figure 4. 4: Training progress with mini batch size 80 and number of epoch 15.

From the confusion matrix Figure 4.5, there is an improvement on the average accuracy percentage after increasing the number of epochs. However, the accuracy for some varieties of leaf lettuce is still below the target. Thus, the number of epochs and mini batch size needs to be re-tuned again until model reached better accuracy percentage. Figure 4.6 shows the model randomly pick one image from the testing file.

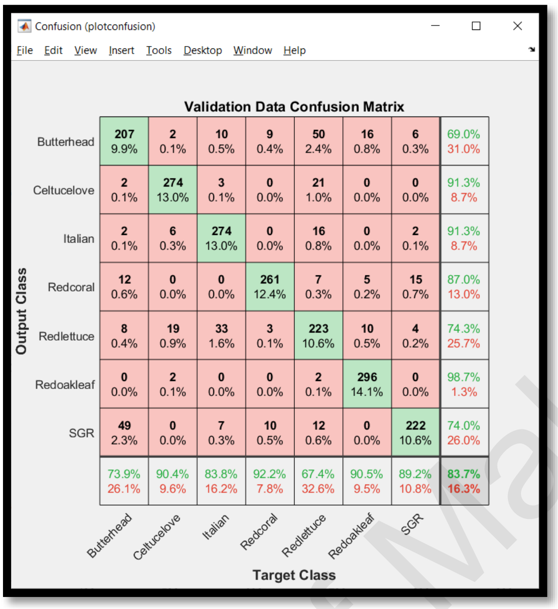


Figure 4. 5: Confusion matrix for mini batch size 80 and number of epoch 15

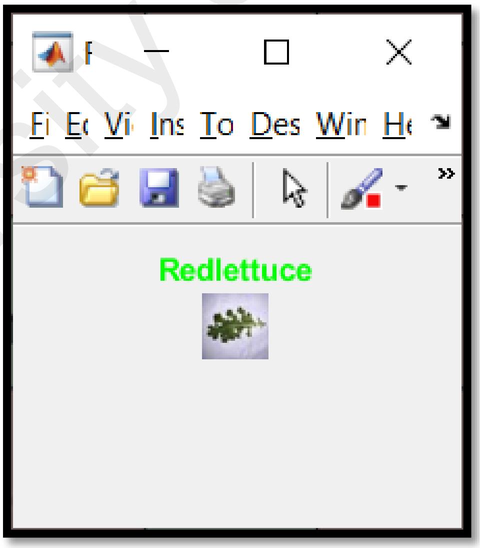


Figure 4. 6: Random image from testing file after training the model

The number of epochs is increased to 20 to see whether it has any affect towards improving the training accuracy or not. From the Figure 4.7, the training accuracy has an improvement compared before and maintaining accuracy percentage above 90% when it reached from epochs 10 to 20. However, it is still not enough to prove that the model can obtain high accuracy percentage because from the confusion matrix Figure 4.8, some of the leaf lettuce still obtained low accuracy percentage below 90%. Figure 3.9 shows the result the model randomly picks one image from testing file and predicted wrongly for the type of leaf lettuce.

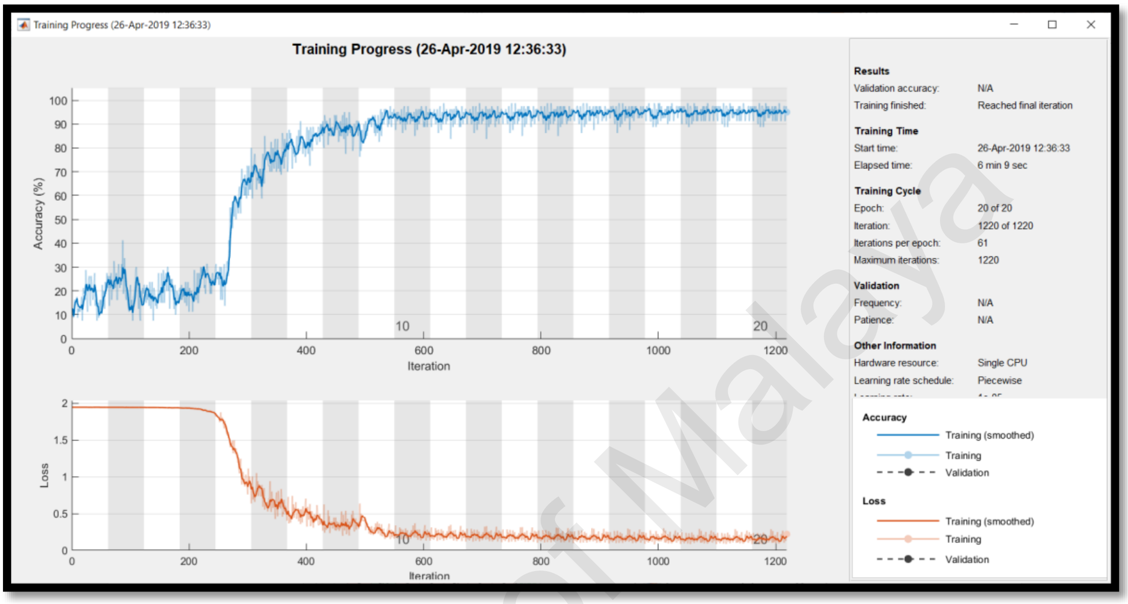


Figure 4. 7: Training progress with mini batch size 80 and number of epoch 20.

