Automated Detection and Monitoring of Vegetation Through Deep Learning

Thesis submitted in fulfilment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

The College of Engineering and Science

Institute of Sustainable Industries and Liveable Cities (ISILC)

Victoria University, Melbourne, Australia

by

ASIM KHAN

Master of Information Technology (Software Engineering) (M.I.T)

Master of Information Systems Management (MISM)

Graduate Certificate in Tertiary Education (GCTE)

Automated Detection and Monitoring of Vegetation Through Deep Learning

by

ASIM KHAN

Master of Information Technology (Software Engineering) (M.I.T)

Master of Information Systems Management (MISM)

Graduate Certificate in Tertiary Education (GCTE)

Supervisory Committee:

Dr. Randall W. Robinson., (Principal Supervisor)

The College of Engineering and Science.,

Institute of Sustainable Industries and Liveable Cities (ISILC).

Dr. Anwaar Ulhaq., (Associate Supervisor)

The College of Engineering and Science.,

Institute of Sustainable Industries and Liveable Cities (ISILC).

Abstract

Healthy vegetation are essential not just for environmental sustainability but also for the development of sustainable and liveable cities. It is undeniable that human activities are altering the vegetation landscape, with harmful implications for the climate. As a result, autonomous detection, health evaluation, and continual monitoring of the plants are required to ensure environmental sustainability. This thesis presents research on autonomous vegetation management using recent advances in deep learning.

Currently, most towns do not have a system in place for detection and continual vegetation monitoring. On the one hand, a lack of public knowledge and political will could be a factor; on the other hand, no efficient and cost-effective technique of monitoring vegetation health has been established. Individual plants health condition data is essential since urban trees often develop as stand-alone objects. Manual annotation of these individual trees is a time-consuming, expensive, and inefficient operation that is normally done in person. As a result, skilled manual annotation cannot cover broad areas, and the data they create is out of date.

However, autonomous vegetation management poses a number of challenges due to its multidisciplinary nature. It includes automated detection, health assessment, and monitoring of vegetation and trees by integrating techniques from computer vision, machine learning, and remote sensing. Other challenges include a lack of analysis-ready data and imaging diversity, as well as dealing with their dependence on weather variability. With a core focus on automation of vegetation management using deep learning and transfer learning, this thesis contributes novel techniques for Multi-view vegetation detection, robust calculation of vegetation index, and real-time vegetation health assessment using deep convolutional neural networks (CNNs) and deep learning frameworks.

The thesis focuses on four general aspects: a) training CNN with possibly inaccurate labels and noisy image dataset; b) deriving semantic vegetation segmentation from the ordinal information contained in the image; c) retrieving semantic vegetation indexes from street-level imagery; and d) developing a vegetation health assessment and monitoring system.

Firstly, it is essential to detect and segment the vegetation, and then calculate the pixel value of the semantic vegetation index. However, because the images in multi-sensory data are not identical, all image datasets must be registered before being fed

into the model training. The dataset used for vegetation detection and segmentation was acquired from multi-sensors. The whole dataset was multi-temporal based; therefore, it was registered using deep affine features through a convolutional neural network. Secondly, after preparing the dataset, vegetation was segmented by using Deep CNN, a fully convolutional network, and U-net. Although the vegetation index interprets the health of a particular area's vegetation when assessing small and large vegetation (trees, shrubs, grass, etc.), the health of large plants, such as trees, is determined by steam. In contrast, small plants' leaves are evaluated to decide whether they are healthy or unhealthy. Therefore, initially, small plant health was assessed through their leaves by training a deep neural network and integrating that trained model into an internet of things (IoT) device such as AWS DeepLens. Another deep CNN was trained to assess the health of large plants and trees like Eucalyptus. This one could also tell which trees were healthy and which ones were unhealthy, as well as their geo-location. Thus, we may ultimately analyse the vegetation's health in terms of the vegetation index throughout time on the basis of a semantic-based vegetation index and compute the index in a time-series fashion.

This thesis shows that computer vision, deep learning and remote sensing approaches can be used to process street-level imagery in different places and cities, to help manage urban forests in new ways, such as biomass-surveillance and remote vegetation monitoring.

Student Declaration

I, *Asim Khan*, declare that this thesis titled *"Automated Detection and Monitoring of Vegetation through Deep Learning"*, is no more than 100,000 words in length, including quotes and exclusive of tables, figures, appendices, bibliography, references, and footnotes. This thesis contains no material that has been submitted and accepted previously, in whole or in part, for the award of any other academic degree or diploma. Except where reference is made in the text of the thesis, this thesis is my own work". No other person's work has been used without due acknowledgment in the main text of the thesis.

I have conducted my research in alignment with the *Australian Code for the Responsible Conduct of Research* and *Victoria University's Higher Degree by Research Policy and Procedures*.

Signed: _____

Date: 11-03-2022

Contents

Publications and Presentations

This thesis includes work by the author that has been published or accepted for publication in the international journals and conferences. These publications are the own work of the author of this thesis, and the author has the permission of the publishers to reproduce the contents of these publications for academic purposes.

In particular, some data, ideas, opinions and figures presented in this thesis have previously appeared or may appear shortly after the submission of this thesis as follows:

1. **Real-time Plant Health assessment via Implementing Cloud-based Scalable Transfer Learning on AWS DeepLens**.
   Khan, A., Nawaz, U., Ulhaq, A., & Robinson, R. W. (2020). *"Real-time plant health assessment via implementing cloud-based scalable transfer learning on AWS DeepLens"*. Plos one, 15(12), e0243243. doi: https://doi.org/10.1371/journal.pone.0243243.

2. **Health Assessment of Eucalyptus Trees Using Siamese Network from Google Street and Ground Truth Images**.
   Khan, A., Asim, W., Ulhaq, A., Ghazi, B., & Robinson, R. W. (2021). *"Health Assessment of Eucalyptus Trees Using Siamese Network from Google Street and Ground Truth Images"*. Remote Sensing, 13(11), 2194. doi: https://doi.org/10.3390/rs13112194.

3. **A Multiview Semantic Vegetation Index for Robust Estimation of Urban Vegetation Cover**.
   Khan, A., Asim, W., Ulhaq, A., & Robinson, R. W. (2022). *"A Multiview Semantic Vegetation Index for Robust Estimation of Urban Vegetation Cover"*. Remote Sensing, 14(1), 228. doi: https://doi.org/10.3390/rs14010228.

4. **A Deep Semantic Vegetation Health Monitoring Platform For Citizen Science Imaging Data**.
   Khan, A.; Asim, W.; Ulhaq, A.; Robinson, R.W. *"A Deep Semantic Vegetation Health Monitoring Platform For Citizen Science Imaging Data"*.
   Under Production with PLOS ONE Journal.

5. **Multi-temporal registration of environmental imagery using affine invariant convolutional features**.
   Khan, A., Ulhaq, A., & Robinson, R. W. (2019, November). *"Multi-temporal registration of environmental imagery using affine invariant convolutional features*. In Pacific-Rim Symposium on Image and Video Technology (pp. 269-280). Springer, Cham. doi: https://doi.org/10.1007/978-3-030-34879-3_21.

6. **Detection of vegetation in environmental repeat photography: a new algorithmic approach in data science**.
   Khan, A., Ulhaq, A., Robinson, R., & Rehman, M. U. (2020). *"Detection of vegetation in environmental repeat photography: a new algorithmic approach in data science"*. In Statistics for Data Science and Policy Analysis (pp. 145-157). Springer, Singapore. doi: https://doi.org/10.1007/978-981-15-1735-8_11.

Other Presentations

1. Research Presentation during VU Open Day Program 2018
2. ISILC 2021 HDR Student Conference
3. Three (3) Minute Thesis

List of Figures

List of Tables

Acknowledgements

First and foremost, I would like to express my sincere gratitude to Allah for His countless blessings upon me, which gave me the patience, strength, and understanding to successfully complete this PhD.

I owe my heartiest gratitude to all those who have made this thesis come to fruition. To begin, I want to express my gratitude to my principal supervisor, Professor Randall W. Robinson, who has served as an incredible mentor. I would like to express my gratitude to him for his patience and encouragement throughout my PhD journey. I am also indebted to Dr. Anwaar Ulhaq, my associate supervisor, for his constant guidance and support. His thoughts and helpful suggestions on how to handle the important areas of this research work are greatly appreciated. That's why I was able to publish my three (3) research articles in Q-1 ranked journals and one (1) research paper in a B-ranked conference. This thesis would not be possible without his assistance.

Professor Ron Adams, Senior Lecturer Dr. Rose Lucas, and Associate Professor Dr. Deborah Zion helped me out when I was stuck by sharing their expertise and knowledge. Their invaluable comments and assistance helped me get through my PhD programme.

Throughout my PhD journey, I am grateful to Victoria University (VU) for providing me support through the Research Training Program from the Australian Government Department of Education, Skills and Employment. I want to thank all the VU staff members with whom I have engaged. You've all been extremely helpful, professional, and above all, fantastic in everything you've done to make our university lives as easy as possible by providing us with all of the services and assistance we've requested. Special thanks go to Elizabeth Smith for her constant reminders about the PhD milestones, Cameron Barrie for library services, Jo Xuereb, Palmina Fichera, and Meika Scholz for administrative assistance.

I would also like to express my gratitude to all of my PhD colleagues and friends who supported and encouraged me throughout my studies. Warda Asim, Bilal Ghazi, Umair Nawaz, Ravinder Singh, Neda Afazaliseresht, and Dinesh Pandey for their unwavering encouragement and support. Talking to these guys and learning from their experiences helped me get back on track on days when I felt shallow and stuck.

Finally, my parents and parents-in-law deserve special thanks for their prayers, love, sacrifices, and encouragement. I would like to express my deepest gratitude

to my very patient wife for her unwavering support and unconditional love; she has made several sacrifices and has always encouraged me along this path.

Dedications

This thesis is dedicated to my parents, wife, and children for their love, encouragement and endless support.

Acronyms

CNN: Convolutional Neural Network

APERS: Affine Parameters Estimation by Random Sampling

AC: Alternating Component

AWS: ANN: Amazon Web Services

ANN: Artificial Neural Network

CBM: Community Based Monitoring

CNN: Convolutional Neural Networks

CR: Crown Radius

DCNN: Deep Convolutional Neural Network

DCDM: DeepLens Classification and Detection Model

DC: Direct Component

DAG-RNN: Directed Acyclic Graphic RNN

FN: False Negatives

FP: False Positives

FPGA: Field-Programmable Gate Array

FFDI: Forest Fire Danger Index

FW: Full-Waveform

FCN: Fully Convolutional Neural Network

GSV: Google Street View

GPU: Graphic Processing Unit

GVI: Green View Index

HD: High Definition

HGVI: Horizontal Green View Index

IoT: Internet of Things

IoU: Intersection Over Union

LOB: Line-Of-Bearing

LRN: Local Response Normalization Layer

ML: Machine Learning

MAD: Mean Absolute Distance

MioU: Mean Intersection Over Union

MED: Median Of Distance

MSVI: Multiview Semantic Vegetation Index

NIR: Near-Infrared Reflectance

NDVI: Normalized Difference Vegetation Index

OA: Overall Accuracy

PSIVT: Pacific-Rim Symposium on Image and Video Technology

PA: Pixel Accuracy

RMSD: Root Mean Squared Distance

SVI: Semantic Vegetation Index

SVM: Support Vector Machine

SCNN: Siamese Convolutional Neural Network

SCGF: Spatially Constrained Gaussian Fields

STD: Standard Deviation of Distance

SGD: Stochastic Gradient Descent

S.I.R.I: Structured-Illumination Reflectance Imaging

TL: Transfer Learning

TN: True Negatives

TP: True Positives

UAV: Unmanned Aerial Vehicle

VGG: Visual Geometry Group

Chapter 1

Introduction

> Experiments upon vegetation give reason to believe that light combines with certain parts of vegetables, and that the green of their leaves, and the various colors of flowers, is chiefly owing to this combination.
>
> *Antoine Lavoisier*

Natural greenery and healthy vegetation are essential characteristics of the urban environment, which offers various advantages, such as improved air quality, human health facilities, storm-water runoff control, carbon reduction, and increased property values. It is widely known that vegetation in urban environments provides a wide range of ecosystem services to the surrounding environment. They improve the quality of the environment, alleviate the adverse effects of human presence, enhance the anthropic environment, and help people identify with their cultural heritage. Plants in general, and trees in particular, cannot be considered to be independent of urban activities and infrastructure, as is often assumed to be true. By providing ecosystem services, urban vegetation can improve the overall quality of life, and it has been claimed that it can help to minimise the negative effects of global warming

on human health. When it comes to Australia, the country is the sixth-largest in terms of land area, with natural vegetation, forests, and woods covering 16% of the country's total land area. Federal and state governments undertake a variety of initiatives to provide people with access to clean, fresh air while also contributing significantly to global warming reduction efforts [4].

Environmental monitoring is now considered a critical task in order to evaluate the effectiveness of environmental policies. When it comes to environmental monitoring, one of the most important aspects is to keep an eye out for information about vegetation, which is essential for predicting ongoing trends at an early stage. In order to preserve and conserve natural resources such as green areas, it is necessary to conduct continual vegetation identification and monitoring throughout the year. Vegetation changes in the urban environment have been shown to be significantly associated with changes in the kind of land cover (for example, building developments). It is necessary to document the change in vegetation in order for land management professionals to work in a good way towards improving the urban environment. Because urban trees are often grown as stand-alone objects, it is necessary to collect data on individual tree health conditions. However, manual annotation of these one-of-a-kind trees is a time-consuming and expensive operation. Thus, manual annotation by specialists is not scalable to wide areas, and the generated data is not up-to-date in a timely manner. This is accomplished through a variety of methods, including the use of drones and UAVs, satellites, and remote sensing, with the assistance of land-care groups, environmental groups, indigenous organisations, and local councils [4].

The effects of plant disease on quantitative and qualitative production [5] are devastating, resulting in a striking blow to farmers, traders, and consumers. Traditionally, farmers detect and diagnose plant diseases through their observations and

rely upon the opinions of local experts and their past experiences. An expert can determine whether or not a plant is healthy [6]. If a plant is found unhealthy, noticeable symptoms on its leaves, stems and fruits are observed and reported. Plant disease diagnosis incorporates a substantially high degree of difficulty through visual examination of the signs on plant leaves. Because of this challenge and the significant number of grown plants and their existing phytopathological issues, even qualified agronomists and plant pathologists sometimes struggle to accurately identify particular diseases. They are consequently driven to wrong assumptions and remedies [7]. Practical plant health assessment and disease diagnosis can improve product quality and prevent production loss. Early detection and classification of crop diseases are significant to securing specific species' production [8]. When a plant gets infected by a particular disease, substantial symptoms are shown on the leaves, which help identify and classify that disease [9]. It is therefore essential to control and assess disease spread [10].

The study of vegetation segmentation, detection, and health assessment is not limited to a single field of science; it is a topic of research in a variety of fields such as computer vision and artificial intelligence (AI). Machine vision is used to examine vegetation segmentation, detection, and health assessment. With the advancement in computer vision, researchers are equipped enough to develop the algorithms to have an automated system for vegetation monitoring.

As in ecology, the vegetation index of some particular sites gives a lot of information regarding their environment. Estimating vegetation cover and biomass is commonly done by calculating various vegetation indexes for automated urban vegetation management and monitoring. However, most of these indexes fail to capture robust estimation of vegetation cover due to their inherent focus on colour attributes, with limited viewpoint and ignoring seasoning variations. It is critical to document

changes in vegetation so that land management professionals may work to improve the urban environment. It turns out that changes in the type of land cover (like building developments) have a big impact on how plants change around cities. However, existing approaches have been highly focused on spectral analysis and colour variations. For instance, the Normalized Difference Vegetation Index (NDVI) tends to amplify atmospheric noise in the near-infrared reflectance (NIR) and red bands and becomes very sensitive to background variation. Therefore, it does not work well for RGB images for street-level vegetation analysis. RGB-based vegetation indexes, on the other hand, aren't very good at predicting how much vegetation there is because they only look at green colours and don't take into account seasonal changes.

Moreover, human observation's cost, time, and logistics limit many studies, yet ecologists have only begun to use automated tools. They are increasingly relying on diverse datasets, from air-borne photographs to deep-sea videos. Without using computer vision and deep learning techniques, their ability to manage and analyse these datasets is limited. Multisensory data, on the other hand, has images that aren't the same. All image datasets must be registered before they can be used in training a model.

Machine learning (ML) [11] algorithms serve a lot in classifying and identifying vegetation. ML helps monitor the health assessment of plants and predict diseases in the plants at early stages [9]. New ML models have evolved over time, such as SVM [12], VGG architectures [13], R-FCN [14], Faster R-CNN [15], SDD [16] and many others. The researchers used them for their experiments in recognising and classifying images. Some of those are used in the automation of agriculture systems [17]. Computer vision is the scientific subject that deals with these areas of vegetation management analysis study.

Deep Learning (DL) is a relatively new subject in machine learning that has the

advantage of automatically extracting intermediate feature representations from raw textual input by building a hierarchical structure [119]. In the field of remote sensing, deep learning-based image segmentation has been effectively employed to segment satellite images, including strategies for urban planning and precision agriculture. Photos taken by drones (UAVs) have also been split up using Deep Learning-based algorithms, which can help solve major environmental problems caused by climate change.

Computer vision is a sub-field of artificial intelligence (AI) concerned with the development of visual perception techniques for computers, which have become an integral aspect of modern life. Computer vision encompasses image processing, machine vision, and pattern analysis, all of which are closely linked research areas. The breadth of approaches and applications they cover overlaps significantly. This means that the fundamental approaches utilised and developed in these domains are nearly the same, with minor changes. Success in one sector encourages success in the other.

This thesis uses a multidisciplinary approach to automate vegetation detection and monitoring. This thesis studies various vegetation traits as a research challenge in computer vision and video understanding. These in-depth traits and characteristics can aid in the development of computer vision algorithms that can assist in separating plants and determining their health. A core focus of this thesis is the use of semantic segmentation, as robust segmentation of urban vegetation accurately and efficiently is critical. When it comes to vegetation management, targeted control and elimination of undesired vegetation, ranging from weeds and shrubs to branches and trees, are used to achieve the desired result. Semantic segmentation assigns a set of object kinds to each image pixel (for example, people, trees, sky, and cars). Pixel-level labelling is the term for this procedure.An image classification task, which predicts a single label for the whole image or frame, is often a more difficult one.

This is because it often takes more time and effort.

The following sections describe motivations, aims , objectives, contributions and organisation of the thesis.

1.1 Motivation

The adoption of green infrastructure schemes is transforming the urban vegetation in many places. These are multibillion-dollar programmes [18] that finance or incentivize the installation of green infrastructure (such as parks, bioswales, street trees, and rain gardens) or the replacement of impervious surfaces (such as asphalt) with pervious ones (e.g., grass). Despite their size and cost, however, programmes are rarely empirically evaluated due to data quality and attribution issues. This research investigates whether very high-resolution remotely sensed aerial imagery can be used to track patterns of urban greening across a metropolis, including differences in dynamics between public and private areas.

Our passions and intended outcomes increase our self-esteem and motivation. The motivation for this thesis' research is based on the artefact of interest to the work and the benefits of the linked research activity. The following issues prompted us to investigate and conduct research in the domain of vegetation segmentation and health assessment:

1. Vegetation segmentation and health is a hot topic in computer vision research [19]. Computer vision experts have solved the majority of low and mid-level vision difficulties, and they are currently focused on higher-level vision challenges. The Semantic Vegetation Index is a high-level computer vision problem that encompasses computer vision's knowledge, scope, accomplishments, and challenges. On the one hand, the answers to this challenge are

based on low and mid-level vision solutions, but they can also assist in the achievement of higher-level computer vision goals.

2. The utility and applicability of a new area in tackling real-world problems is an essential personal motivation for learning it. The applications and applicability of automated vegetation detection and health monitoring are extensive. The Semantic Vegetation Index and Monitoring can be used for a variety of purposes. Automated vegetation index computation, plant health evaluation, and tree geo-location identification are just a few of the application areas. One of the most important motivators for studies on vegetation analysis, detection, understanding, and recognition is the application domains. In addition, computer vision experts are sorting and suggesting several more use areas.

3. Other disciplines of computer vision recognise the issues that robust multi-temporal picture registration, semantic vegetation index, and health assessment encounter. If we can develop solutions to these issues and problems, they will be beneficial to other computer-related professions. The major issues of dealing with noise, occlusion, temporal variations, feature extraction, robust matching, avoiding tracking, intra-class variation handling, and correct classification methods are all well-known in the computer vision, image processing, and machine learning research fields. During the development of their solutions, semantic vegetation based index calculation, multi-sensory data registration, and vegetation health assessment all encounter similar obstacles. As a result, computer vision and research on vegetation segmentation and monitoring helps other parts of computer vision and research.

4. Several approaches to automated semantic vegetation segmentation and health monitoring have been developed in recent years. The accuracy and complex-

ity of the proposed methodologies vary. Despite these answers, our research correctly identifies a number of research gaps, which pushes us to continue working in this area. In this thesis, we attempted to fill in these research gaps with suggestions and answers to these issues. The next section will go into the specifics of these research gaps, as well as our role in suggesting fresh solutions. However, these research gaps are related to the unexplored or under-utilised exploitation and application of significant deep visual elements in video sequences and Google Street View imagery for automated health monitoring and semantic vegetation index computation.

