The Detection of Vegetation Species in Remote Sensing Imaging

Nassr Azeez

A thesis submitted for the degree of

Doctor of Computer Science at the

Department of Computer Science and Electronic Engineering

University of Essex

February 2021

Abstract

Remote sensing classification is a complicated process and requires the perception of many factors. The main image classification factors that can be taken into consideration to improve classification accuracy may include placement of a sufficient classification system, chosen of training samples, feature extraction, and selection of appropriate classification techniques.

The focus of the research in this thesis is on the monitoring of cereal crops, in particular infestation of wheat crops by black grass, a pernicious weed. This is regarded as the major pest for cereal crops, at least in East Anglia, one that causes significant loss of income for farmers. Although annual rather than perennial, a single back grass plant is able to produce many thousands of seeds, which are small enough to be distributed by the wind, though it is believed that the principal means by which it spreads from field to field is by sticking inside farm machinery such as combine harvesters. Anecdotal evidence from farmers is that black grass plants tend to grow directly adjacent to wheat plants, with their roots enveloping those of the wheat and thus effectively strangling the wheat. Black grass is resistant to most weed-killers, and spraying with those weed-killers that are allowed requires a licence — this makes eradicating it difficult. When young black grass plants are potentially detectable because they do not grow in the regular pattern of the wheat plants; but after germination, they tend to be smaller than adjacent wheat plants until they reach maturity, at which time they are slightly taller and with their characteristic black seed-heads.

Thus the purpose of this thesis is to explore the detection of black grass infestations of wheat crops.

Imagery of fields infested with black grass plants is simply not available at the kinds of spatial resolution needed for detection, so an important part of the research was the collection of imagery from a drone. This was done over more than one year and at critical points in the growth of the crop and its 'companion' weed, so that detection at different stages of the growth cycle is potentially possible. However, it should be pointed out that black grass is difficult to detect when immature, except by walking amongst the plants and examining them individually.

The thesis starts by reviewing remote sensing image principles in general terms. It goes on to discuss the use of Image classification in remote sensing and the major steps that may be involved in the image classification process. It sheds the light on the two main approaches that used for classification in remote scening; supervised and unsupervised learning with some detail of the particular techniques used in this thesis, ranging from conventional statistical ones to Genetic Programming and Convolutional Neural Networks, are presented along with a review of their application in remote sensing. The thesis presents two experimental work that used different conventional remote sensing classification techniques. The former sheds light on the regional agricultural land texture classification using grey level cooccurrence matrices (GLCMs) for features extraction and using SVM and Decision Tree Induction techniques for classification. The later focuses on outcomes gained from three conventional supervised techniques that used for remote sensing images classification; random forest, SVMs, and decision tree classifiers. The thesis focuses on analysis by Vegetation Indices (VIs) and sheds light on the fundamental aspects of the most common indices that analysts use in remote sensing; NDVI, CRI2, SG index, and RG Ratio index. The thesis presents two genetic programming toolkits developed at Essex University: the Jasmine vision system builder and ELVS (evolutionary learning vision system) to detect the tree crown regions and consequently measure its efficiency to solve the black grass detection problem. In addition, it presents the relevant fundamentals of convolutional neural networks (CNNs) that used for remote sensing images classification. Five deep convolutional neural networks (Google-NET, VGG16, VGG19, ResNET50, ResNET101) and shallow convolutional neural networks have been trained to five different classes of ROI images including unknown range: wheat, black grass, road, and bushes. Finally, the

conclusions drawn from this research and makes suggestions for further work have been presented.

Acknowledgements

At the forefront, I would like to express my gratitude to the University of Baghdad and the Ministry of Higher Education and Scientific Research of Iraq for offering me a scholarship to obtain the PhD degree, and providing me the opportunity to undertake this experience. I would like also to thank the University of Essex, UK at which I was able to carry out this research.

I am indebted to my supervisor, Dr. Adrian F. Clark, for his continuous support and guidance throughout the thesis stages. He has given his knowledge and put in effort at all times for the benefit of this thesis.

I would also like to acknowledge the immense impact and support that my family has given me. In particular, my father and mother whose encouragement was beyond description.

I am more than grateful to my wife, (Dr. Inas Al-Taie) for her never-ending support throughout this research. Moreover, my special thank to my wonderful boys, Abdulazeez, Mohammed, and my new born baby Ryan for always making me smile and for their patience and understanding whilst doing my PhD thesis work.

Finally, my appreciation also goes out to my friends in Iraq and UK for their support and to all the rest of the family for believing in me and keeping me going. ...

Contents

List of Figures

List of Tables

Chapter 1

Introduction

It is clear facts that a high percent of Earth's land area is agricultural land. It is in farmer interests to be good stewards of the land. In turn, well-managed agricultural land provides food and habitat for wildlife. Nationwide, there is growing demand for locally-grown food. Food production, depends on the availability of farm and ranch land. Therefore, advocates for local food systems must also work to protect agricultural land and take steps to keep the land in active production.

Black-grass is one of the biggest challenges that facing most the arable farmers in the UK. Black-grass is an annual weed of wheat that occurs in the Uk and Europe. Mainly, it is found in the cereal growing areas and it rarely occurs outside of cultivated area. Commonly, the flowers heads show above the cereal crop in May and June. In general, black-grass occurs in autumn. Therefore, the spread of sown cropping in autumn is the main reason that why black-grass is an numerous problem in the UK. In field trials, spring sown crops are much less affected and have given a good reduction in weed infestation. However, establishing crops in spring can be hard especially with heavy soil and the choice of herbicide is more restricted.

Black-grass can seriously reduce arable crop yields through competition for nutrients, especially nitrogen. Effects on crop yield vary greatly due to the relative competitiveness of crop and weed - a

competitive crop can strongly affected by black-grass.

Controlling black-grass is the major challenge that facing most arable farmers in the UK. Black-grass control is a year-round process, using different of cultural and chemical controls. The first step to establish black-grass control is to check black-grass populations in June and July which are the best times to check populations and then plan the control programme. After the black-grass populations step, planning crop rotation, cultivation and establishment technique followed by a herbicide programme in autumn and spring are established to maximise levels of control.

Anecdotal evidence from farmers is that black grass plants tend to grow directly adjacent to wheat plants, with their roots enveloping those of the wheat and thus effectively strangling the wheat. Black grass is resistant to most weed-killers, and spraying with those weed-killers that are allowed requires a licence — this makes eradicating it difficult. it should be pointed out that black grass is difficult to detect when immature, except by walking amongst the plants and examining them individually.

1.1 Why Is Black-Grass A Problem in The UK?

There are three main reasons:

- Increase in autumn sown crops which greatly favour black-grass due to its early autumn emergence pattern.
- Earlier autumn sowing - in 1975 only 5% of winter wheat was sown in September; in recent years this proportion has increased to 50%.
- Herbicide resistance - some degree of resistance now occurs on virtually all farms.

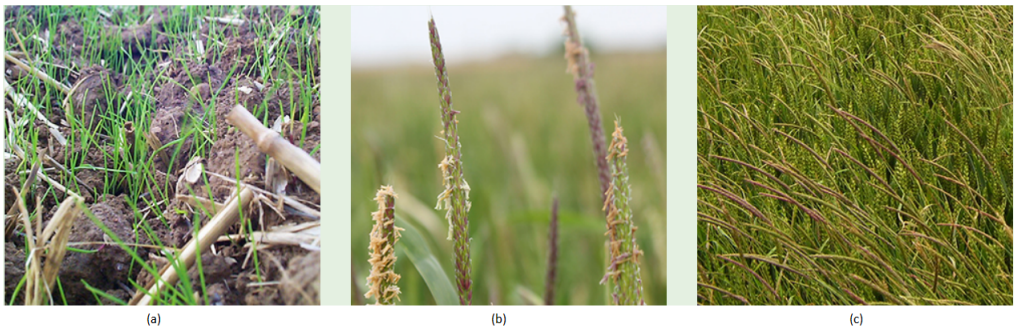

Figure 1.1: Examples of black-grass. (a) Shows seeding germination in April. (b) and (c) Show black-grass heads in July.

1.2 Identifying Black-Grass

Seedling Leaves are fine and smooth with often blue-green upper surface, stem bases are often purple, and leaf blade is twisted with a blunt tip.

Heads Heads compact narrow 2-13cm long and 3-6mm diameter and produce on average 100 seeds per head. july

Characteristics Tall erect annual grass 20-90cm tall forming either a single shoot or clump depending on the competitiveness of the crop; stems round and slender with few nodes. Figure 1.1 shows some examples of black-grass.

1.3 What Is The Economic Impact of Black-Grass?

Researchers from the University of Sheffield and ZSL's Institute of Zoology found that the UK is losing roughly 5% of the UK's domestic wheat consumption, due to black-grass. The research explained that the widespread use of herbicides is resulting to herbicides resistant black-grass problem. Consequently , it costs UK millions in lost profit [SheffieldUniversity, 2019].

Black-grass can seriously reduce arable crop yields through competition for nutrients. Effects on crop

yield vary greatly due to the relative competitiveness of crop and weed.

1.4 Research Objectives

The focus of the research is the monitoring of cereal crops, in particular infestation of wheat crops by black grass, a pernicious weed. This is regarded as the major pest for cereal crops, at least in East Anglia, one that causes significant loss of income for farmers. Although annual rather than perennial, a single back grass plant is able to produce many thousands of seeds, which are small enough to be distributed by the wind, though it is believed that the principal means by which it spreads from field to field is by sticking inside farm machinery such as combine harvesters. Anecdotal evidence from farmers is that black grass plants tend to grow directly adjacent to wheat plants, with their roots enveloping those of the wheat and thus effectively strangling the wheat. Black grass is resistant to most weed-killers, and spraying with those weed-killers that are allowed requires a licence — this makes eradicating it difficult.

When young black grass plants are potentially detectable because they do not grow in the regular pattern of the wheat plants; but after germination, they tend to be smaller than adjacent wheat plants until they reach maturity, at which time they are slightly taller and with their characteristic black seed-heads. Imagery of fields infested with black grass plants is simply not available at the kinds of spatial resolution needed for detection, so an important part of the research was the collection of imagery from a drone. This was done over more than one year and at critical points in the growth of the crop and its 'companion' weed, so that detection at different stages of the growth cycle is potentially possible. However, it should be pointed out that black grass is difficult to detect when immature, except by walking amongst the plants and examining them individually.

In general, wheat is planted in two seasons. Winter wheat is planted in the fall, usually between October and December, and grows over the winter to be harvested in the spring or early summer. Spring wheat is usually planted between March and May and should be harvested between July and September.

Thus the purpose of this thesis is to explore the detection of black grass infestations of wheat crops (Spring wheat) by using two main approaches that used for classification in remote scening; supervised and unsupervised learning and investigate the efficiency of some other techniques, ranging from conventional statistical ones to Genetic Programming and Convolutional Neural Networks.

1.5 Contributions

This research has made a number of contributions, highlighted in the following series of bullet points.

- Data sets have been collected, some are shared from other research groups (UEA and Cherry Farm team) and the rest was captured by us using UAV.

- Conducting two experimental work that used different conventional remote sensing classification techniques to detect Black-Grass and separating the affected area. The former sheds light on the regional agricultural land texture classification using grey level cooccurrence matrices (GLCMs) for features extraction and using SVM and Decision Tree Induction techniques for classification. The later focuses on outcomes gained from three conventional supervised techniques that used for remote sensing images classification; random forest, SVMs, and decision tree classifiers (Chapter 3).

- ENVI 5.3 software has been employed to calculate the most common vegetation indices: NDVI, CRI2, SG index, and RG Ratio index to detect black grass regions (Chapter 4).

- Two genetic programming toolkits developed at Essex University have been tested: the Jasmine vision system builder and ELVS (evolutionary learning vision system) to detect the tree crown regions and consequently measure its efficiency to solve the black grass detection problem (Chapter 5).

- Five deep convolutional neural networks (Google-NET, VGG16, VGG19, ResNET50, ResNET101)

and five shallow convolutional neural networks have been trained to five different classes of ROI images including unknown range: wheat, black grass, road, and bushes to detect the the Black-Grass area (Chapter 6).

The contributions to the knowledge that extracted from this research are summarized as two approaches:

- Detecting The Black Grass Using Manual Methods

  The results show that the extracted texture features using grey level cooccurrence matrices (GLCMs) have high discrimination accuracy and classification using support vector machines outperforms the decision tree technique performance in terms of the number of mis-classifications instances and the classification accuracy with an overall accuracy of 83%, while the decision tree accuracy is 70%, respectively.

  The results of using three conventional supervised techniques for remote sensing images classification; random forest, SVMs, and decision tree classifiers indicate that the random forest classifier performance outperforms the decision tree and SVMs techniques performance in terms of the number of mis-classifications instances and the classification accuracy with an overall accuracy of 86% while the Decision Tree accuracy is 67% and the SVMs accuracy is 56%, respectively.

  The experiment results of using Vegetation Indices (VIs) to detect black grass regions indicate that using NDVI, CRI2, SG index, and RG Ratio index can achieve high precision up to 90%.

  Results indicate that Jasmine is not able to solve many problems concerning remote sensing images classification well, mainly because it lacks operators that able to classify on the basis of colour and texture. In addition, ELVS fail to evolve a program that could segment black grass regions effectively.

- Detecting The Black Grass Using Automated Methods

  The experimental results of using five deep convolutional neural networks (Google-NET, VGG16,

VGG19, ResNET50, ResNET101) and shallow convolutional neural networks show that the classification performance based on deep CNN outperformed the classification performance based on shallow CNN in the majority of cases to detect black grass regions. Regarding to the average accuracy for the four group, deep ResNet50 network achieved the highest accuracy for validating and testing phases (89.4%, 85.0%) in time (00:15:43). While the classification accuracy for validating and testing phases recorded the highest score (87.4%, 81.5%) based on the shallow ResNet101 network in time (00:16:49).

1.6 Publications

There have been 7 published conference papers generated from parts of the thesis contributions and different application areas. Furthermore, one conference paper is under revision. First 3 publications directly relevant to my research, last 5 are in different application areas but the techniques use have contributed. The publications are:

1. Nassr Azeez, Inas Al-Taie, Wafa Yahya, Arwa Basbrain, Adrian Clark. "Regional Agricultural Land Texture Classification Based on GLCMs, SVM and Decision Tree Induction Techniques." In Computer Science and Electronic Engineering (CEEC), 2018.

2. Nassr Azeez, Wafa Yahya, Inas Al-Taie, Arwa Basbrain, Adrian Clark. "Regional Agricultural Land Classification Based on Random Forest (RF), Decision Tree, and SVMs Techniques." In Fourth International Congress on Information and Communication Technology, pp. 73-81. Springer, Singapore, 2020.

3. Nassr Azeez, Inas Al-Taie, Adrian Clark. "Detection of Black-grass of Wheat Crops Using Deep and Shallow Convolutional Neural Networks." (Under Revision).

4. Inas Al-Taie, Nassr Azeez, Arwa Basbrain, and Adrian Clark. "The Effect of Distance Similarity Measures on the Performance of Face, Ear and Palm Biometric Systems." In Digital Image Computing: Techniques and Applications (DICTA), 2017 International Conference on, pp. 1—7. IEEE, 2017.

5. Inas Al-Taie, Nassr Azeez, Wafa Yahya, Arwa Basbrain, Adrian Clark. "Biometric Recognition Systems Based on SVM pca and SVM pca, lda Techniques." In 2019 Third World Conference on Smart Trends in Systems Security and Sustainablity (WorldS4), pp. 158-161. IEEE, 2019

6. Inas Al-Taie, Adrian Clark, and Nassr Azeez. "Improving The Viola-Jones Face Detection Performance by Using The Brightness Channel in HSV and HLS Colour Spaces." In Irish Machine Vision and Image Processing (IMVIP), 2017 International Conference on, PP. 178—185. IPRCS, 2017.

7. Inas Al-Taie, Adrian Clark, and Nassr Azeez. "Similarity Measures and the Performance of Biometric Systems." In Irish Machine Vision and Image Processing (IMVIP), 2017 International Conference on, PP. 115—122. IPRCS, 2017.

8. Basbrain, Arwa M., Inas Al-Taie, Nassr Azeez, John Q. Gan, and Adrian Clark. "Shallow convolutional neural network for eyeglasses detection in facial images." In Computer Science and Electronic Engineering (CEEC), 2017, pp. 157—161. IEEE, 2017.

1.7 Thesis Structure

The first part of chapter 2 starts by reviewing remote sensing image principles in general terms. It goes on to discuss the use of Image classification in remote sensing and the major steps that may be involved in the image classification process. The second part sheds the light on the two main approaches that used for classification in remote scening; supervised and unsupervised learning. Also, some detail of the particular techniques used in this thesis, ranging from conventional statistical ones to Genetic Programming and Convolutional Neural Networks, are presented along with a review of their application in remote sensing.

Chapter 3 presents two experimental work that used different conventional remote sensing classification techniques. The former sheds light on the regional agricultural land texture classification using grey level cooccurrence matrices (GLCMs) for features extraction and using SVM and Decision Tree Induction techniques for classification. The later focuses on outcomes gained from three conventional supervised techniques that used for remote sensing images classification; random forest, SVMs, and decision tree classifiers.

Chapter 4 focuses on analysis by Vegetation Indices (VIs). The first part of this chapter illustrates the study area that been used in research. The second part sheds light on the fundamental aspects of the most common indices that analysts use in remote sensing; NDVI, CRI2, SG index, and RG Ratio index. In addition, the experiment results of using these indices to detect black grass regions have been introduced.

Chapter 5 presents two genetic programming toolkits developed at Essex University: the Jasmine vision system builder and ELVS (evolutionary learning vision system) to detect the tree crown regions and consequently measure its efficiency to solve the black grass detection problem. The first part of this chapter describes the principles underlying the Jasmine and identifies its major shortcomings. In the second part, the tree crown detection problem has been explored using the mentioned toolkits.

Chapter 6 presents the relevant fundamentals of convolutional neural networks (CNNs) that used for remote sensing images classification. Five deep convolutional neural networks (Google-NET, VGG16, VGG19, ResNET50, ResNET101) and shallow convolutional neural networks have been trained to five different classes of ROI images including unknown range: wheat, black grass, road, and bushes. Caffe is a very widely used framework for deep learning. In this chapter, we explain how to get started with CAFFE and the steps that should be set.

Finally, chapter 7 presents the conclusions drawn from this research and makes suggestions for further work.

Chapter 2

Background and Literature Review

2.1 Introduction

Remote sensing classification is a complicated process and requires the perception of many factors. The main image classification factors that can be taken into consideration to improve classification accuracy may include choice of training samples, feature extraction, and selection of appropriate classification techniques. The first part of this chapter starts by reviewing remote sensing image principles in general terms. It goes on to discuss the use of Image classification in remote sensing and the major steps that may be involved in the image classification process. The second part sheds the light on the two main approaches that used for classification in remote sensing; supervised and unsupervised learning. Some detail of the particular techniques used in this thesis, ranging from conventional statistical ones to Genetic Programming and Convolutional Neural Networks, are presented along with a review of their application in remote sensing.

2.2 Weed Detection Using Digital Images

The detection and classification of weeds have technical and economical importance in the agricultural industry. In olden days, weed detection was achieved by checking every place in the field and weeds were removed manually. Later with the technological development, farmers started using herbicides to remove the weeds. Herbicide use is a main method for weed control, however if its volume is excessive there are potential damages to the environment and to other living organisms [Carpenter and Gianessi, 2000]. Usually, herbicides are uniformly distributed over the entire area even though it is well known that they locative distribution is not uniform. Therefore the herbicide quantity can be extremely reduced using with precision farming methods. From this point, image processing techniques were used to detect the weed in the crop [Paikekari et al., 2016]. Weed mapping which distinguished infestations in a production region, has been referred to as one of the tools used for the spacial and specific application of herbicides with fewer environmental damages. [Sansao et al., 2012] proposed a system that based on using image processing to calculate the weed coverage percentage in no-till field of common bean. Images are initially converted in an excess green index (ExG) image, then segmented by a threshold value to indicate the vegetation regions. They employed the feature of appearing crop lines appear in regular lines while weed in a more scattered pattern to discriminate crop regions. Using the estimated vegetation and crop areas, a weed percentage value was allocated to each image. Since these images were geo-referenced, it was possible to construct weed coverage maps by means of a linear interpolation method. They compared between the reference and the automatically generated maps to reveal the capability to coherently distinguish high from low weed infestation in similar regions. [Murdoch et al., 2014] described eyeWeed, a machine vision system, which can be fitted on farm machinery such as sprayers, for automated weed mapping. The eyeWeed project included cameras fitted on groundbased vehicles to capture images of appropriate quality to create farmer acceptable herbicide application maps, images required for mapping need to be captured as an adjunct to normal farming operations such as crop spraying, and weed (black-grass patches) are stable enough so that maps of black-grass heads in June predict patch locations of seedlings for pre-emergence and post-emergence spraying in the

following crop. The researchers indicated that the initial estimations of the economics of automating weed mapping and black-grass patches spraying appear positive. [Paikekari et al., 2016] described weed detection using image processing and how they can detect and separate out weed affected area from the crop plants using image processing. They used color segmentation and edge detection to decrease the handling of herbicides by spraying them only in the areas where weed was present. In the same year, [Desai et al., 2016] proposed removal of weeds in the crop by using image processing. Weeds are extracted from images and described by shape, color and size features. These features were use to classify similar weeds and crop. They described and analyzed different classification techniques; SVM, NN, DA to differentiate weeds and crops. [Aware and Joshi, 2016] proposed crop and weed detection based on texture and size features and automatic spraying of herbicides. Five texture features are used for detection of crop; energy, entropy, inertia, local homogeneity and contrast. Morphological size based features are also used for detection of crop and weed. Compared the all results and taken majority decision for detection of crop and weed.

2.3 Remote Sensing Images Principles

In general, remote sensing is the aggregating and analyzing of information from an Earth's surface object, area, or phenomenon without any direct contact with it, typically from aircraft or satellites [Lillesand et al., 2014]. Remote sensing observation is made from over the object of interest, by using a sensor mounted on a spaceborne platform or airborne [Pellikka and Rees, 2009]. The remote sensing process involves recording, processing, and analyzing reflected or emitted energy from the region of interest area [Rees and Pellika, 2010]. The importance of remote sensing involve the ability; to gather information over large locative areas to identify natural features or objects on the Earth's surface; to recognize surface areas and observe their variation over time; and to integrate the acquired data with other.

Examples of applications of remote sensing include population and demography studies, study of energy using hydrological models, environmental treaty enforcement, and urban planning. One of main

application of remote sensing is agriculture yields in which Satellite and airborne images are used as mapping tools to classify crops, examine their health and viability, and monitor farming practices. Although expensive to collect, remotely sensed images gives an overview that cannot be acquired by any other means [Schmugge et al., 2002] [Kustas et al., 2003]

Different image processing methods can be applied to enhance the remotely sensed images to support visual interpretation and to correct the images if it has been subjected to blurring or degradation distortion by other factors [Pellika et al., 2010]. There are various image analysis methods available used depending on the demands of the specific problem interested. In many cases, image classification techniques are employed to delineate different regions in an image into a thematic map of the study area. Usually, the thematic map can be used with other test area databases for additional analysis and utilization [Rees and Pellika, 2010].

2.4 Image Classification in Remote Sensing

Image classification can be affected by various factors such as the landscape complexity in a study area, selected data, and image processing and classification methods. Therefore, designing an effective image processing procedure is an essential condition for successful classification of remotely sensed data into a thematic map [Lu and Weng, 2007].

Image classification is the operation of allocating land cover classes to pixels. For example, classes involve water, forest, urban, grassland and agriculture. In remote sensing there are three essential image classification techniques; supervised, unsupervised, and object-based image analysis (OBIA) which is the most modern technique in image classification [Weih and Riggan, 2010]. Supervised and unsupervised image classification are the two most common approaches [Geography] [Nath et al., 2014]. The first step in supervised classification is that representative samples for each land cover class need to be selected. In turn, this would create a signature file, which holds all training samples spectral informa-

tion. Finally, the last step would be to employ this signature file to apply a classification algorithm. Many different algorithms that been used for classification such as Maximum likelihood Minimum-distance, Support vector machine (SVM), Principal components, and Iso cluster [Li et al., 2014].

In unsupervised classification, pixels are grouped into clusters depending on their properties then each cluster with a land cover class are classified. Unsupervised classification is the most basic technique due to the fact that it does not need samples for classification. In general, unsupervised classification involves two basic steps; generate clusters(number of groups to be generated) and assign classes (assign land cover classes to each cluster). Some of the popular image clustering algorithms using remote sensing are K-means and ISODATA [Abbas et al., 2016].

Generally, supervised and unsupervised image classification is pixel-based in which each pixel has a class. While, object-based image analysis classification categorizes pixels into representative objects (vectors) with different geometries and sizes. In other words, OBIA involves four steps; perform multiresolution segmentation, select training areas, define statistics, and classify. The two most popular segmentation methods that been used in object-based image analysis classification are multi-resolution segmentation in ECognition and segment mean shift in ArcGIS [Stathakis and Vasilakos, 2006] [Li et al., 2014].

Commonly, remote sensing classification has major steps that may involve choosing of a suitable classification system, picking of training images, image preprocessing, feature extraction, choosing of appropriate classification methods, post processing after classification, and accuracy evaluation. This section sheds the light on the major steps that may be included in image classification [Li et al., 2014].