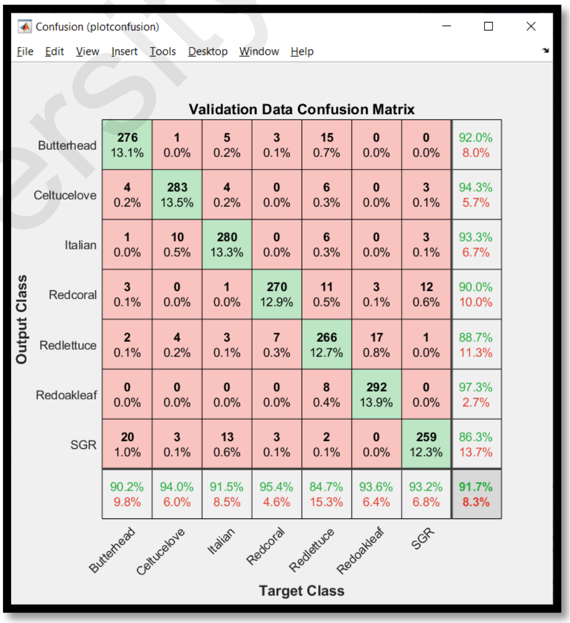


Figure 4. 8: Confusion matrix for mini batch size 80 and number of epoch 20

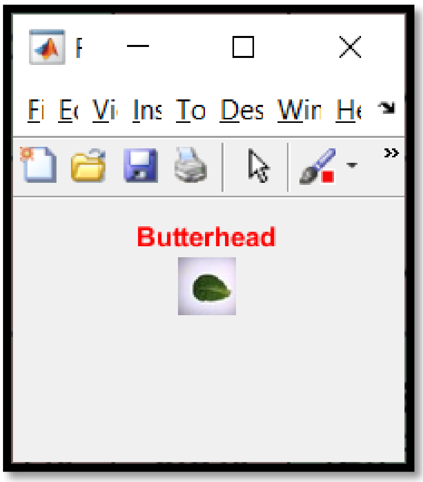


Figure 4. 9: Random image from testing file after training the model

When there is no improvement after changing number of epochs, the size of mini batch size was reduced to 64 and number of epochs was set to 10. Figure 3.10 shows training time was reduced to 2 minutes 20 seconds and training accuracy percentage is above 90% from epochs 9 to 10. Yet, the validation error still high which is about 7.2% when referring to confusion matrix Figure 4.11. Although average accuracy percentage is 92.8%, leaf accuracy percentage for Red Lettuce variety was still low. Thus, number of epochs needs to be tweak for the better result. Figure 4.12 shows the model randomly picks one image form the testing file.