1.2 Aims and Objectives

This thesis aims to answer the following research questions:

1. How are multitemporal imagery such as Fluker post project dataset can be normalized to use it for convolutional neural network training? **(Chapters 2, 3 and 5)**

2. How to estimate vegetation segmentation and evaluate health using the ordinal information provided in image labels for vegetation? **(Chapters 3, 4 and 5)**

3. How are semantic vegetation index extracted using deep CNNs from street level imagery such as Google Street view image dataset and Fluker post? **(Chapters 6 and 7)**

4. How to develop a health assessment and monitoring system for vegetation using deep learning? **(Chapters 4, 5 and 7)**

The integration of the main and sub-research questions in the relevant chapters is illustrated in Figure 1.1.

Figure 1.1: The organizational chart of main and sub research questions.

1.3 Scope and Contribution

1.3.1 Scope

The main goal of this thesis focuses on semantic vegetation index estimation from RGB imagery, vegetation health assessment and monitoring. The main objective is split into interrelated sub-objectives for developing an automated detection system for vegetation, health assessment and monitoring systems through deep learning.

First and foremost, it is necessary to detect and segment the vegetation, after which the pixel value of the semantic vegetation index may be calculated. As a result, because the images in multisensory data are not identical, all image datasets must be registered before they can be used in model training. Secondly, the vegetation index analyses the health of a certain area's vegetation. When assessing small and large vegetation (trees, bushes, grass, and so on), the health of large plants, such as trees,

is evaluated by their stems, while the health of small plants, on the other hand, is evaluated by their leaves to determine whether they are healthy or unhealthy. As a result, we will be able to assess the health of the vegetation in terms of the vegetation index over the course of time. All these fall under the scope of this thesis.

1.3.2 Contribution

Through this research, we discovered significant research gaps in the areas of vegetation segmentation, vegetation index, health evaluation, and unique solutions with far-reaching implications for the future of the relevant field throughout this research. The artefacts of this thesis are the research gaps and their related remedies.

This thesis is presented as a thesis-by-publication, with the following primary methodology chapters produced utilising linked research publications:

1.3.2.1 Data Preparation and Image Registration using Affine Invariant Convolutional Features (Chapter 2)

Direct automation of large multi-temporal image collections is not suitable for any image analysis due to variations in imaging conditions such as variations in viewpoints, scales, luminosity, and camera characteristics. In other words, this multi-temporal image data is not registered. Thus, a robust multi-temporal image registration is urgently required. Chapter 2 proposes a way to make it easier for computers to keep an eye on the environment by registering multiple images taken at the same time. It uses deep convolutional networks to do this.

The practice of overlaying at least two or more images of the same subject obtained at different times and from different views and sensors is known as multi-temporal image registration. This procedure aligns two images, referred to as sensed and reference images, geometrically. Various multi-temporal image registration al-

gorithms for applications such as remote sensing have been proposed over the years. It can be determined by the size of the region (intensity values) or the characteristics of the features (hand-crafted features like SIFT [20]). Because of their robustness against varied image variances, there are many hand-crafted or shallow learning feature-based approaches. Due to end-to-end feature learning, a few robust approaches based on deep features [21, 22] have recently been developed that provide superior accuracy and quality performance compared to hand-crafted feature-based strategies. These methods, on the other hand, are not resistant to affine transformations, which are a necessary element of repeat photography. It's partly due to the fact that deep convolutional networks aren't very adept at generalising minor picture modifications [23]. De Vos et al. [24] proposed a deformable image registration for medical imaging, but the scope differs from the multi-temporal nature of environmental imagery.

The affine invariant deep image registration technique used in this study addresses a gap in the literature by demonstrating how it can control imaging changes in repeat photography while still maintaining good quality. The following research questions were addressed in an attempt to provide answers: The question is: (i). How might deep learning and citizen science be integrated in order to provide automated, community-based environmental monitoring? (ii). In what ways may deep convolutional networks be used to improve the accuracy of multi-temporal image registration while also making it more robust against affine transformations? (iii). Is it possible to improve the performance of convolutional models in image registration by increasing the depth of the models?

This chapter makes the following contributions by addressing the following research questions:

1. Introduced a deep affine invariant network for non-rigid image registration of

multi-temporal repeat photography, which is used for the first time in this paper.

2. Integrate affine invariance and robust outlier detection for image point matching in order to achieve robust multi-temporal image registration using image point matching.

Related publication:

- *Multi-temporal registration of environmental imagery using affine invariant convolutional features.*

  Khan, A., Ulhaq, A., & Robinson, R. W. (2019, November). Multi-temporal registration of environmental imagery using affine invariant convolutional features. In Pacific-Rim Symposium on Image and Video Technology (pp. 269-280). Springer, Cham. doi: https://doi.org/10.1007/978-3-030-34879-3_21.

1.3.2.2 The Vegetation Index Interprets the Health of a Particular Area's Vegetation (Chapters 3 and 6)

It has been found in the literature that automated segmentation is used to extract the vegetation region for the purpose of calculating the vegetation index. There have been several approaches proposed to use remote sensing imagery for this purpose [25, 26]. However, advances in the field of computer vision have enabled us to employ approaches to obtain an automated system for vegetation segmentation, allowing us to calculate quantitative measures relating to vegetation. Fortin et al., [27] proposed an algorithm for estimating landscape composition from repeat photography, but they, like other researchers, relied on a manual segmentation scheme.

In the *chapter 3*, research is being carried out in order to come up with a novel

approach of an automated system for segmentation and then vegetation measurement in repeat photography, which is done with the help of sophisticated computer vision techniques. The proposed algorithm carries out processing to calculate the vegetation index from the images of the same site acquired in different phases of a year as well as in different years. A machine learning algorithm, support vector machine (SVM) was deployed for this job, which was trained using colour and texture features extracted from the patches of the image. The algorithm produced promising results for segmentation as well as for the calculation of quantitative measures for vegetation approximation using the dataset provided. In the field of ecology, the vegetation index of a particular site provides a lot of information about the environment in which it is located.

Related publication:

- *Detection of vegetation in environmental repeat photography: a new algorithmic approach in data science.*

  Khan, A., Ulhaq, A., Robinson, R., & Rehman, M. U. (2020). Detection of vegetation in environmental repeat photography: a new algorithmic approach in data science. In Statistics for Data Science and Policy Analysis (pp. 145-157). Springer, Singapore. doi: https://doi.org/10.1007/978-981-15-1735-8_11.

- *A Multiview Semantic Vegetation Index for Robust Estimation of Urban Vegetation Cover.*

  Khan, A., Asim, W., Ulhaq, A., & Robinson, R. W. (2022). A Multiview Semantic Vegetation Index for Robust Estimation of Urban Vegetation Cover. Remote Sensing, 14(1), 228. doi: https://doi.org/10.3390/rs14010228.

1.3.2.3 Vegetation Health Assessment (Chapters 4 & 5)

The health assessment of vegetation is carried out by the leaves of small plants and by the stems of large plants/trees. It is very obvious that the effects of plant disease on quantitative and qualitative production [5] are devastating, resulting in a striking blow to farmers, traders, and consumers. The health of vegetation is very essential not only for the preservation of plant species but also for their impact on the economy, people, and food production. Traditionally, farmers detect and diagnose plant diseases through their observations and rely upon the opinions of local experts and their past experiences. An expert can determine whether or not a plant is healthy [6]. If a plant is found unhealthy, noticeable symptoms on its leaves and fruits are observed and reported. Plant disease diagnosis incorporates a substantially high degree of difficulty through visual examination of the symptoms on plant leaves. Because of this challenge and the huge number of grown plants and their existing phytopathological issues, even qualified agronomists and plant pathologists sometimes struggle to accurately identify particular diseases and are consequently driven to make wrong assumptions and remedies [7]. Practical plant health assessment and disease diagnosis can improve product quality and prevent production loss. But there were some drawbacks, such as a lack of usability because of hardware complexity issues, inefficient use, and limited real-time use in real-world operational use.

1.3.2.3.1 Small Plants

Automated identification and classification of plant leaf health is critical for both the reduction of economic losses and the conservation of specific plant species. Various machine learning (ML) models have been proposed in the past for the detection and identification of plant leaf diseases. Despite this, their usability is limited because

they are very complicated to use, they can't grow, and they don't work well in real life. In *chapter 4*, DeepLens Classification and Detection Model (DCDM) is a real-time solution to this problem. It can automatically detect and classify leaf diseases in fruit trees (apple, peach, strawberry), as well as vegetable plants (potato and tomato) using scalable transfer learning on Amazon SageMaker and import the results into AWS DeepLens for real-time functional use. Cloud integration enables our approach to be scalable and accessible from anywhere at any time. The experiments on a large image dataset of healthy and unhealthy fruit trees and vegetable plant leaves yielded very impressive results, including the ability to diagnose plant diseases in real-time on the leaves of the plants. Using AWS DeepLens, it takes on average 0.349s to test an image for disease diagnosis and classification. This means that the consumer will get disease information in less than a second from the service.

Related publication:

- *Real-time Plant Health assessment via Implementing Cloud-based Scalable Transfer Learning on AWS DeepLens.*

  Khan, A., Nawaz, U., Ulhaq, A., & Robinson, R. W. (2020). Real-time plant health assessment via implementing cloud-based scalable transfer learning on AWS DeepLens. Plos one, 15(12), e0243243. doi: https://doi.org/10.1371/journal.pone.0243243.

1.3.2.3.2 Large Plants

Our urban lifestyle places a high value on the identification and continuous monitoring of vegetation (trees). The *chapter 5* proposes a deep learning-based network, the Siamese convolutional neural network (SCNN), combined with a modified brute-force-base line-of-bearing (LOB) algorithm that evaluates the health of eucalyptus trees as healthy or unhealthy and identifies their geo-location in real-time

from Google Street View (GSV) and ground truth images. The dataset depicts the various details of eucalyptus trees, including multiple viewpoints, scales, and different shapes and textures, among other things. This research contributes to the automated identification of eucalyptus trees with health issues or dead trees, which can assist urban green management and the local council in making decisions about tree planting and tree care improvements. As a whole, this study shows that a deep learning algorithm can identify the majority of healthy and unhealthy eucalyptus trees in real time, even when the background is complicated.

The key contributions are:

1. classification of trees that are in a healthy or unhealthy state; and

2. identification of the geo-location of the eucalyptus trees.

All of these evaluations are done on the basis of ground truth data collected from the streets by the researchers themselves. It is demonstrated in the experiments that the proposed method is capable of detecting and classifying healthy and unhealthy eucalyptus trees in a variety of datasets with a variety of backgrounds. The proposed method for geo-location identification gives reliable results and could be applied to the geo-identification of other objects on the roadside.

Related publication:

- *Health Assessment of Eucalyptus Trees Using Siamese Network from Google Street and Ground Truth Images.*

  Khan, A., Asim, W., Ulhaq, A., Ghazi, B., & Robinson, R. W. (2021). Health Assessment of Eucalyptus Trees Using Siamese Network from Google Street and Ground Truth Images. Remote Sensing, 13(11), 2194. doi: https://doi.org/10.3390/rs13112194.

1.3.2.4 Semantic Vegetation Index and Multiview Semantic Vegetation Index (Chapter 6)

Vegetation indices (VIs) were created in the 1970s so that satellite sensors could keep an eye on terrestrial landscapes. They have been very successful at measuring things like vegetation condition, foliage, cover, phenology, and processes like evapotranspiration (ET) and primary productivity, which are related to the amount of light a canopy absorbs (fPAR) [28].

Despite the fact that there are a variety of vegetation indexes available in the literature, they are either limited to a specific image modality and colour feature, or they overlook essential flora semantic information. Because of this, they are more susceptible to noise, resulting in inaccurate estimation.

In order to address these shortcomings, a novel semantic vegetation index (SVI) is proposed. This approach incorporates deep semantic segmentation into the process of vegetation index estimation. SVI is robust to the colour, viewpoint, and seasonal variations. Furthermore, it is capable of being applied directly to RGB images. Using deep semantic segmentation and multiview field coverage, it can be integrated into any vegetation management platform. It can be expanded to include multiple views in order to increase exposure and ensure accurate calculations. A multiview semantic vegetation index (MSVI) approach was proposed for this purpose. It is not asserted that the segmentation approach made a significant contribution to this study. Nonetheless, it compares a variety of approaches in order to determine which one is the most appropriate for achieving this goal. For example, the MSVI can be used to look at urban forestry and vegetation biomass in cities. It's a reliable and accurate way to figure out how much plant cover there is on the ground at street level.

Related publication:

- *A Multiview Semantic Vegetation Index for Robust Estimation of Urban Vegetation Cover.*

  Khan, A., Asim, W., Ulhaq, A., & Robinson, R. W. (2022). A Multiview Semantic Vegetation Index for Robust Estimation of Urban Vegetation Cover. Remote Sensing, 14(1), 228. doi: https://doi.org/10.3390/rs14010228.

1.3.2.5 Vegetation Health Monitoring Based on Semantic Vegetation Index (Chapter 7)

The calculation of values of various vegetation indexes over a period of time is frequently attributed to the automated monitoring of vegetation health in a landscape. However, such approaches suffer from the inaccurate estimation of vegetational change as a result of an overreliance on index values based on the colour attributes of vegetation or the availability of multi-spectral bands. However, one common observation is that colour attributes are sensitive to seasonal variations as well as imaging devices, resulting in false and inaccurate change detection and monitoring. An extension of previous work (chapter 6) is added to create a semantic vegetation health monitoring platform that can be used to keep track of vegetation health in a large area/landscape. Initially, a deep learning-based semantic segmentation model is used to classify vegetation in repeated photographs, which is the subject of the proposed article. An index of semantic vegetation is then calculated and plotted in a time series to account for seasonal variations and environmental impacts. According to the results, there is a lot more or less vegetation cover each year. The semantic segmentation model did well when it came to estimating how much vegetation cover each pixel had.

It provides a dependable platform for handling citizen science data in the context

of automated community service initiatives. In general, the use of repeat photography in vegetation monitoring adds significant value to other quantitative data derived from remote sensing and field measurements, as well as to other types of vegetation monitoring. These photos can also help policymakers and the general public become more aware of how the landscape and vegetation are changing. They can also show how land management practices affect the environment.

Related publication:

- *A Deep Semantic Vegetation Health Monitoring Platform For Citizen Science Imaging Data.*

  Khan, A.; Asim, W.; Ulhaq, A.; Robinson, R.W. "A Deep Semantic Vegetation Health Monitoring Platform For Citizen Science Imaging Data."

  Under Review with PLOS ONE Journal.

1.4 Research Datasets

Almost all publicly available vegetation related datasets were used in this study. The utilisation of publicly available datasets enables for comparison of different approaches and provides insight into respective methodologies' limitations. The vegetation analysis community is familiar with these datasets which range in complexity, capture environment, and camera settings. The methods suggested in this thesis are put to the test on these datasets, and compared to current research. This thesis used the following publicly available or self-acquired datasets for conducting this research:

- Google Street View (GSV): Google Street View is a feature of Google Maps and Google Earth that provides interactive panoramas from around the world.It began in numerous locations across the United States in 2007 and has since

spread to encompass cities and rural areas all around the world. On Google Maps, streets with Street View imagery are depicted as blue lines. Google Street View allows users to interact with stitched VR panoramas. The majority of photography is done on a car, but there is also photography done on a tricycle, camel, boat, snowmobile, underwater device, and on foot.

- Fluker Post Project dataset: Fluker Posts are actual wooden posts that serve as photo points in the environment. They are placed in strategic positions on approximately 168 sites. On a Fluker Post, no camera is left. Instead, passers-by used their mobile phones to take photos of the incident using the Fluker Post app. This basic repeat photography approach is a useful tool for long-term natural resource management.

- PlantVillage dataset: There are 54303 healthy and unhealthy leaf photos in the PlantVillage dataset, which are grouped into 38 groups based on species and disease. There are 39 different kinds of plant leaf and background photos accessible in this dataset. There are 61,486 pictures in this dataset. To increase the size of the data collection, we applied six distinct augmentation approaches. Image flipping, gamma correction, noise injection, PCA colour augmentation, rotation, and scaling are among the techniques used.

- Tarnab Farm image dataset: Tarnab farm Peshawar's agricultural research organization has been a driving force behind the province's agricultural production. It has made a significant contribution to the economic prosperity of the farming community over the course of its 109-year history by introducing and evolving high-yielding crop, fruit, and vegetable varieties, standardizing agronomic techniques, and disseminating the latest know-how on crop husbandry, soil management, fertiliser use, and plant protection measures. Farmers are

told about the findings of the research by reading research papers, technological bulletins, and popular publications in their own language.

1.5 Thesis Agenda

This thesis is structured in the format of a "Thesis with publication". The thesis chapters are organised and presented on the basis of their uniqueness, in such a way that they validate the previous claims.

The details of the organisational arrangement of chapters are illustrated in Figure 1.2.

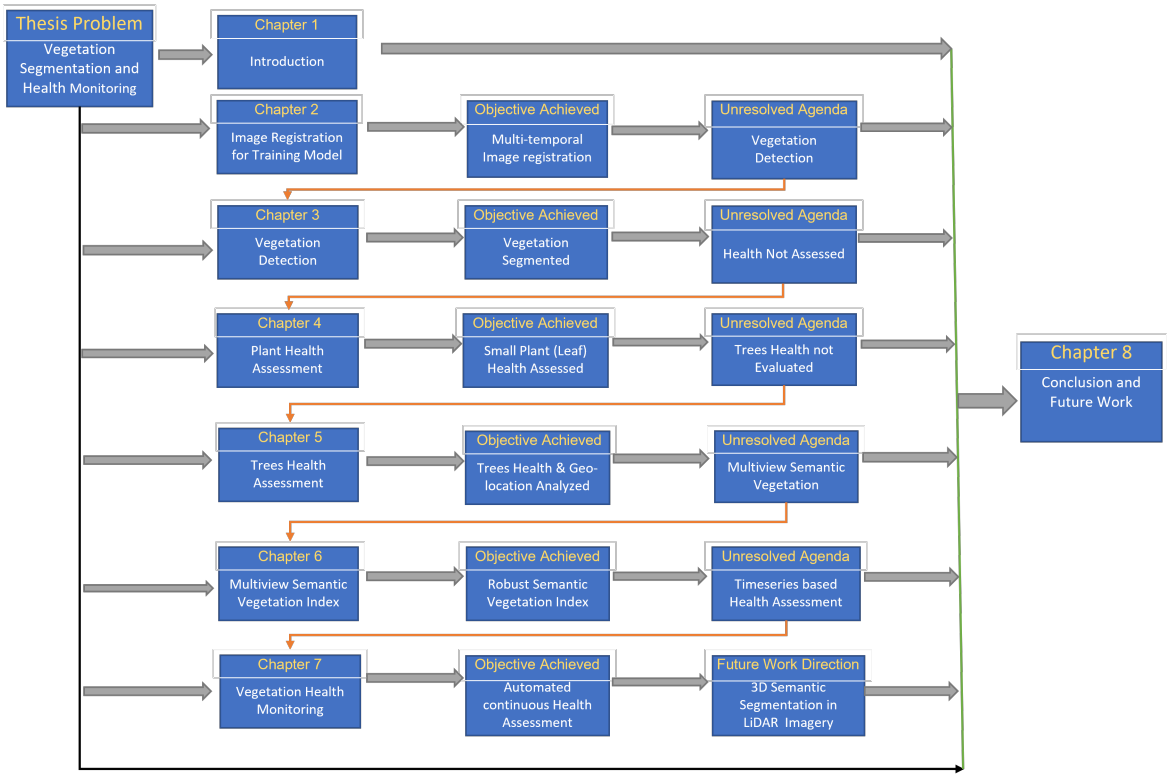


Figure 1.2: The flow of work in this thesis is shown in this organisation chart.

Chapter 1 presents the thesis background and motivation, aims and scope, contribution and significance, followed by the layout of the thesis with a brief

description.

Chapter 2 presents a robust multi-temporal image registration approach based on affine invariance and convolutional neural network architecture. The affine invariant deep image registration techniques can handle imaging variations well in repeat photography while maintaining quality performance. "Multi-temporal Registration of Environmental Images Using Affine Invariant Convolutional Features" is the title of this chapter's research. It was published in Image and Video Technology, PSIVT 2019, doi: https://doi.org/10.1007/978-3-030-34879-3_21, which is a conference on image and video technology.

Chapter 3 represents a novel approach towards automatic environmental analysis where the algorithm will carry out processing to calculate the vegetation index based on binary classification from the images of the same site acquired at different times of the year as well as in different years. The research work for this chapter has been published as "Detection of Vegetation in Environmental Repeat Photography: A New Algorithmic Approach in Data Science", in Statistics for Data Science and policy analysis. Springer, Singapore. Doi: https://doi.org/10.1007/978-981-15-1735-8_11.

Chapter 4 proposed a DeepLens Classification and Detection Model (DCDM) for plant leaf health assessment. In this study, transfer learning techniques are applied in AWS SageMaker, a cloud-based environment, to the proposed model known as the DeepLens Classification and Detection Model (DCDM), to identify and classify various fruits and vegetable leaves as either healthy or not, and to detect their disease based on a deep convolutional neural network. After completion of training DCDM, it was deployed into AWS DeepLens to make it a scalable and efficient real-time classification and identification model. The

related research work has been published as "Real-time plant health assessment via implementing cloud-based scalable transfer learning on AWS DeepLens." Plos one, 15(12), e0243243.doi: https://doi.org/10.1371/journal.pone.0243243.