2.4.1 Selection of Remotely Sensed Data

Data that can be exploited by remote sensing which including both airborne and spaceborne sensor differ in temporal, spatial,spectral, radiometric, directional, and polarisation resolutions. Understanding the

characteristics of each sensor data is important to select suitable remotely sensed data to achieve a successful classification in a specific purpose [Barnsley, 1999]. The selection of suitable sensor data requires considering many factors such as user's need, scale and characteristics of a study area image, the availability of different image data, and time and cost limitations [Wulder and Franklin, 2012].

2.4.2 Selection of Training Samples

A suitable number of training samples are essential for a successful image classification. Three main problems are defined for vegetation classifications when medium spatial resolution data are used; defining sufficient scales for mapping, defining separated land cover units that can be recognized by selected remote sensing data, and selecting training samples [Cingolani et al., 2004].

Usually, training samples are collected from fieldwork, spatial resolution aerial photographs or from satellite images. Different collection strategies may be used, but they would affect classification results [Hubert-Moy et al., 2001] [Landgrebe, 2005]. When the landscape of a study area is complex and diverse, selecting sufficient training samples becomes hard. Especially, when medium or coarse spatial resolution data are used for the classification process due to the occurring of a large volume of mixed pixels. Therefore, selection of training samples must consider the spatial resolution of the remote sensing data being used, availability of ground reference data, and the complexity of landscapes in the study area [Chen and Stow, 2002].

2.4.3 Feature Extraction and Selection

Selecting convenient variables is a key step for successfully image classification. Different variables can be employed in image classification, involoving spectral signatures, transformed images, vegetation indices, textural information, and multisensor images. Due to various capabilities to separate land cover regions, the use of considerable variables in a classification process may reduce classification accur-

acy [Hughes, 1968] [Price et al., 2002]. It is essential to choose only the variables that are beneficial for separating earth's land cover or vegetation classes, mostly when hyperspectral or multisource data are used. Different approaches have been used for feature extraction, such as principal component analysis (PCA), minimum noise fraction transform, linear discriminant analysis (LDA), decision boundary feature extraction, non-parametric weighted feature extraction, wavelet transform [Myint, 2001] [Okin et al., 2001] in order to reduce the data redundancy in remotely sensed data or to extract land cover information [Asner and Heidebrecht, 2002] [Platt and Goetz, 2004].

2.4.4 Selection of a Suitable Classification Method

When selecting a classification method to use in remote sensing, many factors can affect the classification process and outcomes including data availability, spatial resolution of the data used, sources of data, landscape complexity, the classification algorithm used, and researcher's knowledge about the study area and his experience with the classifiers used. Each classification method has its own characteristics. Therefore, is not easy to say which of classification method is suitable for a particular study. Different classification outcomes may be gained depending on the classifier(s) that been chosen [Lu and Weng, 2007].

2.5 Remote Sensing Classification Techniques

In a broad sense, image classification is the process of gathering all pixels in an image or remotely sensed data to a given set of labels or land cover themes [Al-doski et al., 2013] As shown in Figure 2.1.

To the present time, there is still need to introduce regional land cover maps for different purposes of public, private, government, and national security applications [Aplin and Atkinson, 2004]. Various conventional and non conventional classification algorithms have been employed for land use and land cover mapping to improve the accuracy of maps and classified images [Al-doski et al., 2013].

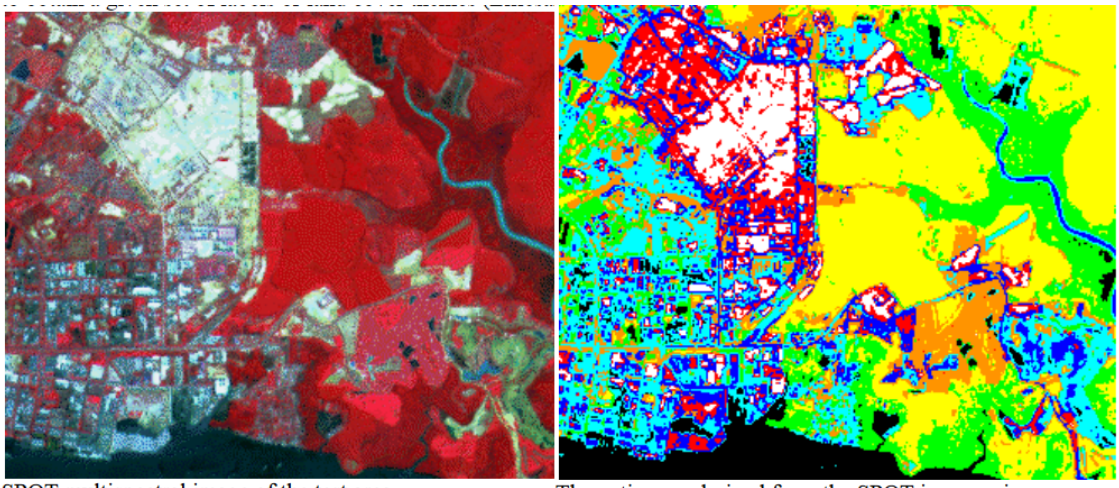

Figure 2.1: Example of Image Classification [Al-doski et al., 2013].

Mullerova [Müllerová, 2005] used multispectral aerial images for mapping changes in sub-alpine vegetation of the Krkonose Mts Plateau, Czech Republic. He used both unsupervised Iterative Self-Organizing Data Analysis technique and supervised Parallelepiped and Maximum Likelihood algorithm to create land cover mapping. The unsupervised classification resulted in 6 classes with a 60.6% overall accuracy. While, the supervised classification resulted in 9 classes with the overall accuracy of 81.1%. [Dewan and Yamaguchi, 2009] employed the ISODATA algorithm with the maximum likelihood method to create land cover classification maps. The overall accuracy of land cover change maps ranged from 85% to 90%. Pabitra et al. Mitra et al. [2004] employed an active support vector learning method for supervised pixel classification in remote sensing image. The goal was to reduce the number of labeled points required to design the classifier. The results indicated that the number of labeled points that required by the active learning algorithm is less compared to the conventional support vector machine. Deilami et al. [Deilami et al., 2014] utilized satellite images to indicate the result of land cover changes in Iskandar Malaysia between 2007 and 2014. The Maximum likelihood supervised classification technique was employed for spotting and monitoring land cover changes. The images were categorized into four types; forest, oil palm, urban and water. The results indicated an increase in the forest and urban area by almost 16.74%, 3.94% and a decrease oil palm by almost 8.57%.

For a specific study, it is still hard to choose a preferred classifier as there is no a guideline for selection

and also for lack of availability of suitable classification algorithms to hand. Consequently, the popular approach is using a comparative analysis various classification algorithms to decide what is preferable for a particular dataset and study. In addition, using combination of various classification algorithms has shown helpful for improving the classification accuracy [Al-doski et al., 2013]. Foody and Mathur [Foody and Mathur, 2004] tested Support Vector Machine against Decision Trees and Neural Network data in an agricultural area in England to create land use/land cover maps and determined the highest accuracy. The experimental results showed that the Support Vector Machine algorithm performance was the better of the three classification methods. A study conducted by Pal and Mather [Pal and Mather, 2005] in which a comparison of Support Vector Machines, Maximum Likelihood, and Neural Network classifiers to classify land cover areas using Landsat 7 ETM+ and hyper spectral data. For both typed of data, results showed that Support Vector Machines produced the most accurate results. [Guo et al., 2008] compared four broad classification methods; Maximum Likelihood Classification, Self Organized Neural Network, Support Vector Machine, and Decision Tree Classification. The experimental results showed that the Decision Tree Classification algorithm outperformed the other three methods.

To achieve high accuracy classified images and maps, some researchers studied the ability to present a Hybrid classification technique by combining some of the classification methods together to take advantage of both the supervised and unsupervised classification. [Zaki et al., 2011] used hybrid classification by combining unsupervised classification (ISODATA) and supervised classification Maximum likelihood to create land cover maps in Egypt. In a similar study, [Alphan et al., 2009] employed the same hybrid combination on multi temporal Landsat and ASTER imagery to assess land cover changes in Turkey. Both studies indicated that using hybrid classification combination was useful to achieve classification accuracy.

[Kantakumar and Neelamsetti, 2015] developed a hybrid classification approach by using a combination of parametric and non-parametric approaches. The developed approach involved the post classification refinement using threshold based knowledge technique and the decision tree approach which is non-parametric and based on the expert knowledge.

Generally, there are many techniques that can be used for solving classification problems, including classification trees, logistic regression, discriminant analysis, neural networks, boosted trees, random forests, deep learning methods, nearest neighbors, support vector machines. In section, we will shed the light on some of these classification techniques.

2.5.1 Support Vector Machines Classifier (SVMs)

SVM is a supervised learning technique with related learning algorithms that analyse data that are used for regression and classification analysis [Drucker et al., 1997] [Chapelle et al., 1999]. Recently, SVMs has become one of important technique in the area of classification due to achieving well predicting the unknown samples with a good degree of accuracy as compared to other traditional classifiers [Suralkar et al., 2012]. In addition, SVMs provide efficient and powerful classification algorithms which have the ability to deal with high-dimensional input features (a factor which limits many efficient classifiers) [Salh and Nayef, 2013]. Originally, SVMs was formulated for linear two-class classification with margin, where margin refers to the minimal distance between the separating hyperplane and the closest data points, see Figure 2.2. In other words, SVMs learning machine is defined by an optimal separating hyperplane, where the margin between the positive and negative examples is maximal. The important feature of this technique is that the solution is based on those data points which are located on the margin [Weston, 1998]. These points are called support vectors.

2.5.2 Decision Tree Classifier

A Decision Tree classifier is a simple and commonly based rule system that used a supervised learning algorithm. This kind of classifier builds a classification or regression model in the form of a tree structure. It breaks down a dataset into smaller subsets while an associated decision tree is developed incrementally at the same time. In other words, these classifiers organized a series of test conditions and questions in a tree structure to be used to perform the prediction on the test dataset [Rokach and

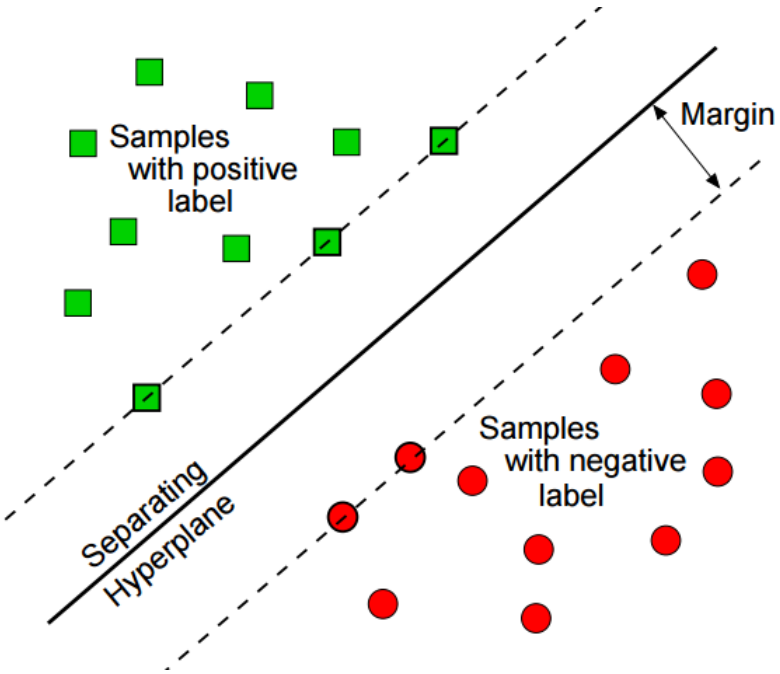

Figure 2.2: The optimal separating hyperplane [Markowetz et al., 2003].

Maimon, 2005].

The basic algorithm for learning decision trees is:

- Starting with whole training data.
- Choose attribute or value along the dimension that gives the best split.
- Create child nodes based on the split.
- Repeat on each child using child data until a stopping condition is reached.

Building an optimal decision tree is a key challenge in decision tree classifier. In general, different decision trees can be constructed from a given set of attributes. While some of these trees are more accurate than others, finding the optimal tree is computationally infeasible due to the exponential size of the search space. However, many efficient algorithms have been introduced to construct a reasonably accurate decision tree in an acceptable amount of time. These algorithms usually use a strategy that grows a decision tree by applying a series of locally optimum decisions about which attribute to use for separating the data [Rokach and Maimon, 2005].

2.5.3 Random Forest Classifier

Random forest algorithm is considered as a supervised classification algorithm. It is an improvement over the decision tree algorithm, in which multiple numbers of small decision trees been generated from random subsets of the data [Gislason et al., 2006]. Each of the decision trees builds a separated classifier where each of these classifiers captures various trends in the data. It then combines the predictions from all the trees to classify a class [Cutler et al., 2007]. The random forest classifier achieves the high accuracy results with the higher the number of decision trees [Rodriguez-Galiano et al., 2012].

The random forests algorithm for regression and classification is summarized as follows [Liaw et al., 2002]:

1. From the original data, draw n_{tree} bootstrap samples.

2. Expand an unpruned regression or classification tree for each of the bootstrap samples with the following modulation: for each node, instead of selecting the best split among all predictors, select a random sample m_{try} of the predictors and choose the best split from those variables.

3. Predict new data by assembling the predictions of the n_{tree} trees. An estimation of the error rate based on the training data can be obtained by the following:

   – For each bootstrap iteration, predict the data which are not in the bootstrap sample, what Breiman [Breiman, 2001] calls out-of-bag or OOB data) by using the tree expanded with the bootstrap sample.

   – Assemble the OOB predictions. Then, calculate the error rate, and call it the OOB estimate of error rate.

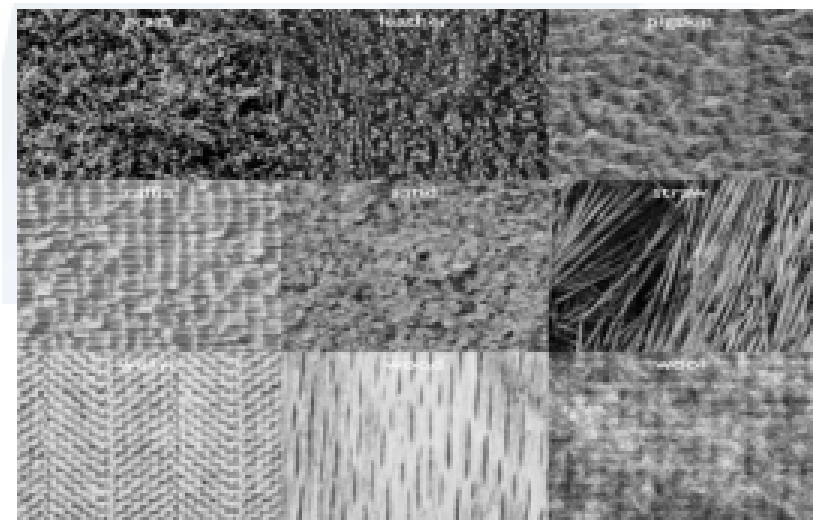

Figure 2.3: Different styles of textures [Kaur and Gupta, 2012]

2.5.4 Classification Based on Texture Features

Texture is the spatial distribution of pixels tones of remotely sensed images. Texture analysis or texture mapping is a popular method for delineating surface features that cause localized variations in the brightness and other spectral properties of the remote sensed image, including shadowing. Primitive or low level features of an image can be general features, such as extraction of shape, color, or texture [Mohanaiah et al., 2013]. Nowadays there has been a great interest in the development of texture based image classification methods in many different areas. Texture is a natural characteristic of all surfaces, which describes the visual patterns, and each has homogenization properties. In other words, texture is one of the most important characteristics employed in identifying objects or ROI (Regions Of Interest) in an image, whether the image be a satellite image, a photomicrograph, or an aerial photograph [Kaur and Gupta, 2012]. Texture is a measurement of variation of surface intensity, quantifying features such as roughness, smoothness, and regularity. Texture describes the local features of the gray level of an image areas [Udupa et al., 2002] [Martínez and Thomas, 2002]. Figure 2.3 shows different styles of texture fields.

Textures are generally categorized into two major categories: structural and statistical [Chianga and Weng, 2013]. Structural textures use basic texture primitives to describe the texture of elements and

placement rules to define the spatial relationship between these elements. While, statistical textures can not be characterized with primitives and deterministic placement rules. Statistical approaches distinguish textures using statistical characteristics of gray levels of pixels of the image [Udupa et al., 2002] [Tsai and Huang, 2003].

Texture classification indicates to the process of gathering test samples of texture into classes, where each producing class contains similar samples depending on some similarity criterion [Zhang and Tan, 2002]. Before classification can take place, some similarity criterion must be defined. These criteria are specified in terms of a set of feature measures, each of them provides a quantitative measurement of a certain texture characteristic. These feature measures are alternatively referred to as texture measures or just features. The sets of feature measures assembled for classification purposes are usually referred to as feature vectors.

Several researchers have worked on finding descriptors and features for texture identification. [Haralick et al., 1973] proposed a technique that use the Gray-level co-occurrence matrices(GLCM) to extract image texture features. Since that time, GLCMs become widely used for texture features extraction in many applications. [Smeulders et al., 2000] proposed texture representation that involves wavelet transform, geometrical method, statistical method, Markovian analysis and methods derived from them. [Zhao and Pietikainen, 2006] introduced a novel simple technique in which dynamic textures are modelled by using Local binary patterns. [Dhongade, 2015] showed that using Gabor features for image retrieval outperforms that using pyramid-structured wavelet transform features, Tree-structured wavelet transform features and multi-resolution simultaneous autoregressive model features. [Liu et al., 2010] used the spatial correlation of Textons to describe the relationships between neighbouring pixels and extraction texture features contrast, entropy, energy and homogeneity. This procedure is effective to detect texture features.

2.6 Classification based on convolutional neural network (CNN)

A convolutional neural network (CNN or ConvNet) is a category of multilayer artificial neural network that achieved highly efficient in image classification and recognition. LeNet by Yann LeCun in 1998 is one of the first CNNs that support the deep learning field [Chen et al.] [LeCun et al., 1998]. Many other architectures have been proposed as improvements over the LeNet, but all based on the concepts of the LeNet. CNN integrates feature extraction with classification [Basbrain et al., 2017], it receives the raw input data and produces the final classification results without any additional process. Furthermore, CNN can handle huge training samples and the features are learned, automatically.

CNN technique has been used in agriculture field to solve various agricultural problems involving classification or prediction, related not only to computer vision and image analysis, but more generally to data analysis. The overall benefits of CNN are encouraging for their further use towards smarter, more sustainable farming and more secure food production. It is remarkable that all papers have been published after 2014, indicating how recent and modern this technique is in the domain of agriculture. The majority of these papers dealt with image classification and identification of areas of interest, including detection of obstacles [Steen et al., 2016] and fruit counting [Rahnemoonfar and Sheppard, 2017], while some other papers focused on predicting future values such as maize yield Kuwata and Shibasaki [2015].

In general, a CNN consists of an input layer, multiple hidden layers, and an output layer. Typically, the hidden layers of a CNN consist of of four main steps: convolution, non linearity (ReLU), pooling or sub sampling, and classification (fully connected layers). These steps are considered as the fundamental building blocks for each Convolutional Neural [Xiao et al., 2016] as shown in figure 2.4:

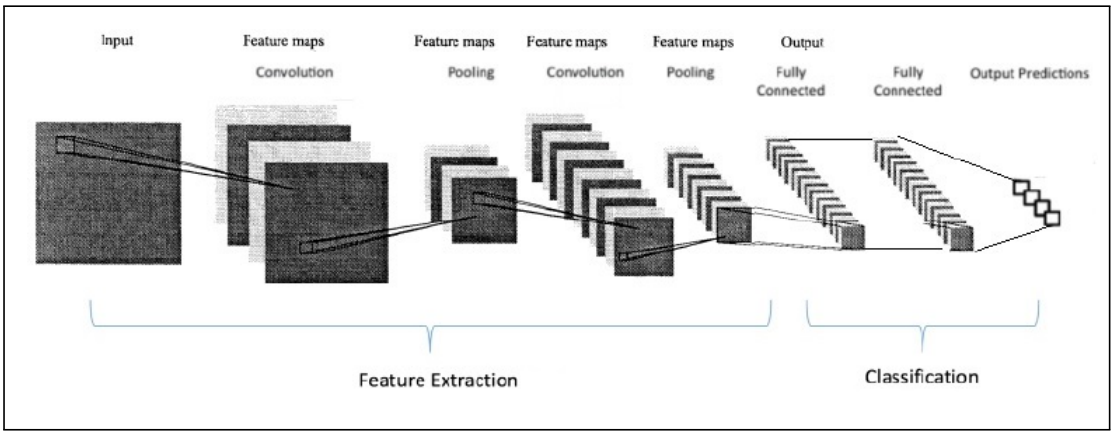

Figure 2.4: Example of CNN architecture

2.6.1 Convolutions

Convolutions are neural networks that use Convolutional layers (Conv layers) which are based on the mathematical operation of convolution. Conv layers consist of a set of filters, which are 2D matrices of numbers. An input image and a filter can be used to produce an output image by convolving the filter with the input image. The main parameter of Conv layer is the number of filters it has. The key purpose of convolution of a CNN is to extract different features from an input image and preserve the spatial relationship between the image pixels.

The convolved feature which is known as 'Feature Map' at each layer represents the output of applying the filters to an input image of that layer. It is computed by sliding a filter over the image and calculating the dot product. Various feature maps can be generated by applying various filters over the same original image. Using more number of filters leads to more image features to be extracted. Consequently, achieving better network performance at recognizing objects in unseen images. The convolution operation is implemented by learning image features using small squares of input image data. Consider a 5×5 image which pixel values are only 0 and 1. Figure 2.5 shows The convolution operation.

There are three key parameters that control the size of the Convolved Feature (Feature Map) and need to be decided before the convolution step is applied:

- Depth: Indicates the number of filters to be used for the convolution operation

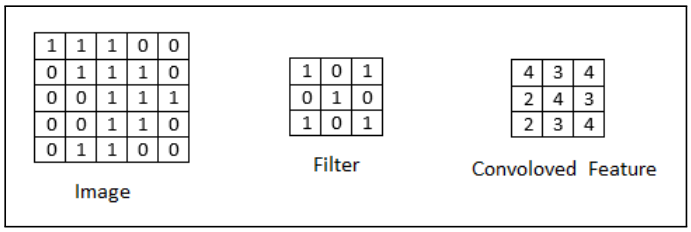

Figure 2.5: The convolution Process

- Stride: Indicates the number of pixels to be jumped by the filter matrix over the input matrix. In other words, if the stride is located to 1 then the filters move one pixel at a time. If the stride value is located to 2, then the filters jump 2 pixels at a time. Therefore, having a big stride gives smaller feature maps.

- Zero-padding: Indicates the procedure of adding zeros round the input matrix border. Using zero padding is useful to control the size of the feature maps [Dumoulin and Visin, 2016] [Ren et al., 2015].

2.6.2 Introducing Non Linearity (ReLU)

The ReLU process is applied after every Convolution step for each pixel to replace all negative pixel values in the feature map by zero [Krizhevsky et al., 2012].

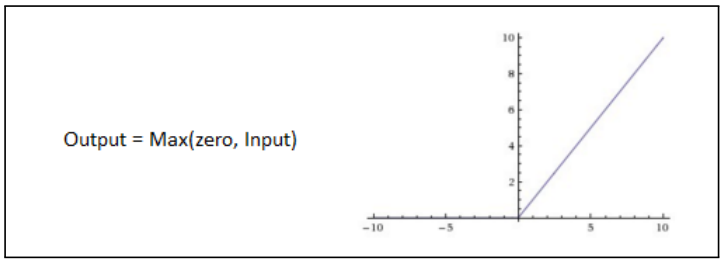

Figure 2.6: The ReLU Process.

2.6.3 The Pooling Process

As it mentioned earlier, applying convolution with different filters to get different feature maps about the image. While, pooling is used to reduce the size of these maps and at the same time preserve the

most significant information of each feature map by using different functions to summarize subregion, for example Average, Max, and Sum function [Harley, 2015] [Dumoulin and Visin, 2016].

Pooling involves selecting a pooling operation, much like a filter to be applied to feature maps. The size of the pooling operation or filter is smaller than the size of the feature map.

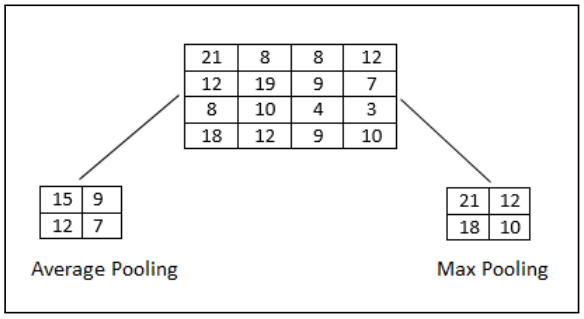

Figure 2.7: The Pooling Process.

2.6.4 Fully Connected Layer

Fully connected layer indicates that each neuron receives input from every element of on the adjacent layer. In a convolutional layer, neurons receive input from only a subarea of the previous layer that called receptive field. While in a fully connected layer the receptive field is the whole previous layer. The aim of the fully connected layer is to use the outcome from the convolutional and pooling layers (high level features) for classifying the input image into various classes depending on the training data-set. As is shown in Figure 2.4, the aim of the convolution and pooling layers is to extract features from the input image while fully connected layer aims to act as a classifier [Krizhevsky et al., 2012].