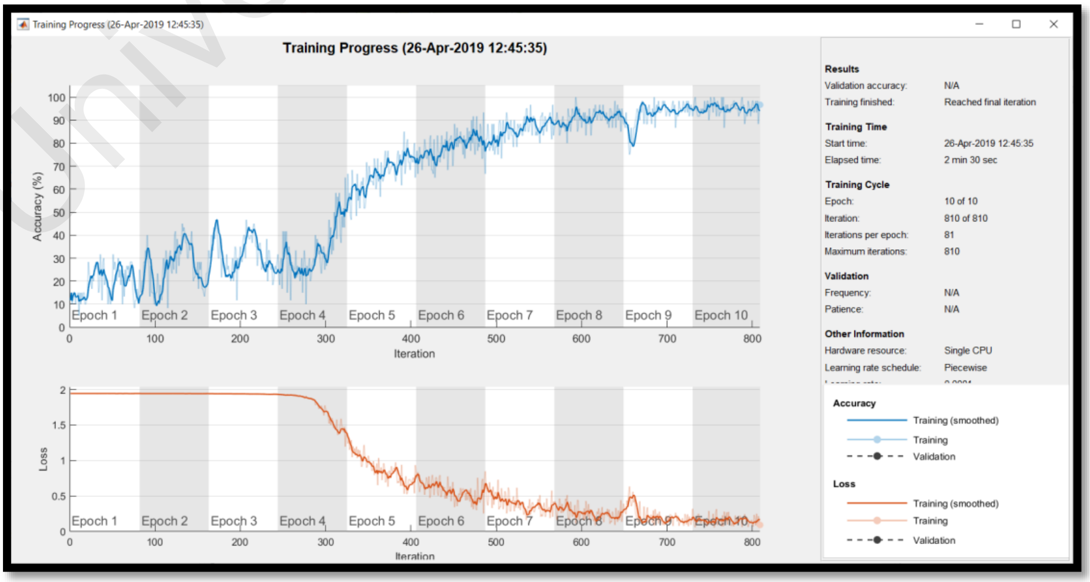


Figure 4. 10: Training progress with mini batch size 64 and number of epoch 10

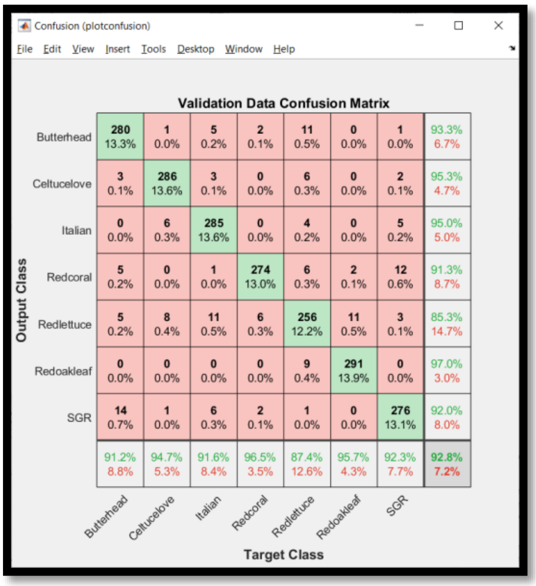


Figure 4. 11: Confusion matrix for mini batch size 64 and number of epoch 10

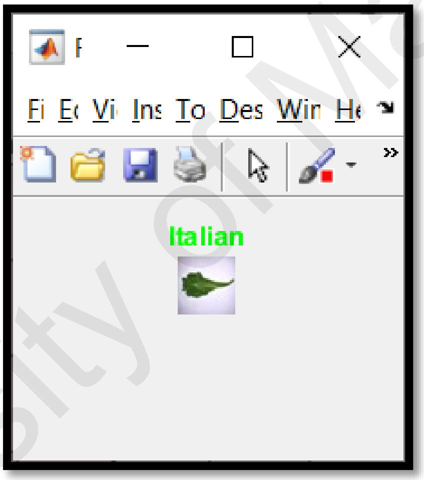


Figure 4. 12: Random image from testing file after training the model

The number of epochs is then increased to 20 and mini batch size maintain at 64. Figure 4.13 shows that training accuracy percentage maintained above 90% when it reached epochs 10. However, training time is also increased to 5 minutes 30 seconds due to increasing number of epochs. Figure 4.14 shows validation error is reduced minimally by 7% compared to before and low accuracy percentage for leaf lettuce, SGR. Figure 3.15 shows the model randomly pick one image form the testing file.

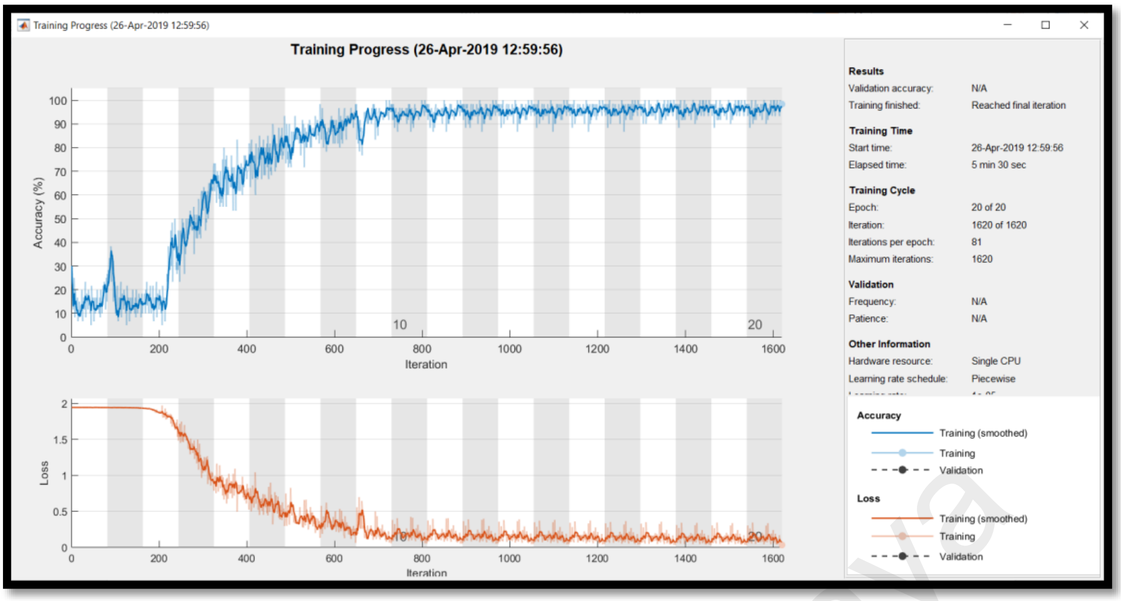


Figure 4. 13: Training progress with mini batch size 64 and number of epoch 20

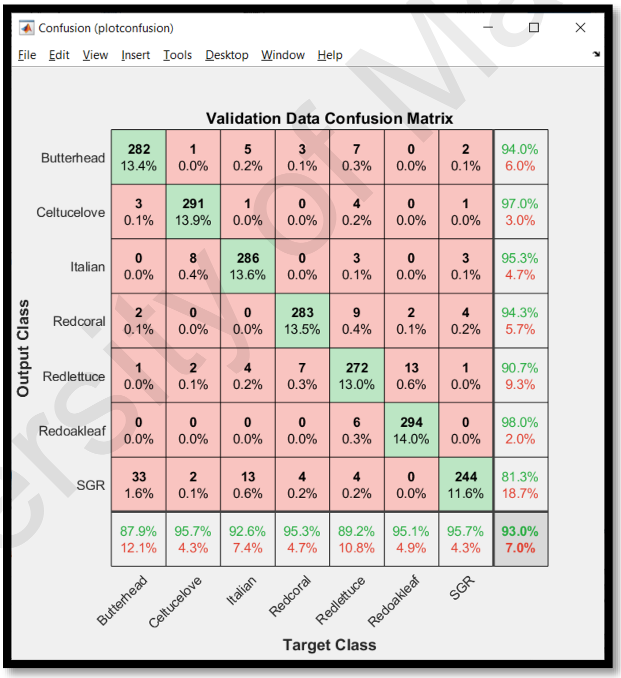


Figure 4. 14: Confusion matrix for mini batch size 64 and number of epoch 20

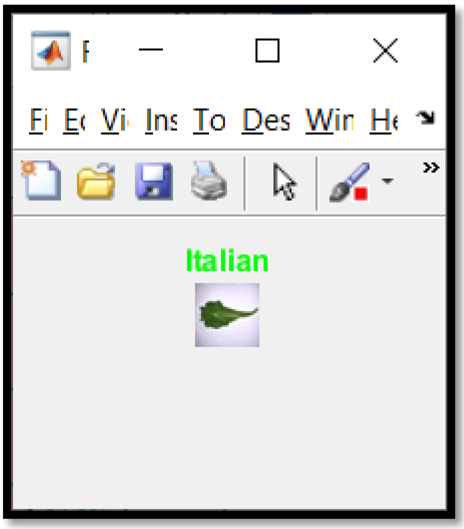


Figure 4. 15: Random image from testing file after training the model

Since mini batch size 64 cannot get better accuracy percentage, it is then reduced to 40 and number of epochs set to 15. Figure 4.16 shows training accuracy percentage is almost achieving 100% when reached from epochs 10 to 15. The training time is about 4 minutes 2 seconds. From Figure 4.17, the validation error was reduced to 3.2% and average accuracy percentage is 96.8%. Figure 3.18 shows the model randomly picks one image form the testing file.