Chapter 5 also contains a paper that proposes a deep learning-based network, the Siamese convolutional neural network (SCNN), combined with a modified brute-force-base line-of-bearing (LOB) algorithm that evaluates the health of eucalyptus trees as healthy or unhealthy and identifies their geo-location in real-time from Google Street View (GSV) and ground truth images. The research paper for this chapter has been published as "Health Assessment of Eucalyptus Trees Using Siamese Network from Google Street and Ground Truth Images", in Remote Sensing, 13(11), 2194. doi: https://doi.org/10.3390/rs13112194.

Chapter 6 presents a robust vegetation index based on semantic segmentation called the multiview semantic vegetation index (MSVI). This MSVI is based on deep semantic segmentation and multiview field coverage and can be integrated into any vegetation management platform. This chapter also consists of a research paper that has been published as "A Multiview Semantic Vegetation Index for Robust Estimation of Urban Vegetation Cover", in Remote Sensing, 14 (1), 228. doi: https://doi.org/10.3390/rs14010228.

Chapter 7 is the extension of Chapter 6, which provides automatic vegetation health monitoring using repetitive photography. In this article, we build upon our previous work on the development of a Semantic Vegetation Index (SVI) and expand it to introduce a semantic vegetation health monitoring platform to monitor vegetation health in a large landscape. The research work for this

chapter has been submitted to a journal, "A Deep Semantic Vegetation Health Monitoring Platform for Citizen Science Imaging Data" (under production).

Chapter 8 provides a conclusion to the work that has been presented as well as directions for future exploration.

1.6 Chapter Summary

In this chapter, we presented our research topic, its background, significance, and applications, our motivation to work on this problem, and the thesis's organizational structure.

In the next chapter, we'll show how deep affine features can be used to register multi-temporal vision using a convolutional neural network. In addition, we would establish a link between our research contributions and the elimination of flaws in earlier techniques.

Chapter 2

Multi-Temporal Registration of Environmental Imagery using Affine Invariant Convolutional Features

As mentioned in list of publications, this chapter with same title was published as an original research paper in Pacific-Rim Symposium on Image and Video Technology (PSIVT) - 2019. Lecture Notes in Computer Science, vol 11854, (pp. 269-280),Springer, Cham. doi: https://doi.org/10.1007/978-3-030-34879-3_21. The contents are the same, with the exception of certain layout adjustments to ensure consistency in the presentation across the thesis.

Abstract

Repeat photography is the practise of collecting multiple images of the same subject at the same location but at different timestamps for comparative analysis. The visualisation of such imagery can provide a valuable insight for continuous monitoring and change detection. In Victoria, Australia, citizen science and environmen-

tal monitoring are integrated through visitor-based repeat photography of national parks and coastal areas. Repeat photography, however, poses enormous challenges for automated data analysis and visualisation due to variations in viewpoints, scales, luminosity, and camera attributes. To address these challenges brought by data variability, this paper introduces a robust multi-temporal image registration approach based on affine invariance and convolutional neural network architecture. Our experimental evaluation on a large repeat photography dataset validates the role of multi-temporal image registration for better visualisation of environmental monitoring imagery. Our research will establish a baseline for the broad area of multi-temporal analysis.

2.1 Introduction

Australia is the 6th largest country in the world by land, with 16% area covered by naive vegetation, forests, and woodlands. There are more than 500 national parks with almost 4% land. The Australian government is seriously taking every step to take care of the natural environment and keep this country green. Federal and state governments organise various projects to give fresh and healthy air to the people and play a vital role in reducing global warming. Environmental monitoring is essential for all these natural resources, especially to protect and conserve the green areas and national parks. This is being achieved with the help of land-care groups, environmental groups, indigenous organisations and local councils by various means, such as drones, UAVs, satellites and remote sensing. However, such efforts and approaches have their pros and cons and require extensive resources for operation.

An alternative approach to environmental monitoring could be based on citizen science, which actually engages local communities and visitors to look after the nat-

ural resources. It is achievable by taking photos of visited areas with their smart phones and cameras and sharing them with environmental scientists. Those photographs can be helpful for important observations, and appropriate interventions could be designed if any significant change is found during manual inspection. One example of such a project is the "The Fluker Post Project" [29], which involves numerous points in more than 150 locations all over Australia. Visitors and local communities are encouraged to take photos and send them back to the main website. Such projects are based on the concept of repeat photography, which is an approach for comparing photos of the same location taken at different timestamps [30]. To date, the project is successfully running and has collected valuable imaging data about vegetation, parks, catchments and waterways. One example is shown in Figure 2.1.

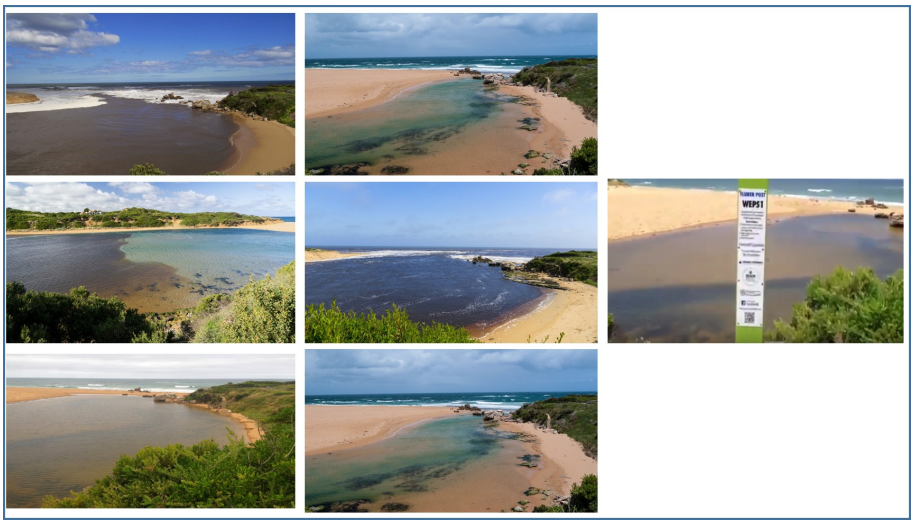


Figure 2.1: Illustrative repeat photography from Fluker Post community based image collection: Right image is reference post (coded as WEPS1) installed at Point Richie, Warrnambool, Victoria, Australia, left six photos were taken by different visitors at different times of the year. It can be observed that there is little control how visitors capture images from post or its surroundings. Variations in viewpoints, scales, lighting conditions and seasonal variations pose enormous challenges for automated multi-temporal image registration.

This extensive image data is currently being monitored manually to suggest appropriate interventions by state organisations like Parks Victoria. This manual approach, however, is less effective due to the abundance of data, and thus more automation is required to facilitate and speed up the data analysis. It is even more challenging in the case of repeat photography data, as manual image analysis is so cumbersome and inefficient. However, direct automation of such a large image collection is not suitable for any image analysis due to variations in imaging conditions such as variations in viewpoints, scales, luminosity, and camera characteristics. In other words, this multi-temporal image data is not registered. Thus, robust multi-temporal image registration is urgently required. This paper proposes a robust multi-temporal image registration technique for environmental repeat photography data based on deep convolutional networks to facilitate automated environmental monitoring.

Multi-temporal image registration is the process of overlaying at least two or more images of the same subject but taken at different times and from different view points and sensors. Actually, this process aligns geometrically two images known as sensed and reference images, respectively. Over the years, various multi-temporal image registration approaches have been proposed for applications such as remote sensing. It can be based on area (intensity values) or features (hand crafted features like SIFT [20]). Due to their robustness against different imaging variations, there is an abundance of hand-crafted or shallow learning feature based techniques. Most recently, a few robust approaches based on deep features [21, 22] that provide better accuracy and quality performance compared to hand-crafted feature based techniques due to end-to-end feature learning. However, these approaches are not robust against affine transformations which is an imperative feature of repeat photography. It is partially due to the fact that deep convolutional networks do not provide good generalisation for small image transformations [23]. De Vos et al., [24]

introduced a deformable image registration for medical images, but the scope is different from the multi-temporal nature of environmental imagery. To address this gap in the literature, this paper has introduced an affine invariant deep image registration technique that can handle imaging variations well in repeat photography while maintaining quality performance.

We attempt to answer the following research questions: (i) How can deep learning and citizen science be combined to achieve automated, community based environmental monitoring? (ii) How can multi-temporal image registration be simultaneously made accurate with deep convolutional networks and robust against affine transformations? (iii) Can the deepness of convolutional models increase the performance of image registration?

By addressing these research questions, this paper aims to make the following contributions:

(i) We introduce a deep affine invariant network for non-rigid image registration of multi-temporal repeat photography;

(ii) We integrate affine invariance and robust outlier detection for image point matching for robust multi-temporal image registration.

The rest of this paper is organized as follows: Section 2 reviews the related work. Section 3 provides a detailed description of our proposed network design and discusses how robust point matching is performed. Section 4 outlines our experimental set-up and interprets the results. Section 5 summarises this paper.

2.2 Prior Work

Community Based Monitoring (CBM): Community-based monitoring (CBM) is defined by Whitelaw et al. [31] as "a process where concerned citizens, government

agencies, industry, academia, community groups, and local institutions collaborate to monitor, track, and respond to issues of common community (environmental) concern". Over the last few years, a large number of CBM projects have been initiated around the world, mostly located in the USA, Australia, Canada, and the Russian Federation. Lawrence and Pretty [32, 33] have reported that since the 1990s, up to 500,000 community-based groups have been established in varying environmental and social contexts only in the USA and Canada. Mobile services and technology for community-based monitoring are covered [34]. Smart cities will benefit from the Internet of Things (IoT) and related CBM architecture [35]. Notable reviews [36, 37, 38, 39, 34, 35] provide valuable insight into community-based monitoring projects. These projects collect abundant multi-model data for analysis.

Repeat Photography: It used in CBM research studies due to availability of low cost cameras and smart phones. Different methods and applications of repeat photography are presented by Webb [40]. Various projects are available in the literature. An analysis of vegetation change in the San Juan Mountains using repeat photography was given by Zier et al. [41] and in the Appalachian Mountains by Hendrick et al. [42]. Digital repeat photography for phenological research in forest ecosystems was conducted by Sonnentag et al. [30]. A tourist-based environmental monitoring system based on principles of repeat photography is introduced by Augar and Fluker [29] with the help of Fluker posts at various locations in Australia. School kids were involved in collecting environmental imagery, [43]. Repeat photography has rarely been used in Australia. Pickard et al.,[44] presented data on vegetation changes in Australia over a century using repeat photography. However, the majority of all the approaches mentioned entail cumbersome manual inspection work and lack automation. Recent advances in computer vision, image processing, and deep learning enable us to automate the analysis of repeat photography. Deep learning

techniques especially convolutional neural networks, have appeared as revolutionary development tools for automated recognition and image analysis [45].

Image Registration And Deep Learning: Repeat photography-based environmental monitoring requires preprocessing of image collection captured over time. One of the most important steps is image registration; more specifically, in the context of the present study, it is multi-temporal image registration. Image registration seeks to remove the two-date images' geometric position inconsistent, making the same image coordinates reflect the same objects. Notable surveys [46, 47, 48, 49] cover various techniques for image registration. Image registration techniques can be classified based on area (e.g., raw intensity values) or features (e.g., SIFT [20]). The success of deep learning has also affected the registration process. Recently, deep approaches are proposed [21, 50, 22] that provide better accuracy and quality performance compared to hand-crafted feature-based techniques due to end-to-end feature learning. However, these approaches are not robust against affine transformations, which is an imperative feature of repeat photography. Although our work is related to a similar domain, we attempt to address shortcomings in the previous work in our proposed approach.

2.3 The Proposed Deep Image Registration Framework:

In this section, we describe the architecture of the proposed deep image registration framework for multi-temporal image registration of repeat photographic data.

Feature-based image registration usually comprises the following stages: First, a pair of images (sensed and reference) are submitted for feature detection and feature descriptors are calculated. Preliminary point-wise correspondence or point

sets are then estimated according to distance based metrics as feature pre-matching is refined by inlier detection. It is then followed by optimal transformation parameters estimation that guides the re-sampling and image transformation of the sensed image.

The first goal is to find deep discriminative descriptors. For this purpose, we make use of a deep convolutional network that comprises two parts. The first part spatially transforms the input image by generating an affine transformation for each input sample. It then extracts local key points from feature maps of a convolutional neural network. The second part outputs local descriptors given patches cropped around the key points. We name these parts the affine-invariant deep feature extractor and the deep feature descriptor.

Generation of Feature Maps: We first use ResNet [51] to generate a rich feature map from an image I, which can be used to extract deep keypoint locations. The reason for selecting ResNet is its effectiveness in building deep networks without overfitting by injecting identity mappings or so-called shortcut connections. It also solves the vanishing gradient problem, which refers to the situation where the gradients from where the loss function is calculated, when the network is too deep, would shrink to zero after several iterations of the chain rule. Such a problem causes the weights to never update their values and thus results in no learning. ResNet has three blocks, block of convolutional filters of 5×5 followed by batch normalization, ReLU activations, and another group of 5×5 convolutions. These convolutions are zero-padded to get the same output size as the input, with 16 output channels.

Affine-Invariant Deep Feature Extractor: After each block, we integrate a spatial transformer module [52] into the network that introduces explicit spatial transformations on feature maps, without making any changes to the CNN loss function. During spatial transformer implementation, we use 2D spatial affine transformations that help during affine invariant keypoint detection. Suppose that the

output pixels are defined to lie on a regular grid $G = G_i$ of elements of a generic feature map, τ_θ is affine transformations, a_i^s, b_i^s as the source coordinates in the input feature map, and a_i^t, b_i^t as the target coordinates of the regular grid in the output feature map. We can write the transformation introduced by the affine transformer as:

$$\begin{bmatrix} a_i^s \\ b_i^s \end{bmatrix} = \tau_\theta(G_i) = \tau_\theta \begin{bmatrix} a_i^t \\ b_i^t \\ 1 \end{bmatrix} = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix} \begin{bmatrix} a_i^t \\ b_i^t \\ 1 \end{bmatrix} \tag{2.1}$$

The first part of the spatial transformer is a localisation network that takes the feature map as an input and outputs the parameters of the spatial transformation that should be applied to the feature map. In the second part, the grid generator takes the predicted transformation parameters and uses them to generate a sampling grid (a set of points where the input map should be sampled to produce the transformed output). Finally, the sampler takes the feature map and the sampling grid as inputs to the sampler, generating the output map sampled from the input at the grid points. For more implementation details of the spatial transformer, please refer to our previous work [52]. The transformed feature maps are robust to affine-transformations and we use them to extract affine-invariant deep discriminative features.

To capture scale-space response, we resize each feature map n times, at uniform intervals between $1/t$ and t, where $n = 3$ and $t = 2$. The resulting maps are then convolved with n 5×5 filters, resulting in n score maps. It is then followed by non-maximum suppression by using a softmax operator over 15×15 windows in a convolutional manner, which results in n saliency maps. Due to their scale-invariant nature, these saliency maps are again resized to the original image size, and the

score is unaffected. Using another softmax function, all of these saliency maps are then merged into one map M. Similarly, orientation map θ is calculated using the arctan function,whis is based on an existing framework [53]. We choose the top K elements of M as keypoints and we have information of the form $\{x, y, s, \theta\}$ about these keypoints. These keypoints are invariant to affine transformations, and we call them affine-invariant deep discriminative features.

Deep Feature Descriptors: For this goal, we crop image patches around the selected key point locations. We crop and resize them to 32×32. Our descriptor network is an L2 Net [54]. It comprises convolutional layers followed by Local Response Normalization layer (LRN) as the output layer to produce unit descriptors. It results in 128 dimensional descriptor D for every patch.

Feature Pre-Matching And Outlier Removal: For feature matching, we use Euclidean-distance d between the respective feature descriptors of feature points p and q as:

$$d(p, q) = d(D(p), D(q)) \tag{2.2}$$

These matches may have many outliers for multi-temporal image registration. In other words, we are interested in detecting among F nearest points in a given feature map, $(p_k)_{k=1,\ldots,F}$ and their predicted corresponding matches. $r(p_k)_{k=1,\ldots,F}$ in the corresponding reference feature map,that is, the largest subset of key points that follow the same affine model. To achieve such outlier removal, we use an algorithm, namely the Affine Parameters Estimation by Random Sampling (APERS) [55], which detects the outliers in a given set of matched points.

Non-Rigid Transformation Estimation: Finding point sets and their correspondences is the most crucial step. Point sets are represented by sensed set, $(x_i | x_i \in R^d, i = \{1, \ldots, N)\}$ and target set $(t_i | t_i \in R^d, i = \{1, \ldots, M)\}$, where the

sensed set is faced with a transformation, τ. Due to repeated photography, we assume that there exists a significant overlap between the sensed and reference point sets. If we know the feature descriptor D that can be used to get correspondence, following the approach described in [56], the point set registration can be formulated as follows:

$$E(\tau) = \lambda\phi(\tau) + \sum_{i=1}^{N}\sum_{j=1}^{M} exp(\frac{||t_j - x'_j||^2}{\sigma^2} - \frac{(D(t_j) - D(x'_j))^T \sum^{-1} D(t_j) - D(x'_j)}{\xi^2}) \tag{2.3}$$

where $x'_i = x_i = \tau$ is the transformed point, $\sum$ denotes the covariance matrix of the feature descriptors, $\sigma.\xi$ are scales, $\phi(.)$, the regularization, and λ is used to control the level of smoothness. We use spatially constrained Gaussian Fields (SCGF) for non-rigid transformation estimation [56].

Image Resampling and Transformation: The estimated transformation parameters finally guide the re-sampling and warping of the sensed image for getting registered image. Due to artefacts and simplicity of pixel level registration, the refinement of coarsely registered images to sub-pixel accuracy is performed. However, rather interpolating image grayscales, we uses functions of the graylevels like [57].

2.4 Experimental Results and Discussion

We conducted experimental studies based on publicly available image datasets. In this section, we will provide a brief description of the dataset (Fluker Post Vegetation Dataset), registration accuracy, and other performance comparisons with state-of-the-art techniques. Our network is based on Keras (with TensorFlow backend) and

the Nvidia P2000 Quadro GPU.

Fluker Post Vegetation Dataset: This image dataset is collected as a part of the "The Fluker Post Project" [29]. This project is based on citizen science, and a part of the project is to place photo-capturing posts in over 150 locations all over Australia. Visitors and local communities are encouraged to take photos and send them back to the main website. In each location, multiple images are taken from the same point, called the Fluker post. This project now has more than 2000 images, all of which are organised into different albums related to different locations across Australia. The web address of this project is http://www.flukerpost.com. However, the problem is that all those photos are taken from different viewpoints and camera sensors. The image resolution also varies across the dataset. Photos are captured in different seasons as well as different timestamps of the day. All these make the work challenging. Therefore, it becomes very difficult to compare and extract some valuable information like vegetational change detection.

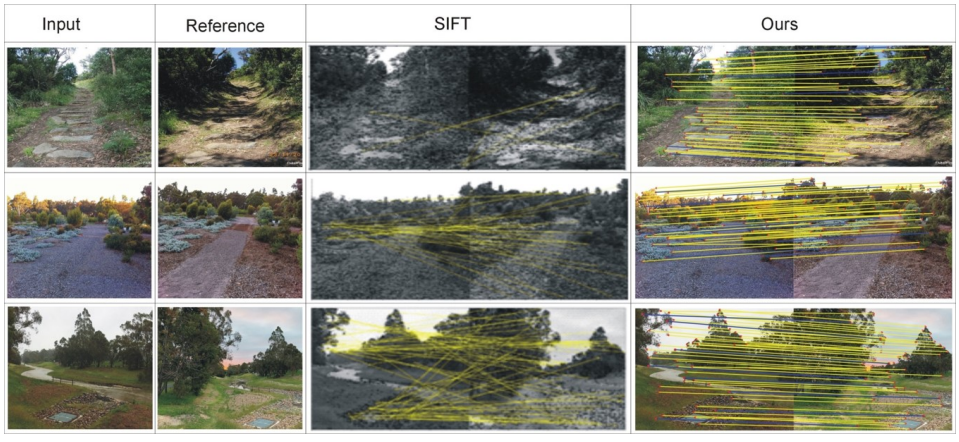


Figure 2.2: Illustrative repeat photography image registration from Fluker Post community based image collection: First Column: All images are input images of different locations in Australia, Second Column, these images are reference images from relevant Fluker Post, third column shows the point correspondences after SIFT matching and the last Column shows the point correspondences obtained after automated deep multi-temporal image registration.

Subjective Comparison: We developed a baseline image registration based on SIFT hand-crafted features. For evaluation, we compared our convolutional feature with SIFT. For subjective comparison, we have displayed feature correspondences. Figure 2.2 displays a scenario of image registration and feature matching. We calculated the matching features and warped the images according to the match.