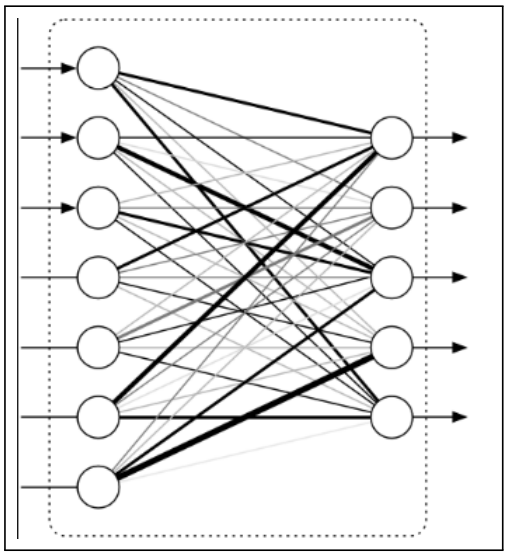

Figure 2.8: The Fully Connected Layer.

2.7 Classification Based on Genetic Programming

Genetic programming (GP) is an extension of a machine learning– Genetic Algorithms (GA). Generally, GP aims to generate computer programs which have the ability to solve a problem without having to identify exactly how to do it [Koza et al., 1999]. General GP hypothesis are derived from GA hypothesis; however, a key difference between them is the individuals structure. GP individuals are computer programs represented as trees with arbitrary size, while fixed length vectors are established for genetic algorithms [Luke, 2013].

Genetic programming creates a solution automatically without needing the user to identify the solution structure earlier. Operators are established randomly into a tree called "individual" or "computer program". An initial population of programs is created, and then a cost function is set to determine how well an individual solves the problem. Individuals of the better-performing will take part in a generation of a new population. Then these individuals are evaluated using the same cost function, and so on. The GP evolution occurs iteratively till it obtains a solution with good fitness, or reaches to a limit on the number of generations [Yimyam, 2015]. The obtained solution is not guaranteed to be the the global optimum but it is definitely the best solution of the GP evolution. Furthermore, many GP execute on

the same training dataset may give different solutions.

Many fields of applications could benefit greatly from GP like biotechnology, data-mining, optimization, signal, processing, image processing, pattern recognition, computer graphics, electrical engineering circuit design, and financial market [Chaudhari et al., 2009]. In the last two decades, GP has been largely employed to classification, optimization, and automatic feature selection related tasks. The common use of GP is mainly due to its flexibility and comprehensible tree structure. Similarly, research is also active in the field of image processing, because of its promised results over wide areas ranging from medical to multispectral imaging. Although genetic algorithms have often been used for classification problems, only recently some attempts have been made to solve such problems using GP. Genetic programming processes are illustrated in Figure 2.9.

2.7.1 Terminologies used in GP

In this section, the various terminologies associated with GP are discussed.

Population The population initialization is The first step of a GP run . A GP population includes computer programs that involving of root, internal nodes, and leaf nodes. Functions are randomly chosen for root and internal nodes while terminals are randomly set to leaves. A maximum size of a tree is defined to control tree size. The root node is presumed to be at depth 0 and the depth of the deepest leaf of a tree is the depth of the tree.

Terminal set Terminal nodes are leaves of a tree, they lies at the end of each branch of a tree. The terminal set may consist of:

- constants: They are a real numbers set. A constant is chosen in a random way from a pre-specified domain of real numbers to put in a computer program.

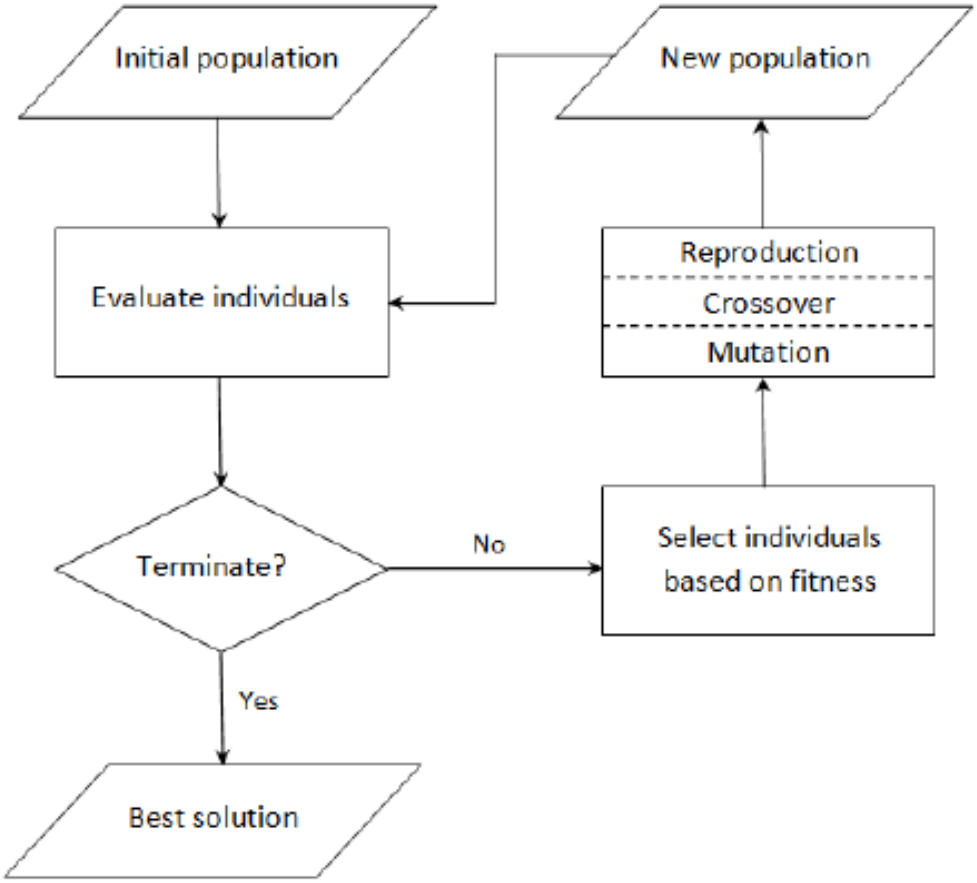

Figure 2.9: Overview of genetic programming processes [Yimyam, 2015]

- variables: They are external inputs recovered from input data features
- zero-argument functions: They are functions that take no argument but return a value.

Function set The function set can comprise of all sorts of functions that used in computer programs, for example mathematical, arithmetic, condition, loop, and boolean functions.

Fitness function A fitness function is used to evaluate how well individuals have learnt to predict outputs from inputs. Highly fit individuals have more opportunity to participate in producing a population in the next generation, whilst individuals with lower fitness are removed from the population.

The fitness function employed depends on the problem. Some fitness functions are based on the calculation of the output accuracy of the computer program. While, some other functions calculate the value of error between the obtained outputs and the desired outputs.

Genetic programming parameters GP parameters play a main role in controlling evolution. There are basic and important parameters used by GA, those include: number of generations, population size, the maximum size of programs, the probabilities of performing the genetic operations and other details of the run.

Termination criteria A run of a GP engine terminates by obtaining a satisfying solution or reaching a maximum number of generations. However, if no program is fit enough, a maximum number of generations is used to terminate the evolution.

Genetic operations A population of programs is iteratively produced for every new generation of the evolution. Three strategies, crossover, mutation and reproduction, are employed to generate individuals from the previous population. Note that these operations proceed based on copies without disrupting

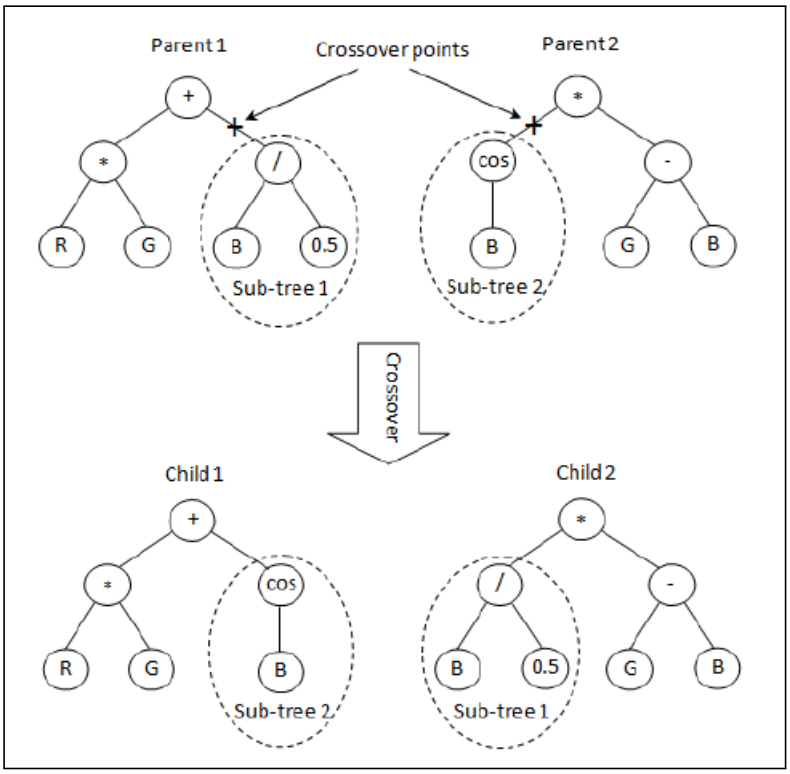

Figure 2.10: Example of sub-tree crossover [Yimyam, 2015].

the original individuals. Thus, they are able to be selected to create offspring multiple times by selection. There are three main operation used in GP:

- Crossover: It is a genetic operator used to combine the genetic information of two parents to generate new offspring with the hope that they create new and efficient offspring. Sub-tree crossover is the most commonly employed form for crossover. Two parents are chosen from the population on the basis of their fitness and a crossover point of them is randomly selected then swapping the parents'sub-trees creates an offspring program. The crossover probability rate is normally declared with a highest probability than the other genetic operators. Tree-based crossover is illustrated in Figure 2.10.

- Mutation: Sub-tree mutation form is generally used the most. One parent is chosen from the population based on fitness, and a mutation point is randomly chosen. After that, a child is generated by replacing a parent's sub-tree with a randomly created sub-tree. Mutation is applied to most GP systems with a much lower probability than crossover. the mutation operation is

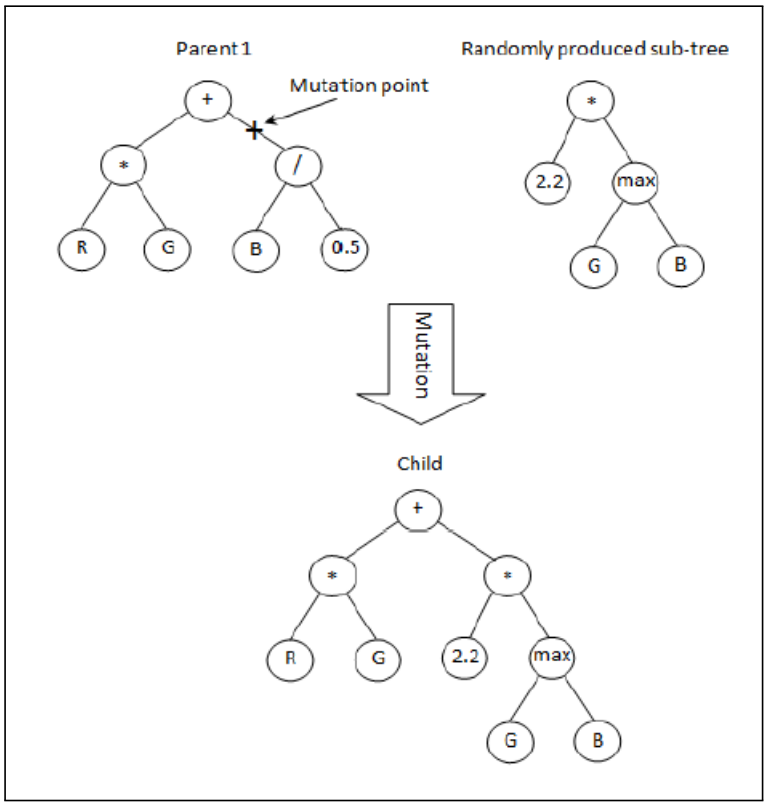

Figure 2.11: Example of sub-tree mutation [Yimyam, 2015].

illustrated in Figure 2.11.

- Selection: To generate computer programs for a new generation, individuals of the previous population are mostly be chosen to produce a new population. The selection process is based on fitness because better individuals are more appropriate for breeding. The tournament selection technique is widely used for choosing individuals [Poli et al., 2008]. The tournament size (t), the quantity of individuals, are randomly chosen from the previous population, then their fitnesses are compared. The individual of the highest fitness is selected and assigned to create a new offspring. However, the best individual is not be selected all the time, and the worst one is not be ignored always from the selection process [Koza et al., 1999].

2.8 Summary

The first part of this chapter starts by reviewing remote sensing image principals in general terms. It goes on to discuss the use of Image classification in remote sensing and the major steps that may be involved in the image classification process. The next part sheds the light on the two main approaches that used for classification in remote scening; supervised and unsupervised learning. The remainder of the chapter examines classification based on conventional statistical techniques, classification based on texture features, classification based on Convloutional Neural Network, and classification based on Genetic Programming.

The Next chapter will present two experimental work that used different conventional remote sensing classification techniques.

Chapter 3

Analysis by Conventional Remote Sensing Approaches

Land cover observation using remote sensing data requires robust classification techniques which give the accurate complex land cover mapping. Remote sensing images classification based on the idea that various earth's surface feature types have a different spectral reflectance, these features are recognized by using the classification process [Al-doski et al., 2013]. In other words, image classification is defined as the operation of classifying all image pixels or remote sensing satellite data to gain a set of labels or land cover themes. Scientists and researchers have made great efforts in improving classification accuracy considerably

This chapter introduces two experimental work that used different conventional remote sensing classification techniques. The former sheds light on the regional agricultural land texture classification using grey level cooccurrence matrices (GLCMs) for features extraction and using SVM and Decision Tree Induction techniques for classification. The later focuses on outcomes gained from three conventional supervised techniques that used for remote sensing images classification; random forest, SVMs, and decision tree classifiers.

Figure 3.1: Davidsons Farm 11.06.2016.

3.1 The Study Area

The study area is Davidsons Farm in Peldon, Colchester (51°48′02.1”N 0°53′57.7”E, 51°47′51.9”N 0°54′29.6”E). During our visiting to this area, the effect of black grass was very clear in most of this area. The images are captured using UAV with two different sensors Visible and NIR (MAPIR Survey2 Cameras). The near infrared filter is used to capture a single band of reflected near infrared light. The red image channel is equal to Near Infrared (850nm) Light, the green and blue image channels are not used as they not contain useful data compared to the red channel . The output images will be a single image band, meaning black and white. A white pixel will be high NIR reflectance, and a black pixel low NIR reflectance. See figures 3.1, 3.2, and 3.3.

Figure 3.2: Davidsons Farm Visible Image.

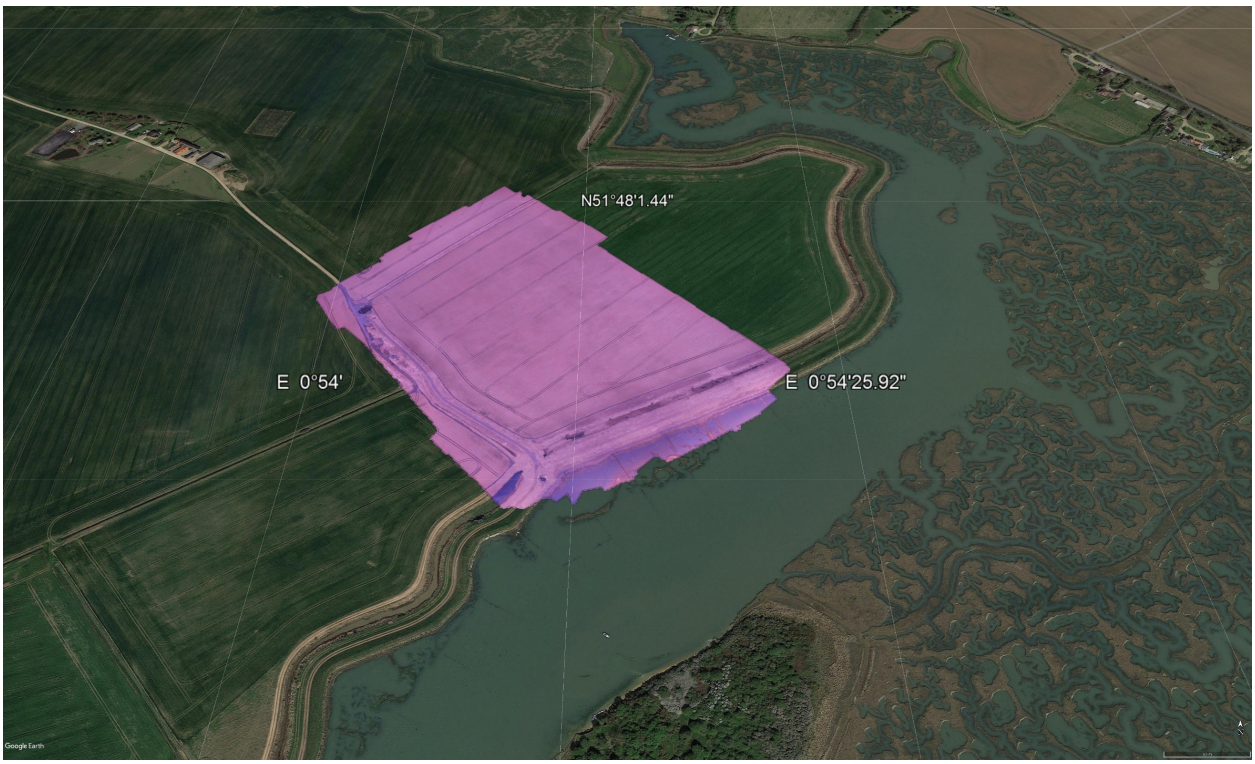

Figure 3.3: Davidsons Farm NIR Image.

3.2 Regional Agricultural Land Texture Classification Based on GLCMs, SVM and Decision Tree Induction Techniques

Nowadays there has been a great interest in the development of texture based image classification methods in many different areas. Texture is a natural characteristic of all surfaces, which describes the visual patterns, and each has homogenization properties. In this section, our concern is with the regional agricultural land texture classification using grey level cooccurrence matrices (GLCMs). The GLCM is a matrix of how often different combinations of pixel brightness values (grey levels) occur in an image. The texture discrimination is performed to partition a textured image into areas, each related to a homogeneous texture where samples of four different textures are extracted from the image. For each patch, the four features of the GLCM matrices namely, dissimilarity, correlation, angular second moment, and homogeneity are computed. Finally, for texture classification we have used two well-known methods: Support Vector Machine and decision tree induction to distinguish different types of ground cover– and black grass from wheat in particular. The results show that these texture features have high discrimination accuracy and classification using support vector machines gives better results as compared to the decision tree induction classifier.

3.2.1 Gray Level Co-ocurrence Matrix (GLCM)

The Gray level co-occurrence matrix (GLCM) is also referred to as a cooccurrence distribution. It is one of the most known Statistical texture analysis methods that has been used in a number of applications. The gray level co-ocurrence matrix (GLCM) method is a way of extracting second order statistical texture features [Udupa et al., 2002] [Martínez and Thomas, 2002]. It is defined over an image to be the distribution of co-occurring values at a given offset. GLCM takes into consideration the relation between two pixels at a time, called the reference and the neighbor pixel [Mohanaiah et al., 2013]. In other words, GLCM is commonly used to characterize the texture of an image by determining how often

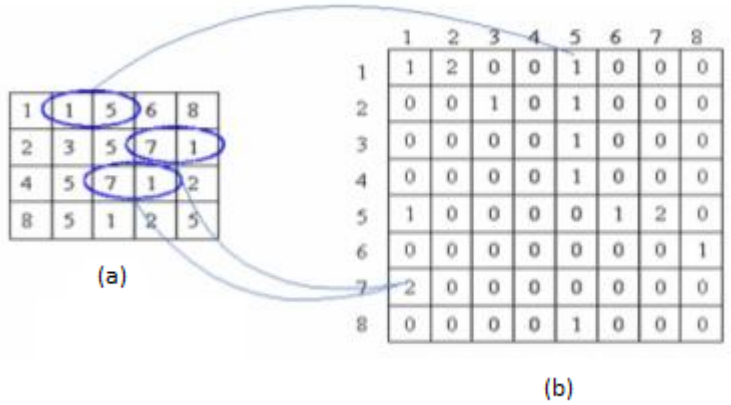

Figure 3.4: GLCM Calculation

pairs of pixel with particular values and in a specified spatial relationship exist in an image [Zulpe and Pawar, 2012]. A GLCM is a matrix in which the number of rows and columns is similar to the number of gray levels G in the image. The matrix element $P(i, j|\Delta x, \Delta y)$ is the relative frequency of two pixels separated by a distance $(\Delta x, \Delta y)$ occur within a given neighborhood, one with intensity i and the other with intensity j [Benčo and Hudec, 2007]. Figure 3.4 illustrates the GLCM calculation. In this figure, (a) represents the input image and (b) represents the GLCM matrix. The number of gray levels in the image determines the size of the GLCM.

In the output GLCM, element (1,5) contains the value 1 because there is only one instance in the input image where two horizontally adjacent pixels have the values 1 and 5, respectively. glcm(7,1) contains the value 2 because there are two instances where two horizontally adjacent pixels have the values 7 and 1. Element (1,6) in the GLCM has the value 0 because there are no instances of two horizontally adjacent pixels with the values 1 and 6.

There are different statistical measures from the GLCM that can be used to estimate the similarity between different gray level co-occurrence matrices. In our experiment, four statistical measures have been employed to provide information about the texture of the input image; ASM, Correlation, Dissimilarity, and IDM.

Angular Second Moment (ASM) Angular Second Moment is also called Energy or Uniformity. It is the summation of squares of entries in the GLCM Angular Second Moment that measures the image homogeneity. Angular Second Moment is high when image has very good homogeneity or when pixels have high similarity.

$$ASM = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i,j)^2 \tag{3.1}$$

Where G is the number of gray levels used and i, j are the spatial coordinates of the function $P(i, j)$.

Correlation Correlation is a measure of gray level linear dependence between the pixels at the specified positions relative to each other.

$$Correlation = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{\{i \times j\} \times P(i,j) - \{\mu_x \times \mu_y\}}{\sigma_x \times \sigma_y} \tag{3.2}$$

Where G is the number of gray levels used. μ is the mean value of P. μ_x, μ_y, σ_x and σ_y are the means and standard deviations of P_x and P_y.

Dissimilarity Dissimilarity is the measurement of the local intensity variation defined as the mean absolute difference between the neighbouring pairs.

$$Dissimilarity = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P_{i,j} |i - j| \tag{3.3}$$

Inverse Difference Moment (IDM) Inverse Difference Moment (IDM) measures the local homogeneity, the tightness of the components in GLCM. IDM weight value is the opposite of the contrast weight. The weight used in contrast is $(i-j)^2$ and $1/1+(i-j)^2$ in homogeneity [Chaki and Dey, 2020]. The

result is a low IDM value for inhomogeneous images, and a relatively higher value for homogeneous images.

$$IDM = sum_{i,j=0}^{G-1} \frac{P_{i,j}}{1+(i-j)^2} \tag{3.4}$$

3.2.2 The Experimental Work

Four different classes of ROI (Region Of Interest) images are used for the experimental work with total 80 samples: wheat, black grass, road, and bushes. Each class contains 20 samples. Every image is having the exact size of 16 × 16 in axial view. Figures 3.5 on page 43 and figure 3.6 on page 43 show the original image and the classes used in the experimental work, respectively.

The Gray Level Co-ocurrence Matrix (GLCM) method is used to extract four statistical texture parameters of remote sensing images: Dissimilarity, Inverse Difference Moment (homogeneity), Angular Second Moment and Correlation, see section 3.2.1 for more details. These statistics provide information about the texture of the input image. SVM and Decision Tree Induction techniques have been used to classify these four different classes of ROI (Region Of Interest). Figures 3.7 and 3.8 represent classification using GLCM based on SVM and decision tree learning techniques, respectively. The results show that GLCM have good performance in texture feature extraction and these texture features have high discrimination accuracy. To calculate the accuracy of classification, we applied the following equation which is defined as [Kanwal et al., 2016]:

$$\frac{TP+TN}{N} \tag{3.5}$$

where TP is the number of true positive detections, TN is the number true negative detections, and N represents the number of images tested. Classification using support vector machines gives better results as compared to the decision tree induction classifier in terms of the number of mis-classifications instances and the classification accuracy with an overall accuracy of 83%, while the decision tree accuracy

is 70%, respectively.