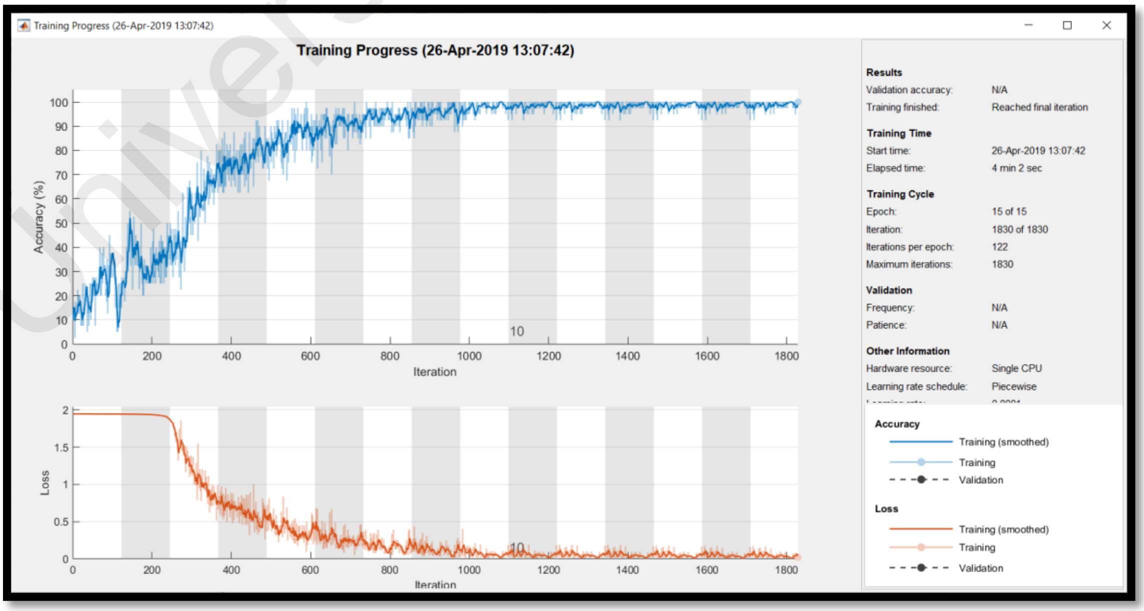


Figure 4. 16: Training progress with mini batch size 40 and number of epoch 15

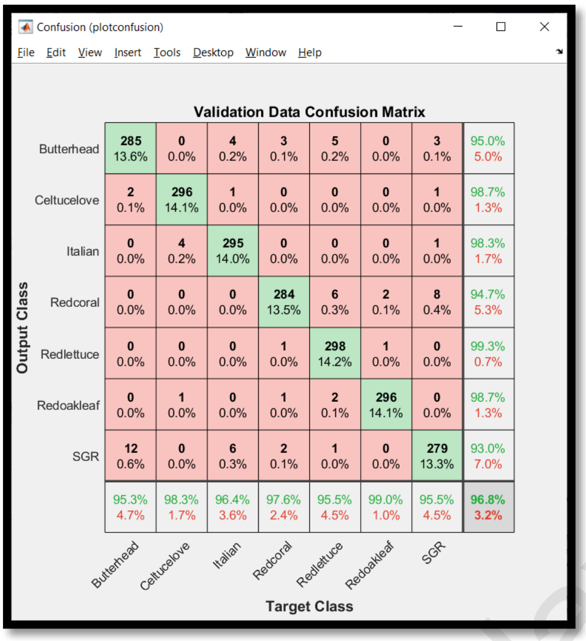


Figure 4. 17: Confusion matrix for mini batch size 40 and number of epoch 15

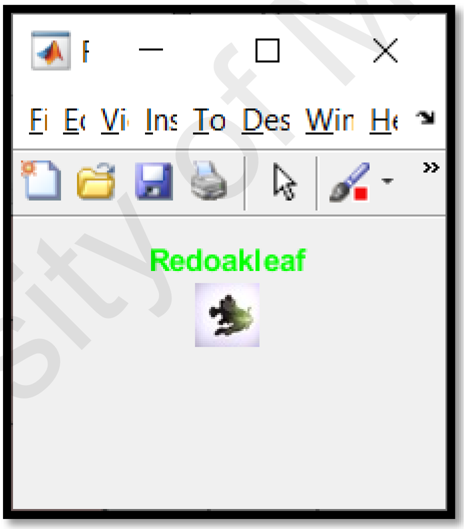


Figure 4. 18: Random image from testing file after training the model

From the Figure 4.19 below, the accuracy percentage is consistently above 95% when reached epochs 10 and mini batch size used was 40. The number of epoch is enough to be set to 10 because it reached highest accuracy percentage and does not have much difference if added a greater number of epoch. The training time taken is about 3 minutes 21 seconds.