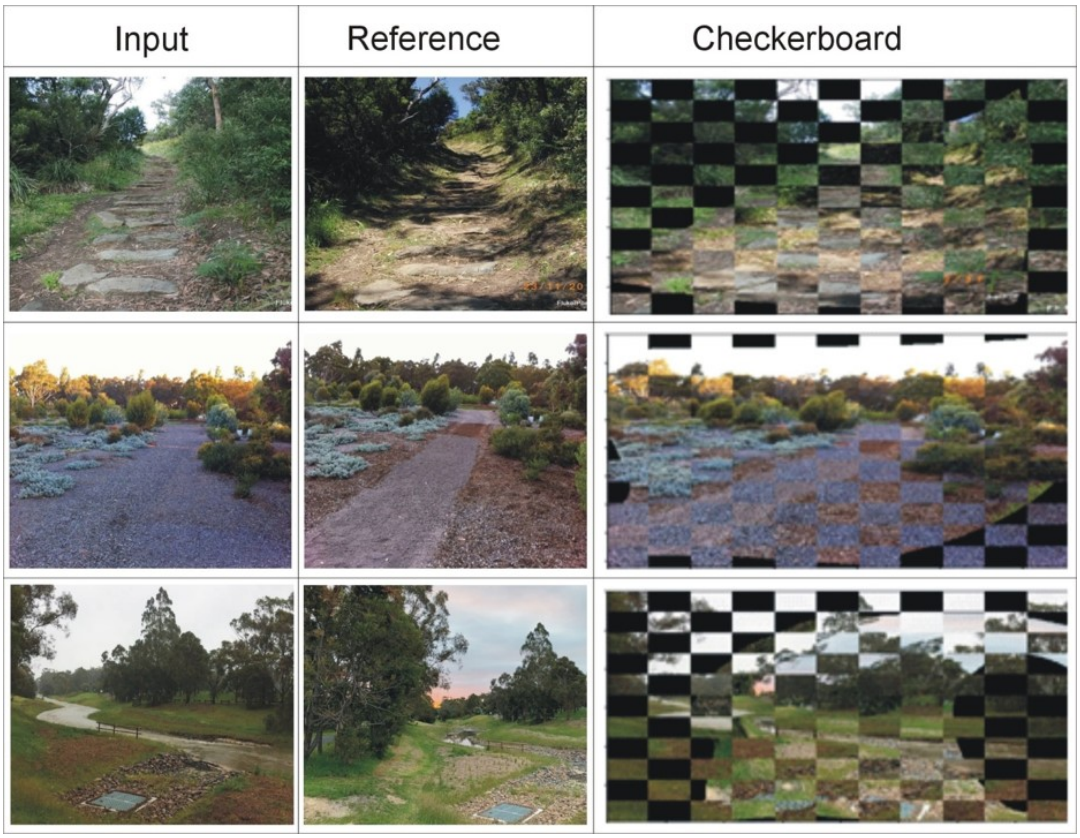


Figure 2.3: Illustrative repeat photography image registration from Fluker Post community based image collection: First Column: All images are input images of different locations in Australia, Second Column, these images are reference images from relevant Fluker Post, last Column shows the registered image with varied sections shown by checkerboard after automated deep multi-temporal image registration.

In Figure 2.3, we also displayed a checkerboard to show the parts of the images that suffered the transformational effects. Both images clearly show that our feature matching is dense and robust to different variations compared to SIFT.

Objective Comparison: For objective comparison, we used different metrics. In each corresponding image pair, we extract feature points using both methods and use the most reliable 95-105 pairs of matches to measure precision. P as

$$P = \frac{\text{TP}}{\text{TP+TF}} \tag{2.4}$$

where TP is true positive TF is true false.

Table 2.1 shows the comparison of our approach vs. SIFT. Additionally, we mark landmarks with 15 points and record the locations of the landmarks for measuring errors like root mean squared distance (RMSD), mean absolute distance (MAD), median of distance (MED), and the standard deviation of distance (STD). Table 2.2 shows the performance of our algorithm compared to other approaches. It can be observed that our approach is more reliable than all other approaches.

Table 2.1: Feature pre-matching precision test result for Fluker Post Vegetation Dataset. The used unit is percentage.

Index	SIFT	Ours
Average	70.75 %	96.35%
Minimum	35.41%	89.76%
Maximum	92.50%	99.6%

Table 2.2: Comparative Analysis in terms of registration accuracy test result on Fluker Post Vegetation Dataset in terms of different error matrices

Method	RMSD	MAD	MED	STD
CPD [58]	12.67	15.38	5.67	8.75
GLMDTPS [59]	26.87	28.92	8.60	11.48
GL-CATE [60]	10.94	14.89	4.25	8.57
Deep Features [21]	9.79	9.23	7.41	5.12
Ours	8.62	8.67	5.68	3.67

Algorithm 1 Point Set registration Algorithm

input: p, q
output: p'
Initialize: $iterations = 30, \phi, \alpha = 0$
for $k = 0, k < iterations, k++$ **do**
Get descriptors $d(p), d(q)$
Assign the correspondences weight C by Eq: 5,6,7 in [56]
Construct the RBF kernel matrix $\mathcal{K}$
Compute the transformation parameter α using Spatially Constrained Gaussian Fields (SCGF)[56]
Update the model point set $p' = p + \mathcal{K}\alpha$
end for
Return p'

2.5 Conclusion

In this paper, we investigated the integration of citizen science and deep learning-based image registration to facilitate automated image analytics for environmental monitoring. We proposed a novel deep affine invariant network for non-rigid image registration of multi-temporal repeat photography. We also integrated robust point matching and affine invariance into our framework for robust multi-temporal image registration. Extensive experimental studies have been conducted to evaluate the underlying design based on public datasets. The experimental results indicated that the proposed approach delivered higher quality performance than the existing techniques. This work would set new research directions to achieve a fully automated environmental monitoring system.

Chapter 3

Detection of vegetation in environmental repeat photography: a new algorithmic approach in data science

As mentioned in list of publications, this chapter with same title was published as an original research paper in the Statistics for Data Science and Policy Analysis, (2020), (pp 145-157). Springer, Singapore. doi: https://doi.org/10.1007/978-981-15-1735-8_11. The contents are the same, with the exception of certain layout adjustments to ensure consistency in the presentation across the thesis.

Abstract

The environment, being one of the major issues in today's world, needs special attention from researchers. With the advancement in the field of computer vision, researchers are equipped enough to come up with the algorithms to have an automated system for environmental monitoring. This paper proposes an algorithm in this regard, which can be used to monitor the change in vegetation in a specific loca-

tion. This would assist environmental professionals in directing their efforts towards improving the environmental situation. An algorithm is proposed which registers the image so that comparison can be carried out in an accurate manner using a single framework for all the images. The registration algorithm aligns the new image with the already present previous image by performing a transformation. The registration process is followed by a segmentation process that separates out the vegetation region from the image. A novel approach towards segmentation has been proposed, which works on a machine learning-based algorithm. The proposed algorithm showed promising results, with an *F-measure* of 85.36%. The segmentation result leads us to an easy calculation of the vegetation index, which can be used to create a vegetation record for a specific site.

3.1 Introduction

In today's world, the population has increased rapidly with the advancement of technology, which has created a need and importance for environmental monitoring. Humans have affected the environment to such an extent that the time has come to pay attention to this issue. The change in climate and the change in land are interconnected powers of global change prompted by human beings [61], [62].

Environment monitoring is now an important task to do so that information can be processed to assess environmental effectiveness. It has been proven to be helpful in monitoring humidity and temperature. One of the most important parts of environmental monitoring is paying attention to the information about vegetation, which helps us figure out what's going to happen in the future at an early stage.

Nowadays, satellite and aerial podiums are used to get remote sensing images that can be used for environmental monitoring. But for a very long time, photographs

taken from the ground have been used to record the change in the environment and ecosystem [44]. Repeat photography is one of the most common methods used for monitoring any change, where images are taken repeatedly of a particular location at different times. So, by aring the repeated photographs, analysis can be carried out regarding the change. While in the case of analysing environmental change, the time difference between image repetition is measured in months and years. This repeated photography conveys the change in the environment over time [63]. The advancement in remote sensing images has made these repeating photography methods lose their importance. But if environmental analysis needs to be carried out in a transparent manner, then revitalization of this technology is necessary so that researchers can utilise these images in their research. As repeated photographs give detailed information regarding the transformation in vegetation regarding some particular sites.

For decades, repeated photography has been used for various monitoring tasks, such as geomorphological development [64] [65] [66], change in trees arrangement [67] [68], coastal locales [69], plant phenology [70] [71] [72] [73] and many others. Literature shows that researchers have immensely utilised repeat photography for vegetation cover [74] [75] [42] [42] [76]. Repeat photography involves many issues, like background clutter, high inter-class variation, and illumination variance [77] [78]. However, with the advancement of technology and the introduction of new approaches, these limitations are encountered. Repeated photography is used to carry out different analyses. Hall et al., [79] and Roush et al., [67] have proposed an algorithm where an image is divided into rectangular grids and then, using those rectangular grids, the percentage of vegetation cover is calculated. Point sampling is also used to quantify the vegetation change [77]. Where classification is performed to classify each image into different cover types, which are treated as different classes.

Moreover, different cover types represent different quantitative measures. In literature, papers can be found that utilise automated segmentation for extracting the vegetation region for calculating the vegetation index. But the approaches proposed utilise remote sensing imagery for this purpose [25] [26]. However, the advancement in the computer vision field enables us to use approaches to get the automated system for the segmentation of vegetation so that quantitative measures regarding vegetation can be calculated. Fortin et al., [27] have proposed an algorithm for estimation of landscape composition from repeat photos, but like other researchers, they have used a manual segmentation scheme.

The purpose of the study was to develop an approach for an automated system for segmentation and vegetation measurement in repeat photography using advanced computer vision techniques. The proposed algorithm showed promising results for segmentation as well as the calculation of quantitative measures for vegetation approximation. As in the field of ecology, the vegetation index of some particular sites gives a lot of information regarding their environment.

The rest of the paper is organised in the following manner: Section 2 discusses the dataset used in this paper. Section 3 discusses the main methodology, and each step is discussed in detail. Section 4 carries details regarding the results of the proposed algorithm on the dataset discussed in Section 2. Section 5 presents the conclusion of the research.

3.2 Dataset

A dataset called Fluker Post [80] is used in the paper. This dataset has different images of the same scene taken at different points in time. Figure 3.1 shows some of the example images of two different scenes and their three different time span images.

The dataset contains images of 22 locations in total, and in each location there are images of multiple scenes. The dataset is the first initiative towards monitoring more than 150 important sites.

Figure 3.1: Figure shows dataset images where row figures shows different time spans and columns represent different scenes

3.3 Proposed Methodology

The proposed methodology involves multiple steps which lead to the calculation of the vegetation index present in an image. Figure 3.2 shows the general flow diagram followed for the extraction of vegetation index for any input image with regard to previous images of the particular scene the input image belongs to. This section

further discusses all the building blocks of the proposed methodology.

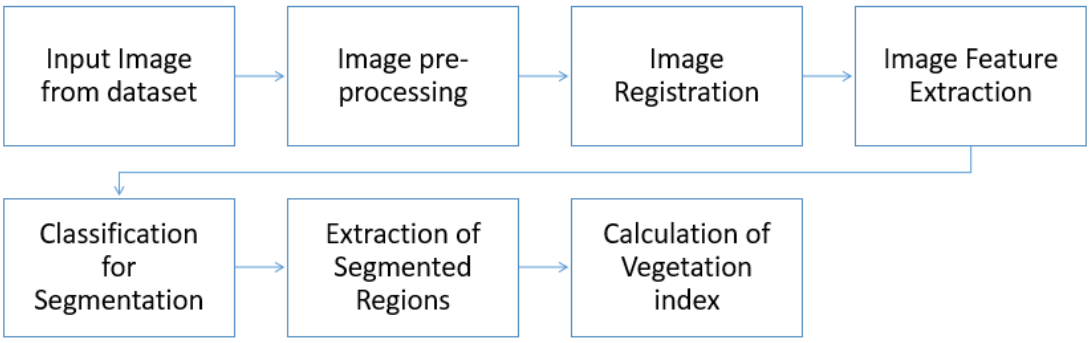


Figure 3.2: General Flow Diagram

3.3.1 Image Pre-processing

As the images in the dataset are taken at different time spans with different cameras, the images in the dataset have varying light intensities. Therefore, the dataset images need to be preprocessed before their registration process. For this process, image histogram equalisation is adopted to encounter the said problem. The image was divided into R, G, and B channels, after which image histogram equalisation was applied to the individual channels. The resultant channels were combined together to get the RGB image.

3.3.2 Image Registration

Acquired images are of the same scene, but they are not on a single coordinate system because they are captured from different viewpoints, as can be seen in Figure 3.1. Image registration is the process that can be used to tackle this problem. Image Registration transforms one image with respect to the other in such a manner that they can both be compared. Barbara et al., [47] have described the general steps

that need to be followed for image registration in any research problem. Those steps are,

- Feature Detection
- Feature Matching
- Model Estimation for Transformation
- Image Transformation

3.3.2.1 Feature Detection:

Gradients in an image hold great importance as they represent information regarding edges. Similarly, in the case of mitotic and non-mitotic cells, gradients can play a vital role. As discussed earlier as well, mitotic and non-mitotic cells differentiate from each other on the basis of texture and shape. So gradients give us information about edges, and edges represent the texture or shape of the image. Therefore, using HOG can be significant for the good performance of cell detection. HOG generates the histogram of the orientation of a window or patch selected. The working steps of HOG are discussed below.

- In the first step, the image is divided into small patches, and each patch is known as the detection window. The size of the window NxN is dependent on the input image size and the number of gradients we want to extract.
- Colour and gamma normalisation are performed for the extraction of strong gradients in the image.
- A Gaussian derivative is used to compute the gradients fx and fy in horizontal and vertical directions, respectively. The following filter kernels are used for

$$[-1, 0, 1] \quad and \quad [-1, 0, 1]^T \tag{3.1}$$

- Now orientation binning is performed in which cell histograms are developed. In a spatial cell, each pixel holds a specific weight which is derived from the gradient calculated. Histogram channels are spread evenly with a difference of 20 degrees.

- Contrast Normalization is performed on the overlapping blocks in cells. The normalisation factor used can be seen in the below equation.

$$Norm : L = \frac{B}{\sqrt{||B||_2^2 + t^2}} \tag{3.2}$$

  where "B" represents the un-normalized vector and "t" represents the threshold value which will remain constant for all windows.

- The last step is the collection of all HOGs in each block. This predicts our feature vector, which can now be used for classification..

3.3.2.2 Feature Matching:

In this step, the features of two images are matched together to calculate correspondence between them. We have used the Brute-Force Matcher method for calculating similarity between detected features. The Brute-Force Matcher works on a simple phenomenon where a distance between K1 (keypoint features extracted from image 1) and K2 (keypoint features extracted from image 2) is calculated. The minimum euclidean distance between K1 and K2 will be considered as matched points.

3.3.2.3 Model Estimation for Transformation:

After the feature matching task, the most important task is to remove the outliers. Outliers are those feature points which do not fit in our model. Removing them is necessary so that an accurate transformation parameter can be calculated. The proposed architecture utilises RANSAC for outlier removal tasks. The RANSAC algorithm works on four main steps, which are,

- From the dataset containing outliers, select a random subset to start the process.
- Model fitting is performed on the designated subset.
- Number of outliers is calculated for the fitted model.
- Repeat all three of the above steps for the selected number of iterations. The desired model with the check on the number of outliers against it is considered the best fit model.

For model estimation, we have used the similarity transform as a global mapping model. This model tackles three main transformation constraints, which are translation, rotation, and scaling. This model is also known as the "shape-preserving mapping model" because it conserves the angle as well as the curvature present in an image that needs to be transformed. So in our case, it is really important to conserve the angles because we need to segment out the output image of the registration block.

Equations 3 and 4 show the formulation used to calculate u and v, which are transformation coordinates. Where i and j are the coordinates of the original image, s, p, and t show scale factor, rotation, and translation factor, respectively, which are

calculated using the global mapping model.

$$u = s * (i * \cos(p) - j * \sin(p)) + t_i \tag{3.3}$$

$$v = s * (i * \sin(p) - j * \cos(p)) + t_j \tag{3.4}$$

3.3.2.4 Image Transformation:

A mapping function was constructed in the previous step, which would be used to transform the image so that two images can have the same scale, rotation, and translation. which is necessary so that we can know the change in vegetation index between two images in an accurate manner. Each pixel is individually transformed using equations 3 and 4 discussed in the previous step.

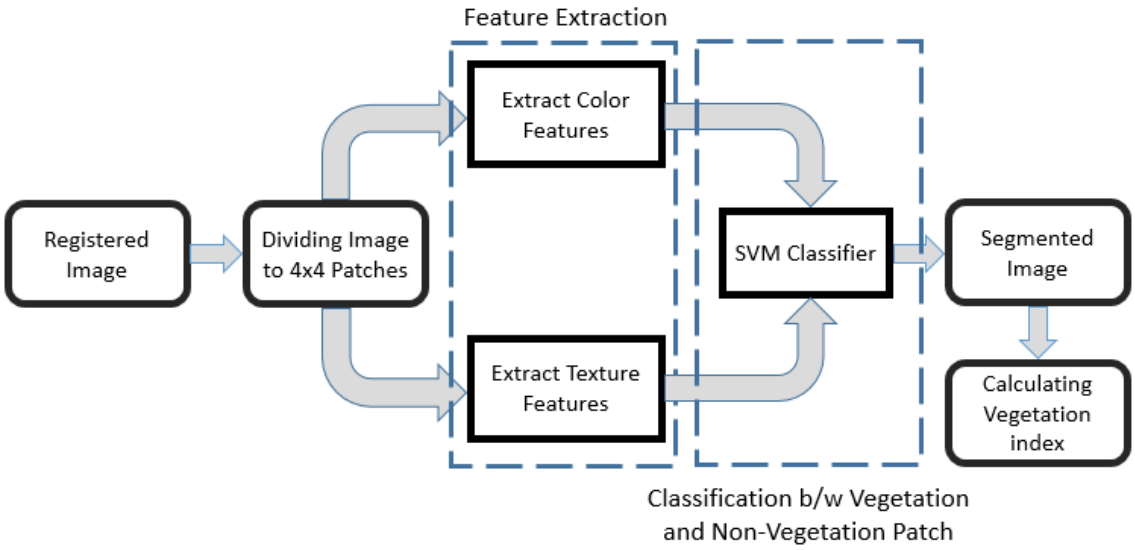


Figure 3.3: Proposed Segmentation Algorithm for Vegetation Index Calculation

3.3.3 Image Feature Extraction

After we register an image, we need to find the features in the image. To do this, we break the image into small chunks of 4x4 pixels. From each chunk or patch, colour and texture features were extracted. Figure 3.3 shows the algorithm adopted after image registration for the process of segmentation and finally achieving the vegetation index. The features extracted from the image are.

3.3.3.1 Colour Features:

Three colour features (red, green, and blue) pixel values from a block of 4x4 were extracted and presented as the mean values of all three channels in a block.

3.3.3.2 Texture Features:

Five texture features were extracted, which were: energy, entropy, contrast, homogeneity, and correlation. Equations 5–9 are used for the extraction of these features after converting the patch from RGB to grey scale. Where "I" is the pixel value of a gray-scale patch, l and m are the coordinates of the patch.

$$Energy = \sum_{l} \sum_{m} I^2(l, m) \tag{3.5}$$

$$Entropy = -\sum_{l} \sum_{m} I(l, m) log(I(l, m)) \tag{3.6}$$

$$Contrast = \sum_{l} \sum_{m} (l - m)^2 I(l, m) \tag{3.7}$$

$$Homogeneity = \sum_{l} \sum_{m} \frac{I(l, m)}{1 + |l - m|} \tag{3.8}$$

$$Correlation = \frac{\sum_l \sum_m (l - \mu_x)(m - \mu_y) I(l, m)}{\sigma_x \sigma_y} \tag{3.9}$$

3.3.4 Classification for Segmentation

After feature extraction of each patch of the image, we need to classify the patches into vegetation and non-vegetation classes. Each chunk was labelled with vegetation or non-vegetation using the Image Labelling application in MATLAB. A Support Vector Machine was used as a classifier. SVM was trained using 100 different scene images taken from Fluker post project dataset [80]. An SVM is a linear classifier that generates a boundary between the two classes for classification purposes. The boundary is created using the optimal hyperlane, which is the best fit model for classification of classes P and Q.

$$p \,.\, x + q = 0 \tag{3.10}$$

The optimal hyperlane equation 3.10 is required to full-fill two conditions which are,

- Hyperlane should be such that it satisfies below two equations,

$$f(x) = p \,.\, x + q \text{ should only be positive if } x \,\epsilon\, P \tag{3.11}$$

$$f(x) \leq 0 \quad \text{if } x \,\epsilon\, Q \tag{3.12}$$

- The hyperlane should be at the maximum possible distance from all the observations, which would depict the robustness of the system. The hyperlane

distance from the observation is defined as.

$$\frac{|p \,.\, x + q|}{||p||} \tag{3.13}$$

After classification of all patches, vegetation patches are segmented out from the image and combined together in a single image

3.3.5 Calculating Vegetation Index

After the segmentation process, an image is achieved that only contains vegetation content in the image. So we can now calculate the vegetation index with the help of pixels being segmented out. The equation used for calculating the vegetation index is shown below.

$$\text{Vegetation Index} = \frac{\text{Number of pixels segmented}}{\text{Total number of pixels in original image}} \tag{3.14}$$

3.4 Results

As discussed earlier, 100 images from the Fluker Post dataset were taken into account for training purposes. For testing purposes, we took another 50 images from the same dataset. On those images, registration was applied on the basis of the model image from each source cite. Then the developed algorithm was applied to segment the vegetation region in the image. Example images can be seen in Figure 3.4. It clearly shows that high performance segmentation is performed for the extraction of vegetation regions.