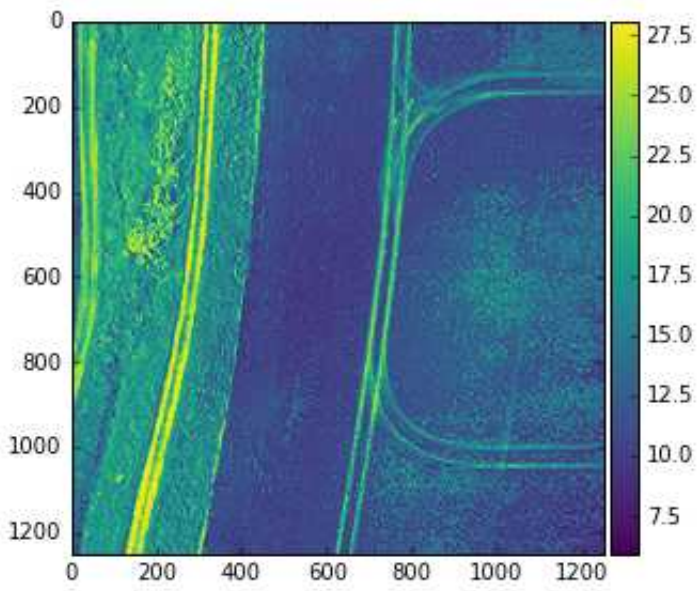

Figure 3.5: The original image.

Figure 3.6: Samples of four different textures are extracted from the image

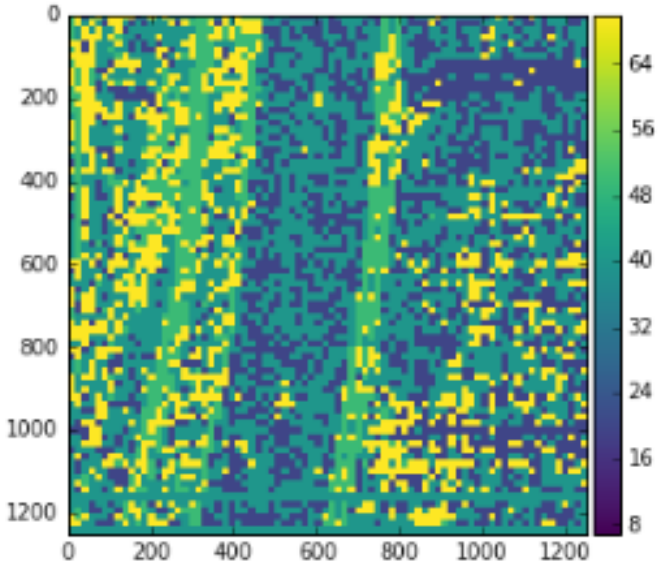

Figure 3.7: Classification Using GLCM based on SVM learning technique.

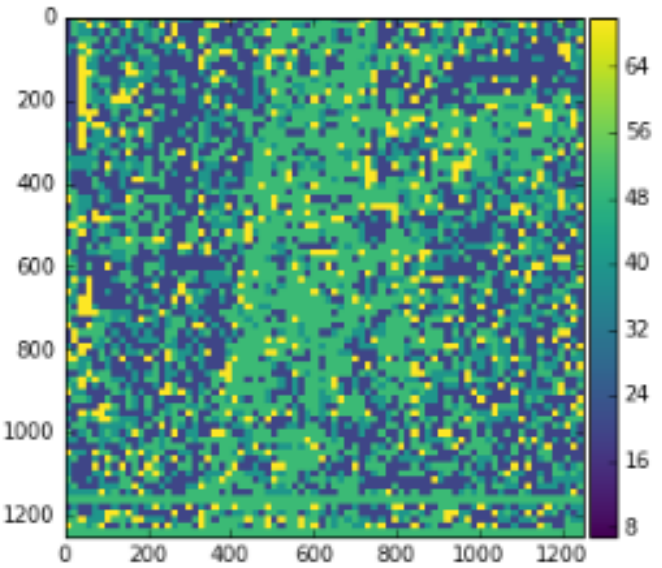

Figure 3.8: Classification Using GLCM based on Tree classification learning technique.

3.3 Regional Agricultural Land Classification Based on Random Forest (RF), Decision Tree, and SVMs Techniques

The objective of this section is to present results obtained with the Random Forest (RF) classifier and Decision Tree Classifier to compare their performance with the Support Vector Machines (SVMs) tech-

nique. The mentioned techniques are applied over the imagery we have captured with different classes of ROI (Region Of Interest).

The Davidsons Farm, Peldon, Colchester is the study area chosen for this project. It is located in the south of Colchester near Mersea Island. RGB and NAR images are taken by a drone to characterize variation in the state of vegetal cover. Six different classes of Region Of Interest (including Unknown range) images are used for the experimental work with total 1600 samples: wheat, black grass, road, bushes, and stones. A mosaic image including 1401 samples has been tested. Figure 3.9 on page 46 represents the outcomes of remote sensing images classification based on RF. Figure 3.10 on page 46 represents the outcomes of classification based on Decision Tree and figure 3.11 on page 47 represents the outcomes of classification based on SVMs classifier.

For our first evaluation of the three model's performance, we used an accuracy score. This score measures how many labels the model got right out of the total number of predictions. Results showed that the random forest classifier performance outperforms the Decision Tree and SVMs techniques performance in terms of classification accuracy and training time with an overall accuracy of 86% while the Decision Tree accuracy is 67% and the SVMs accuracy is 56%.

For the second evaluation , we used a confusion matrix to express how many of a classifier's predictions were correct, and when incorrect. In the confusion matrices, the rows represent the true labels and the columns represent predicted labels. Values on the diagonal represent the number of times where the predicted label matches the true label. Values in the other cells (off-diagonal cells) represent instances where the classifier mislabeled an observation; the column tells us what the classifier predicted, and the row tells us what the right label was. We can see that the random forest classifier performance outperforms the other models in terms of the number of times where the predicted label matches the true label

```
We have 1401 samples
The training data include 6 classes: [1 2 3 4 5 6]
predict    1    2    3    4    5   6   All
truth
1        262   18    4    9   17   3   313
2          7  228    6   20   35   0   296
3          0    3  286    3    5   1   298
4          1    7    3  186   13   0   210
5          8   19    1   11  211   3   253
6          1    6    0    3   10  11    31
All      279  281  300  232  291  18  1401

Accuracy     0.85
```

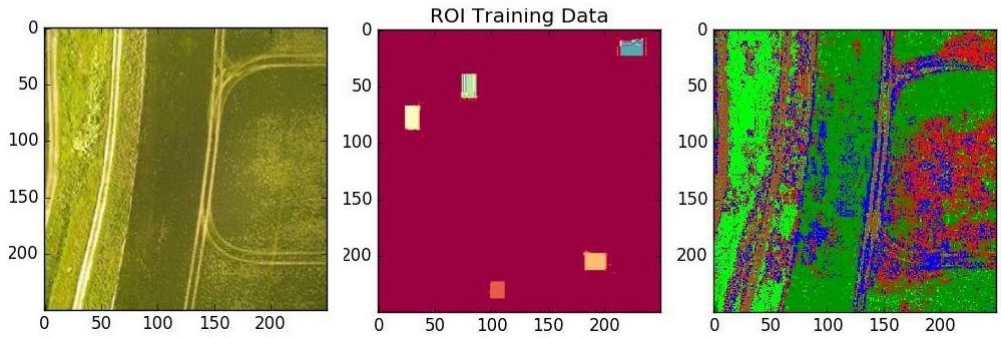

Figure 3.9: Remote sensing images classification based on the RF classifier

```
We have 1401 samples
The training data include 6 classes: [1 2 3 4 5 6]
predict    1    2    3    4    5   6   All
truth
1        223   29   17   18   26   0   313
2         20  199   22   18   37   0   296
3          5   11  259   18    5   0   298
4          7   50   20  112   21   0   210
5         21   57    6   19  150   0   253
6          5   18    0    3    5   0    31
All      281  364  324  188  244   0  1401

Accuracy    0.67
```

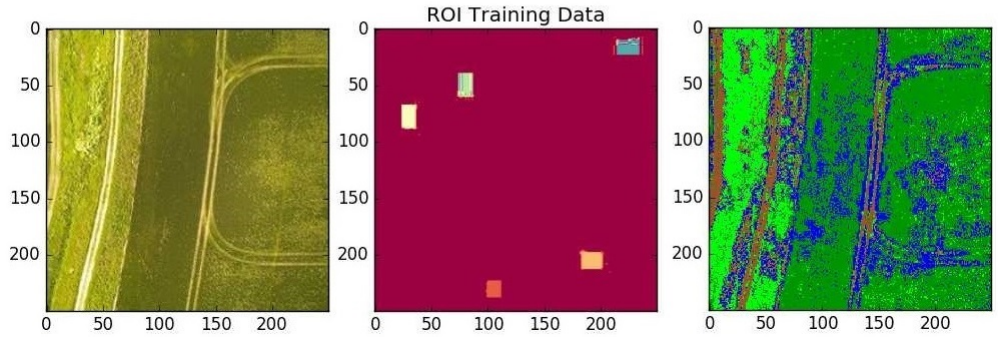

Figure 3.10: Remote sensing images classification based on the Decision Tree classifier

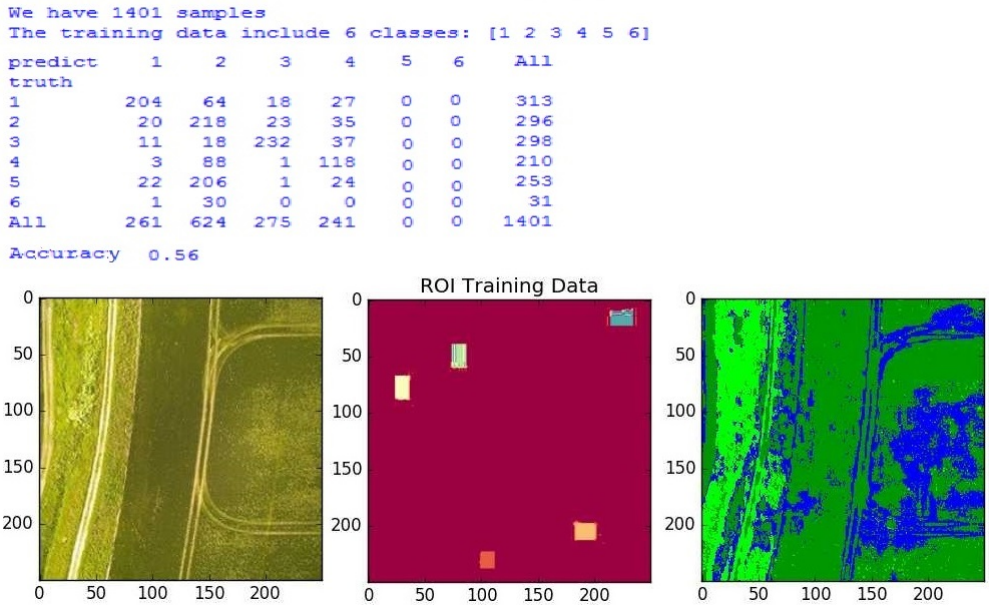

Figure 3.11: Remote sensing images classification based on the SVMs classifier

3.4 Conclusions

Texture is a natural characteristic of all surfaces, which describes the visual patterns and each has a homogenization properties. In this chapter, our concern is with regional agricultural land texture classification using grey level co-occurrence matrices (GLCMs).

Texture discrimination is performed to partition a textured image into regions, each corresponding to a perceptually homogeneous texture where samples of four different textures are extracted from the image. For each patch, the four features of the GLCM matrices namely, dissimilarity, correlation, angular second moment, and homogeneity are computed. Finally, for texture classification we have used two well known methods: support vector machine and decision tree induction to distinguish different types of ground cover– and black grass from wheat in particular. The results show that these texture features have high discrimination accuracy. Classification using support vector machines outperforms the decision tree technique performance in terms of the number of mis-classifications instances and the classification accuracy with an overall accuracy of 83%, while the decision tree accuracy is 70%, respectively.

Different supervised classification algorithms: Random Forest (RF), SVMs, and decision tree classifiers have been employed in the remote sensing field. The experimental work presented results obtained with the Random Forest (RF) and the Decision Tree classifier and to compare their performance with the Support Vector Machines (SVMs) technique. The mentioned techniques are applied over the imagery we have captured with six different classes of ROI (Region Of Interest) images including Unknown range. Results indicated that the random forest classifier performance outperforms the decision tree and SVMs techniques performance in terms of the number of mis-classifications instances and the classification accuracy with an overall accuracy of 86% while the Decision Tree accuracy is 67% and the SVMs accuracy is 56%, respectively.

The first part of the next chapter will illustrate the study area that been used. The second part will shed light on the fundamental aspects of the most common indices that analysts use in remote sensing; NDVI, CRI2, SG index, and RG Ratio index. In addition, the experiment results of using these indices will be introduced.

Chapter 4

Analysis by Vegetation Indices (VIs)

4.1 Introduction

In the remote sensing applications area, experts have developed VIs (vegetation indices) to evaluate vegetative covers qualitatively and quantitatively by using spectral measurements. Vegetation indices allow to describe the distribution of vegetation based on the characteristic reflectance patterns of green vegetation. VI quantifies vegetation biomass for each individual pixel in a remote sensing image and is calculated from the spectral bands that are sensitive to biomass and vigor [Dutta, 2016] [Payero et al., 2004]. A wide range of vegetation indices have been published in scientific literature to extract information of surfaces, but only a small subset have been systematically tested.

UAV (Unmanned aerial vehicles) offer the opportunity for precision agriculture to efficiently monitor agricultural land. A VI derived from an aerially observed multispectral image can measure the crop health, moisture and nutrient content. The vegetation indices have the potential to differentiate healthy from diseased plants. The challenge in this research is to use the appropriate vegetation index based upon their robustness, scientific basis, and applicability to detect the black grass and recognise its separate pattern early to start terminate it and avoiding the mass losing of crop. The first part of this chapter il-

Figure 4.1: The Study Area

lustrates the study area that been used in this research. The second part sheds light on the fundamental aspects of the most common indices that analysts use in remote sensing; NDVI, CRI2, SG index, and RG Ratio index. In addition, the experiment results of using these indices to detect black grass regions have been introduced.

4.2 The Study Area

Our study area is Cherry Tree Farm in Harleston, Suffolk (52°18′2.27”N 1°24′10.73”E , 52°17′20.27”N 1°25′12.59”E), figure 4.1 represents the study area. During our visiting to this area, the effect of black grass was very clear in most of this area. Figure 4.2 shows the Black-grass affected areas in Cherry tree farm field.

The challenge is to detect the black grass and recognise its separate pattern early to start terminate it and avoiding the mass losing of crop. The idea is simply to starting take images for the affecting area using UAV with different sort of sensors (visible and NIR). As a first step a visible and NIR images was

Figure 4.2: Cherry Tree Farm Images 24.05.2016.

Figure 4.3: Cherry Tree Farm Visible Image.

taken and ground truth data was collected with range of two weeks provided as with the location of some black grass affected area to compare it with our result images. As it mentioned, two types of images have been used in this research:

Visible image Mosaic image taken using UAV and generate by Pix4Dmapper software, the camera used was Panasonic Lumix DMC-LX7 Resolution: 0.035 m. See figure 4.3.

NIR Image Mosaic image taken using UAV and generate by Pix4Dmapper software, the camera used was Panasonic Lumix DMC-LX7 + 720 nm Filter Resolution: 0.035 m X 3 Bands. See figure 4.4.

Table 4.1 represents the Black-Grass GCP (Ground Control Point). The GCPs are the locations in the field where the black grass is located. The * symbol indicates there is an abundance plants in that area. The others points are where there is only a few plants.

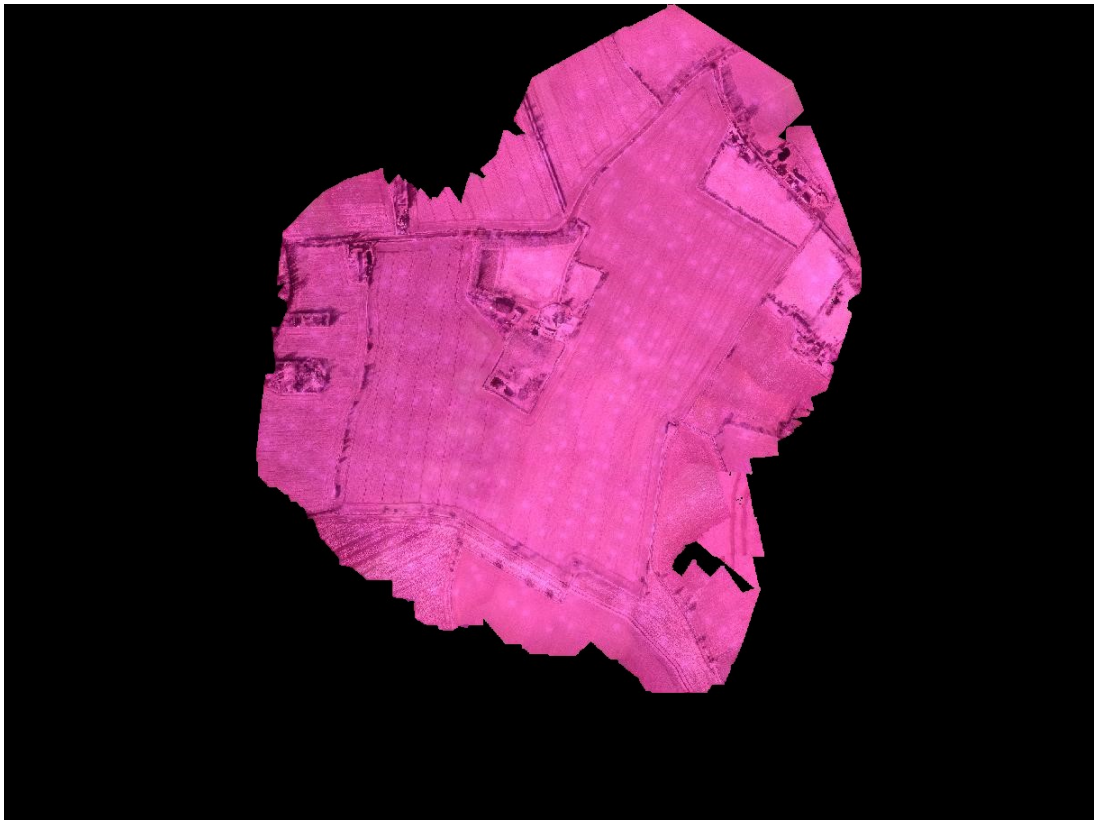

Figure 4.4: Cherry Tree Farm NIR Image.

Table 4.1: Black-Grass GCP (Ground Control Point)

	North	east	
1	5.2E+07	1244643	
2	5.2E+07	1244733	
3	5.2E+07	1244741	
4	5.2E+07	1244664	*
5	5.2E+07	1244583	
6	5.2E+07	1244405	*
7	5.2E+07	1244359	*
8	5.2E+07	1243010	*
9	5.2E+07	1242989	*
10	5.2E+07	1242966	*
11	5.2E+07	1242941	*
12	5.2E+07	1242933	
13	5.2E+07	1242928	
14	5.2E+07	1242900	
15	5.2E+07	1242575	*
16	5.2E+07	1242800	*
17	5.2E+07	1242781	
18	5.2E+07	1242743	*
19	5.2E+07	1242386	
20	5.2E+07	1242380	

4.3 Vegetation Indices in ENVI

Over time and through many scientific studies, remote sensing experts have come to understand how combinations of the measured reflectance properties at two or more wavelengths reveal specific vegetation characteristics, also known as VIs. There are more than 150 existing VIs, with additional indices emerging as sensors advance and provide new information. ENVI exposes 27 of these indices which were selected based upon their robustness, scientific basis, and applicability.

ENVI provides VIs that can be used to detect the presence and relative abundance of various vegetation properties. The VIs provided in ENVI are not designed to quantify the exact concentration or abundance of any given vegetation component. Instead, they are intended for use in geographically mapping relative amounts of vegetation components, which can then be interpreted in terms of ecosystem conditions. In this research, we used four of the most common indices that analysts use in remote sensing; NDVI, CRI2, SG index, and RG Ratio index. To compare the result with the GCP and to find the best match, we used the raster slicer, where it divides the image into layers according to its point values and places it within ranges that are easy to isolate. Thus, each layer can be compared separately with the GCP and find the best range of values that match the areas of black grass. This technique was used with all the indices as they facilitate the comparison process and find the relative accuracy.

4.4 NDVI (Normalized Difference Vegetation Index)

The NDVI is perhaps the most well known and often used vegetation index. Generally, NDVI is used to measure vegetation change over time. It is a way to measure healthy vegetation. In other words, when the NDVI values are high that indicates there is healthier vegetation. When the NDVI values are low that means there is less or no vegetation. NDVI quantifies vegetation by detecting the difference between near-infrared which vegetation strongly reflects and red light which vegetation absorbs [Nouri et al., 2014].

Green vegetation reflects much light in the near infrared (NIR) band yet very little in the red band (RED). Along these lines, the most straightforward vegetation index is ($NIR - RED$) because that distinction has a positive and great value in most cases, so that would be acquired for the other land covers. Since the NIR space is imperceptible to the human eye, the higher reflectance in the green band makes us see vegetation as green. Its clearly, The low reflectance and transmittance is created by the solid assimilation of chlorophyll pigment. The effect of absorption is less powerful in the near infrared ($700-1100nm$) yet reflectance and transmittance are higher. The sort of pigment, leaf water substance, cell-structure, thickness of the leaves and their introduction in the sun-sensor path impact on the spectral signature details. The condition of improvement of plants and trees through the seasons and inside a season as an element of rainfall and temperature is, obviously, a source of variation in the growth of vegetation and subsequently of the reflectance attributes. To complicate matters, the soil foundation impacts reflectance when canopies don't cover the area. The distinction in reflection (communicated by DNs (digital numbers)) of the NIR and RED groups is much more noteworthy for green vegetation than the distinction for exposed soils, outcrops water, dried out vegetation, etc. A picture containing pixel estimations of the distinction is termed a vegetation record picture. The 'Normalized Difference Vegetation Index ($NDVI$)' is broadly utilized, where normalization is given by dividing the difference by the sum, so as to constrain the scope of $NDVI$ qualities to -1 and 1.

$$NDVI = \frac{(NIR - Red)}{(NIR + Red)} \tag{4.1}$$

Pixels with rich vegetation in a NDVI picture will show up as whitish tones (if such tones are allotted to high values, as is standard) and pixels with no or poor vegetation in dull dim to dark tones. Regularly colour coding is utilized to part up the DNs (Digital Numbers) of a solitary $NDVI$ (or other) picture by slicing. Generally, dim to light blue colours are allocated to low $NDVI$ qualities, green colours to the centre qualities, and red shades to the high values. Such images look alluring however, have the disservice that gradual transition and patterns may not be clear.

To investigate which NIR bands have the best result, we applicate the NDVI index for all of them, one by one. So we had three $NDVI's$ result that we compare between them using the GCP (Ground Control Point) (with range of 3 pixels). Figure 4.5 in page 57 illustrates the NDVI result images with color slice.

Using $ENVI$ 5.3 software we start to prepare the data to process by combining the two images the visible (3 bands) and the NIR (3 bands, after seperating the image bands into NIR_R, NIR_G, and NIR_B) to produce multiband image (6 bands), using the ENVI Build Band Stack Task.

1. Visible + NIR_R best range $(0.0 to 0.125)$ 40% match the GCP.

2. Visible + NIR_G best range $(-0.25 to -0.125)$ 55% match the GCP.

3. Visible + NIR_B best range $(0.0 to 0.125)$ 95% match the GCP.

We Used the Raster Color Slices tool to select data ranges and colors to highlight areas of an image. The results showing that the best range (best matching) with the GCP in NIR_B (from NIR image) in this slice (3) which is the blue light range.