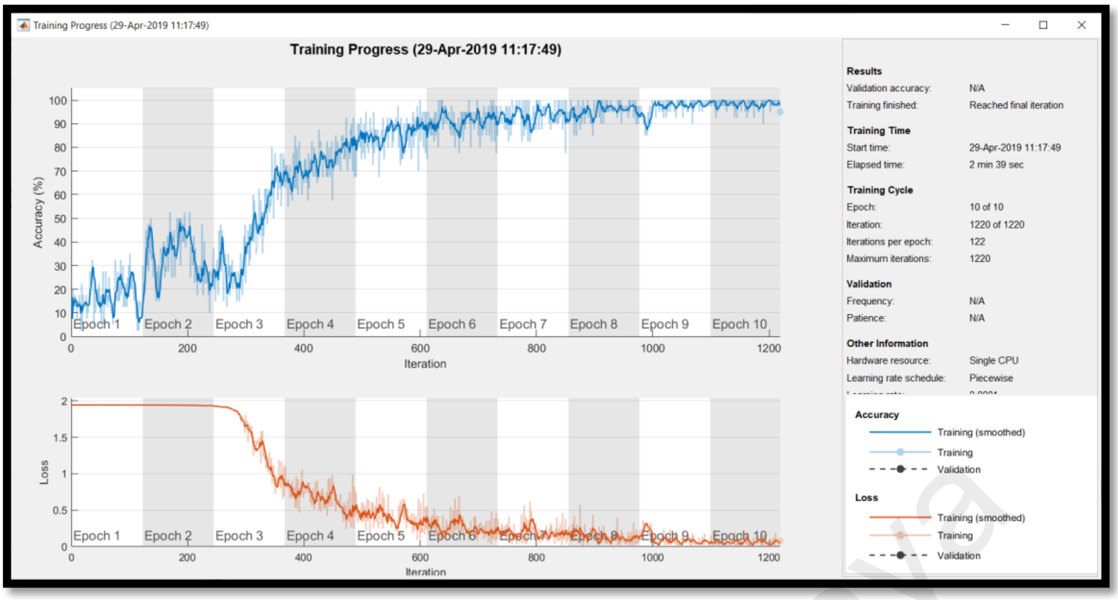


Figure 4. 19: Training progress with mini batch size 40 and number of epoch 10

Figure 4.20 below shows confusion matrix for each varieties of leaf lettuce. Class Celtuce Love obtained highest percentage accuracy with 99%, the lowest is class SGR with accuracy of 92.3% and average percentage accuracy for leaf recognition was 96.5%. According to the confusion matrix below, for class SGR has large number of misclassification which is out of 300 images, the model misclassified 14, 5 and 4 images of SGR as Butterhead, Italian and Red Coral respectively. Figure 4.21 shows the result of leaf recognition prediction image after training and testing the model.

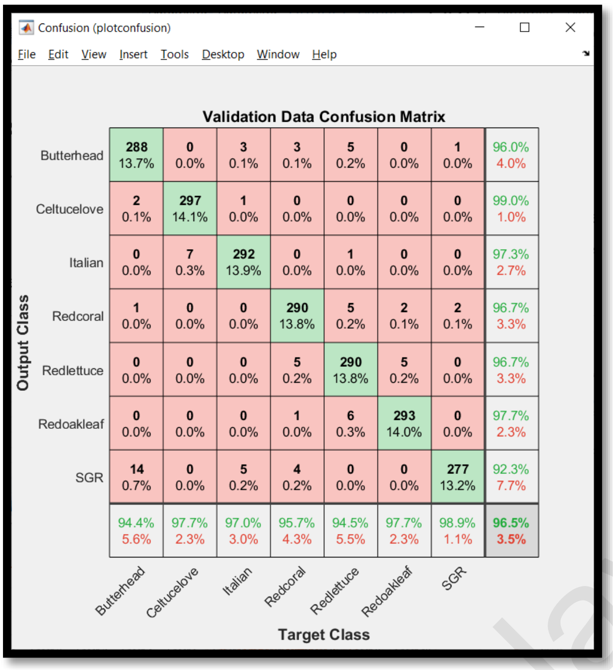


Figure 4. 20: Confusion matrix for mini batch size 40 and number of epoch 10

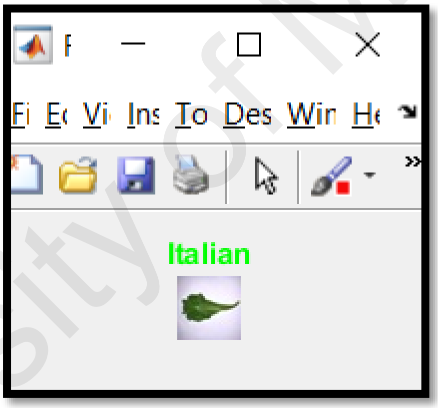


Figure 4. 21: Random image from testing file after training the model

Although the result for the parameter set before achieving better accuracy percentage, validation error is still more than 3%. Thus, the parameter was adjusted for epochs was set to 10 and mini batch size is reduced to 32. From the Figure 4.22, it shows that the parameter setting was the best since accuracy percentage achieved for each varieties of lettuce was above 90%. Figure 4.23 is the confusion matrix of the model and it show that the validation is reduced to 2.2%. The total number of correct prediction was also improved with five varieties of lettuce have the correct prediction more than 290 out of 300 leaf sheets and the others two type are correctly predicted above 282 out of 300 leaf sheets of testing image. Figure 3.24 shows the result of leaf recognition prediction image after training and testing the model.