For further evaluation of the proposed segmentation algorithm, a few numerical

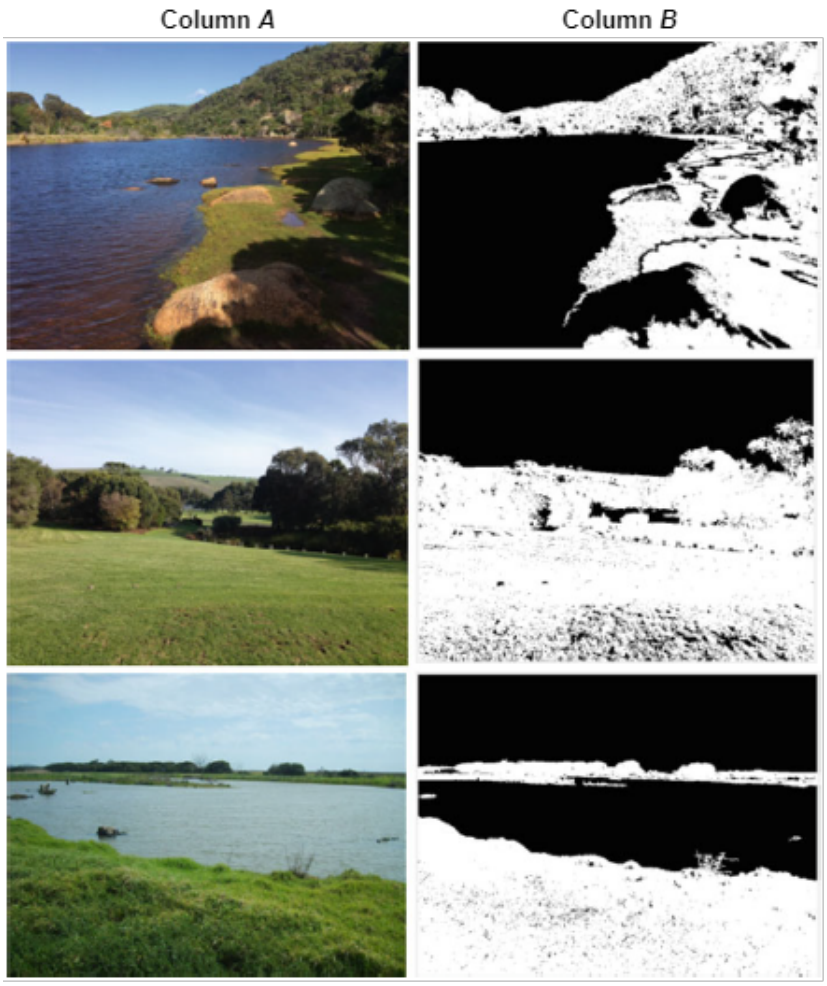


Figure 3.4: Column *A* shows the example registered imaged followed by the segmentation mask for vegetation region extraction in column *B*

approaches were adopted. The *F-measure* of the classification process carried out by SVM based on given features was evaluated on 50 test images. The equation of *F-measure* is shown below,

$$F\text{-}measure = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.15}$$

Where precision is the ratio of correctly classified vegetation patches to the total number of patches classified as vegetation region by SVM, and recall that the ratio

between the number of correctly classified vegetation and the number of vegetation patches present in the whole labelled image.

The *F-measure* achieved for the 50 test images from the Fluker post dataset is 85.36%. Moreover, we have calculated the vegetation index of the images as well. As shown in Figure 4, the calculated vegetation index is 0.47, indicating that 47.6% of the image is covered by vegetation.

3.5 Conclusion

Authors have proposed a novel approach to automatic environmental analysis. The proposed algorithm will carry out processing to calculate the vegetation index from images of the same site acquired in different phases of the year as well as in different years. The proposed algorithm for the segmentation of vegetation regions has shown promising results. A machine learning algorithm was trained on images using colour and texture features extracted from the patches of the RGB image. The segmentation algorithm is analysed using precious and recall values, where the calculated *F-measure* is 85.36%.

Chapter 4

Real-Time Plant Health Assessment via Implementing Cloud-Based Scalable Transfer Learning on AWS DeepLens

As mentioned in list of publications, this chapter with same title was published as an original research paper in the Plos one, (2020), 15(12): e0243243. doi: https://doi.org/10.1371/journal.pone.0243243. The contents are the same, with the exception of certain layout adjustments to ensure consistency in the presentation across the thesis.

Abstract

The control of plant leaf diseases is crucial as it affects the quality and production of plant species with an effect on the economy of any country. Automated identification and classification of plant leaf diseases is, therefore, essential for the reduction of economic losses and the conservation of specific species. Various Machine Learning (ML) models have previously been proposed to detect and identify plant leaf disease;

however, they lack usability due to hardware sophistication, limited scalability and realistic use inefficiency. By implementing automatic detection and classification of leaf diseases in fruit trees (apple, grape, peach and strawberry) and vegetable plants (potato and tomato) through scalable transfer learning on Amazon Web Services (AWS) SageMaker and importing it into AWS DeepLens for real-time functional usability, our proposed DeepLens Classification and Detection Model (DCDM) addresses such limitations. Scalability and ubiquitous access to our approach is provided by cloud integration. Our experiments on an extensive image data set of healthy and unhealthy fruit trees and vegetable plant leaves showed 98.78% accuracy with a real-time diagnosis of diseases of plant leaves. To train DCDM deep learning model, we used forty thousand images and then evaluated it on ten thousand images. It takes an average of 0.349s to test an image for disease diagnosis and classification using AWS DeepLens, providing the consumer with disease information in less than a second.

4.1 Introduction

The effects of plant disease on quantitative and qualitative production [5] are devastating, resulting in a striking blow to farmers , traders and consumers. A 14.1% relative disease loss across all crops was observed in a US-based study conducted by the U.G.A. Center for Agribusiness and Economic Growth [81]. A description of losses due to plant disease reported by the University of Georgia Extension in the 2017 Georgia Farm Gate Value Study (AR-18-01) [81].

Traditionally farmers detect and diagnose plant diseases through their observations and rely upon the opinions of local experts and their past experiences. An expert can determine whether or not a plant is healthy [6]. If a plant is found unhealthy,

noticeable symptoms on its leaves and fruits are observed and reported. Diagnosis of plant disease incorporates a substantially high degree of difficulty through visual examination of the symptoms on plant leaves. Because of this challenge and the huge number of grown plants and their existing phytopathological issues, even qualified agronomists and plant pathologists sometimes struggle to accurately identify particular diseases and are consequently driven to wrong assumptions and remedies [7]. Practical plant health assessment and diseases diagnosis can improve product quality and prevent production loss. Early detection and classification of crop disease are significant to secure the specific species production [8]. Various research studies have found that early detection of plant diseases is crucial as over the period, diseases start affecting the growth of their species, and their symptoms appear on the leaves [17]. When a plant got infected by a specific disease, and then significant symptoms are shown on the leaves, which help in the identification and classification of that particular disease [9]. It is therefore essential to control and assess disease outspread [10]. A specific fungus or bacterium is frequently associated with the colour, scale, form, and margins of spots and blight (lesions). Many fungi develop disease "signs", such as mould growth or fruiting bodies that appear in the dead area as dark specks. Early stages of bacterial infections that develop during humid weather on leaves or fruits sometimes appear as dark and water-soaked spots with a separate margin and often a halo, a lighter-coloured ring around the site. As in peach plant, for instance, the decayed area is small and looks similar in appearance to neighbouring healthy tissue at an early stage; therefore, it is tough to detect diseases [82].

In the exploration of the agricultural field, technology plays a vital role. With the use of various machine learning and image processing techniques, researchers are trying to explore plant disease detection and classification. It is difficult, time-consuming and unreliable to detect plant diseases manually. Since a health evaluation

is tedious and time-consuming for an individual plant in a large plot, this testing procedure is replicated over time [6]. A single plant may have different diseases having the same pattern of symptoms; moreover, various conditions of the plant show similar signs and symptoms [83], making it challenging to identify the specific disease. For instance, the key Grapevine Yellow (GY) symptoms are very common and outstanding in late summers, such as leaf discolouration, bunch drying and abnormal wood ripening, allowing GY to be recognized and, by and large, differentiated from other grapevine disorders that may exhibit similar alterations (e.g. leafroll or direct damage due to feeding of leafhopper). However, the expression of symptoms among different GYs is very standardized, so symptomatology is not helpful to distinguish one GY from another. Since phytoplasmas are poorly transmitted by grafting on woody plants and because the symptomatic response induced by various GY agents in Baco 22A is the same, even indexing on the hybrid Baco 22A, used in the past, did not help much [84][85].

Machine learning (ML) [11] algorithms are serving a lot in the process of classification and identification of plant diseases automation. ML helps in monitoring of health assessment of plant and predicting diseases in the plant at early stages [9]. With the time progression, new ML models evolved, such as SVM [12], VGG architectures [13], R-FCN [14], Faster R-CNN [15], SDD [16] and many others. The researchers used them for their experiments in the field of recognising and classifying images. Some of those are used in automation of Agriculture systems [17].

The advancement in deep learning (DL) [86] has provided promising results and solutions in crop disease diagnosis and classification. Islam et al., [87] presented the integration of machine learning and image processing for the detection and classification of leaf disease images. They developed an SVM model for potato disease detection and used potato leaves dataset, consisting of healthy leaves and diseased

leaves. For performance, they used performance parameters such as accuracy, sensitivity, recall and F1-score. Dubey et al., [88] came up with an image processing technique by using the K-Means algorithm for the detection and classification of apple fruit disease and then used multiclass SVM for training and testing images. Al-Amin et al., [9] trained their model for potato disease detection through Deep CNN, and they computed performance for analysing the result using parameters such as recall, precision and F1-score. This model achieved an accuracy of 98.33% in experiments. According to Sladojevic et al., [89] to learn features, CNN must be trained on a large dataset of a large number of images. They developed a CNN model for classification of leaves diseases of apple and tomato plants and the experimental accuracy findings of their research for numerous diseases trial with an accuracy of 96.3%. Miaomiao et al., [90] presented an effective solution for grape diseases detection as they mentioned that two entirely different basic models integrated, it would be more useful to obtain remarkable results and improve the accuracy of detection. Therefore, they proposed a UnitedModel based on the integration of GoogLeNet with ResNet, whereas GoogLeNet raises the total units for all layers of a network and ResNet to increase the total number of layers in a network. Ye Sun et al., [82] developed a model based on structured-illumination reflectance imaging (S.I.R.I.) for identification of peach fungal diseases. In their work, CNN and three image classification methods used for processing of ratio images, alternating component (AC) images and direct component (DC) images to detect the diseases and area of peach. As a result, they found that A.C. images performance is better than D.C. images in peach diseases detection and ratio images gave a high accuracy rate. Hyeon Park et al., [8] developed a CNN network of two convolutional and three fully connected layers, for disease detection in the strawberry plant. They worked on a small dataset of leaves images consisting of healthy leaves and a powdery mildew strawberries disease class.

Xiaoyue et al., [91] worked on four typical grapes diseases, and for detection, they proposed a Faster DR-IACNN detector, based on deep learning. They reported that their proposed detector automatically detects the diseased spots on grapes leaves, thus giving an excellent result for the detection of diseases in real-time. In order to detect leaves diseases in vegetables, Zhang et al., [92] come up with an RGB model colours based three channels CNN. Konstantinos et al., [7] detected and classified 25 plant diseases by using different CNN based architectures. They trained and tested their model on the open-source dataset named PlantVillage. However, the results obtained in terms of accuracy may differ from using the same dataset for both training and testing purposes.

According to the above-discussed studies, CNN [93][94] always played a significant role and is widely used in the detection and classification of different plant diseases and provided agreeably results. There were some limitations, however, such as a lack of usability due to hardware complexity problems, minimal scalability, inefficiency and minimal real-time inferences in real-world operational use. The recent development in cloud-based services and efficient deep learning has motivated us to devise a practical and scalable solution to agricultural problems, and this paper lies in the similar domain. We found that most of the images in the PlantVillage dataset are either white or grey background; however, the real-world situation is different and may contain other colours in the background. Thus model trained only on uniform background colour may result in low accuracy or wrong prediction. Therefore, to address this research issue, we used a combination of publically available PlantVillage dataset [95] and images collected from Tarnab Farm (an agriculture research institute, Pakistan) real-cultivation environment to achieve high accuracy and a robust model. For training and testing, we used AWS SageMaker, a Cloud-based environment for our proposed model known as DeepLens Classification and Detection Model

(DCDM) to identify and classify various fruits and vegetables leaves diseases, based on Deep Convolutional Neural Network (DCNN) [96]. After completion of training DCDM, it was deployed in the Internet of Things (IoT) device known as AWS DeepLens to make it a scalable and efficient real-time classification and identification model. AWS Deeplens is DL based high definition (H.D) video camera with 4 Mega-Pixel sensors for ML related projection integration and implementation.

With our DCDM, we evaluated seven different CNN architectures using accuracy results and computation time. Those CNN architectures include ResNet-50 [51], AlexNet [97], VGG-16 [13], VGG-19 [13], DenseNet [98], SqueezeNet [99] and DarkNet [100]. All these architectures were trained and tested keeping the environment constantly. Our DCDM model out-performed all other architectures in terms of computation time as well as performance-wise. It obtained an average accuracy rate of 98.78% on test images. Our findings are the first step towards a system based on an AWS DeepLens camera for plant disease diagnosis. Moreover, in our work, we also extracted feature maps [89] of an input image after passing through the CNN model and applied filters to visualise the activations through the CNN layers [101]. The overall flow of the proposed DCDM model is illustrated in Figure 4.1.

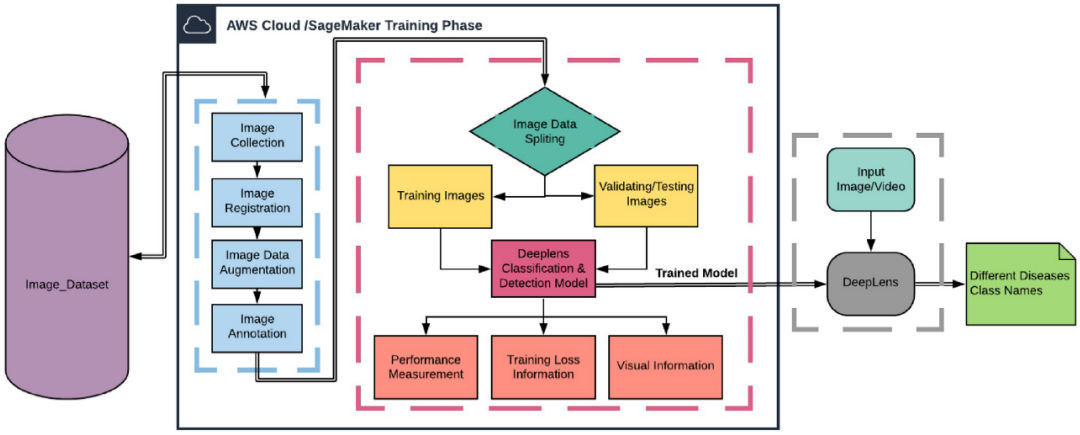


Figure 4.1: The data flow diagram of the DCDM that illustrates the process of our proposed disease diagnosis.

The rest of the paper is organised as follow: Section 2 explains the materials and methodology, including pre and post-processing datasets, describing the proposed CNN model, AWS DeepLens transfer learning, and performance assessment. A detailed overview of experimental results is given in section 3. Section 4 introduces the discussion, while Section 5 offers recommendations for conclusion and future work, followed by references part.

4.2 Materials and Methodology

The development process of the DCDM model for plant leaves disease detection, and classification involved various stages, i.e. starting with data collection along with data pre-processing and preparation, training model in AWS Cloud (SageMaker Studio) [102] and implementing in AWS DeepLens for inferences purpose. A strawberry plant is chosen for real-time disease assessment shown in Figure 4.2.

Figure 4.2: Identification & classification of strawberry plant leaf disease by AWS DeepLens in real-time.

4.2.1 Dataset Preparation

We used around 50,000 of plant leaves images (including both healthy and infected leaf images for fruit trees and vegetable plants) from local farmlands and publicly available dataset known as PlantVillage [95]. The dataset was categorised into different classes and assigned labels where each label is representing either a plant-leaf disease class or a healthy plant (leaf). A sample image for each class label shown in Figure 4.3.

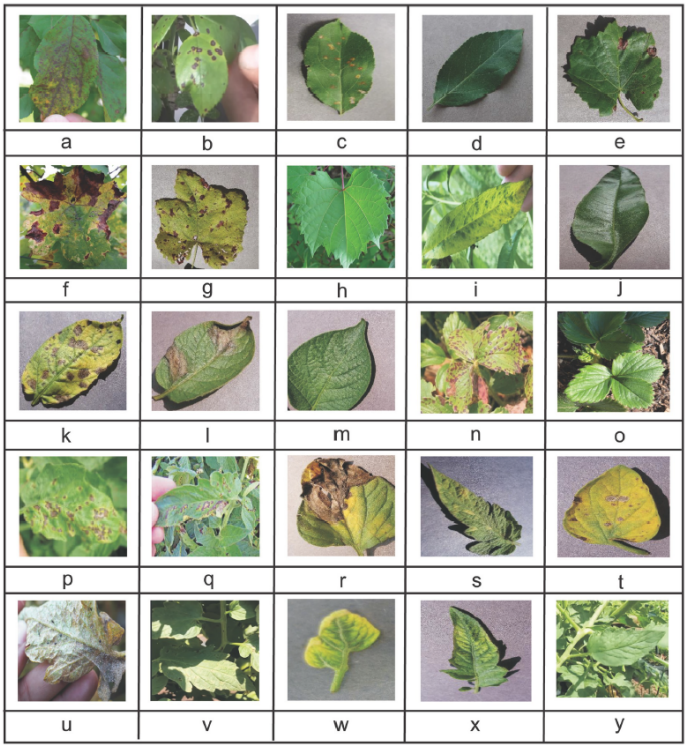


Figure 4.3: Sample images from dataset: (a). Apple Scab, (b). Black Rot, (c). Cedar Apple Rust, (d). Apple Healthy, (e). Grape Black Rot, (f). Grape Esca, (g). Grape Leaf Blight, (h). Grape Healthy, (i). Peach Bacterial Spot, (j). Peach Healthy, (k). Potato Early Blight, (l). Potato Late Blight, (m). Potato Healthy, (n). Strawberry Leaf Scorch, (o). Strawberry Healthy, (p). Tomato Bacterial Spot, (q). Tomato Early Blight, (r). Tomato Late Blight, (s). Tomato Leaf Mold, (t). Tomato Septoria Leaf Spot, (u). Tomato Spider Mites, (v). Tomato Target Spot, (w). Tomato Leaf Curl Virus, (x). Tomato Mosaic Virus, (y). Tomato Healthy.
From PlantVillage: (c), (d), (e), (g), (j), (k), (l), (m), (r), (s), (t), (w) and (z).
From Tarnab Farm: (a), (b), (f), (h), (i), (n), (o), (p), (q), (u), (v) and (y).

4.2.2 Data Augmentation

A large number of images are used to train a DCNN model to achieve highly precise prediction and accuracy. In our case, some of the plants leaves disease classes had fewer images in number; therefore, the process of data augmentation (technique) applied to those limited number of image diseases classes.

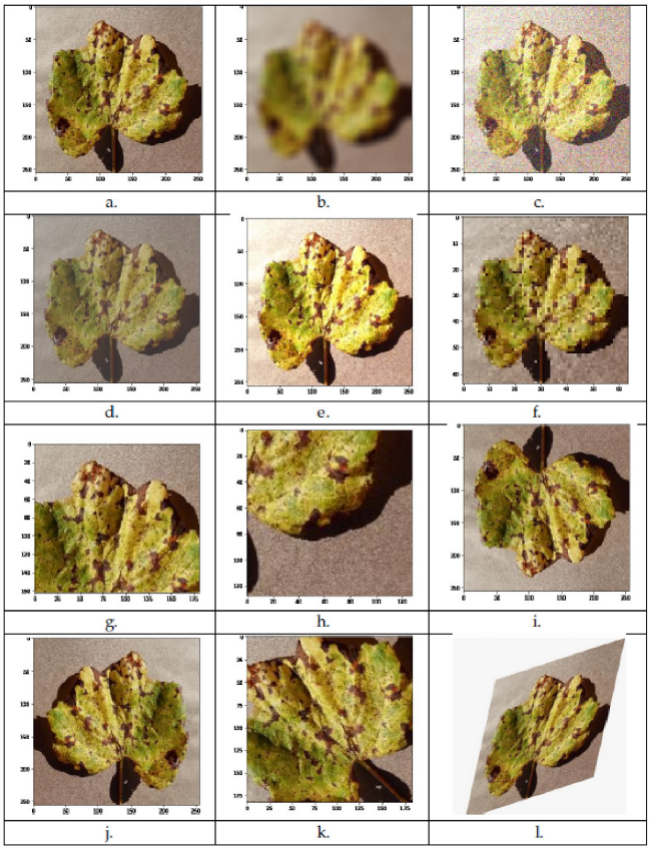


Figure 4.4: Data augmentation technique examples: (a). Original Image, (b). Blur, (c) Random Gaussian Noise, (d). Random Contrast, (e). Random Bright, (f). Scale Proportionality, (g). Random Crop, (h). Deterministic Crop, (i). Vertical Flip, (j). Horizontal Flip, (k). Rotate Without Padding, (l). Y-Sheared.