4.5 Carotenoid Reflectance Index 2

The Carotenoid Reflectance Index 2 ($CRI2$) is a measure of reflection that is sensitive to the carotenoid pigments in the plant leaf and supplies better results in the high concentration carotenoid areas. Higher $CRI2$ values refer to higher of carotenoid concentration (Gitelson, 2002). $CRI2$ equation is:

$$CRI2 = (1/p510) - (1/p700) \tag{4.2}$$

Where P510 and p700 represent the pigment reflectance at 510 and 700 nm, respectively. After comparing the GCP with the results of CRI2 (Carotenoid Reflectance Index 2), we located the best matching

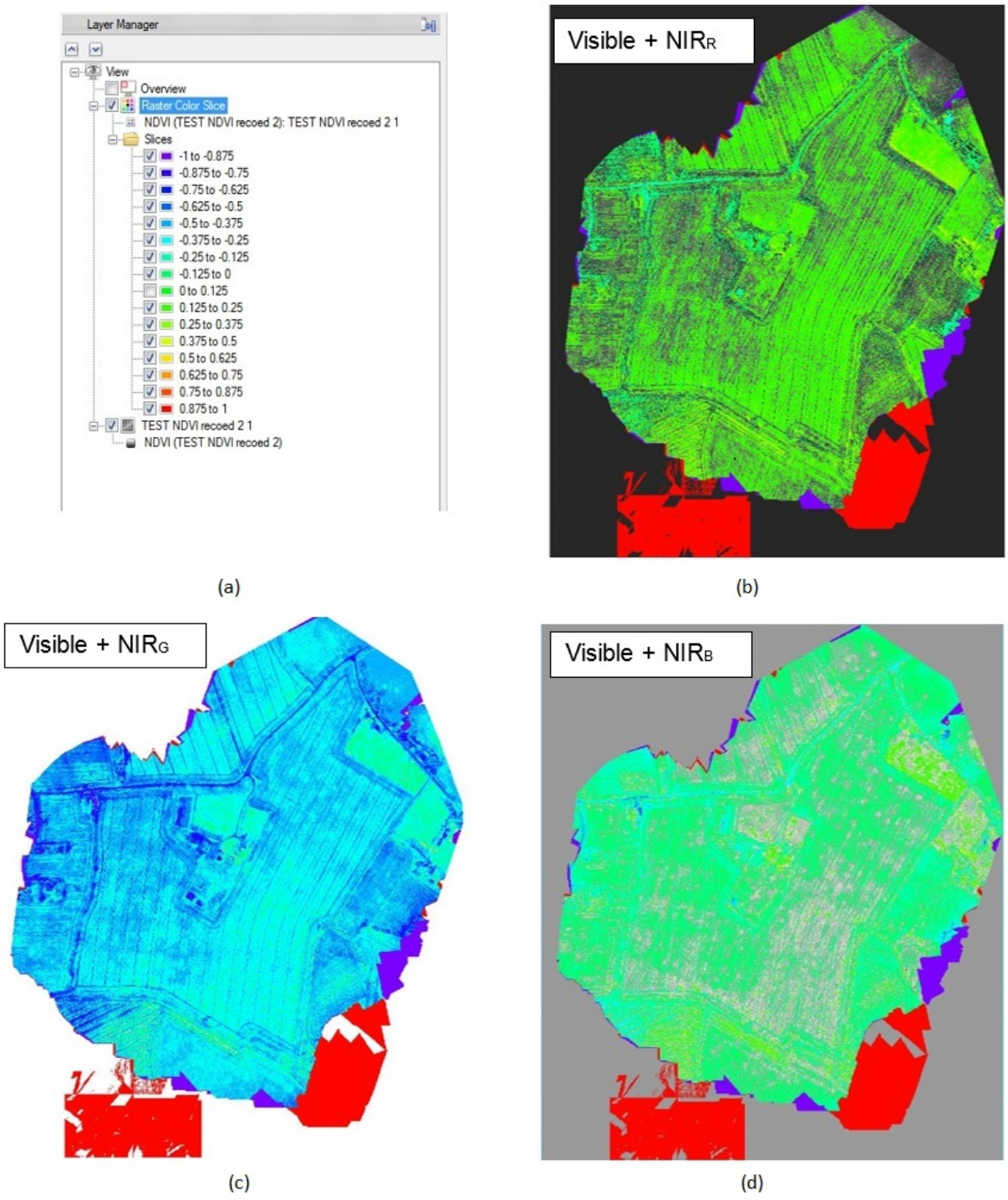

Figure 4.5: NDVI result images with color slice

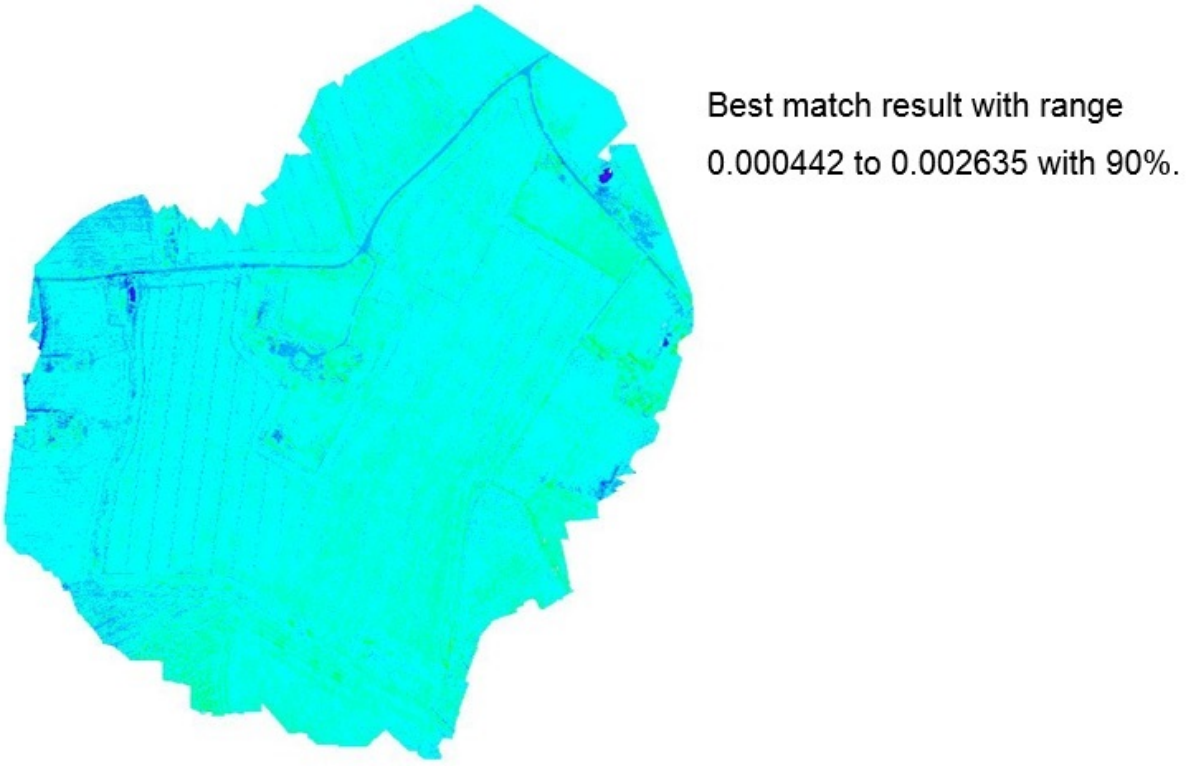

Figure 4.6: Carotenoid Reflectance Index 2

in range $0.000442 - 0.002635$ slice with 90%. Figure 4.6 in page 58 shows the Carotenoid Reflectance Index2.

4.6 Sum Green Index

(SG) Sum Green index, simplest plants indicators used change detection in vegetation greenness. The SG is the mean of reflectance across the 500 nm to 600 nm portion of the spectrum. This sum is then normalized by the number of bands to convert it back to units of reflectance. Because the light is absorbed strongly by green plants in this spectral region, SG index is quite sensitive to small changes in the vegetation canopy across the $500nm$ to $600nm$ range of the spectrum. The ranges value of this index is from 0 to more than 50 (Lobell, 2003). After comparing the GCP with the results of SG (Sum Green index), we located the best matching in range $143.4375 - 159.375$ slice with 90% Figure 4.7 in page 59 shows the sum green ratio index.

Figure 4.7: Sum Green Ratio Index

4.7 Red Green Ratio Index

The Red Green Ratio Index (*RGRI*) is a reflectance measurement that indicates the proportional expression of anthocyanin to chlorophyll which caused leaf redness. *RGRI* equation is:

$$RGRI = mean(RED)/mean(GREEN) \tag{4.3}$$

Where mean(RED) represents all bands with wavelengths in the red range of the electromagnetic spectrum and mean(GREEN) represents all bands with wavelengths in the green range of the electromagnetic spectrum. The RGRI is a guide of leaf production and stress. The common value range for green plants is 0.7 to 3 (Gamon, 1999). After comparing the GCP with the results of RGRI (Red Green Ratio), we located the best matching in range $1.034341 - 1.129808$ slice with 90%. Figure 4.8 in page 60

shows the red green ratio index

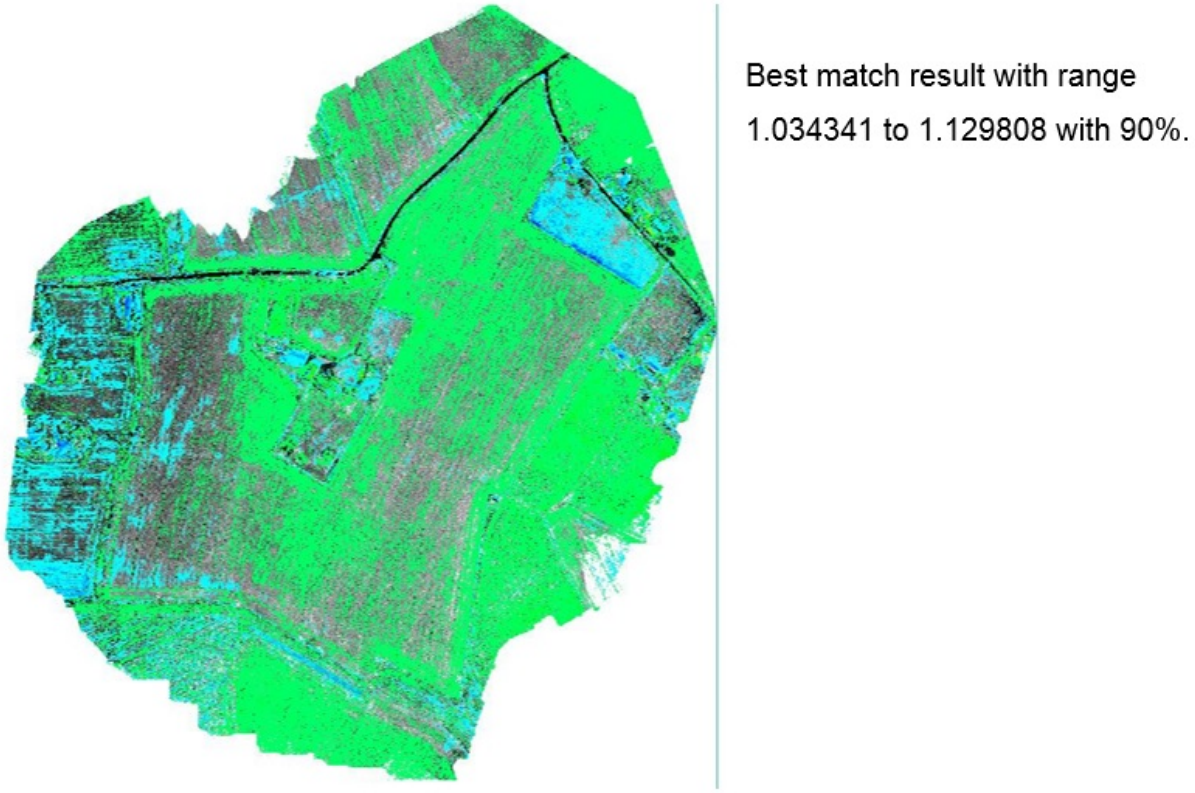

Figure 4.8: Red Green Ratio Index

4.8 Summery

The Earth's surface information is required in various applications. Land cover classification using remote sensing images is one of the common applications in remote sensing. A vegetation index measures vegetation biomass and vigor for each pixel in a remote sensing image. VIs are ratios or linear combinations of reflectance measurements in various spectral bands existing within the visible and near infrared bands.

NDVI, CRI2, SG index, and RG Ratio index are widely used to measure vegetation change over time. By manipulating images and picking the right slice and band we can achieve high precision up to 90% using NDVI, CRI2, SG index, and RG Ratio index. Our aim is to find a system that detect black grass automatically, deal with large data, and can detect the black grass in all growing stage. However, the

mentioned VIs still a manual process, consumes a lot of time and effort, needs an expert to deal with, and it is not effective with the data of large size. In addition, It has not been used with black grass in early stage

Next chapter will present two genetic programming toolkits developed at Essex University: the Jasmine vision system builder and ELVS (evolutionary learning vision system) to detect the tree crown regions and consequently measure its efficiency to solve the black grass detection problem.

Chapter 5

Genetic Programming

5.1 Introduction

Genetic Programming (GP) is an evolutionary algorithm that has the ability to learn relationships that hidden in data and express them in a mathematical manner automatically [Espejo et al., 2009]. In the last two decades, GP has been largely employed to classification, optimization, and automatic feature selection related tasks. The common use of GP is mainly due to its flexibility and comprehensible tree structure. Similarly, research is also active in the field of image processing, because of its promised results over wide areas ranging from medical to multispectral imaging. Although genetic algorithms have often been used for classification problems, There is almost no research that used genetic programming for this purpose. Two genetic programming toolkits developed at Essex University have been tested: the Jasmine vision system builder and ELVS (evolutionary learning vision system) to detect the tree crown regions and consequently measure its efficiency to solve the black grass detection problem.

Jasmine [Essex, 2017] is a system that uses GP to evolve programs that solve the individual stages of a complete system from a set of generic vision components using training data. As will be illustrated in this chapter, Jasmine described in [Essex, 2017] is not able to solve the black grass detection problem

well, mainly because it lacks operators able to classify on the basis of colour and texture.

The first part of this chapter describes the principles underlying the Jasmine and identifies its major shortcomings. In the second part, the tree crown detection problem has been explored using Jasmine and ELVS.

5.2 The Jasmine, Vision System Learning

Jasmine (Java, A Segmented IMage Notation Environment) employs GP to build vision systems of the components. Those improved vision systems usually have 3 steps: segmentation of the exciting features from the background, extraction of these features, and classification. Jasmine provides a graphical user interface that allows the user to capture training data for these steps, and to develop the steps independently. Jasmine creates as many classes as are appropriate for the vision problem, and then paint them onto images; the places that have painted are used as the training data, first for segmentation and subsequently for classification. After having a trained system, Jasmine allows the application of individual test images. Figure 5.1 illustrates the separation of training and testing and the stages of the evolved vision systems. Each stage is defined by a character and number (e.g., A7 is the object classifier generated by GP), which are mentioned in the descriptions later in this chapter. These stages mainly involve interaction with the user.

A1, a training dataset is allocated in the system. The program is able to support photos and videos. Only images are used as an input set in the research.

A2, each image should have a background and/or objects of interest that have been identified (This process must done manually). If objects and background of images are consistent, and there is a big number of input images, the manual process is not required to be applied for all photos in order to decrease the time cost.

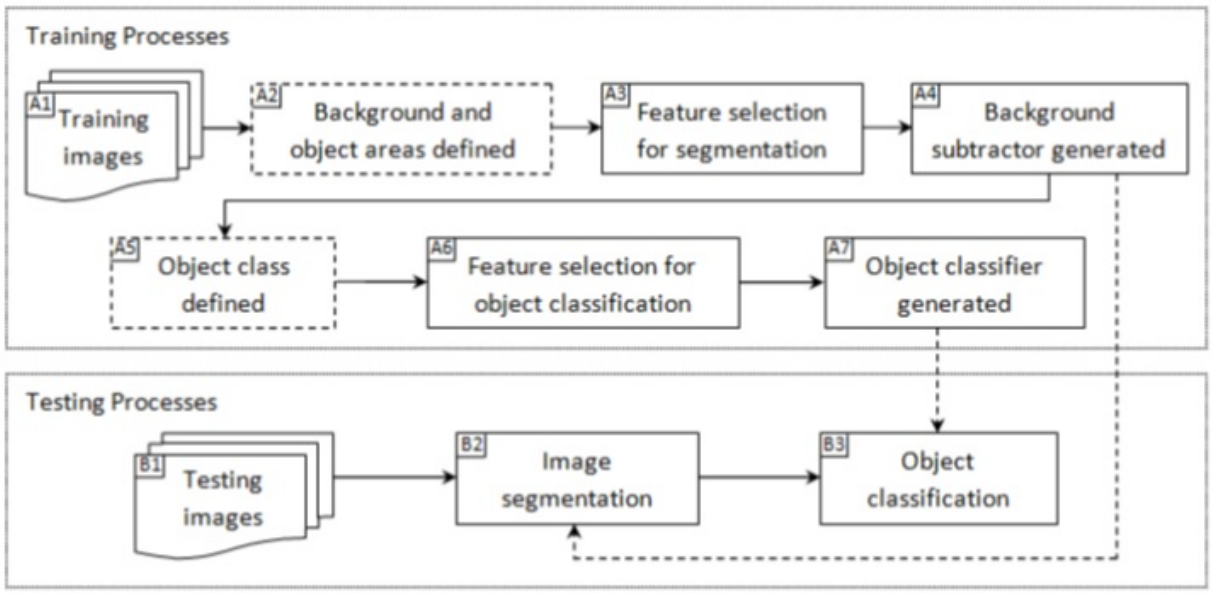

Figure 5.1: Jasmine framework

A3, feature extraction functions supplied in the system are evaluated and selected to produce a segmenter. Because it is not clear which features are active, many extraction functions can be constructed in the system. However, the impact of a large number of operations leads to increase the computing time requirements, and the precision does not necessarily increase.

Due to providing various extraction functions on HSI, RGB, texture, and binary modes, feature operator selection is employed to reduce the number of unnecessary operators. All features are assessed depending on their effectiveness for background subtraction, then the best extraction functions are used to produce a segmenter. Three feature selection methods can be a choice for the function assessment: linear discriminant analysis (LDA), program classification map (PCM), and information gain (IG).

Step A4 is to create a background-subtractor, genetic programming is the basis for the task. Four GP representations are provided to produce a segmenter. They involve two dynamic range selection techniques consisting of original dynamic range selection (DRS) and improved dynamic range selection (DRS2C) established by Oechsle [Oechsle, 2009], furthermore, two threshold hypotheses comprising of entropy thresholding and fluctuation thresholding.

As a computer program gives a numeric value as an output, and the output will be explained as a class

label. Different GP classification strategies are used to transform program output into a class label, for example static range selection (SRS) and dynamic range selection (DRS). The SRS class boundaries are predefined before development begins, and do not change during development. The zero point is given as the class boundary for binary classification problems. Class 1 is labeled to negative output; in contrast; positive program output is labled as class 2;

DRS is an alternative way to SRS. Class sectors of DRS can be identified by slots that are calculated dynamically during the evolutional process; therefore, the class boundaries may vary in each run. In practical terms, the range of $[-250, 250]$ is applied for the segmentation. The produced output values are rounded to the nearest number. Output values less than -250 are represented as -250, and values bigger than 250 are represented as 250 [Oechsle, 2009] [Loveard and Ciesielski, 2001].

Step A5 is to define classes, after the training images are segmented each segmented subject is defined a class. The class label needs to be identified by the user, and the user just selects a class and clicks on an subject area to define the class.

Step A6 is to select features for object classification. The functions include feature extraction based on shape, location-based and grey-scaled properties.

A7, the last essential process for the training part, is creating an object classifier. This process is based on genetic programming.

In steps $B1 - B3$, testing on unseen data is implemented. For the testing part, the processes comprise background subtraction and subject classification. The segmenter and the subject classifier are performed on the testing images. After an image is segmented, and a segmented subject area is classified, the result is presented via the interface.

5.3 The Experimental Work

The experimental work in this chapter has explored the efficiency of two genetic programming frameworks (Jasmine and ELVS). We used Jasmine to create a vision system that can locate and classify trees crowns in images. A database of collected data consisting of 30 images are used. The dataset has been split into 20 images for training, 10 images for testing. Figure 5.2 in page 67 shows samples of the tree database used in the experiment. The first step is to generate a program which have the ability to differentiate between different parts of the image depending on their colour and texture. We need to create two classes- one for the background, and one for the trees. Figure 5.3 in page 67 shows the two classes that been used in this experiment. After Having generated appropriate training data, we need to do some feature selection. The 20 most useful features would be selected by default) and we can choose others. Figure 5.4 shows the score associated with each feature, and whether it is enabled or disabled. The features we have selected will be used by the GP engine to evolve programs that perform the segmentation. The second stage in this vision system is to decide whether or not a region identified by the segmenter is a tree crown or not.

The fitness function that has been used in this research to measure the error of computer is presented in equation 5.1.

$$error = \frac{FP + FN}{N} \tag{5.1}$$

where FP is the number of false positives, FN is the number of false negatives, and N is the number of training samples. If the fitness value of a computer program is estimated and its error is equal to zero, that means the computer program is effective as a solution for the evolution. It was found that Jasmine performs poorly on this problem by detecting the shadow of the tree as a tree crown and failing to evolve a program that could segment trees crowns regions effectively. Consequently measure its efficiency to solve the black grass detection problem. Jasmine employs only shape, location-based and grey-scale

Figure 5.2: Samples of the database used in the experiment.

Figure 5.3: Samples of the two classes used in the experiment.

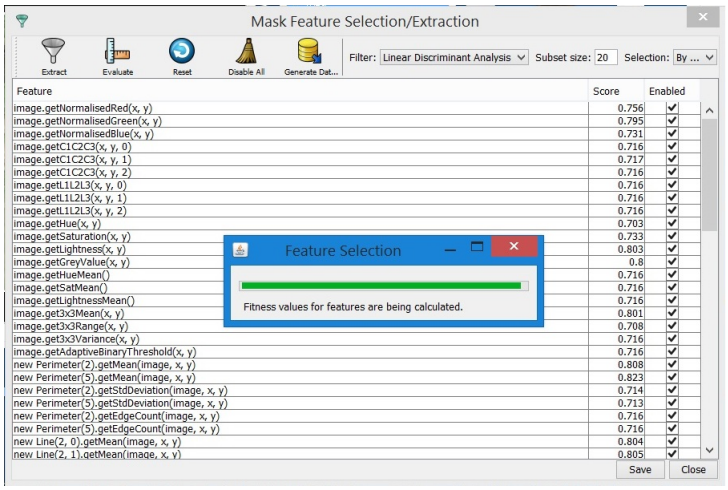

Figure 5.4: The features selected for the experiment.

feature extraction for object classification. Hence, we should expect it to be an appropriate to recognize some types of material with similar shape but different colour or texture. Figure 5.5 in page 68 and figure 5.6 in page 69 show some examples of the tree crown classification results using Jasmine.

Figure 5.5: Examples of the tree crown classification results using Jasmine.

Figure 5.6: Examples of the tree crown classification results using Jasmine.

Jasmine able to face many classification problems. However, Jasmine is not able to solve many problems concerning remote sensing images classification well, mainly because it lacks operators able to classify on the basis of colour and texture. When Jasmine applied to research experiments focus on remote sensing application, several difficulties very disturbing.

1. It is not appropriate for use on large data sets because some processes require user interaction to deal with each image or object (identification, segmentation and classification). It's a huge amount of work for the user especially with big data projects. This problem makes training part of process seriously difficult and critic.

2. The system provide only limited extraction functions for object classification: shape, grey-scaled, and location-based features as a function for object classification, to deal with a complex texture images like satellite images or aerial images, the user need more efficient function that can recognize the different between each pattern.

3. Applying a segmenter and classifier on test data is not automatic. The user has to apply the segmenter to every test image and then the classifier to manage segmented object(s) in each image. Therefore, it is a hard process for the training part and a huge responsibility for the

testing.

4. The classification results difficult to analysis. The interface shown the result using single frame, so it's not easy to collect all the result to check the classification accuracy.

We tried to exceed the above shortcomings by using Genetic Programming toolkit (ELVS) which is presented in the next sections.

5.4 Evolutionary Learning Vision System (ELVS)

The black grass detection problem has also been explored using a state-of-the-art Genetic Programming toolkit developed at Essex known as ELVS ("evolutionary learning vision system"), developed from the research of Oechsle [Oechsle, 2009] and Yimyam [Yimyam, 2015]. ELVS is known to work on a wide variety of problems, ranging from detecting skin lesions and classifying them as cancerous or benign through to detecting whales in satellite images. One of the key characteristics of the toolkit is that it learns from very few images, usually less than ten — a very desirable feature for this research as the amount of training is limited. ELVS contains operators that allow it to segment and classify on the basis of colour, shape and texture. The latter are likely to be the most important for this work; operators include grey-level co-occurrence matrices, Law's masks, Local Binary Patterns, and wavelets.

It was found that ELVS performs poorly on this problem, failing to evolve a program that could segment black grass regions effectively. This is one of a small number of problems on which ELVS has proven ineffective; others include detecting lesions in mammograms and classification of galaxy images. All three of these require classification by texture. As other approaches described in this thesis work somewhat better than this, we conclude that the texture operators — even though they are representative of the state of the art — are not particularly effective. The authors of ELVS are now working on improved ways of describing texture, so the datasets from this thesis may help them achieve improved texture segmentation and classification at some point in the future.

5.5 Summery

Jasmine able to face many classification problems. However, Jasmine is not able to solve many problems concerning remote sensing images classification well, mainly because it lacks operators that able to classify on the basis of colour and texture. Genetic Programming toolkit developed at Essex known as ELVS, it contains operators that allow it to segment and classify on the basis of colour, shape and texture. It was found that ELVS failed to evolve a program that could segment black grass regions effectively.

Next chapter will present the relevant fundamentals of convolutional neural networks (CNNs) that used for remote sensing images classification. Five deep convolutional neural networks (Google-NET, VGG16, VGG19, ResNET50, and ResNET101) and shallow convolutional neural networks have been applied to five different classes of ROI images including unknown range.

Chapter 6

Classification Based on Convolutional Neural Networks (CNNs)

6.1 Introduction

In the last few years, the interest for convolutional neural network (CNN or ConvNet) has been increasing rapidly due to their impressive results in image classification and recognition and how they have revolutionized the field of computer vision including remote sensing images [Castelluccio et al., 2015] [Krizhevsky et al., 2012].

CNN is a kind of neural network model that extracts higher representations for the image data. Unlike the classical recognition methods that need to define the image features in advance, CNN uses raw pixel data of the image, trains the model, then extracts the features automatically for better classification. Most of the existing approaches for the monitoring and detecting of cereal crops employ handcrafted feature extraction methods. Several projects have demonstrated that deep learning features produce beneficial information about the relationships between raw data and learnt features. In recent years, CNN has become the widely used approach in computer vision. The main advantage of CNN is that

it combines feature extraction and classification and produces the final classification results without any additional process. CNNs can be considered as a powerful automatic feature extractor because it can handle huge number of training samples, and features are learned automatically by the neural networks [Sharif Razavian et al., 2014]. Motivated by the success of CNN in various computer vision processes, five deep and shallow convolutional neural networks have been trained to explore the detection of black grass infestations of wheat crops.