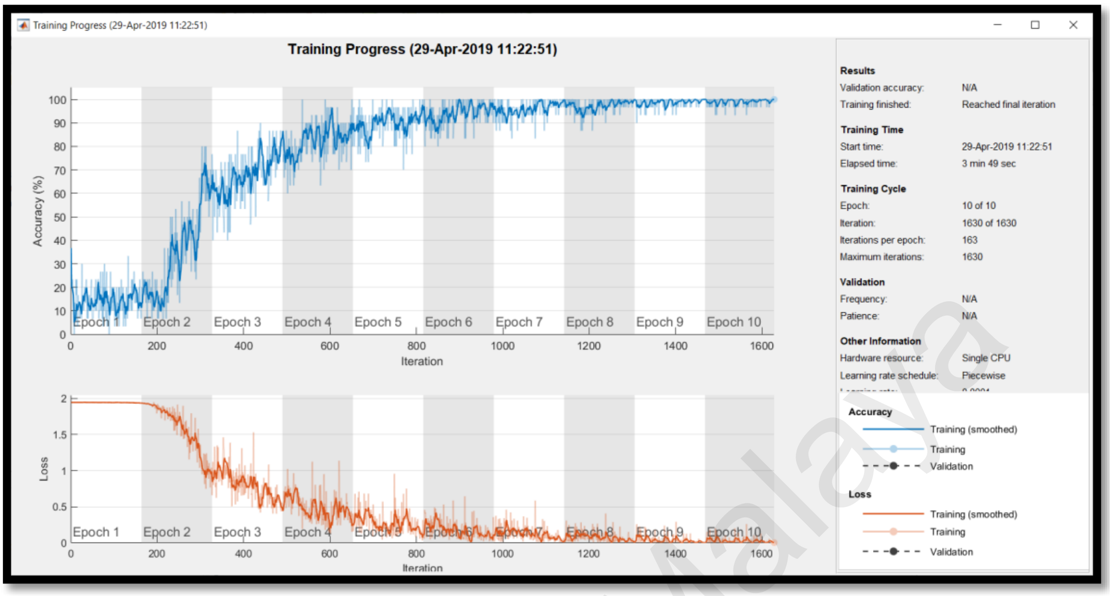


Figure 4. 22: Training progress with mini batch size 32 and number of epoch 10

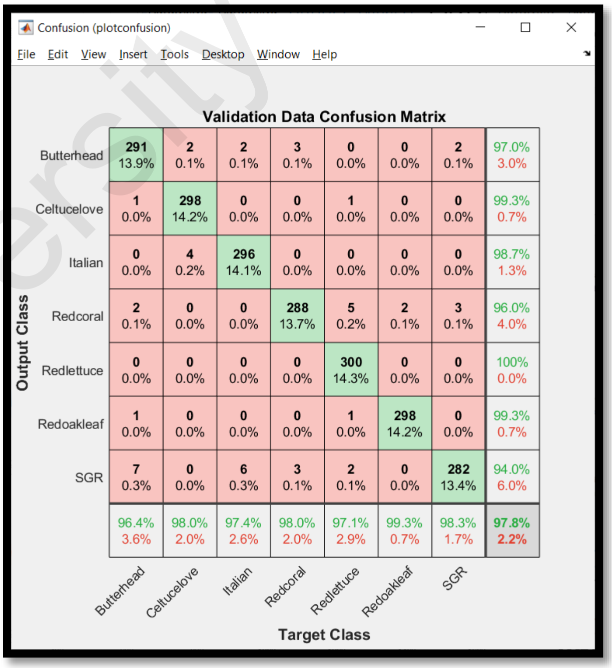


Figure 4. 23: Confusion matrix for mini batch size 32 and number of epoch 10

Figure 4. 24: Random image from testing file after training the model

Table 4.1 below shows the summary of all parameter setting for the model. The parameter needs to be tuned in order to get the better accuracy of the model. Figure 4.25 shows the result of the average accuracy and validation error percentage. It shows that the accuracy is increased when the size of mini batch is reduced while the validation error can reduced until 2.2%. Therefore, the parameter setting is suitable to the model since the average accuracy and average of each varieties of lettuce is more than 90%.

Table 4. 1: Summary of all parameter setting for the model

No of epochs	Mini batch size	Average Accuracy (%)	Validation error (%)	Training time
10	80	79.8	20.2	2 min 44 sec
15	80	83.7	16.3	6 min 20 sec
20	80	91.7	8.3	6 min 9 sec
10	64	92.8	7.2	1 min 30 sec
20	64	93.0	7.0	5 min 30 sec
10	40	96.5	3.5	2 min 39 sec
15	40	96.8	3.2	4 min 02 sec
10	32	97.8	2.2	3 min 49 sec

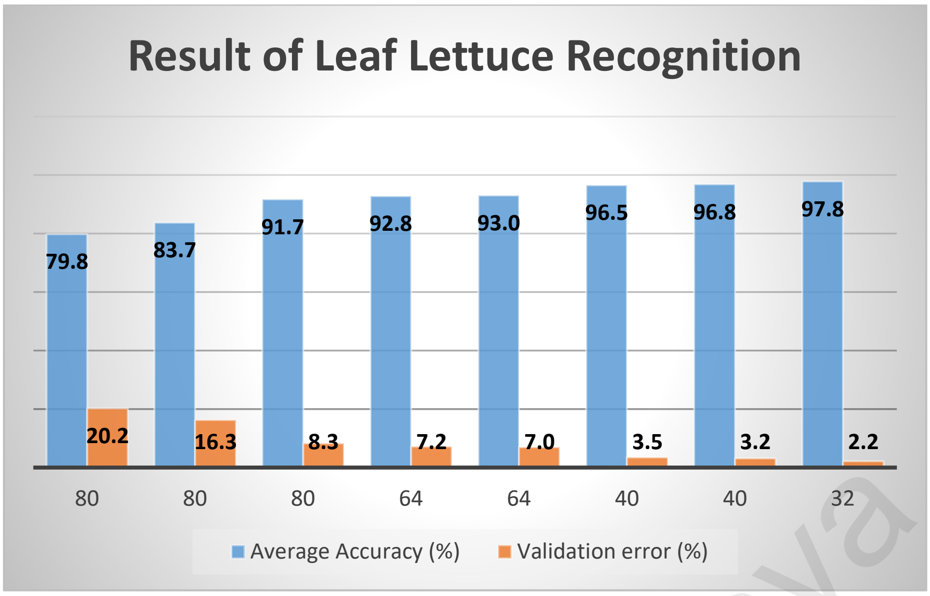


Figure 4. 25: Result of leaf lettuce recognition

4.2.2 Bouquet Lettuce Recognition

For this session, mini batch size is set to 20 because the images for bouquet lettuce is only 200, 400 and 400 images for Butterhead lettuce, Red Oakleaf lettuce and Red Coral lettuce respectively. Figure 4.26 shows the training progress of the model. The system is in process of learning from the epoch 0 until 15. From the figure below, the training accuracy was consistently above 95% when reached epochs 7. The training time taken is about 55 seconds.