The process of data augmentation [103] provided us with new images from our existing images. Different augmentation techniques like blurriness, rotation, flipping (horizontal and vertical), shearing (horizontal and vertical), and the addition of noise

were applied accordingly. An illustration of different augmentation techniques shown in Figure 4.4. By using this technique, the number of images in our dataset increased, which is essential for obtaining more accurate results after the training stage of CNN.

4.2.3 Image Registration and Classes Annotation

After completion of data augmentation process, we had to re-register the images in the same dimensions, as we used two types of a dataset having different dimensions. Image registration is an essential step in image processing whenever two or more images are processed and analysed [104]. It is a method of overlaying images (two or more) of the same scene taken at different times, from different points of view and/or from different sensors. It aligns two images (reference and sensed images) geometrically [47][4]. We resized all the images into 272 x 363 pixels and annotated all the images before putting image as an input to any model/network for pre-training CNN structures. The classes of leaf diseases for fruits and vegetables that we used in our training and testing dataset are listed in the Table 4.1 with both regular and botanical names.

Table 4.1: The dataset for leaf disease classes.

Class No.	**Pant Name**	**Plant Botanical Name**	**Disease Name**	**Disease Botanical Name**	**Total Images**
1	Apple	Malus domestica	Scab	Venturia inaequalis	183
2	Apple	Malus domestica	Black rot	Botryosphaeria obtusa	182

3	Apple	Malus domestica	Cedar apple rust	Gymnosporangium juniperivirginianae	67
4	Apple (Healthy)	Malus domestica			172
5	Grapes	Vitis vinifera	Black rot	Guignardia bidwellii	218
6	Grapes	Vitis vinifera	Esca	Phaeomoniella chlamydospora	138
7	Grapes	Vitis vinifera	Leaf blight	Pseudocercospora vitis	207
8	Grapes (Healthy)	Vitis vinifera			182
9	Peach	Prunus persica	Bacterial spot	Xanthomonas campestris	229
10	Peach (Healthy)	Prunus persica			186
11	Potato	Solanum tuberosum	Early blight	Alternaria solani	200
12	Potato	Solanum tuberosum	Late blight	Phytophthora infestans	200
13	Potato (Healthy)	Solanum tuberosum			165

14	Straw-berry	Fragaria spp.	Leaf scorch	Diplocarpon earlianum	223
15	Straw-berry (Healthy)	Fragaria spp.			185
16	Tomato	Lycopersicum esculentum	Bacte-rial spot	Xanthomonas campestris pv. Vesicatoria	213
17	Tomato	Lycopersicum esculentum	Early blight	Alternaria solani	225
18	Tomato	Lycopersicum esculentum	Late blight	Phytophthora infestans	190
19	Tomato	Lycopersicum esculentum	Leaf mold	Fulvia fulva	225
20	Tomato	Lycopersicum esculentum	Septoria leaf spot	Septoria lycopersici	187
21	Tomato	Lycopersicum esculentum	Spider mites	Tetranychus urticae	167
22	Tomato	Lycopersicum esculentum	Target spot	Corynespora cassiicola	160
23	Tomato	Lycopersicum esculentum	Leaf curl virus		385
24	Tomato	Lycopersicum esculentum	Mosaic virus	Tomato mosaic virus	237

25	Tomato (Healthy)	Lycopersicum esculentum			165

4.2.4 CNN and DeepLens Classification and Detection Model (DCDM)

A typical CNN consists of various layers. Each layer consists of multiple nodes with some activation function attached. The first layer is the input layer that takes input data, whereas, the last layer is the output layer that generates output. A random number of layers exists between the input and output layer, referred to as hidden layers (i.e. convolutional or convo, pooling, dense or fully connected and softmax layer) [93][94]. If CNN contains two or more than two hidden layers, it is known as Deep Convolutional Neural Network (DCNN) [96].

We designed our DCDM using deep learning TensorFlow framework [105] and Keras [106] library. Keras is an open-source deep-learning library used to perform different deep learning applications. We used it for the implementation of DCDM architecture, inspired by Visual Geometry Group (VGG) Neural Networks. It is an advanced model of object-recognition supporting up to 16-19 weight layers [13]. Constructed as a deep CNN, VGG also out-performs baselines outside of ImageNet on several tasks and datasets. There are two variants of VGG Neural Networks namely VGG-16, which comprises of 16 convolutional layers and VGG19 comprises of 19 convolutional layers. VGG is also one of the most used architectures for image recognition today. This architecture uses filters of the same width and height for all the convolutional layers. The architecture of VGG-16 and VGG-19 out-performed than the other state-of-the-art architectures like ResNet-50, DenseNet, Inception-

VNet [107] as they converge very quickly and score over 90% accuracy during the first epochs of training. The VGG-19 architecture consists of roughly about 138 million parameters [108] while VGG-16 has less number of parameters due to less number of layers, however, a large number of parameter makes computationally expensive for training purpose.

Our proposed architecture has the same sequential structure as of VGG Neural Network but with some less number of layers, thus, the numbers of parameters are extensively low, which makes it computationally less expensive and fast.

Our DCDM architecture contains a total of nine layers with six convolutional layers and three fully connected layers shown in Figure 4.5.

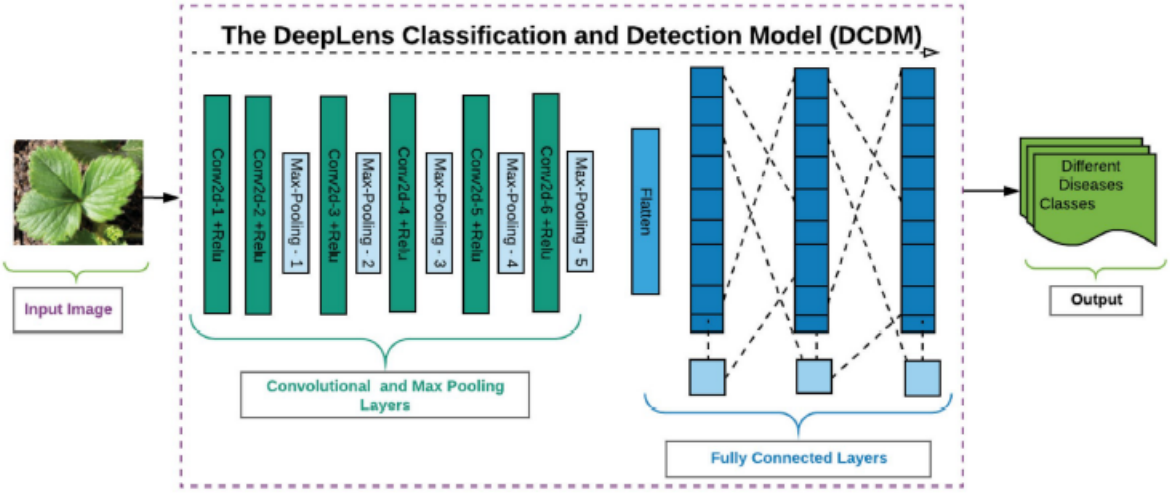


Figure 4.5: The representation of DeepLens classification and detection model (DCDM) architecture.

Convolutional layers are having non-linearity activation units following by max-pooling layers. The non-linearity activation is often used with convolutional layers. This activation is also known as a ramp function which has a shape of the ramp and transfers the output once it is a positive value; else it results in 0. The last layer, which is also known as a SoftMax layer comprising of 25 nodes in the output layer where each node specifies an individual class of our dataset.

The details of these layers are described below and shown in Table 4.2.

Convolutional Layer: The above stated proposed model used six convolutional layers. There are two types of characteristics in each layer, i.e., input and numeral filters. The filter numbers are then convolved on each layer which extracts the useful features and passes it to the next connected layer. For an RGB image, each filter is applied to all three colour channels, and thus, a corresponding matrix is obtained accordingly. We used a filter size of 3 x 3 for all convolutional layers. Pooling Layer: Most commonly, the pooling layer follows each convolutional layer. There are five max-pooling layers in the proposed method. The pooling layers are often used to minimise computational cost as it reduces the size of each convolutional layer output. The max-pooling has an activated filter which slides on the input and based on the size of the filter, and the max value is selected as an output. We used a filter of 2 x 2 for all max-pooling layer.

Dense Layer: It is also known as an artificial neural network (ANN) classifier. Our model has three dense or fully connected layers. In fully-connected layers, each node is connected with only one node of another layer. The first two fully-connected layers have ReLu activation during the last layer, which is also known as the output layer, has a softmax activation. The softmax activation works by finding the node with the highest probability value of prediction being made. Hence the node with the higher value is forwarded as an output.

Dropout: The overfitting issue is prevented by the addition of a dropout of 0.5. It is added to the dense layers of the model.

Parameters: The total model parameters of our model are 51,161,305.

The model takes the image data as an input, then processes that input data by extracting features from the image and then classifies it either healthy or a diseased leaf, if it is an infected leaf then it further predicts the disease class name, the most

resemble one. The expected class then results as an output.

Table 4.2: The summary Of DCDM layered architecture.

Layer (type)	**Output Shape**	**Param #**
conv2d (Conv2D)	(None, 272, 363, 64)	1792
$conv2d\_1(Conv2D)$	(None, 272, 363, 64)	36928
$max\_pooling2d(MaxPooling2D)$	(None, 136, 181, 64)	0
$conv2d\_2(Conv2D)$	(None, 136, 181, 128)	73856
$max\_pooling2d\_1(MaxPooling2D)$	(None, 68, 90, 128)	0
$conv2d\_3(Conv2D)$	(None, 68, 90, 256)	295168
$max\_pooling2d\_2(MaxPooling2D)$	(None, 34, 45, 256)	0
$conv2d\_4(Conv2D)$	(None, 34, 45, 512)	1180160
$max\_pooling2d\_3(MaxPooling2D)$	(None, 17, 22, 512)	0
$conv2d\_5(Conv2D)$	(None, 17, 22, 512)	2359808
$max\_pooling2d\_4(MaxPooling2D)$	(None, 8, 11, 512)	0
$flatten(Flatten)$	(None, 45056)	0
$dense(Dense)$	(None, 1024)	46138368
$dense\_1(Dense)$	(None, 1024)	1049600
$dense\_2(Dense)$	(None, 25)	25625
Total parameters: 51,161,305		
Trainable parameters: 51,161,305		
Non-trainable parameters: 0		

We made changes to the hyper-parameters shown in the Table 4.3 to optimise our model. We selected the optimizer of Stochastic Gradient Descent (SGD), proving to be an optimal trade-off between accuracy and effectiveness [109]. The SGD is clear and reliable. The hyper-parameters model to be tuned, in particular the initial

Table 4.3: Hyper-parameters of the experiments.

Hyper-Parameters	**Value**
Optimizer	SGD
Momentum	0.9
Epochs	50
Batch Size	32
Dropout rate	0.5
No. of Layer	9
Learning Rate	$1.0x10^{-3}$
Loss Function	Cross Entropy

learning rate, which is used in optimization as it explains how rapidly the weights are altered to achieve a minimum of the local or global loss function. The momentum (= 0.9) tends to accelerate SGD in the correct direction and dampens the oscillations [110]. In addition, regularisation is a very effective method to avoid over-fitting. The most common way of regularisation is L2 Regularization, where the combination with SGD results in weight decay, in which the weights for each update are scaled by a factor slightly smaller than one [111]. A total of 50 epochs are performed in each experiment, where each epoch is the number of training iterations. Finally, DCDM is trained at a batch size of 32 and stopped training on epoch-50.

4.2.5 Transfer Learning in AWS Cloud

Transfer learning (TL) is a concept in the ML which simply means that a method learns basic knowledge in solving a particular problem and later reusing that knowledge for other more or less similar problem solution [112]. This technique encourages us to use for solving any relevant problem for which there is not sufficient data available. Thus it relaxed the assumption of training and testing data, should be both distributed identically and independently [113]. It takes a long time and large-sized dataset for training CNN from scratch. Hence in certain situations where the

dataset is limited, then TL is a helpful method. We used TL from scratch for DCDM training. Amazon's Cloud platform and AWS DeepLens were selected to address the scalability constraints. The Amazon cloud infrastructure offers data collection , data transfer and computing resources for the application development and deployment. AWS provides many services and several different applications. They also have a platform for building, preparation and rollout, as well as validating models of machine learning. On AWS Services or some other compatible systems , for example, AWS DeepLens, the trained model can be deployed.

4.2.6 Lambda Function on DeepLens

AWS DeepLens is a deep-learning-based high definition (H.D), 4-mega-pixel video camera that is designed specifically for machine learning models developments and implementation. It has a built-in 8GB memory and 16GB storage capacity with 32 GB SD card (extendable). It has more than 100 GFLOPS computing power so it can process machine learning projects independently as well as those integrated with AWS Cloud [114]. It has a straightforward usage process as the user can take picture/ image through DeepLens camera, then store it and process it to use in machine learning projects [115]. There are a large number of pre-trained models, built-in to it, but a customised model can also be used with DeepLens camera. For instance, any custom based model can be trained or imported into in SageMaker and then can be implemented in AWS DeepLens through various deep learning frameworks such as Tensorflow or Caffe [114],[115]. A lambda function is used to establish a successful connection to access the DeepLens on a local computer. The lambda functions are the pre-defined functions executed by DeepLens once the project has been deployed [116]. Lambda function streamlines the development process by managing the servers necessary to execute code. They serve as the connection between the AWS DeepLens

and Amazon Sagemaker for the camera to generate a real-time inference [117]. It controls various resources such as computing capability and power, networking. It has a user-specified function embedded in code, and Lambda function invoke that user code when it is executed. The code returns a message containing data from the event received as input [117]. The visual illustration of the AWS DeepLens work-flow is shown in Figure 4.6.

After completing the training stage in SageMaker, we implemented the subsequent trained Model in AWS DeepLens camera for inferences of Leaves health assessment.

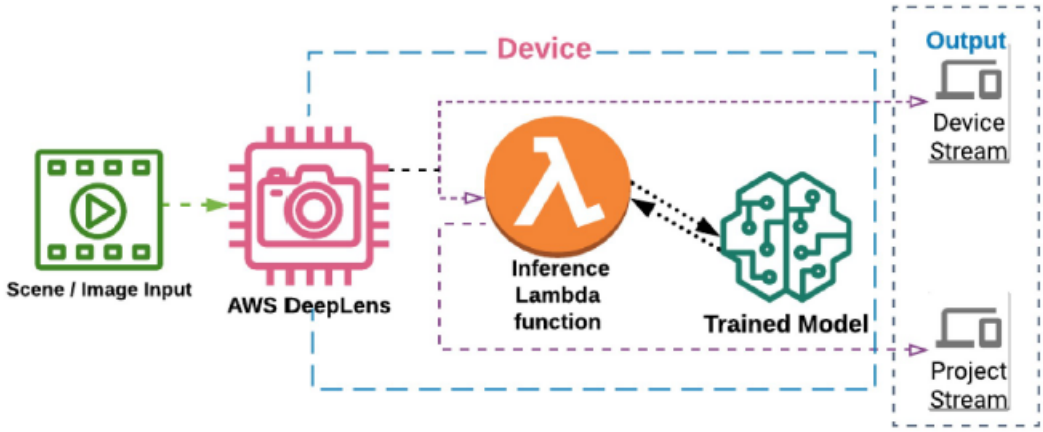


Figure 4.6: Basic workflow of a deployed AWS DeepLens project [1].

4.2.7 Evaluation and Performance Measurement

Several methods are used to measure the efficiency of neural networks, including precision, recall, accuracy, and f1-score. The precision tells us about the correct predictions made out of false-positive while recall tells us about the accurate predictions made out of false negatives. The accuracy is the number of correct predictions out of both false positives and false negatives. All the performance metrics for our trained model have been determined using the formulas in Eq (1), (2), (3), and (4) are listed.

We calculated the values from the confusion matrix shown in Figure 4.10.

$$Precision = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4.1}$$

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.2}$$

$$Accuracy = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}} \tag{4.3}$$

$$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4.4}$$

Where TP is true positives, TN is true negatives, FP is false positives and FN is false negatives. Here the TP and TN are the correct predictions while the FP and FN are the wrong predictions made by our model.

4.2.8 Features Maps Extraction and Filters Visualization in CNN Layers

4.2.8.1 Extraction of Feature Maps

Feature maps [118] are used to present the local information passing through the CNN Layers. In an ideal feature mapping of CNN, they are sparse and help in the understanding of the classical model. In convolutional layer, to extract feature maps from the source image, several mathematical computations are carried out [119]. In Figure 4.7, a visual representation for the extraction of feature maps presented for various layers of our model. It also provides information about each layer, i.e. what

and how a particular layer of CNN gains information from other layers, such piece of information can help the developer to make proper adjustments in the developing model for best results. From our visualisation images, we found that our model is gaining information in the hierarchical order. It means that the high-level layers present more specific features and vice versa.

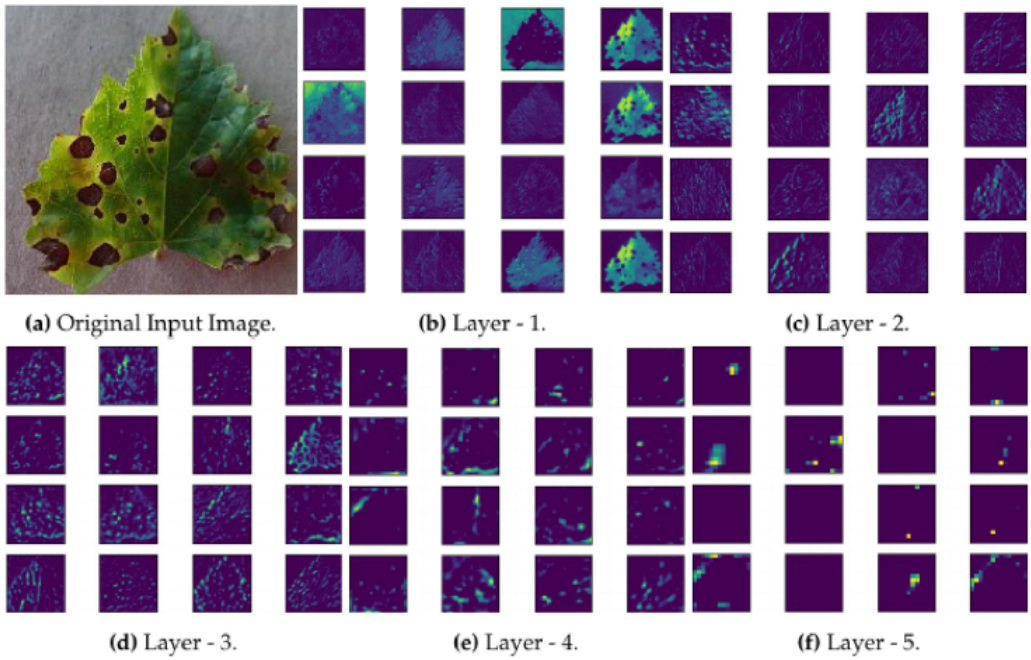


Figure 4.7: Visualization of feature map from DCDM convolutional layer for a sample leaf.

Similarly, if the dimensions are higher than feature maps, images would also be more accurately classified. For instance, in an image, the edge corners and some abstract colour features are presented by a deep layer Figure 4.7, while other corners and edges represented in shallows layers. Moreover, the middle layers are usually responsible for capturing the same textures because these layers are having complex invariance and more layers in number, after extracting higher-level abstract features, the striking posture of the entire image shown by the high-level feature map.

The feature maps extracted in the first layer represents the overall physical ap-

pearance of the leaf image. In the middle layers, the patterns of disease are extracted as can be seen in Figure 4.7. The last layers in Figure 4.7 often extract the delicate features as they are then used to finalise the predicted class.

4.2.9 Filter Visualization in Model Layers

Generally, filters are used for the detection of unique patterns in an input image. It is done by detecting the change in the intensity values of the image. Thus, each filter has its particular importance for feature extraction [120]. As an example, a high pass filter detects the existence of edges in an image. In our DCDM model, various filters are used to extract features like edges, shape, the colour of the leaf, and many more useful features.

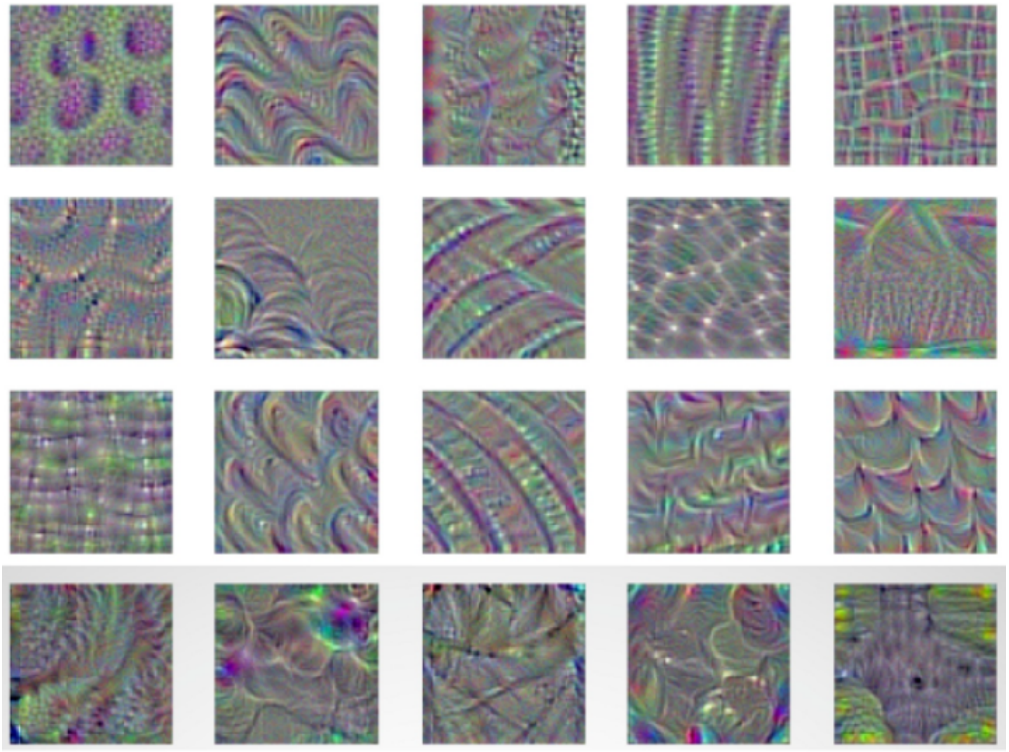


Figure 4.8: Visualisation of filter activation in DCDM convolution layers.