This chapter presents the relevant fundamentals of convolutional neural networks (CNNs) that used for remote sensing images classification. Five deep convolutional neural networks (Google-NET, VGG16, VGG19, ResNET50, ResNET101) and shallow convolutional neural networks have been trained to five different classes of ROI images including unknown range: wheat, black grass, road, and bushes. Caffe is a very widely used framework for deep learning. In this chapter, we explain how to get started with CAFFE and the steps that should be set.

6.2 Motivation for Using Convolutional Neural Networks (CNNs)

There are three main reasons for using Convolutional Neural Networks:

- A CNN is a category of multilayer artificial neural network that can automatically learn feature within multiple levels from original images instead of using original images as feature input that might involve noises or undesirable information that reduce the efficiency of a classifier.

- Nowadays, images used for computer vision problems are usually of large scale with high resolution. To process a colored image of size 224×224 including the 3 color channels (RGB) in the image, that comes out to $(224 \times 224) \times 3 = 150.528$ input features. A typical hidden layer in such a network might have 1024 nodes, so it needs to train 150.528×1024 which equals to more than 150 million weights for the first layer alone. In this case, the network would be huge and nearly impossible to train. Consequently, it seems wasteful for every node in the first hidden layer to

look at every pixel of the image.

- To train a network to detect an object in an image, it needs to detect the object regardless of where it appears in the image. Imagine training a network that works well on a certain object image, but then feeding it a slightly shifted version of the same image. The object would not activate the same neurons, so the network would react completely differently. Using a CNN can help mitigate these problems.

6.3 The CNN Architectures

CNNs are the most common type of deep learning method which can employ an input image, assign learnable weights and biases to various objects in the image and then be able to differentiate these objects. Large amounts of data can be trained in the deep learning methods which considered as the main advantage of deep learning method. While, the key drawback of deep learning methods is that CNNs need to be trained with huge datasets that include enough variations in the training data. Many architectures have been proposed to achieve high and efficient image classification and recognition. There is no design methodology for CNNs: the seem to be designed on the basis of image size and experience.

6.3.1 VGG16

VGG16 is a convolutional neural network, it is 16 layers deep and has the ability to classify images into 1000 object categories. There are 16 layers with learnable weights: 13 convolutional layers, and 3 fully connected layers (fc6, fc7, and fc8) as shown in figure 6.1. Figure 6.2 depicts the architecture of the VGG16 layers [Russakovsky et al., 2015], in which the input to cov1 layer is of fixed size 224×224 RGB image. The image is passed through a stack of convolutional (conv.) layers, where the filters were used with a very small receptive field: 3×3. It also utilizes 1×1 convolution filters, which can be seen

as a linear transformation of the input channels (followed by non-linearity). The convolution stride is fixed to 1 pixel; the spatial padding of conv. layer input is such that the spatial resolution is preserved after convolution, the padding is 1-pixel for 3×3 conv. layers. Spatial pooling is carried out by five max-pooling layers, which follow some of the conv. layers (not all the conv. layers are followed by max-pooling). Max-pooling is performed over a 2×2 pixel window, with stride 2.

Three fully-connected (fc) layers follow a stack of convolutional layers (which has a different depth in different architectures): the first two have 4096 channels each, the third performs 1000 channels. The final layer is the soft-max layer. The configuration of the fully connected layers is the same in all networks.

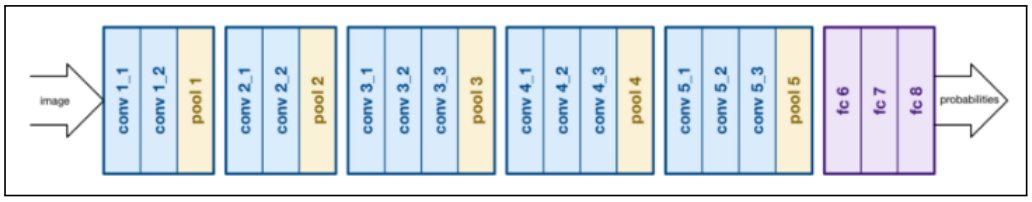

Figure 6.1: The 16 Layers Deep of VGG16 Network

```
 1   'input'     Image Input             224x224x3 images with 'zerocenter' normalization
 2   'conv1_1'   Convolution             64 3x3x3 convolutions with stride [1  1] and padding [1  1  1  1]
 3   'relu1_1'   ReLU                    ReLU
 4   'conv1_2'   Convolution             64 3x3x64 convolutions with stride [1  1] and padding [1  1  1  1]
 5   'relu1_2'   ReLU                    ReLU
 6   'pool1'     Max Pooling             2x2 max pooling with stride [2  2] and padding [0  0  0  0]
 7   'conv2_1'   Convolution             128 3x3x64 convolutions with stride [1  1] and padding [1  1  1  1]
 8   'relu2_1'   ReLU                    ReLU
 9   'conv2_2'   Convolution             128 3x3x128 convolutions with stride [1  1] and padding [1  1  1  1]
10   'relu2_2'   ReLU                    ReLU
11   'pool2'     Max Pooling             2x2 max pooling with stride [2  2] and padding [0  0  0  0]
12   'conv3_1'   Convolution             256 3x3x128 convolutions with stride [1  1] and padding [1  1  1  1]
13   'relu3_1'   ReLU                    ReLU
14   'conv3_2'   Convolution             256 3x3x256 convolutions with stride [1  1] and padding [1  1  1  1]
15   'relu3_2'   ReLU                    ReLU
16   'conv3_3'   Convolution             256 3x3x256 convolutions with stride [1  1] and padding [1  1  1  1]
17   'relu3_3'   ReLU                    ReLU
18   'pool3'     Max Pooling             2x2 max pooling with stride [2  2] and padding [0  0  0  0]
19   'conv4_1'   Convolution             512 3x3x256 convolutions with stride [1  1] and padding [1  1  1  1]
20   'relu4_1'   ReLU                    ReLU
21   'conv4_2'   Convolution             512 3x3x512 convolutions with stride [1  1] and padding [1  1  1  1]
22   'relu4_2'   ReLU                    ReLU
23   'conv4_3'   Convolution             512 3x3x512 convolutions with stride [1  1] and padding [1  1  1  1]
24   'relu4_3'   ReLU                    ReLU
25   'pool4'     Max Pooling             2x2 max pooling with stride [2  2] and padding [0  0  0  0]
26   'conv5_1'   Convolution             512 3x3x512 convolutions with stride [1  1] and padding [1  1  1  1]
27   'relu5_1'   ReLU                    ReLU
28   'conv5_2'   Convolution             512 3x3x512 convolutions with stride [1  1] and padding [1  1  1  1]
29   'relu5_2'   ReLU                    ReLU
30   'conv5_3'   Convolution             512 3x3x512 convolutions with stride [1  1] and padding [1  1  1  1]
31   'relu5_3'   ReLU                    ReLU
32   'pool5'     Max Pooling             2x2 max pooling with stride [2  2] and padding [0  0  0  0]
33   'fc6'       Fully Connected         4096 fully connected layer
34   'relu6'     ReLU                    ReLU
35   'drop6'     Dropout                 50% dropout
36   'fc7'       Fully Connected         4096 fully connected layer
37   'relu7'     ReLU                    ReLU
38   'drop7'     Dropout                 50% dropout
39   'fc8'       Fully Connected         1000 fully connected layer
40   'prob'      Softmax                 softmax
41   'output'    Classification Output   crossentropyex with 'tench' and 999 other classes
```

Figure 6.2: The VGG16 layers

6.3.2 VGG19

VGG19 is a convolutional neural network, it is 19 layers deep and can classify images into 1000 object categories. The network has 47 layers, there are 19 layers with learnable weights: 16 convolutional layers, and 3 fully connected layers. Figure 6.3 depicts the architecture of the VGG19 layers [He et al., 2016]. A fixed size of 224 × 224 RGB image is given as input to this network which means that the matrix is of shape (224,224,3). The only preprocessing that was done is that they subtracted the mean RGB value from each pixel, computed over the whole training set. Used kernels of 3 × 3 size with a stride size of 1 pixel, this enabled them to cover the whole notion of the image. Spatial padding is used

1	'input'	Image Input	224x224x3 images with 'zerocenter' normalization
2	'conv1_1'	Convolution	64 3x3x3 convolutions with stride [1 1] and padding [1 1 1 1]
3	'relu1_1'	ReLU	ReLU
4	'conv1_2'	Convolution	64 3x3x64 convolutions with stride [1 1] and padding [1 1 1 1]
5	'relu1_2'	ReLU	ReLU
6	'pool1'	Max Pooling	2x2 max pooling with stride [2 2] and padding [0 0 0 0]
7	'conv2_1'	Convolution	128 3x3x64 convolutions with stride [1 1] and padding [1 1 1 1]
8	'relu2_1'	ReLU	ReLU
9	'conv2_2'	Convolution	128 3x3x128 convolutions with stride [1 1] and padding [1 1 1 1]
10	'relu2_2'	ReLU	ReLU
11	'pool2'	Max Pooling	2x2 max pooling with stride [2 2] and padding [0 0 0 0]
12	'conv3_1'	Convolution	256 3x3x128 convolutions with stride [1 1] and padding [1 1 1 1]
13	'relu3_1'	ReLU	ReLU
14	'conv3_2'	Convolution	256 3x3x256 convolutions with stride [1 1] and padding [1 1 1 1]
15	'relu3_2'	ReLU	ReLU
16	'conv3_3'	Convolution	256 3x3x256 convolutions with stride [1 1] and padding [1 1 1 1]
17	'relu3_3'	ReLU	ReLU
18	'conv3_4'	Convolution	256 3x3x256 convolutions with stride [1 1] and padding [1 1 1 1]
19	'relu3_4'	ReLU	ReLU
20	'pool3'	Max Pooling	2x2 max pooling with stride [2 2] and padding [0 0 0 0]
21	'conv4_1'	Convolution	512 3x3x256 convolutions with stride [1 1] and padding [1 1 1 1]
22	'relu4_1'	ReLU	ReLU
23	'conv4_2'	Convolution	512 3x3x512 convolutions with stride [1 1] and padding [1 1 1 1]
24	'relu4_2'	ReLU	ReLU
25	'conv4_3'	Convolution	512 3x3x512 convolutions with stride [1 1] and padding [1 1 1 1]
26	'relu4 3'	ReLU	ReLU
27	'conv4_4'	Convolution	512 3x3x512 convolutions with stride [1 1] and padding [1 1 1 1]
28	'relu4_4'	ReLU	ReLU
29	'pool4'	Max Pooling	2x2 max pooling with stride [2 2] and padding [0 0 0 0]
30	'conv5_1'	Convolution	512 3x3x512 convolutions with stride [1 1] and padding [1 1 1 1]
31	'relu5_1'	ReLU	ReLU
32	'conv5_2'	Convolution	512 3x3x512 convolutions with stride [1 1] and padding [1 1 1 1]
33	'relu5_2'	ReLU	ReLU
34	'conv5_3'	Convolution	512 3x3x512 convolutions with stride [1 1] and padding [1 1 1 1]
35	'relu5_3'	ReLU	ReLU
36	'conv5_4'	Convolution	512 3x3x512 convolutions with stride [1 1] and padding [1 1 1 1]
37	'relu5_4'	ReLU	ReLU
38	'pool5'	Max Pooling	2x2 max pooling with stride [2 2] and padding [0 0 0 0]
39	'fc6'	Fully Connected	4096 fully connected layer
40	'relu6'	ReLU	ReLU
41	'drop6'	Dropout	50% dropout
42	'fc7'	Fully Connected	4096 fully connected layer
43	'relu7'	ReLU	ReLU
44	'drop7'	Dropout	50% dropout
45	'fc8'	Fully Connected	1000 fully connected layer
46	'prob'	Softmax	softmax
47	'output'	Classification Output	crossentropyex with 'tench' and 999 other classes

Figure 6.3: The VGG19 layers [He et al., 2016].

to preserve the spatial resolution of the image. max pooling was performed over a 2×2 pixel windows with stride 2. This is followed by (ReLu) to introduce non-linearity to make the model classify better and to improve computational time. Implemented three fully connected layers from which first two were of size 4096 and after that a layer with 1000 channels and the final layer is a softmax function.

6.3.3 ResNet50 and ResNet101

ResNet50 and ResNet101 [Challenge, 2012] are convolutional neural networks, the ResNet50 network is 50 layers deep and it has 171 layers. while the ResNet101 network is 101 layers deep and it has 347 layers. Both of these networks can classify images into 1000 object classes. Figure 6.4 shows the ResNet network layers. The ResNet50 architecture contains the following element:

- A convoultion with a kernel size of 7×7 and 64 different kernels all with a stride of size 2 giving 1 layer.

- Max pooling with also a stride size of 2.

- In the next convolution there is a $1 \times 1, 64$ kernel following this a $3 \times 3, 64$ kernel and at last a $1 \times 1, 256$ kernel, These three layers are repeated in total 3 time so giving 9 layers in this step.

- In the next convolution there is a kernel of $1 \times 1, 128$ after that a kernel of $3 \times 3, 128$ and at last a kernel of $1 \times 1, 512$ this step is repeated 4 time so giving 12 layers in this step.

- After that there is a kernal of $1 \times 1, 256$ and two more kernels with $3 \times 3, 256$ and $1 \times 1, 1024$ and this is repeated 6 time giving a total of 18 layers (In ResNet101, this is would be repeated for 23 time). And then again a $1 \times 1, 512$ kernel with two more of $3 \times 3, 512$ and $1 \times 1, 2048$ and this is repeated 3 times giving a total of 9 layers.

- Finally, there is a average pool and end it with a fully connected layer containing 1000 nodes and at the end a softmax function so this gives 1 layer. So totaling this, it gives a $1+9+12+18+9+1 = 50$ layers Deep Convolutional network (In ResNet101, it gives $1 + 9 + 12 + 69 + 9 + 1 = 101$ layers).

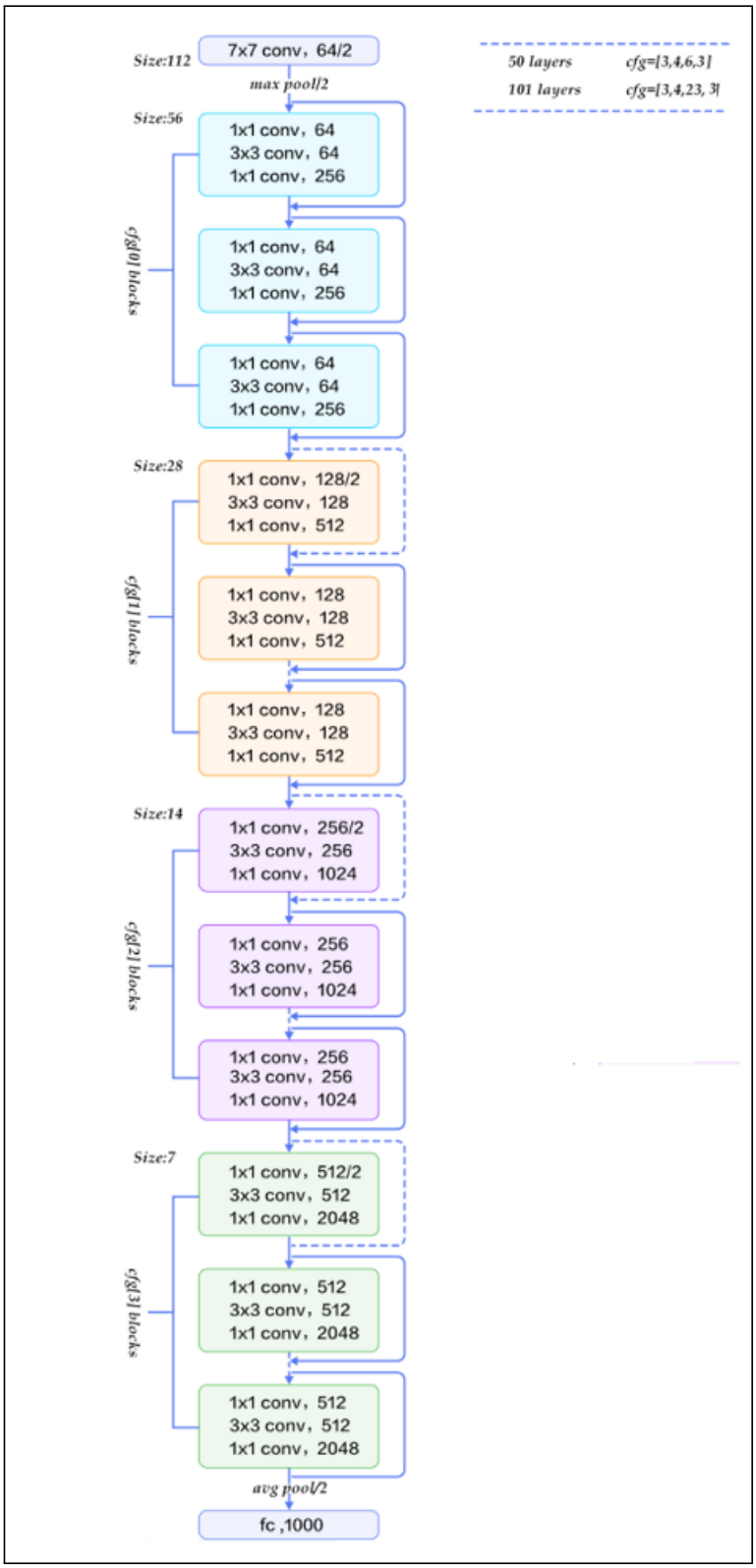

Figure 6.4: The ResNet Network Layers.

6.3.4 GoogleNet

GoogleNet is a convolutional neural network, it is 22 layers deep and it has 144 layers [Szegedy et al., 2015]. This architecture takes image of size 224×224 with RGB color channels. All the convolutions inside this architecture uses Rectified Linear Units (ReLU) as their activation functions. In the previous architecture, the fully connected layers are used at the end of the network. These fully connected layers contain the majority of parameters of many architectures that causes an increase in computation cost. In GoogLeNet architecture, there is a method called global average pooling is used at the end of the network. This layer takes a feature map of 7×7 and averages it to 1×1. This also decreases the number of trainable parameters to 0 and improves the accuracy. One of the main features of GoogleNet is using Inception module. In this architecture, there is a fixed convolution size for each layer. In the Inception module 1×1, 3×3, 5×5 convolution and 3×3 max pooling performed in a parallel way at the input and the output of these are stacked together to generated final output. The idea behind that convolution filters of different sizes will handle objects at multiple scale better. Figure 6.5 shows the GoogleNet network layers.

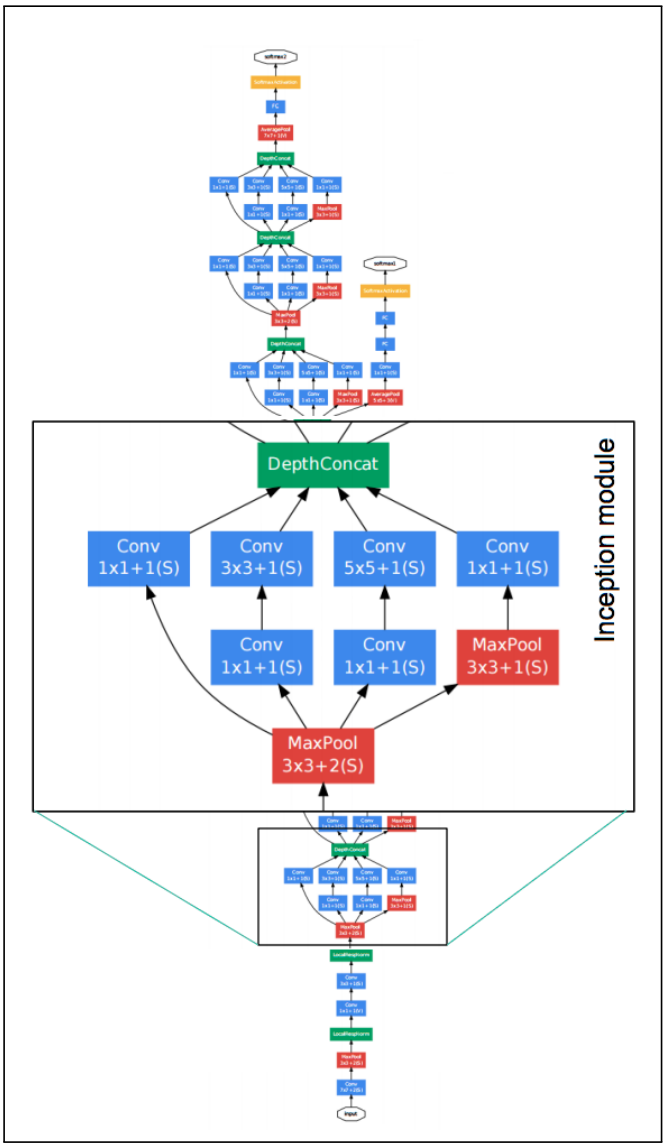

Figure 6.5: The GoogleNet Network Layers.

6.3.5 The ConvNets Training Process

The overall training process of the Convolution Network may be summarized as below:

Step 1 Initialize all filters and parameters/weights with random values

Step 2 The network takes a training image as input, applies the forward propagation step (convolution, ReLU and pooling operations along with forwarding propagation in the Fully Connected layer) and finds the output probabilities for each class.

Step 3 Calculate the total error at the output layer:

$$TotalError = \sum \frac{1}{2}(targetprobability - outputprobability)$$

Step 4 Use Backpropagation to calculate the gradients of the error with respect to all weights in the network and use gradient descent to update all filter values/weights and parameter values to minimize the output error.

- The weights are adjusted in proportion to their contribution to the total error.
- This means that the network has learnt to classify a particular image correctly by adjusting its weights/filters such that the output error is reduced.
- Parameters such as the number of filters, filter sizes, architecture of the network have all been fixed before Step 1 and do not change during training process — only the values of the filter matrix and connection weights get updated.

Step 5 Repeat steps 2 — 4 with all images in the training set.

6.4 CAFFE

Caffe is a common and high performance deep learning framework, there are 4 steps that should be set to get started with CAFFE :

- Data preparation: Images are prepared and stored in a format that can be used by Caffe (a Caffe dataset).
- Model definition: A CNN architecture and its parameters are chosen in a configuration file with extension .prototxt.

- Solver definition: The solver is used for model optimization. The solver parameters are defined in a configuration file with extension .prototxt.

- Model training and finetuning: In this step, the model is trained. The trained model is produced in a file of extension .caffemodel. After this step, the .caffemodel trained model is used to make predictions of new unknown data.

6.5 The Experimental Work

In this section, we first describe the image databases used in the training phase. Thereafter, we examine the effect of training five deep convolutional neural networks (Google-NET, VGG16, VGG19, ResNET50, ResNET101) and five shallow convolutional neural networks on the classification performance on the validation dataset. Finally, we compare the performance of the motioned methods on the validation and testing datasets. Five different classes of ROI images including unknown range: wheat, black grass, road, and bushes have been employed for this experiment.

6.5.1 The Study Area

To cover the need of a big database for training and because the Blackgrass shape and print change during the growing period, it was necessary for as to monitor and capture the change as long as we can. The monitoring start in 16th of April and the last viewing was in 4th of Jun, the area that chosen place in East mersea near Colchester (51°47′45.0”N 0°56′29.6”E , 51°47′35.2”N 0°56′40.2”E), It's a wheat field that affected by blackgrass. The viewing and image capturing arranging in a week period. Both RGB and IR cameras was used to extend the database and improve the training progress. The images are captured using UAV with two different sensors Visible and NIR (MAPIR Survey2 Cameras). See figures 6.6, 6.7, and 6.8.

Figure 6.6: Chosen Area.

6.5.2 Experiment Design and Database

The targeting area of (wheat crops field) was monitored during different growth stages on different seasons and at different times. The images have been taken by using MAPIR survay2 cameras (IR and visible light) attached to 3DR solo drone A database of the collected data consisting of 4100 images are used. The dataset has been split into 3000 images for training, 1000 images validation, and 100 images for testing. The training dataset images which are used for fine-tuning have been scaled to 224x224x3 images to fit the CNN model input requirement. To accomplish four-fold cross-validation, the original sample set is randomly subject-partitioned into four equal sized subsets. Of the four subsets, one is retained as the validation dataset and the remaining three subsets are used as training dataset. There are no overlapping subject images between the training, validating, and testing data. With the five shallow networks, features are extracted using $inception5b-output$ layer for GoogleNet, $conv5-2$ layer for VGG16, $conv5$—4 layer for VGG19, add—11 layer for RezNet50, and $res5b$ layer for RezNet101.