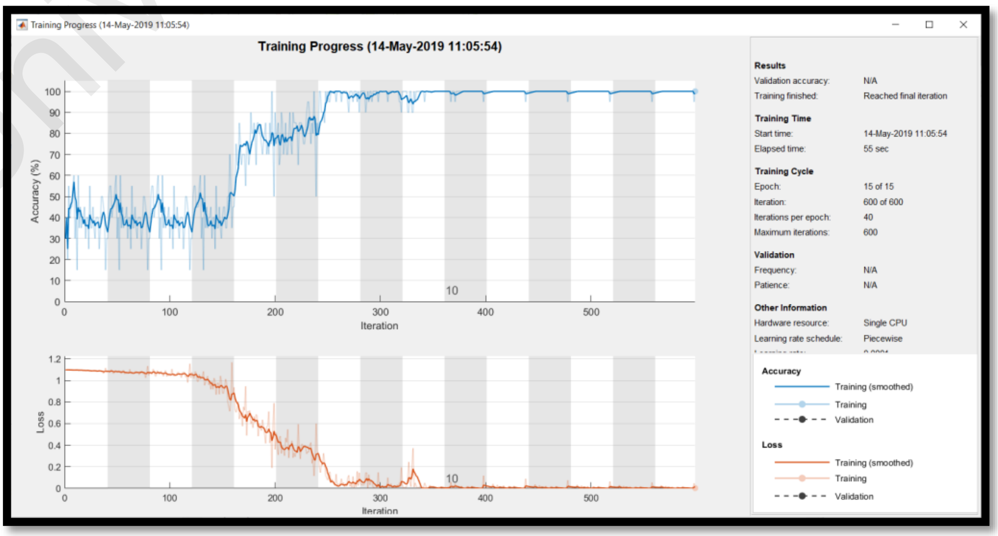


Figure 4. 26: Training progress with mini batch size 20 and number of epoch 15

Figure 4.27 below shows the single and average accuracy for the recognition of bouquet lettuce. Class Red Oakleaf and Butterhead obtained 100% accuracy while class Red Coral achieved 97.5% accuracy and the average accuracy for bouquet recognition is 99%. The validation error of the model is only 1%. Figure 4.28 shows the result of bouquet lettuce recognition prediction image after training and testing the model.

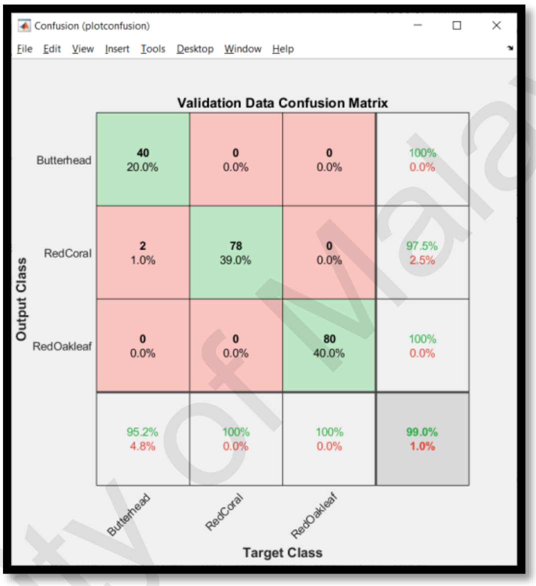


Figure 4. 27: Confusion matrix for mini batch size 20 and number of epoch 15

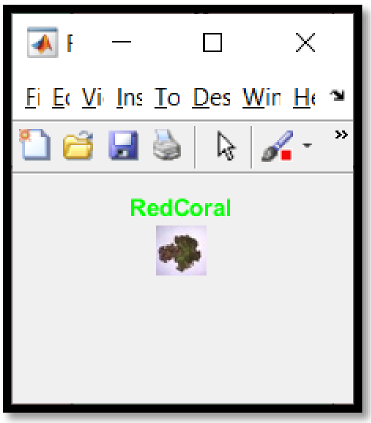


Figure 4. 28: Random image from testing file after training the model

4.3 Summary

Based on the result obtained for leaf lettuce recognition, improvement on the model's performance of the model can be done by tuning the parameters such as epoch's number and mini batch size. The best parameter, i.e. mini batch size for leaf lettuce recognition in this experiment is 32. The number of epoch is 10 because the training accuracy becomes stable as it reached epoch 8. For bouquet lettuce recognition, the total image sample is 1000 and they are only to recognize for three varieties of lettuce. Result obtained for this experiment is 99% for average accuracy. Mini batch size for bouquet lettuce recognition is 20 and it may happen because of the total of image training is quite small. Both model can be applied in real condition since both of them get the accuracy more than 95%. However, an image sample for training and testing need to add on further to get better and accurate result for bouquet lettuce recognition.