In Figure 4.8, a visual representation for a few filters presented where each filter

has its application for extracting leaf features. After detecting the specific feature of the image by a filter, it is then passed to the next layer where other filters extract the additional feature. This process continues until the last layer, and thus integrating all together helps to define the predicted class for an input image.

4.3 Experimental Results

The entire dataset was distributed into different training sets (80%, 70% and 60%) and testing data (20%, 30% and 40%) for performance evaluation, as shown in Table 4.4. The model was using 10% of the each training set split for validation purpose during its training phase.

Table 4.4: Dataset split for training and testing.

Train - Test Data Split (%)	**Training Images**	**Testing Images**
80 - 20	40000	10000
70 - 30	35000	15000
60 - 40	30000	20000

The performance indicator, accuracy for each data split is shown in the Table 4.5. After every ten epochs of preparation, the values are presented. However, in comparison with another train-test dataset splits for DCDM model performance evaluation, the data split of 80% – 20% performed very well at the epoch scale of 50 with the maximum accuracy of 98.78% as shown in Table 4.5.

Table 4.5: Dataset split for training/testing and accuracy obtained per epoch.

Dataset (Train/Test) Split in %	**Accuracy [%]**				
	10 Epochs	**20 Epochs**	**30 Epochs**	**40 Epochs**	**50 Epochs**
80 – 20	92.31	95.84	96.86	97.39	**98.78**
70 – 30	91.23	94.89	96.15	96.77	97.46
60 – 40	90.70	94.92	95.04	95.98	96.21

In Figure 4.9 (a) (b), the accuracy and loss for both training and testing/validating are presented for each epoch. These graphs were generated for the data split of 80% – 20%. The accuracy graph visually shows that accuracy increases gradually for both training and testing, and then tends to converge on a specific point. It also shows that after 40 epochs, the change in accuracy reduces as the validation accuracy appears to be equivalent to training accuracy. Similarly, the right graph shows how the loss starts decreasing gradually as the model learns on a given dataset. The loss of validation data becomes stable after 43 epochs and thus tends towards a specific value.

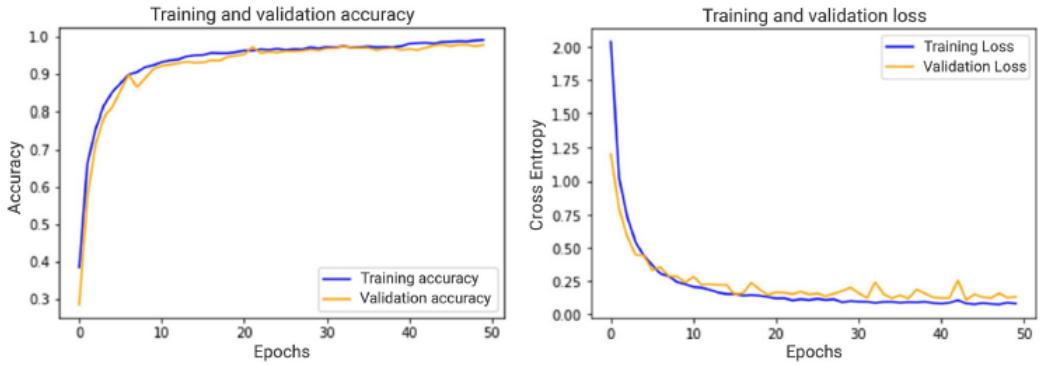


Figure 4.9: Trend graph for accuracy and loss in training and validation.

The validation process gives a confusion matrix shown in Figure 4.10. After

computing values from the confusion matrix, the results are shown for the 80%-20% split ratio in Table 4.6.

Table 4.6: DCDM performance report.

Evaluation Metrics	**Value in %**
Precision	98.38%
Recall	97.98%
Accuracy	98.78%
F1-Score	98.17%

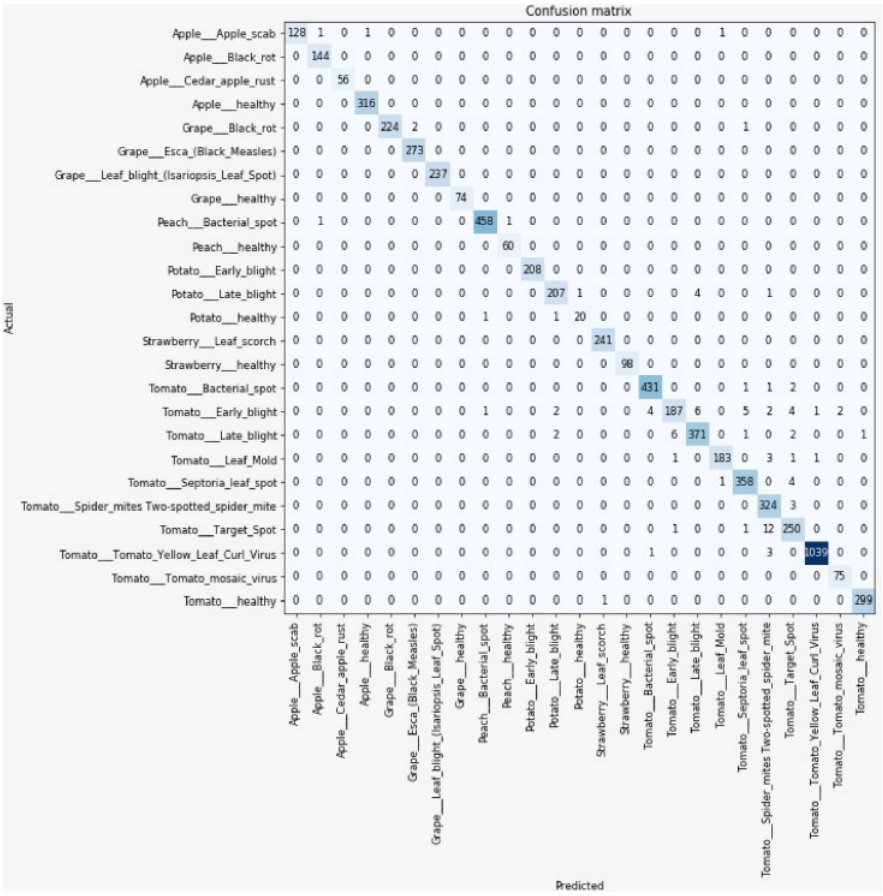


Figure 4.10: Confusion matrix for 80 -20 % dataset split set.

The confusion matrix shows the predictions made by 80-20 dataset split are presented in the Figure 4.10. The matrix displays the number of predictions that

are true and false. It also offers the information for which class is more reliably predicted and vice versa. The groups of Apple Cedar Rust, Grape Leaf Blight, Grape Healthy, Potato Early Blight, and Strawberry Healthy have been correctly predicted, so these groups have not been wrongly predicted. While the Tomato Early Blight, Tomato Late Blight, Tomato Spider Mites, and Tomato Goal Spot classes have the most inaccurate predictions from other classes. Likewise, the Potato Late Blight and Tomato Septoria Leaf Spot groups have an average number of false predictions. The remaining groups are expected with a minimum number of incorrect predictions, such as Apple Scab, Apple Black Rot, Apple Safe, etc.

Some of the sample output images with an AWS DeepLens are shown in Figure 4.11.

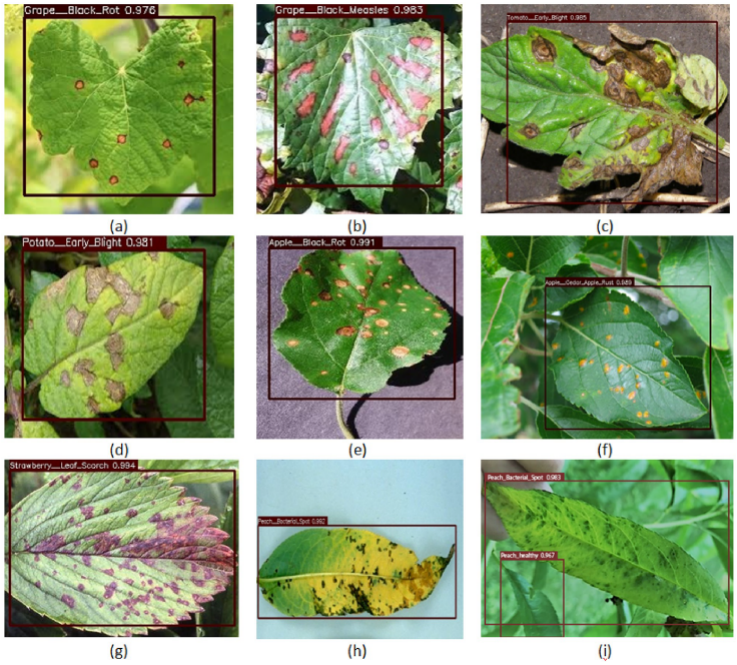


Figure 4.11: Sample results from real field and controlled environment images.

4.3.1 Comparative Analysis

A comparative overview of different CNN architectures with the DCDM, is given in this section. Training the model on different architectures is a crucial approach used to define the best architecture for targeted application. The architectures we used for the identification and classification problem are the highest performing architectures. We compared the performance of DCDM, ResNet-50 [51], DensNet [98], VGG-16 [13], VGG-19 [13], AlexNet [97], SqueezeNet [99] and DarkNet [100] architecture for each training and testing dataset split using same hyper-parameters. An evaluation metric of accuracy was used for comparison, based on Equation 3.

For each CNN architecture, we obtained an output accuracy of more than 90%. AlexNet architecture works with the lowest precision of 92.43%. This architecture is considered to be the smallest and most simple architecture of all. However, it still provides us with a accuracy of over 90%. With some slight changes and a different number of layers, the VGG-16 and VGG-19 architectures are the same. For the classification challenges, they have an important record of doing very well. They provide us with an accuracy of 94.05% and 96.89% respectively for our research dataset. Similarly, SqueezeNet and DenseNet architecture also performed with an accuracy of 94.67% and 96.59%. The ResNet-50 architecture is well-known for good performance on large datasets. It has a bulk of 50 layers with different inter-connections. Thus, performing with an accuracy of 97.85% and being able to score the position of the third-best architecture in our list. The architecture of DarkNet provides an accuracy close to DCDM model. It results from the accuracy of 98.21%, scoring the position of second-best architecture while DCDM architecture performed outstanding and stood with the position of best architecture with an accuracy of 98.78%. The results for each architecture based on accuracy is visually represented in Figure 4.12.

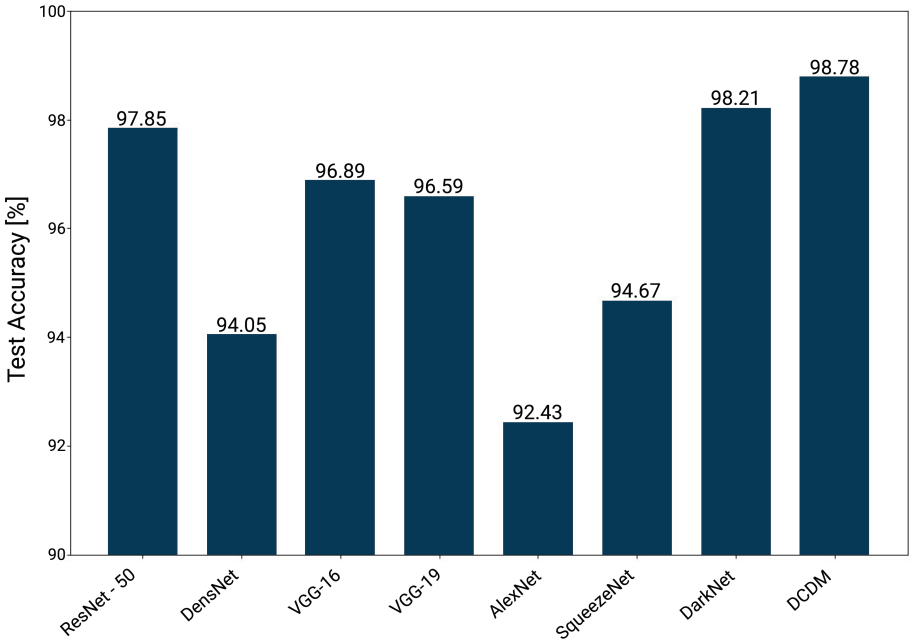


Figure 4.12: Average accuracy obtained by each CNN model.

The comparison of each architecture concerning time consumed has also been made which implies the time required for training. The time consumed by our architecture requires less computation time, thus having the lowest average time during training per epoch. It testifies that our architecture is the most efficient both performance as well as computation wise. The results for each architecture on the basis of computation time taken per epoch is shown in Table 4.7.

Table 4.7: Average time consumed by CNN's per epoch.

Trained CNN Models	**Average Time Per Epoch (in Minutes)**
ResNet-50	2:03
DensNet	2:38
VGG-16	1:53
VGG-19	1:59
AlexNet	1:44
SqueezeNet	2:32
DarkNet	2:13
DCDM	**1:26**

The DCDM model realises higher convergence speed and greater accuracy during the training phase relative to the regular CNN architectures (ResNet-50, VGG-16, VGG-19, DensNet, AlexNet, SqueezeNet, DarkNet, etc.). The findings of this study show that end-to-end classification of plant leaf diseases is realised by the proposed algorithm and offers a solution and a reference for the implementation of deep learning approaches in plant disease classification.

4.4 Discussion

The conventional approach to image classification methods focused on hand-engineered features such as SIFT [121], HoG [122] and SURF [123] etc., has previously been used to remove features from pictures. Thus, relying heavily on the pre-defined features underlying them, added to the success of all these methods. Function descriptors themselves are a complicated and repetitive process that may be revisited if there is a substantial change in the topic at hand or the parameters of the corresponding dataset. In all conventional attempts to diagnose plant diseases utilizing image recognition, this difficulty has arisen because they rely solely on hand-engineered features, techniques of image enhancement, and a variety of several other challenging and exhaustive methodologies [101].

DL has significantly advanced in many research areas. Deep Neural Network for Convolution (CNN) Architectures [124] have become famous recently as they eliminate the dependency on explicit hand-crafted features instead, learn strong feature representations directly, from raw data. The integration of these deep neural networks features at different specificity levels (ranging from low-level functions such as edges to abstract high-level features such as objects) [125] and comprehensive classifiers, fashion from end-to-end. Indeed, the architecture of Deep CNN has

state-of-the-art performance obtained on image classification tasks [126],[97].

We find that most of the images in the PlantVillage dataset are either white or grey background, but the real-world situation is different and can include other background colours. Thus, the only uniform background colour trained in the model will result in low accuracy of false prediction. To achieve high accuracy and a stable model, we used a mix of PlantVillage dataset and images gathered from Tarnab Farm, Pakistan the real-cultivation and research environment. We applied various data augmentation techniques to the training data to maximize the number of those leaf disease classes where they were less in number. Thus the processed dataset comprised of around fifty thousand images of twenty-five different infected and healthy plant leave classes from six plants i.e. apple, grapes, peach, strawberry, potato and tomato.

We proposed a DeepLens Classification and Detection Model (DCDM) to recognise and diagnose multiple fruit trees and vegetable plant leaf diseases. We used a cloud-based environment for DCDM training and testing to address the concerns of scalability and applicability. It was deployed on AWS DeepLens after completion of DCDM training. For ML projects, AWS Deeplens is a DL-based camera with a 4 mega-pixel high definition (HD) sensor.

We compared DCDM with seven different CNN architectures utilising performance accuracy and computation time. ResNet-50 [51], AlexNet [97], VGG-16 [13], VGG-19 [13], DenseNet [98], SqueezeNet [99] and DarkNet [100] are included in these CNN architectures. All of these models have been trained and tested under the same environment, i.e. same dataset set was used for training and testing phases using the same hyper-parameters for all. All other architectures exceeded our DCDM model in terms of computing time, as shown in Table 4.7. On real field and test images, DCDM obtained an overall accuracy rate of 98.78%, which is higher than others as shown in Figure 4.12. Our study findings are the first step towards a system for

plant disease diagnosis based on an AWS DeepLens camera.

At the current point, however, there are a range of weaknesses that need to be dealt with in future work. Firstly, in addition to AWS Deeplens, it can be easily implemented in the future on multiple mobile platforms such as iOS, Android or Windows-based mobile applications due to the fast classification process of our model. Secondly, More plant species will be introduced to make this model more scalable in the future. As there are few plant species at present, they are included and evaluated. Lastly, in future work, modern techniques such as Multi-spectral and Hyper-spectral images should also be tested for the detection and classification of plant diseases.

4.5 Conclusion

With this proposed deep model applied on AWS DeepLens, 25 separate disease classes in Apple, Grape, Peach, Potato, Strawberry and Tomatoes can be predicted in real-time. In real-time predictions and classifications for field experiments, our model gained 98.78% accuracy. This practical method would facilitate the practitioners and society relevant to agriculture by contributing to the enhancement of the agri-economy, as the severe issue of plant (leaves) diseases, can be instantly recognised and classified. In addition, this approach is scalable, and it can also be used as an online repository for plant leaves disease identification and classification. More classes of other vegetables and fruit leaves can also be added in future. To improve its usability and applicability, we will incorporate our model into various mobile platforms such as iOS, Windows and Android-based applications in our future work. Thus, due to regular smartphone use, the functionality would become more flexible and easy to use. Moreover, new techniques such as multi-spectral and hyper-spectral

images should also be evaluated in future work for the identification and classification of plant diseases.

Chapter 5

Health Assessment of Eucalyptus Trees using Siamese Network from Google Street and Ground Truth Images

As mentioned in list of publications, this chapter with same title was published as an original research paper in the Remote Sensing, (2021), 13(11), 2194, an official journal of MDPI, doi: https://doi.org/10.3390/rs13112194. The contents are the same, with the exception of certain layout adjustments to ensure consistency in the presentation across the thesis.

Abstract

Urban greenery is an essential characteristic of the urban ecosystem, which offers various advantages, such as improved air quality, human health facilities, storm-water run-off control, carbon reduction and an increase in property values. Therefore, identification and continuous monitoring of the vegetation (trees) has vital importance in our urban lifestyle. This paper proposes a deep learning-based network, Siamese

convolutional neural network (SCNN), combined with a modified brute-force-base line-of-bearing (LOB) algorithm that evaluates the health of eucalyptus trees as healthy or unhealthy and identifies their geo-location in real-time from Google Street View (GSV) and ground truth images. Our dataset represents eucalyptus trees' various details from multiple viewpoints, scales and different shapes to texture. The experiments carried out in the Wyndham city council area in the state of Victoria, Australia. Our approach obtained an average accuracy of 93.2% in identifying healthy and unhealthy trees after training on around 4500 images and testing on 500 images. This study helps in identifying the eucalyptus tree with health issues or dead trees in an automated way that can facilitate urban green management and the local council to decide about the plantation and improvement in looking after trees. Overall, this study shows that even in a complex background, most healthy and unhealthy eucalyptus trees detected by our deep learning algorithm in real-time.

5.1 Introduction

Street trees are an essential feature of urban or metropolitan areas, although relatively ignored. Their benefits include air filtering, water interception, cooling, minimising energy consumption, erosion reduction, pollution management, and run-off detection [127], [128]. Various trees are planted in urban areas due to street trees' social, economic and environmental advantages. One such tree, eucalyptus, is a valuable asset for communities in urban areas (Australia). Eucalyptus trees are icons of the Australian flora, often called gum trees. They dominate the Australian landscape with more than 800 species, forming forests, woodlands and shrub-lands in all environments, except for the aridest deserts. Evidence from DNA sequencing and fossil discovery shows that eucalyptus had its evolutionary roots in Gondwana when

Australia was still linked to Antarctica [129]. Traditionally, indigenous Australians have used almost all parts of eucalyptus trees. Leaves and leaf oils have medicinal properties, and saps may be used as adhesive resins; bark and wood were used to make vessels, tools and weapons, such as spears and clubs [130]. For the conservation of Australia's rich biodiversity, eucalyptus native forests are significant.

There are two factors that are detrimental to the health of street trees. Firstly, urban trees are under persistent strain, i.e. excessive soil moisture and soil mounding in nurseries on roots that have an adverse effect on their health [131]. Secondly, urban ecosystem distinguished by elevated peak temperatures relative to nearby rural areas [132], soil compaction, limited growth of roots, pollution of groundwater [133], and high air pollution concentrations caused by community activities. Usually, urban soil contains a significant volume of static building waste, contaminants, de-icing salts, low soil quality and a significant degree of volume density, thus maintaining a low natural activity and the inferior organic material substance provided [134], [135]. Both of these reasons raise the likelihood of water and nutrient pressure, which degrades the metabolism and development of a tree and reduces its capacity to provide ecosystem services. Urban tree conditions are adversely affected due to soil compaction, low hydraulic conductivity, low compaction aeration and mostly insufficient available rooting space [135]. In addition, inadequate conditions at the site raise the threat of insect disease, and infestation [132].