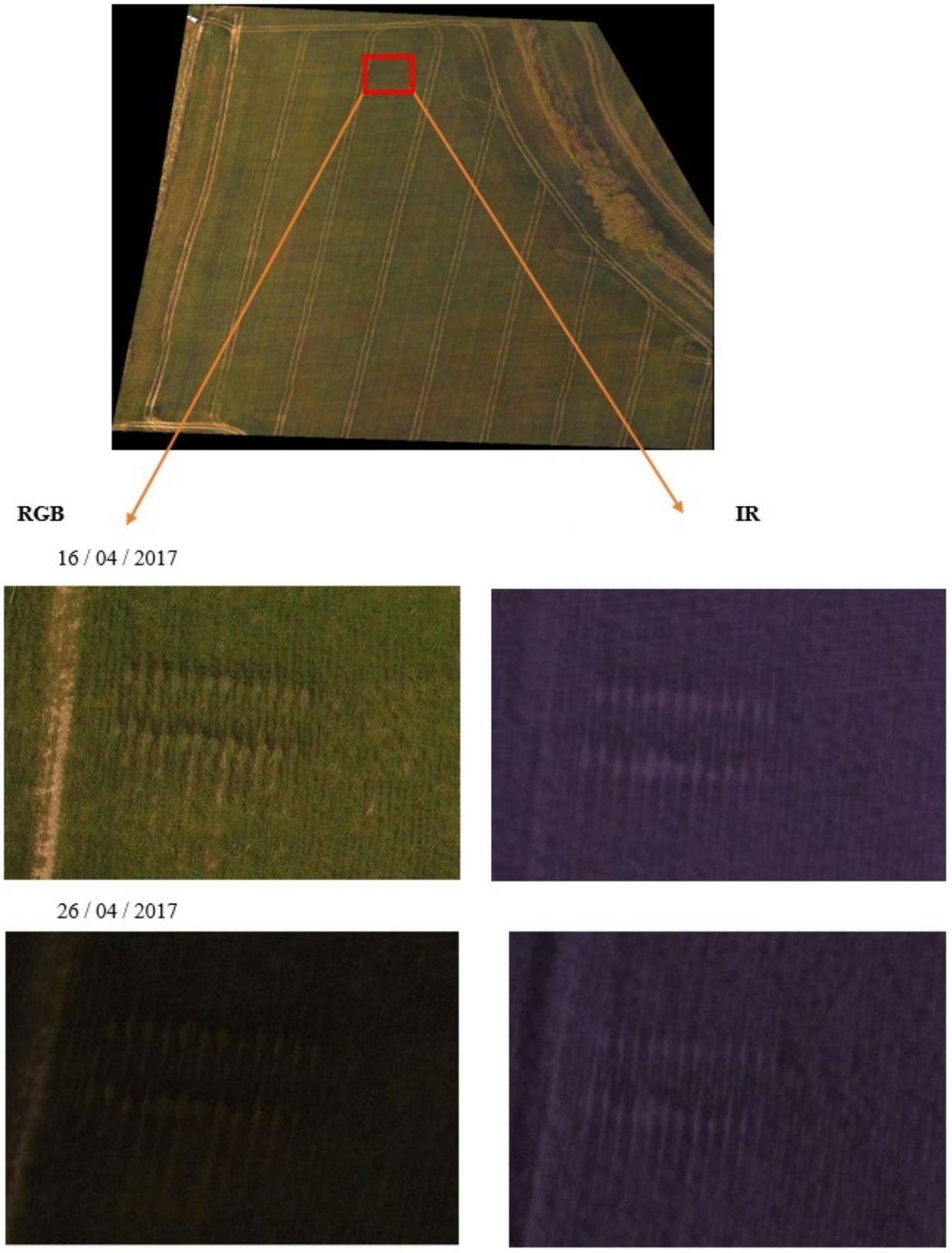

Figure 6.7: Sample from Blackgrass Spot

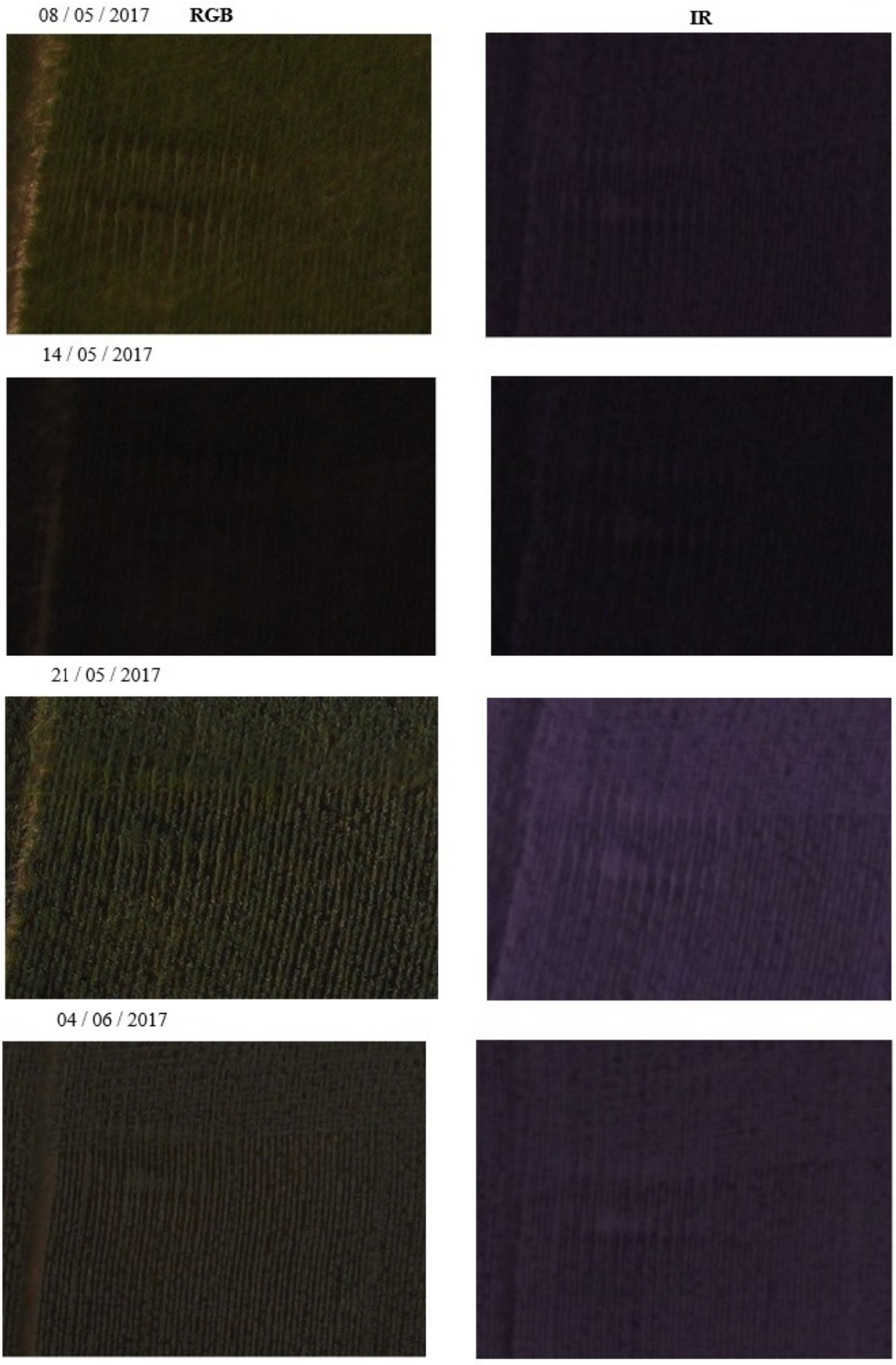

Figure 6.8: Sample from Blackgrass Spot

6.5.3 CNN Architecture Setup

The Caffe toolkit [Jia et al., 2014] on NVIDIA GeForce GTX 1080 GPU was applied to fine-tune the pre-trained GoogleNet, VGG16, VGG19, RezNet50, and RezNet101 deep CNN models. The parameters of the applied training algorithm were as follows: momentum=0.9, weight decay=0.0001, initial learning rate=0.0001, the solver for training network is 'sgdm'. Before starting the testing, the 4-fold cross-validation was adopted to find the optimal extracted features which have the best average accuracy on the validation datasets.

6.5.4 Results and Discussion

Before starting the testing, the 4-fold cross-validation was examined to find the optimal extracted features which have the best average accuracy on the validation datasets. To analyse the efficiency of the deep and shallow networks, we conducted two comparative evaluations to compare the accuracy and speed of the networks that been used for this experiment. Table 6.1 shows the classification accuracy based on deep CNNs and shallow CNNs. Table 6.2 shows the the average accuracy for the deep and shallow CNNs, the training time for the training images, and the testing timing speeds per image. To calculate the accuracy, we applied the following equation which is defined as [Kanwal et al., 2016]:

$$\frac{TP + TN}{N} \tag{6.1}$$

where TP is the number of true positive detections, TN is the number true negative detections, and N represents the number of images tested. In general, the experimental results show that the classification performance based on deep CNN outperformed the classification performance based on shallow CNN in the majority of cases to detect black grass regions. Regarding to the average accuracy for the four group, deep ResNet50 network achieved the highest accuracy for validating and testing phases (89.4%, 85.0%) in time (00:15:43). While the classification accuracy for validating and testing phases recorded

the highest score (87.4%, 81.5%)based on the shallow ResNet101 network in time (00:16:49).

Table 6.1: Classification Accuracy Based on CNNs and Shallow CNNs

					CNN		Shallow CNN		
		Group No	CNN Name	Layer	Validation	Test	Validation	Test	Training Time
1	1	1	'googlenet'	'inception_5b-output'	82.0%	80.0%	82.6%	78.0%	00:06:44
	2	1	'vgg16'	'conv5_2'	87.5%	83.0%	57.6%	53.0%	00:08:46
	3	1	'vgg19'	'conv5_4'	85.0%	89.0%	82.9%	77.0%	00:10:53
	4	1	'resnet50'	'add_11'	88.7%	85.0%	86.3%	81.0%	00:15:35
	5	1	'resnet101'	'res5b'	86.6%	81.0%	86.5%	84.0%	00:18:43
2	1	2	'googlenet'	'inception_5b-output'	85.4%	84.0%	84.3%	78.0%	00:06:38
	2	2	'vgg16'	'conv5_2'	90.0%	79.0%	76.0%	66.0%	00:09:11
	3	2	'vgg19'	'conv5_4'	89.6%	71.0%	88.1%	78.0%	00:10:45
	4	2	'resnet50'	'add_11'	88.9%	84.0%	86.2%	78.0%	00:15:49
	5	2	'resnet101'	'res5b'	85.1%	59.0%	86.7%	83.0%	00:18:36
3	1	3	'googlenet'	'inception_5b-output'	80.3%	76.0%	84.2%	80.0%	00:06:38
	2	3	'vgg16'	'conv5_2'	85.5%	75.0%	72.3%	26.0%	00:09:28
	3	3	'vgg19'	'conv5_4'	87.2%	73.0%	88.2%	82.0%	00:11:06
	4	3	'resnet50'	'add_11'	89.5%	86.0%	83.9%	81.0%	00:15:37
	5	3	'resnet101'	'res5b'	90.7%	81.0%	89.7%	79.0%	00:19:43
4	1	4	'googlenet'	'inception_5b-output'	83.1%	85.0%	83.6%	75.0%	00:07:37
	2	4	'vgg16'	'conv5_2'	85.9%	77.0%	68.0%	77.0%	00:09:29
	3	4	'vgg19'	'conv5_4'	90.1%	83.0%	83.2%	73.0%	00:11:01
	4	4	'resnet50'	'add_11'	90.3%	85.0%	85.7%	77.0%	00:15:51
	5	4	'resnet101'	'res5b'	88.5%	78.0%	86.7%	80.0%	00:20:08

Table 6.2: Average Accuracy For the Four Groups

			Average accuracy for the four group							
			CNN		Shallow CNN		Training Time for 3000 Images		Testing Time Speed S/Image	
	CNN Name	Layer	Validation	Test	Validation	Test	CNN	Shallow CNN	CNN	Shallow CNN
1	'googlenet'	'inception_5b-output'	82.7%	81.3%	83.7%	77.8%	00:06:54	00:04:33	0.0040	0.0031
2	'vgg16'	'conv5_2'	87.2%	78.5%	68.5%	55.5%	00:09:13	00:07:45	0.0112	0.0102
3	'vgg19'	'conv5_4'	88.0%	79.0%	85.6%	77.5%	00:10:56	00:08:49	0.0129	0.0118
4	'resnet50'	'add_11'	89.4%	85.0%	85.5%	79.3%	00:15:43	00:13:51	0.0086	0.0060
5	'resnet101'	'res5b'	87.7%	74.8%	87.4%	81.5%	00:19:18	00:16:49	0.0107	0.0087

6.6 Conclusions

CNNs is a category of feedforward artificial neural network that has proven highly effective in various areas such as classification and image recognition. This chapter has presented the relevant fundamentals of convolutional neural networks (CNNs) that used for remote sensing images classification. Five deep

convolutional neural networks (Google-NET, VGG16, VGG19, ResNET50, ResNET101) and five shallow convolutional neural networks were trained to five different classes of ROI images including unknown range: wheat, black grass, road, and bushes. The experimental results showed that the classification performance based on deep CNN outperformed the classification performance based on shallow CNN in the majority of cases to detect black grass regions. Regarding to the average accuracy for the four group, deep ResNet50 network achieved the highest accuracy for validating and testing phases in time (00:15:43). While the classification accuracy for validating and testing phases recorded the highest score based on the shallow ResNet101 network in time (00:16:49).

Next chapter will introduce the conclusions drawn from this research and make suggestions for further work.

Chapter 7

Conclusions and Future Works

7.1 Thesis Contributions

Black-grass is one of the biggest challenges that facing most the arable farmers in the UK. Black-grass is an annual weed of wheat that occurs in the Uk and Europe. Mainly, it is found in the cereal growing areas and it rarely occurs outside of cultivated area. Commonly, the flowers heads show above the cereal crop in May and June.

Although annual rather than perennial, a single black grass plant was able to produce many thousands of seeds, which were small enough to be distributed by the wind, though it was believed that the principal means by which it spreads from field to field was by sticking inside farm machinery such as combine harvesters. Anecdotal evidence from farmers was that black grass plants tend to grow directly adjacent to wheat plants, with their roots enveloping those of the wheat and thus effectively strangling the wheat. Black grass was resistant to most weed-killers, and spraying with those weed-killers that were allowed requires a licence — this made eradicating it difficult. When young black grass plants were potentially detectable because they did not grow in the regular pattern of the wheat plants; but after germination, they tended to be smaller than adjacent wheat plants until they reach maturity, at which time they were

slightly taller and with their characteristic black seed-heads.

The focus of this research was on the monitoring of cereal crops, in particular infestation of wheat crops by black grass, a pernicious weed. This was regarded as the major pest for cereal crops, at least in East Anglia, one that caused significant loss of income for farmers. Imagery of fields infested with black grass plants was simply not available at the kinds of spatial resolution needed for detection, so an important part of the research was the collection of imagery from a drone. This was done over more than one year and at critical points in the growth of the crop and its 'companion' weed, so that detection at different stages of the growth cycle is potentially possible. However, it should be pointed out that black grass is difficult to detect when immature, except by walking amongst the plants and examining them individually.

The first part of chapter 2 started by reviewing remote sensing image principles in general terms. It went on to discuss the use of Image classification in remote sensing and the major steps that may be involved in the image classification process. The second part shed the light on the two main approaches that used for classification in remote sensing; supervised and unsupervised learning. Also, some detail of the particular techniques used in this thesis, ranging from conventional statistical ones to Genetic Programming and Convolutional Neural Networks, presented along with a review of their application in remote sensing.

Chapter 3 presented two experimental work that used different conventional remote sensing classification techniques. The former shed light on the regional agricultural land texture classification using grey level cooccurrence matrices (GLCMs) for features extraction and using SVM and Decision Tree Induction techniques for classification. The results showed that these texture features have high discrimination accuracy and classification using support vector machines outperforms the decision tree technique performance in terms of the number of mis-classifications instances and the classification accuracy with an overall accuracy of 83%, while the decision tree accuracy is 70%, respectively.The later focused on outcomes gained from three conventional supervised techniques that used for remote sensing images

classification; random forest, SVMs, and decision tree classifiers. Results indicated that the random forest classifier performance outperforms the decision tree and SVMs techniques performance in terms of the number of mis-classifications instances and the classification accuracy with an overall accuracy of 86% while the Decision Tree accuracy is 67% and the SVMs accuracy is 56%, respectively

Chapter 4 focused on analysis by Vegetation Indices (VIs). The first part of this chapter illustrated the study area that been used in research. The second part shed light on the fundamental aspects of the most common indices that analysts use in remote sensing; NDVI, CRI2, SG index, and RG Ratio index. In addition, the experiment results of using these indices to detect black grass regions were introduced. The results indicated that using NDVI, CRI2, SG index, and RG Ratio index can achieve high precision up to 90%. The aim in this chapter was to find a system that detect black grass automatically, deal with large data, and can detect the black grass in all growing stage. However, the mentioned VIs still a manual process, consumed a lot of time and effort, needed an expert to deal with, and it was not effective with the data of large size. In addition, It was not used with black grass in early stage.

Chapter 5 presented two genetic programming toolkits developed at Essex University: the Jasmine vision system builder and ELVS (evolutionary learning vision system) to detect the tree crown regions and consequently measure its efficiency to solve the black grass detection problem. The first part of this chapter described the principles underlying the Jasmine and identifies its major shortcomings. In the second part, the tree crown detection problem was explored using the mentioned toolkits. Results indicated that Jasmine is not able to solve many problems concerning remote sensing images classification well, mainly because it lacks operators that able to classify on the basis of colour and texture. In addition, It was found that ELVS failed to evolve a program that could segment black grass regions effectively.

Chapter 6 presented the relevant fundamentals of convolutional neural networks (CNNs) that used for remote sensing images classification. Five deep convolutional neural networks (Google-NET, VGG16, VGG19, ResNET50, ResNET101) and shallow convolutional neural networks were trained to five dif-

ferent classes of ROI images including unknown range: wheat, black grass, road, and bushes. The experimental results showed that the classification performance based on deep CNN outperformed the classification performance based on shallow CNN in the majority of cases to detect black grass regions. Regarding to the average accuracy for the four group, deep ResNet50 network achieved the highest accuracy for validating and testing phases in time (00:15:43). While the classification accuracy for validating and testing phases recorded the highest score based on the shallow ResNet101 network in time (00:16:49).

The contributions to the knowledge that extracted from this research to help detecting the black grass of Spring wheat which is planted between March and May and should be harvested between July and September are summarized as two approaches:

- Detecting The Black Grass Using Manual Methods

  The results showed that the extracted texture features using grey level cooccurrence matrices (GLCMs) have high discrimination accuracy and classification using support vector machines outperforms the decision tree technique performance in terms of the number of mis-classifications instances and the classification accuracy with an overall accuracy of 83%, while the decision tree accuracy is 70%, respectively.

  The results of using three conventional supervised techniques for remote sensing images classification; random forest, SVMs, and decision tree classifiers indicated that the random forest classifier performance outperforms the decision tree and SVMs techniques performance in terms of the number of mis-classifications instances and the classification accuracy with an overall accuracy of 86% while the Decision Tree accuracy is 67% and the SVMs accuracy is 56%, respectively.

  The experiment results of using Vegetation Indices (VIs) to detect black grass regions indicated that using NDVI, CRI2, SG index, and RG Ratio index can achieve high precision up to 90%.

  Results indicated that Jasmine is not able to solve many problems concerning remote sensing

images classification well, mainly because it lacks operators that able to classify on the basis of colour and texture. In addition, It was found that ELVS failed to evolve a program that could segment black grass regions effectively.

- Detecting The Black Grass Using Automated Methods

  The experimental results of using five deep convolutional neural networks (Google-NET, VGG16, VGG19, ResNET50, ResNET101) and shallow convolutional neural networks showed that the classification performance based on deep CNN outperformed the classification performance based on shallow CNN in the majority of cases to detect black grass regions. Regarding to the average accuracy for the four group, deep ResNet50 network achieved the highest accuracy for validating and testing phases (89.4%, 85.0%) in time (00:15:43). While the classification accuracy for validating and testing phases recorded the highest score (87.4%, 81.5%) based on the shallow ResNet101 network in time (00:16:49).

7.2 Future Directions

The contributions from this research lay the groundwork for future research into the detection of black grass infestations of wheat crops in order to explore a wide range of feature extraction and classification problems. Some specific issues should be considered for future research:

- An important part of the research was the collection of imagery from a drone. So, the datasets could be extended and captured during all the crop growing periods to explore its effectiveness on the algorithms performance. The ground sampling distance is an important metric to consider for photogrammetry and measurements in the images. As a general consideration, the ground resolution (related to the ground pixel size) is inversely proportional to the distance, so the closer the better. However, the closer you are to the object, the more pictures you need to shoot, and the more time will take the processing, so this is a trade-off researchers should keep in mind.

- All the used images in the experimental work was captured by drone (RGB and NIR images). So, using a higher resolution camera with different sensors (such as thermal sensor) and different filters will make the dataset more flexible to be used for different analysing and training techniques.

- Continue developing the genetic programming toolkit (EILVS) using our dataset and extended colour and texture operations for object classification. So, it can deal with the black grass detection problems more efficiently.

- CNN integrates feature extraction with classification, it receives the raw input data and produces the final classification results without any additional process. In the future work, we suggest to use CNNs as feature extraction and use another technique for classification such as SVMs to explore its effectiveness on the performance accuracy.

- Using a combination of different vegetation indices for features exaction with different CNN architectures for object classification and comparing their performances to detect the black grass more efficiently.

Deep CNNs Classification Results																							
		googlenet				vgg16				vgg19				resnet50				resnet101					
Test Classification	Test Class	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4		
1(1).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 4		
1(100).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 3	Class 1	Class 1	Class 3	Class 3	Class 1	Class 1	Class 1	Class 3	Class 3	Class 3	Class 3	Class 1	Class 1		
1(11).jpg	Class 1	Class 3	Class 3	Class 3	Class 2	Class 3	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 1	Class 1	Class 1	Class 1	Class 2	Class 4	Class 4	Class 2		
1(12).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(17).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 3	Class 3	Class 3	Class 1	Class 3	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 4	Class 3		
1(18).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(19).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(2).jpg	Class 1	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2		
1(22).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 2		
1(23).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(26).jpg	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 1	Class 1	Class 2	Class 1	Class 3	Class 4	Class 2	Class 2		
1(28).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(29).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 4	Class 4	Class 1		
1(33).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(4).jpg	Class 1	Class 1	Class 2	Class 1	Class 4	Class 2	Class 2	Class 4	Class 4	Class 4	Class 4	Class 2	Class 2	Class 1	Class 1	Class 2	Class 2	Class 2	Class 4	Class 2	Class 4		
1(43).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(44).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(48).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1		
1(54).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1		
1(57).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(58).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(6).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(60).jpg	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 4	Class 1	Class 2	Class 1		
1(64).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(65).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(68).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 3	Class 3	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1		
1(7).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(70).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1		
1(74).jpg	Class 1	Class 2	Class 2	Class 3	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 4	Class 2		
1(75).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 4	Class 1	Class 3	Class 3	Class 4	Class 1	Class 1	Class 1	Class 2	Class 1	Class 4	Class 1	Class 1		
1(76).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 2	Class 1	Class 1	Class 2	Class 2	Class 2	Class 1	Class 4	Class 1	Class 2	Class 4	Class 4	Class 4	Class 2		
1(8).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1		
1(80).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1		
1(81).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 3	Class 3	Class 3	Class 1	Class 3	Class 3	Class 1	Class 3	Class 3	Class 1	Class 3	Class 3	Class 3	Class 3	Class 3		
1(85).jpg	Class 1	Class 3	Class 2	Class 3	Class 4	Class 1	Class 3	Class 3	Class 4	Class 1	Class 3	Class 3	Class 4	Class 1	Class 4	Class 1	Class 2	Class 1	Class 4	Class 1	Class 1		
1(86).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 3	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1		
1(9).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 4		
1(90).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1		
1(91).jpg	Class 1	Class 1	Class 3	Class 1	Class 1	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 1	Class 3	Class 3	Class 1	Class 3	Class 4	Class 4	Class 4	Class 4		
1(95).jpg	Class 1	Class 1	Class 1	Class 4	Class 2	Class 1	Class 3	Class 2	Class 2	Class 1	Class 2	Class 2	Class 2	Class 4	Class 4	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2		

1(96).jpg	Class 1	Class 2	Class 2	Class 4	Class 1	Class 1	Class 3	Class 1	Class 4	Class 1	Class 3	Class 3	Class 1	Class 1	Class 2	Class 1	Class 2	Class 1	Class 4	Class 1	Class 1
1(97).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 3	Class 1	Class 3	Class 3	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(10).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2
1(13).jpg	Class 2	Class 3	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(14).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(15).jpg	Class 2	Class 3	Class 3	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(16).jpg	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 1	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3
1(20).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 4	Class 2	Class 2
1(24).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(25).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 4	Class 2	Class 4
1(27).jpg	Class 2	Class 3	Class 3	Class 3	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 2	Class 3	Class 3	Class 4	Class 4	Class 3	Class 3
1(3).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(30).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(32).jpg	Class 2	Class 3	Class 2	Class 3	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 3	Class 2	Class 3	Class 4	Class 4	Class 4	Class 4
1(34).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(35).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2
1(36).jpg	Class 2	Class 1	Class 2	Class 3	Class 2	Class 1	Class 2	Class 3	Class 3	Class 2	Class 3	Class 2	Class 3	Class 1	Class 2	Class 1	Class 2	Class 2	Class 4	Class 2	Class 2
1(37).jpg	Class 2	Class 3	Class 2	Class 3	Class 3	Class 3	Class 2	Class 3	Class 2	Class 2	Class 3	Class 3	Class 2	Class 4	Class 2	Class 2	Class 2	Class 3	Class 3	Class 3	Class 3
1(40).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(42).jpg	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 2	Class 2	Class 2	Class 3	Class 3	Class 3	Class 2	Class 3	Class 3	Class 3	Class 2	Class 3
1(45).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(46).jpg	Class 2	Class 3	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(47).jpg	Class 2	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 3	Class 3	Class 3	Class 3	Class 3	Class 4	Class 4	Class 4	Class 4	Class 4	Class 4	Class 4	Class 4
1(49).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4
1(5).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(50).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(53).jpg	Class 2	Class 3	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 4	Class 2
1(55).jpg	Class 2	Class 2	Class 2	Class 2	Class 1	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2
1(56).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(59).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(63).jpg	Class 2	Class 3	Class 2	Class 3	Class 2	Class 3	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 3	Class 4	Class 2	Class 2
1(66).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2
1(67).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(69).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(77).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(78).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(79).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(87).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2
1(88).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(89).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(98).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(99).jpg	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(21).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 2	Class 2	Class 2	Class 3	Class 3	Class 3	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2
1(31).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3