CHAPTER 5: CONCLUSION AND FUTURE WORK

CNN approach to learn features from leaf or bouquet lettuce images with classifiers for lettuce recognition was studied. Based on the results obtained, it is justified CNN was able to determine and obtained high accuracy percentage prediction of leaf and bouquet lettuce images in the MATLAB simulation window. Based on leaf images, the model was able to differentiate seven different varieties of lettuce with high classification accuracy. Originally, the project was proposed to differentiate ten varieties of lettuce instead of seven. The reduction for varieties of lettuce to differentiate was due to space constraint to cultivate the lettuce and time consumption as one variety of lettuce takes about a month to grow. Bouquet lettuce experimentation limited to only three varieties of lettuce because other varieties of lettuce did not produce good output. Both leaf lettuce and bouquet lettuce image recognition experiment, obtained high accuracy percentage above 90%. However, based on my observation, bouquet lettuce is the best image recognition method in order to differentiate varieties of lettuce since the model predict the images was 100% correct for 2 types of lettuce whereas the other one just misclassified 2 images only. In reality or practical use, when applying the model to differentiate varieties of lettuce, user will take an image of the bouquet lettuce and not leaf lettuce.

For future improvement, more images are needed for testing and training the model to evaluate and verify that the model able to predict varieties of lettuce accurately. From the model, the application to recognize varieties of lettuce can be develop further and other information related to lettuce such as significant name, family name, benefit of lettuce or market price can be incorporated. Besides that, other features can be included also to differentiate size of lettuce based on growth day. Farmers can observe the growth of lettuce real-time or notified best time to harvest.

REFERENCES

Adams Wei Yu, Lei Huang, Qihang Lin, Ruslan Salakhutdinov, and Jaime Carbonell (2018). Block-Normalized Gradient Method: An Empirical Study for Training Deep Neural Network. arXiv:1707.04822v2

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton (2017). ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM.* 84-90.

Andreas Kamilaris and Francesc X. Prenafeta-Boldu (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 70-90.

ArunPriya C., Balasaravanan T. andAntony Selvadoss Thanamani (2012). An Efficient Leaf Recognition Algorithm for Plant Classification Using Support Vector Machine. *Proceedings of the International Conference on Pattern Recognition, Informatics and Medical Engineering,* 428-432.

Avneet Pannu (2015). Artificial Intelligence and its Application in Different Areas. *International Journal of Engineering and Innovative Technology (IJEIT),* 79-84.

Bishop, C. M. (2006). Pattern Recognition and Machine Learning. *Springer, New York.*

Chathurika Sewwandi Silva and Upul Sonnadara (2013). Classification of Rice Grains Using Neural Networks. *Proceedings of Technical Sessions,* 9-14.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich (2015). Going Deeper with Convolutions. *IEEE*, 1-9.

Jiazhi Pan and Yong He (2008). Recognition of plants by leaves digital image and neural network. *International Conference on Computer Science and Software Engineering,* 906-909.

Jixiang Du, Deshuang Huang, Xiaofeng Wang, and Xiao Gu (2005). Shape Recognition Based on Radial Basis Probabilistic Neural Network and Application to Plant Species Identification. Berlin: Springer Verlag, 281–285.

John Murphy (2016). *An Overview of Convolutional Neural Network Architectures for Deep Learning.* Microway

Jyotismita Chaki and Ranjan Parekh (2011). Plant Leaf Recognition using Shape based Features and Neural Network classifiers. *International Journal of Advanced Computer Science and Applications,* 41-47.

Juncheng Maa, Keming Dua, Feixiang Zhenga, Lingxian Zhangb, Zhihong Gongc and Zhongfu Suna (2018). A Recognition Method for Cucumber Diseases using Leaf Symptom Images Based on Deep Convolutional Neural Network. *Computers and Electronics in Agriculture,* 18-24.

Karen Simonyan and Andrew Zisserman (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *International Conference on Learning Representations.* 1-14.

Konstantinos P. Ferentinos (2018). Deep Learning Models for Plant Disease Detection and Diagnosis. *Computers and Electronics in Agriculture*, 311-318.

Mads Dyrmann (2017). Automatic Detection and Classification of Weed Seedlings under Natural Light Conditions.

Qian, N. (1999). On The Momentum Term in Gradient Descent Learning Algorithms. *Neural Networks: The Official Journal of the International Neural Network Society, 12(1)*, 145–151

Ronnel R. Atole and Daechul Park (2018). A Multiclass Deep Convolutional Neural Network Classifier for Detection of Common Rice Plant Anomalies. *International Journal of Advanced Computer Science and Applications,* 67-70.

Sue Han Lee, Chee Seng Chan, Paul Wilkin and Paolo Remagnino (2015). Deep-Plant: Plant Identification with Convolutional Neural Networks. *ICIP 2015,* 452-456.

Stephen Gang Wu, Forrest Sheng Bao, Eric You Xu, Yu-Xuan Wang, Yi-Fan Chang and Qiao-Liang Xiang (2017). A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network. *IEEE International Symposium on Signal Processing and Information Technology,* 11-16.

Stuart J. Russell and Peter Norvig (2010). Artificial Intelligence: A Modern Approach. *Pearson Education,* 1-5.

Tom Fawcett (2006). An Introduction to ROC Analysis. *Pattern Recognition Letters. 27 (8)*: 861–874.

Vaibhav Amit Patela and Manjunath V. Joshib (2017). Convolutional Neural Network with Transfer Learning for Rice Type Classification. *Proceedings of the SPIE,* 89-96.

Vijay Satti, Anshul Satya and Shanu Sharma (2013). An Automatic Leaf Recognition System for Plant Identification Using Machine Vision Technology. *International Journal of Engineering Science and Technology (IJEST),* 874-879.

Weiwei Shi, Yihong Gong and Jinjun Wang (2016). Improving CNN Performance with Min-MaxObjective. *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence,* 2004-2010.

Xiao-Feng, Wang, De-Shuang, Huang, Ji-Xiang, Dua Huan Xu and Laurent Hutte (2008). Classification of Plant Leaves with Complicated Background. *Applied Mathematics and Computation,* 916–926.

Yann LeCun, Leon Bottou, Yoshua Bengio and Patrick Haffner (1998). Gradient-based Learning Applied to Document Rocognition. *Proceeding of the IEEE,* 1-46.