Shallow CNNs Clasification Results

Test Classification	Test Class	googlenet_inception_5b				vgg16_conv5_2				vgg19_conv5_4				resnet50_add_11				resnet50_res5b			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1(1).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 4	Class 4
1(100).jpg	Class 1	Class 3	Class 3	Class 1	Class 1	Class 3	Class 2	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 3	Class 3	Class 1
1(11).jpg	Class 1	Class 3	Class 3	Class 3	Class 1	Class 3	Class 2	Class 1	Class 1	Class 3	Class 2	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 4	Class 2	Class 4	Class 2
1(12).jpg	Class 1	Class 1	Class 1	Class 1	Class 4	Class 4	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(17).jpg	Class 1	Class 4	Class 1	Class 3	Class 4	Class 3	Class 1	Class 3	Class 1	Class 3	Class 1	Class 1	Class 4	Class 1	Class 1	Class 3	Class 1	Class 2	Class 1	Class 2	Class 3
1(18).jpg	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(19).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(2).jpg	Class 1	Class 2	Class 2	Class 1	Class 2	Class 1	Class 3	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 1	Class 2	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2
1(22).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(23).jpg	Class 1	Class 1	Class 1	Class 3	Class 3	Class 1	Class 4	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 4	Class 1	Class 1
1(26).jpg	Class 1	Class 1	Class 1	Class 3	Class 3	Class 1	Class 1	Class 3	Class 1	Class 3	Class 3	Class 1	Class 3	Class 1	Class 1	Class 1	Class 4	Class 3	Class 3	Class 3	Class 4
1(28).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(29).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(33).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(4).jpg	Class 1	Class 1	Class 1	Class 4	Class 4	Class 1	Class 4	Class 4	Class 1	Class 1	Class 4	Class 2	Class 4	Class 4	Class 4	Class 4	Class 4	Class 4	Class 2	Class 2	Class 4
1(43).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(44).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1
1(48).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(54).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(57).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(58).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(6).jpg	Class 1	Class 1	Class 1	Class 1	Class 2	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(60).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 2	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 4	Class 2	Class 1	Class 2	Class 2	Class 4
1(64).jpg	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(65).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(68).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 3	Class 1	Class 3	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(7).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(70).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(74).jpg	Class 1	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 4	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(75).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(76).jpg	Class 1	Class 1	Class 1	Class 4	Class 1	Class 3	Class 3	Class 1	Class 3	Class 3	Class 1	Class 3	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(8).jpg	Class 1	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4
1(80).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(81).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 3	Class 1	Class 3	Class 3	Class 3	Class 3	Class 4	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(85).jpg	Class 1	Class 2	Class 1	Class 2	Class 4	Class 3	Class 3	Class 4	Class 3	Class 3	Class 3	Class 3	Class 2	Class 1	Class 2	Class 1	Class 1	Class 2	Class 1	Class 2	Class 1
1(86).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(9).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 4	Class 1	Class 4	Class 1	Class 4
1(90).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(95).jpg	Class 1	Class 2	Class 2	Class 1	Class 2	Class 4	Class 2	Class 3	Class 1	Class 3	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 2	Class 2	Class 2	Class 1
1(96).jpg	Class 1	Class 2	Class 2	Class 2	Class 1	Class 1	Class 1	Class 3	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 2	Class 1	Class 2	Class 1	Class 2	Class 2	Class 2
1(97).jpg	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1
1(10).jpg	Class 2	Class 4	Class 4	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2	Class 4	Class 2	Class 4	Class 4	Class 4	Class 2
1(13).jpg	Class 2	Class 3	Class 3	Class 2	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2
1(14).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(15).jpg	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 2	Class 2	Class 3	Class 3	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2
1(16).jpg	Class 2	Class 3	Class 3	Class 3	Class 3	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 4	Class 4	Class 3	Class 2	Class 2	Class 2	Class 3
1(20).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2	Class 3
1(24).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 3	Class 3	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(25).jpg	Class 2	Class 2	Class 2	Class 2	Class 3	Class 3	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(27).jpg	Class 2	Class 3	Class 3	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 2	Class 3	Class 3

1(3).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(30).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(32).jpg	Class 2	Class 3	Class 2	Class 3	Class 2	Class 3	Class 3	Class 3	Class 2	Class 2	Class 3	Class 2	Class 4	Class 3	Class 2	Class 2	Class 3	Class 4	Class 2	Class 4	Class 2
1(34).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 1	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(35).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(36).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 1	Class 2	Class 3	Class 1	Class 3	Class 3	Class 2	Class 2	Class 4	Class 4	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2
1(37).jpg	Class 2	Class 3	Class 2	Class 2	Class 3	Class 3	Class 3	Class 3	Class 2	Class 3	Class 2	Class 3	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(40).jpg	Class 2	Class 2	Class 2	Class 2	Class 4	Class 4	Class 2	Class 4	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(42).jpg	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 2	Class 3	Class 4	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(45).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 3	Class 3	Class 3	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(46).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 3
1(47).jpg	Class 2	Class 1	Class 1	Class 1	Class 1	Class 1	Class 1	Class 3	Class 1	Class 3	Class 3	Class 1	Class 3	Class 1	Class 1	Class 1	Class 1	Class 4	Class 4	Class 3	Class 4
1(49).jpg	Class 2	Class 4	Class 2	Class 2	Class 4	Class 4	Class 2	Class 3	Class 2	Class 2	Class 4	Class 2	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4
1(5).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2
1(50).jpg	Class 2	Class 2	Class 2	Class 2	Class 4	Class 4	Class 2	Class 4	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(53).jpg	Class 2	Class 2	Class 2	Class 3	Class 2	Class 3	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2
1(55).jpg	Class 2	Class 2	Class 1	Class 2	Class 1	Class 4	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 4	Class 1	Class 1	Class 2	Class 4	Class 2	Class 2	Class 2	Class 2
1(56).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(59).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(63).jpg	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 2	Class 2	Class 3	Class 4	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2
1(66).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(67).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(69).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(77).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(78).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(79).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(87).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(88).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(89).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(98).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 4	Class 3	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(99).jpg	Class 2	Class 2	Class 2	Class 2	Class 2	Class 3	Class 3	Class 3	Class 2	Class 3	Class 2	Class 2	Class 3	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2	Class 2
1(21).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(31).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(41).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(51).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(52).jpg	Class 3	Class 3	Class 2	Class 3	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(61).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(62).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 4	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(71).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(72).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(73).jpg	Class 3	Class 3	Class 3	Class 3	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 2	Class 2	Class 3	Class 3	Class 3	Class 3	Class 3
1(82).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(83).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(84).jpg	Class 3	Class 4	Class 2	Class 2	Class 2	Class 3	Class 3	Class 3	Class 3	Class 2	Class 2	Class 3	Class 4	Class 2	Class 2	Class 2	Class 2	Class 3	Class 3	Class 3	Class 3
1(92).jpg	Class 3	Class 4	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(93).jpg	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3	Class 3
1(94).jpg	Class 3	Class 3	Class 2	Class 2	Class 2	Class 3	Class 3	Class 3	Class 3	Class 2	Class 2	Class 3	Class 3	Class 2	Class 2	Class 2	Class 2	Class 3	Class 3	Class 3	Class 3
1(38).jpg	Class 4	Class 4	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 1	Class 1	Class 1	Class 1	Class 4	Class 1	Class 4	Class 4	Class 1	Class 4
1(39).jpg	Class 4	Class 2	Class 2	Class 2	Class 2	Class 4	Class 2	Class 4	Class 4	Class 4	Class 2	Class 2	Class 4	Class 2	Class 4	Class 4	Class 4	Class 4	Class 4	Class 4	Class 4

Bibliography

AW Abbas, N Minallh, N Ahmad, SAR Abid, and MAA Khan. K-means and isodata clustering algorithms for landcover classification using remote sensing. *Sindh University Research Journal-SURJ (Science Series)*, 48(2), 2016.

Jwan Al-doski, Shattri B Mansorl, and Helmi Zulhaidi Mohd Shafri. Image classification in remote sensing. *Department of Civil Engineering, Faculty of Engineering, University Putra, Malaysia*, 2013.

Hakan Alphan, Hakan Doygun, and Yüksel I Unlukaplan. Post-classification comparison of land cover using multitemporal landsat and aster imagery: the case of kahramanmaraş, turkey. *Environmental monitoring and assessment*, 151(1-4):327–336, 2009.

Paul Aplin and Peter M Atkinson. Predicting missing field boundaries to increase per-field classification accuracy. *Photogrammetric Engineering & Remote Sensing*, 70(1):141–149, 2004.

Gregory P Asner and Kathleen B Heidebrecht. Spectral unmixing of vegetation, soil and dry carbon cover in arid regions: comparing multispectral and hyperspectral observations. *International Journal of Remote Sensing*, 23(19):3939–3958, 2002.

Amruta A Aware and Kavita Joshi. Crop and weed detection based on texture and size features and automatic spraying of herbicides. *Int J Adv Res Comput Sci Softw Eng*, 6(1), 2016.

M Barnsley. Digital remotely-sensed data and their characteristics. *Geographical Information Systems*, 1:451–466, 1999.

Arwa M Basbrain, Inas Al-Taie, Nassr Azeez, John Q Gan, and Adrian Clark. Shallow convolutional neural network for eyeglasses detection in facial images. In *Computer Science and Electronic Engineering (CEEC), 2017*, pages 157–161. IEEE, 2017.

Miroslav Benčo and Robert Hudec. Novel method for color textures features extraction based on glcm. *Radioengineering*, 16(4):65, 2007.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Janet Carpenter and Leonard Gianessi. Herbicide use on roundup ready crops. *Science*, 287(5454): 803–803, 2000.

Marco Castelluccio, Giovanni Poggi, Carlo Sansone, and Luisa Verdoliva. Land use classification in remote sensing images by convolutional neural networks. *arXiv preprint arXiv:1508.00092*, 2015.

Jyotismita Chaki and Nilanjan Dey. *Texture Feature Extraction Techniques for Image Recognition*. Springer, 2020.

Large Scale Visual Recognition Challenge. Imagenet http://www. image-net. org/challenges. *LSVRC/2012/results. html*, 2012.

Olivier Chapelle, Patrick Haffner, and Vladimir N Vapnik. Support vector machines for histogram-based image classification. *IEEE transactions on Neural Networks*, 10(5):1055–1064, 1999.

Narendra S Chaudhari, Aruna Tiwari, and Anuradha Purohit. Genetic programming for classification. *Int. J. Comput. Electron. Eng. IJCEE*, 1:69–76, 2009.

D Chen and Douglas Stow. The effect of training strategies on supervised classification at different spatial resolutions. *Photogrammetric Engineering and Remote Sensing*, 68(11):1155–1162, 2002.

Yu-Sheng Chen, Guangjun Su, and Haihong Li. Machine learning for calligraphy styles recognition.

Yu-Min Chianga and Jo-Mien Weng. A texture classification method based on gray-level co-occurrence matrix and support vector machines. *European International Journal of Science and Technology*, 2(10): 77–79, 2013.

Ana M Cingolani, Daniel Renison, Marcelo R Zak, and Marcelo R Cabido. Mapping vegetation in a heterogeneous mountain rangeland using landsat data: an alternative method to define and classify land-cover units. *Remote sensing of environment*, 92(1):84–97, 2004.

D Richard Cutler, Thomas C Edwards Jr, Karen H Beard, Adele Cutler, Kyle T Hess, Jacob Gibson, and Joshua J Lawler. Random forests for classification in ecology. *Ecology*, 88(11):2783–2792, 2007.

Bashir Rokni Deilami, Baharin Bin Ahmad, Malik RA Saffar, and Hafiz Aminu Umar. Using remote sensing and gis to detect and monitor land use and land cover change in iskandar malaysia during 2007–2014. *Middle-East Journal of Scientific Research*, 22(3):390–394, 2014.

Riya Desai et al. Removal of weeds using image processing. *IJACT*, 2016.

Ashraf M Dewan and Yasushi Yamaguchi. Using remote sensing and gis to detect and monitor land use and land cover change in dhaka metropolitan of bangladesh during 1960–2005. *Environmental monitoring and assessment*, 150(1-4):237, 2009.

Manisha Dhongade. Survey on texture classification methods. *International Journal of Computer Science Trends and Te chnology (I JCST)*, 3(5), 2015.

Harris Drucker, Christopher JC Burges, Linda Kaufman, Alex J Smola, and Vladimir Vapnik. Support vector regression machines. In *Advances in neural information processing systems*, pages 155–161, 1997.

Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*, 2016.

Rishiraj Dutta. Review of normalized difference vegetation index (ndvi) as an indicator of drought. 2016.

Pedro G Espejo, Sebastián Ventura, and Francisco Herrera. A survey on the application of genetic programming to classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(2):121–144, 2009.

Essex. The jasmine vision system builder, 2017. URL `http://vase.essex.ac.uk/software/jasmi`

Giles M Foody and Ajay Mathur. A relative evaluation of multiclass image classification by support vector machines. *IEEE Transactions on geoscience and remote sensing*, 42(6):1335–1343, 2004.

GIS Geography. Image classification techniques in remote sensing.

Pall Oskar Gislason, Jon Atli Benediktsson, and Johannes R Sveinsson. Random forests for land cover classification. *Pattern Recognition Letters*, 27(4):294–300, 2006.

Jian Guo, Jixian Zhang, Yonghong Zhang, and Yinxuan Cao. Study on the comparison of the land cover classification for multitemporal modis images. In *2008 International Workshop on Earth Observation and Remote Sensing Applications*, pages 1–6. IEEE, 2008.

Robert M Haralick, Karthikeyan Shanmugam, et al. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6):610–621, 1973.

Adam W Harley. An interactive node-link visualization of convolutional neural networks. In *International Symposium on Visual Computing*, pages 867–877. Springer, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Laurence Hubert-Moy, Adeline Cotonnec, L Le Du, Anabelle Chardin, and Patrick Pérez. A comparison

of parametric classification procedures of remotely sensed data applied on different landscape units. *Remote Sensing of Environment*, 75(2):174–187, 2001.

Gordon Hughes. On the mean accuracy of statistical pattern recognizers. *IEEE transactions on information theory*, 14(1):55–63, 1968.

Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.

Lakshmi N Kantakumar and Priti Neelamsetti. Multi-temporal land use classification using hybrid approach. *The Egyptian Journal of Remote Sensing and Space Science*, 18(2):289–295, 2015.

Nadia Kanwal, Erkan Bostanci, and Adrian F Clark. Evaluation method, dataset size or dataset content: how to evaluate algorithms for image matching? *Journal of Mathematical Imaging and Vision*, 55(3): 378–400, 2016.

Amandeep Kaur and Savita Gupta. Texture classification based on gabor wavelets. *International Journal of Research in Computer Science*, 2(4):39, 2012.

John R Koza, David Andre, Martin A Keane, and Forrest H Bennett III. *Genetic programming III: Darwinian invention and problem solving*, volume 3. Morgan Kaufmann, 1999.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

William P Kustas, Andrew N French, Jerry L Hatfield, Tom J Jackson, M Susan Moran, Al Rango, Jerry C Ritchie, and Tom J Schmugge. Remote sensing research in hydrometeorology. *Photogrammetric Engineering & Remote Sensing*, 69(6):631–646, 2003.

Kentaro Kuwata and Ryosuke Shibasaki. Estimating crop yields with deep learning and remotely sensed

data. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 858–861. IEEE, 2015.

David A Landgrebe. *Signal theory methods in multispectral remote sensing*, volume 29. John Wiley & Sons, 2005.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Miao Li, Shuying Zang, Bing Zhang, Shanshan Li, and Changshan Wu. A review of remote sensing image classification techniques: The role of spatio-contextual information. *European Journal of Remote Sensing*, 47(1):389–411, 2014.

Andy Liaw, Matthew Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

Thomas Lillesand, Ralph W Kiefer, and Jonathan Chipman. *Remote sensing and image interpretation*. John Wiley & Sons, 2014.

Guang-Hai Liu, Lei Zhang, Ying-Kun Hou, Zuo-Yong Li, and Jing-Yu Yang. Image retrieval based on multi-texton histogram. *Pattern Recognition*, 43(7):2380–2389, 2010.

Thomas Loveard and Victor Ciesielski. Representing classification problems in genetic programming. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, volume 2, pages 1070–1077. IEEE, 2001.

Dengsheng Lu and Qihao Weng. A survey of image classification methods and techniques for improving classification performance. *International journal of Remote sensing*, 28(5):823–870, 2007.

Sean Luke. Essentials of metaheuristics. lulu, 2013. *Available for free at http://cs. gmu. edu/˜ sean/book/metaheuristics*, 2013.

Florian Markowetz, Lutz Edler, and Martin Vingron. Support vector machines for protein fold class prediction. *Biometrical Journal*, 45(3):377–389, 2003.

Judit Martínez and Federico Thomas. Efficient computation of local geometric moments. *IEEE Transactions on Image Processing*, 11(9):1102–1111, 2002.

Pabitra Mitra, B Uma Shankar, and Sankar K Pal. Segmentation of multispectral remote sensing images using active support vector machines. *Pattern recognition letters*, 25(9):1067–1074, 2004.

P Mohanaiah, P Sathyanarayana, and L GuruKumar. Image texture feature extraction using glcm approach. *International Journal of Scientific and Research Publications*, 3(5):1, 2013.

Jana Müllerová. Use of digital aerial photography for sub-alpine vegetation mapping: A case study from the krkonoše mts., czech republic. *Plant Ecology*, 175(2):259–272, 2005.

ALISTAIR Murdoch, CARL Flint, ROBERT Pilgrim, PN de la Warr, J Camp, B Knight, P Lutman, B Magri, P Miller, T Robinson, et al. Eyeweed: automating mapping of black-grass (alopecurus myosuroides) for more precise applications of pre-and post-emergence herbicides and detecting potential herbicide resistance. *Aspects of Applied Biology*, 127:151–158, 2014.

Soe Win Myint. A robust texture analysis and classification approach for urban land-use and land-cover feature discrimination. *Geocarto international*, 16(4):29–40, 2001.

Siddhartha Sankar Nath, Girish Mishra, Jajnyaseni Kar, Sayan Chakraborty, and Nilanjan Dey. A survey of image classification methods and techniques. In *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, pages 554–557. IEEE, 2014.

Hamideh Nouri, Simon Beecham, Sharolyn Anderson, and Pamela Nagler. High spatial resolution worldview-2 imagery for mapping ndvi and its relationship to temporal urban landscape evapotranspiration factors. *Remote sensing*, 6(1):580–602, 2014.

Olly Oechsle. *Towards the automatic construction of machine vision systems using genetic programming*. PhD thesis, University of Essex, 2009.

Gregory S Okin, Dar A Roberts, Bruce Murray, and William J Okin. Practical limits on hyperspectral vegetation discrimination in arid and semiarid environments. *Remote Sensing of Environment*, 77(2): 212–225, 2001.

Ajinkya Paikekari, Vrushali Ghule, Rani Meshram, and VB Raskar. Weed detection using image processing. *International Research Journal of Engineering and Technology (IRJET)*, 3(3):1220–1222, 2016.

Mahesh Pal and PM Mather. Support vector machines for classification in remote sensing. *International Journal of Remote Sensing*, 26(5):1007–1011, 2005.

JO Payero, CMU Neale, and JL Wright. Comparison of eleven vegetation indices for estimating plant height of alfalfa and grass. *Applied Engineering in Agriculture*, 20(3):385, 2004.

P Pellika, W Gareth Rees, and P Pellikka. Glacier parameters monitored using remote sensing. In *Remote sensing of glaciers*, pages 41–62. CRC Press, 2010.

Petri Pellikka and W Gareth Rees. *Remote sensing of glaciers: techniques for topographic, spatial and thematic mapping of glaciers*. CRC Press, 2009.

Rutherford V Platt and Alexander FH Goetz. A comparison of aviris and landsat for land use classification at the urban fringe. *Photogrammetric Engineering & Remote Sensing*, 70(7):813–819, 2004.

Riccardo Poli, William B Langdon, Nicholas F McPhee, and John R Koza. *A field guide to genetic programming*. Lulu. com, 2008.

Kevin P Price, Xulin Guo, and James M Stiles. Optimal landsat tm band combinations and vegetation indices for discrimination of six grassland types in eastern kansas. *International Journal of Remote Sensing*, 23(23):5031–5042, 2002.

Maryam Rahnemoonfar and Clay Sheppard. Real-time yield estimation based on deep learning. In *Autonomous Air and Ground Sensing Systems for Agricultural Optimization and Phenotyping II*, volume 10218, page 1021809. International Society for Optics and Photonics, 2017.

W Gareth Rees and P Pellika. Principles of remote sensing. *Remote Sensing of Glaciers. London*, 2010.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

Victor Francisco Rodriguez-Galiano, Bardan Ghimire, John Rogan, Mario Chica-Olmo, and Juan Pedro Rigol-Sanchez. An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 67:93–104, 2012.

Lior Rokach and Oded Maimon. Top-down induction of decision trees classifiers-a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35(4):476–487, 2005.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Thair A Salh and Mustafa Z Nayef. Face recognition system based on wavelet, pca-lda and svm. *Computer Engineering and Intelligent Systems Journal*, 4(3), 2013.

JP Sansao, MS Jnior, Leonardo A Mozelli, FA Pinto, and Daniel M Queiroz. Weed mapping using digital images. In *Proceedings of International Conference on Agricultural Engineering-CIGR*, number 1931, 2012.

Thomas J Schmugge, William P Kustas, Jerry C Ritchie, Thomas J Jackson, and Al Rango. Remote sensing in hydrology. *Advances in water resources*, 25(8-12):1367–1385, 2002.

Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 806–813, 2014.

SheffieldUniversity. Overuse of herbicides costing uk economy Âč400 million a year, 2019. URL `https://www.agriland.co.uk/farming-news/overuse-of-herbicides-costing-`

Arnold WM Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on pattern analysis and machine intelligence*, 22(12):1349–1380, 2000.

Demetris Stathakis and Athanassios Vasilakos. Comparison of computational intelligence based classification techniques for remotely sensed optical image classification. *IEEE Transactions on Geoscience and remote Sensing*, 44(8):2305–2318, 2006.

Kim Arild Steen, Peter Christiansen, Henrik Karstoft, and Rasmus Nyholm Jørgensen. Using deep learning to challenge safety standard for highly autonomous machines in agriculture. *Journal of Imaging*, 2(1):6, 2016.

SR Suralkar, AH Karode, Priti W Pawade, et al. Texture image classification using support vector machine. *International Journal of Computer Applications in Technology*, 3(1):71–75, 2012.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

Du-Ming Tsai and Tse-Yun Huang. Automated surface inspection for statistical textures. *Image and Vision computing*, 21(4):307–323, 2003.

Jayaram K. Udupa, Punam K. Saha, and Roberto de Alencar Lotufo. Disclaimer:" relative fuzzy con-

nectedness and object definition: theory, algorithms, and applications in image segmentation". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(11):I–1500, 2002.

Robert C Weih and Norman D Riggan. Object-based classification vs. pixel-based classification: Comparative importance of multi-resolution imagery. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38(4):C7, 2010.

Jason Weston. Support vector machine (and statistical learning theory) tutorial. *NEC Labs America*, 4, 1998.

Michael A Wulder and Steven E Franklin. *Remote sensing of forest environments: concepts and case studies*. Springer Science & Business Media, 2012.

Kecong Xiao, Zishuai Zhang, and Jun Wu. Chinese text sentiment analysis based on improved convolutional neural networks. In *Software Engineering and Service Science (ICSESS), 2016 7th IEEE International Conference on*, pages 922–926. IEEE, 2016.

Panitnat Yimyam. *Agricultural Produce Grading by Computer Vision Based on Genetic Programming*. PhD thesis, University of Essex, 2015.

Rafat Zaki, Abotalib Zaki, Saad Ahmed, et al. Land use and land cover changes in arid region: The case new urbanized zone, northeast cairo, egypt. *Journal of Geographic Information System*, 3(03):173, 2011.

Jianguo Zhang and Tieniu Tan. Brief review of invariant texture analysis methods. *Pattern recognition*, 35(3):735–747, 2002.

Guoying Zhao and Matti Pietikainen. Local binary pattern descriptors for dynamic texture recognition. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 2, pages 211–214. IEEE, 2006.

Nitish Zulpe and Vrushsen Pawar. Glcm textural features for brain tumor classification. *IJCSI International Journal of Computer Science Issues*, 9(3):354–359, 2012.