The evaluation of tree health conditions is highly critical for biodiversity, forest management, global environmental monitoring and carbon dynamics. Unhealthy tree features are identifiable and can build a detection and classification model using deep learning to intelligently diagnose eucalyptus in a healthy and unsanitary/dead tree. To consider the importance of urban trees to the community, they should be adequately maintained, including obstacle prevention, regeneration and substi-

tution of dead or unhealthy trees. Ideally, skilled green managers need to monitor the precise and consistent spatial data on tree's health. About 60% of the riparian tree vegetation in extensive wetlands and floodplains reported being in poor health, or extinct [136]. Chronic decreases are associated with extreme weather conditions due to human resources management, various pathogens, pests and various parasites. Trees are stressed [137] in the landscape, where the soil has a poor drainage mechanism, also resulting in low growth of trees. The most common factors such as soil erosion, nutrient deficiency, allelopathy, biodiversity, pests, and diseases affect eucalyptus species' health.

Detection and recognition of eucalyptus tree health presents a challenging task since many trees have few pixels across input images, and some trees are also overshadowed by other trees and cannot be found due to weather conditions or lighting. For addressing these challenges and achieving high accuracy and precise prediction, a large amount of labelled training data for feature extraction of healthy and unhealthy class features is required. For this purpose, we used GSV imagery and ground truth images were obtained from various viewpoints and at different times. This study uses the Siamese Convolutional Neural Network (SCNN) [138], to develop an automated model for identification and classification and a line-of-bearing measurement approach paired with a spatial aggregation approach is used to estimate the geolocation of the eucalyptus tree. We concentrated on the identification of healthy and unhealthy eucalyptus trees along the streets and roads in the Wyndham city council area [139]. This study aims to use a self-created ground truth and GSV [140] imagery for finding the geo-location, identification and classification of healthy and unhealthy eucalyptus trees to prevent damage that can significantly reduce ecosystem harm and financial loss. GSV is an open image series of streetwise panoramic views with approximate precise geo-location details acquired on mobile platforms

using GPS, wheel encoder, and inertial navigation sensor (using multiple sources such as cars, trekkers and boats) [141]. This GSV has been widely used to increase geographical information in a variety of areas of interest, including urban greenery [2], [142], land use classification [143],[144] and tree shade provision [145].

Our key contributions are a.) classification of trees that are in a healthy or unhealthy state and b.) identification of geo-location of the eucalyptus trees. All these evaluations are done based on and self-gathered ground truth data from streets. Our experiments show that this proposed method can effectively detect and classify healthy and unhealthy eucalyptus trees with various dataset and complex backgrounds. Our proposed method for geo-location identification gives us reliable results and could be applied for geo-identification of other objects on the roadside. Figure 5.1 shows the overall visual representation of this study.

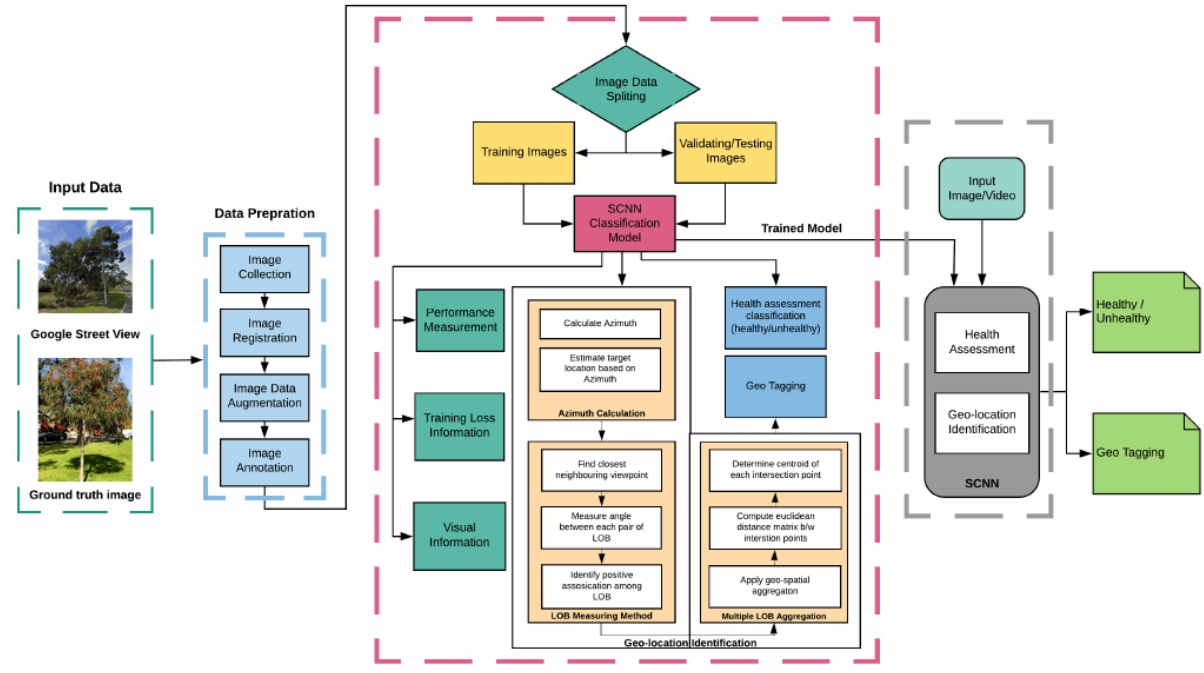


Figure 5.1: General workflow diagram of classifying healthy or unhealthy and geo-tagging eucalyptus trees from GSV and ground truth images.

5.2 Related Work

Numerous work has been done on detection and recognition in various areas such as fruits and vegetable plant leaves disease detection [146], vegetation detection [147], pedestrian detection [148], face detection [149], object detection [150], using various deep learning algorithms [151]. Automatic data analysis in the remote sensing (RS) [152] and computer vision [153] field is of vital significance. RS data has been used in urban areas to assess trees health. A large volume of the study shows various RS techniques used to determine the current condition of trees. In contrast, on the other side, a minimal amount of research shows interest in the identification and classification of dead trees. Milto Miltiadou et al., [154] presented a new way to detect dead eucalyptus camaldulensis with the introduction of DASOS (feature vector extraction). They tried to explore the probability of dead trees detection without tree delineation from Voxel-based full-waveform (FW) LiDAR. Shendryk et al., [155] suggested a bottom-up algorithm to detect eucalyptus tree trunks and the delineation of individual trees with complex shapes. Agnieszka Kamińska et al., [156] used remote sensing techniques, including airborne laser scanner and colour infrared imagery, to classify between living or dead trees and concluded that only airborne laser scanner detects dead tree at the single tree level.

Martin Weinmann et al., [157] proposed a novel two-step approach to detect a single tree in heavily sampled 3D point cloud data obtained from urban locations and tackled semantical classification by assignment of semantic class labelling to irregularly separated 3D points and semantic segmentation by separating individual items within the named 3D points. S. Briechle et al., [158] worked on the PointNet++ 3D deep neural network with the combination of imagery data (LiDAR and multispectral) to classify various species as well as standing dead tree crowns. The

values of laser echo pulse width and multispectral characteristics were also introduced into the classification process, and individual tree's 3D segments were created in a pre-processing stage of a 3D detection system. Yousef Taghi Mollaei et al., [159] developed an object-oriented model using high-resolution images to map the pest-dried trees. The findings confirm that the object-oriented approach can classify the dried surfaces with precise detail and high accuracy. W. Yao et al., [160] proposed an approach to individual dead tree identification using LiDAR data in mountain forests. The three-dimensional coordinates were derived from laser beam reflexes, pulse intensity and width using waveform breakdowns and 3D single trees were then detected by an optimized method that describes both the dominated trees and small under-story trees within the canopy model.

According to Xiaoling Deng et al., [161],[162] machine learning has been used to set several benchmarks in the field of agriculture. W. Yao et al.[160] and Shendryk et al., [163] published their prior work on the identification of dead trees is performed by individual tree crown segmentation prior to the health assessment. Meng R. et al., [164], Shendryk et al., [155], López-López M et al., [165], Barnes et al., [166], Fassnacht et al., [167], mentioned that most of the current tree health studies centred either on evaluating the defoliation of the tree crown or the overall health status of the tree, although there was minimal exposure to the discolouration of the tree crown. Dengkai et al., [168] used a group of fields assessed tree health indicators to define tree health that was classified with a Random Forest classifier using airborne laser scanning (ALS) data and hyperspectral imagery (HSI). They compared the outcomes of ALS data and HIS and also their combination and then analysed the accuracy degree of classification. Nasi et al., [169], [170] reported in two different pieces of research that the potential of UAV-based photogrammetry and HSI for mapping bark beetle in an urban forest, damage at tree level. Degerickx et al., [171]

performed tree health classification based on chlorophyll and leaf area index derived from HSI, where for individual tree crown segmentation, they used ALS data. Xiao et al., [172] used normalised difference vegetation index (NDVI) to detect healthy and unhealthy trees. They found it challenging to map tree health across various species or in places where many tree species coexist. Goldbergs et al., [173] evaluated local maxima and watershed models for the detection of individual trees, and they found the efficient performance of these models for dominant and co-dominant trees. Fabian et al., [174] presented their work on random forest regression to predict total trees using local maxima and a classification process to identify a tree, soil and shadow. Li et al., [175] introduced a Field-Programmable Gate Array (FPGA) for tree crown detection, significantly rapid calculations without loss of functioning.

Siamese network [138] has been used in a variety of applications, including object tracking [176], plant leaves disease detection [177], signature verification [178], railway track switches [179], coronavirus diseases detection [180]. H Bromley et al., [181] proposed a neural network model for signature matching by introducing for the very first time Siamese network. Bin Wang et al., [182] presented a few-shot learning method for leaf classification with a small sample size based on the Siamese network. However, we are using a Siamese convolutional neural network (SCNN) combined with a modified brute-force-based line-of-bearing (LOB) algorithm to classify eucalyptus trees as healthy or unhealthy and to find their geo-location.

5.3 Material and Methods

5.3.1 Study Area and GIS Data

The Wyndham city council (VIC, Australia) area [139] was chosen as the study area, as shown in Figure 5.2. It is located on Melbourne's western outskirts and covers

an area of 542 km^2 and has a coastline of 27.4 km. It has an estimated resident population of 270,478, according to the 2019 census. Wyndham is currently the

Figure 5.2: a.) Location of the study area in Victoria, Australia. b.) Suburbs in the Wyndham city council.

third fastest-growing local council in Victoria. Wyndham's population is growing and diverse, and the community forecasts indicate the population will be more than 330,000 by 2031 [139]. There are 19 suburbs (Cocoroc, Eynesbury, Hoppers Crossing, Laverton North, Laverton RAAF, Little River, Mambourin, Mount Cottrell, Point Cook, Quandong, Tarneit, Truganina, Werribee, Werribee South, Williams Landing, Wyndham Vale) in Wyndham [183]. Wyndham City Council is committed to enhancing the environment and liveability of residents. As part of this commitment, thousands of new trees are planted each year to increase Wyndham's tree canopy cover through the street tree planting program.

5.3.2 Google Street View (GSV) Imagery

The orientation of eucalyptus trees (healthy and unhealthy) in a 360° GSV can be identified by GSV images. Images of the static street view have been downloaded via the GSV image application programming interface (API) [184] by supplying the corresponding parameter information with uniform resource locators (URLs) [185].

The GSV API snaps the requested coordinates automatically to the nearest GSV viewpoint. We have taken four GSV images with the fov of 90° and headings of 0°, 90°, 180°, 270°, respectively as shown in Figure 5.3.

Figure 5.3: GSV images were obtained from 4 different viewpoints.

The "street-view" python package [186] was used for acquiring accurate latitude and longitude values for each GSV viewpoint to convert the coordinates requested to the nearest available Panorama IDs (i.e., unique panorama ID with purchased date [year, month], latitude and longitude). The latest Panorama ID was then used as the input location parameter as shown in the Figure 5.4.

Figure 5.4: Different location images of the study area with latitude, longitude values and panorama IDs.

We built a Python script to create the URLs and download 1000 GSV images to

cover the study field automatically. In order to remove the Google logos, we cropped the downloaded images.

5.3.3 Annotation Data

For deep supervised learning algorithms to be practical, large image data is essential. From GSV images acquired with screen captures on Google Maps, we created 1000 images data points by manually tagging eucalyptus trees, as can be seen in Figure 5.5. To increase the methodology's transferability, random eucalyptus trees' around 3500 images at the Wyndham city council, Victoria, Australia, were also taken for training, validation and testing of the model. We used "labelling" [187] for ground truth and panorama images. It is a tool written in Python for graphical image annotation and uses Qt for its graphical interface. Annotations are stored in PASCAL VOC, the format used by ImageNet, as XML files. We used the PASCAL VOC format because the Siamese network supports it. In DL, training an algorithm requires an ample training and validation dataset to minimise and prevent overfitting the model. At the same time, a test dataset is required to assess the trained model's performance. In total, 4500 images from GSV and self-gathered images were annotated and used as a dataset for training, 500 for validation and the other 500 for testing (accuracy) evaluation.

5.3.4 Training Siamese CNN

We trained Siamese CNN based on its central idea that if we use two input images from the same class, then their feature vectors must also be identical, and if two input images are not of the same class, then their feature vectors must also be different. Depending on the input image types, the vector features must be very different in these situations and the similarity score will also be different.

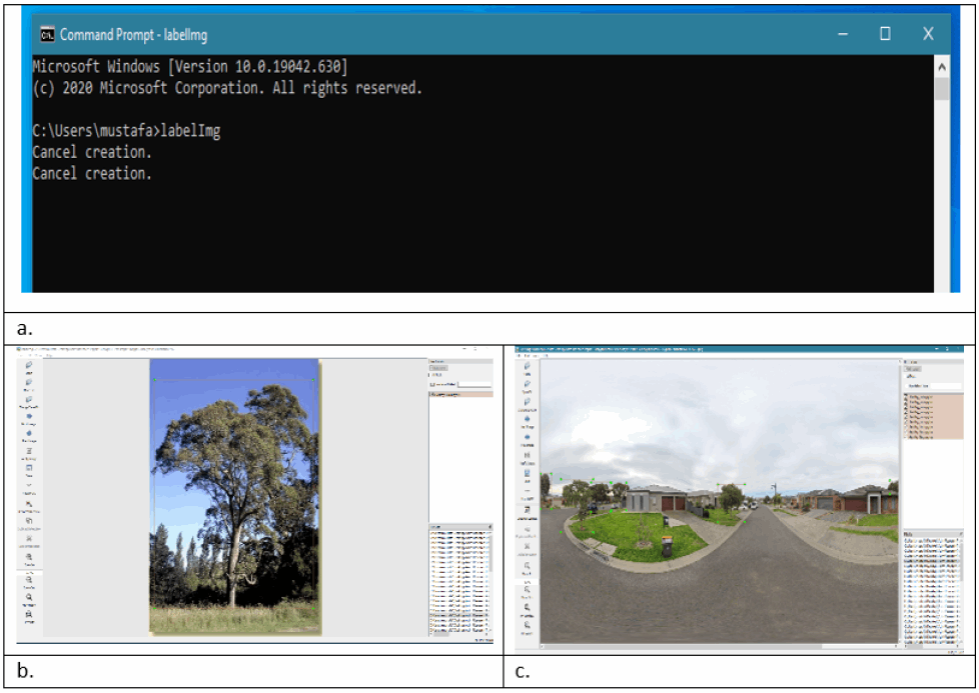


Figure 5.5: a.) Command prompt-LabelmeImg screenshot, b.) Annotating single tree image, c.) Annotating panorama image.

5.3.5 Siamese CNN Architecture

The word Siamese refers to twins [138]. Siamese Neural Network is a sub-class of neural network architecture that comprises of two or more networks [188]. These networks must be two copies of the same network, i.e., having the same configuration with the same parameters and weights.

We used the Siamese network, consisting of two identical convolutional neural networks (CNN) [189]. The network architecture is the same as in our previous work [146], where an individual CNN is comprising of 6 convolutional layers and three fully connected or Dense layers. Each convolution layer contains two feature types, input and numeral filters. We used a 3x3 filter size for all convolution layers. The number of the filter is transformed into the next linked layer for each layer, which extracts the valuable features. One of the key benefits of the convolutional network is that the

input image to the network can be much bigger than the size of the candidate image. Furthermore, in one evaluation, it will measure the similarity in all translated sub-windows on a dense grid. We search multiple scales in one forward-pass by assembling a mini-batch of scaled images. The output of this network performance is a score chart. For enhancing convergence speed, batch normalization [190] is applied to all convolutional layers except the last layer. We used five max-pooling layers that follow each convolutional layer to minimize the computational cost. The max-pooling has an active filter of 2x2 that slides on the input image and, based on the filter size; then the maximum value is selected as an output. The first two layers of the fully connected layers have ReLU activation [191] while the last layer (also known as the output layer) has a SoftMax activation [192]. The SoftMax activation finds the maximum probability value node and forwarded it as an output. A dropout of 0.5 is added to the fully connected layers to prevent over-fitting issues in the model. The total model parameters of our model are 51,161,305. Figure 5.6 is the visual representation of our Siamese network.

5.3.5.1 Contrastive Loss Function

Features extracted by the subnetworks are fed into the decision-making network component, which determines the similarity. This decision-making network can be a loss function [193], i.e., contrastive loss function [194].

We trained Siamese CNN with contrastive loss function [194]. Contrastive loss is a distance-based loss function used to find embeddings where the Euclidean distance is small in two related points and high in two separate points, [194]. Therefore, if input images are of same class, then loss function allows the network to output features close to feature space and if the input images are not similar then the output features are away. The similarity feature function is:

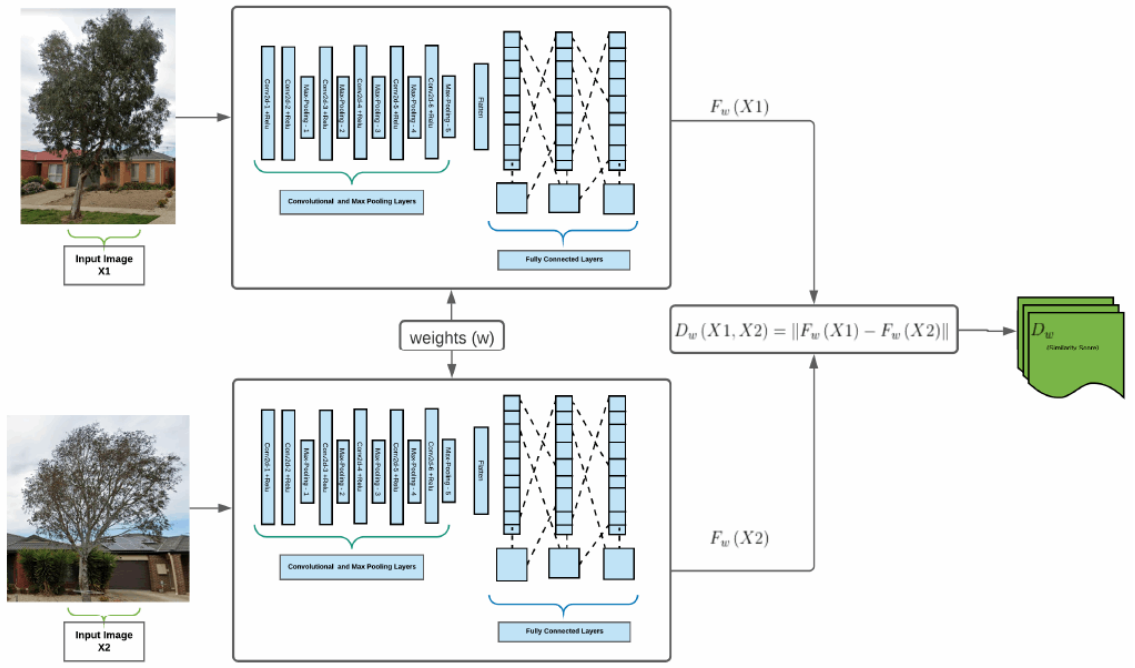


Figure 5.6: Visual representation of Siamese network architecture that takes two different inputs and provides the inference.

$$D_w(x1, x2) = \|F_w(x1) - F_w(x2)\| \tag{5.1}$$

Where x1 and x2 are the input images that shares the parameter vector w and Fw(x1), Fw(x2) represents the input mapping in the feature space and Dw is the Euclidean distance. By calculating the Euclidean distance, Dw, between the feature vectors, the co-evolutionary Siamese network can be seen as a measuring function that measures the similarity between x1 and x2.

We use contrastive loss function defined by Chopra et. al., [195], [196], in Siamese network training, defined as follow:

$$L(w, y, x1, x2) = \frac{y}{2} D_w(x1, x2)^2 + \frac{1-y}{2} (max\{0, m - D_w(x1, x2)\})^2 \tag{5.2}$$

where y is a binary label assigned to input images x1 and x2, y=1 if both the inputs are of the same class and y=0 if both inputs are of different class, while m > 0 is a margin value, must be chosen experimentally depending on the application domain.

Minimizing L(w, y, x1, x2) with respect to w will then result in a small value of Dw(x1, x2) for images of the same species and a high value of Dw(x1,x2) for images of different species. This is visually represented in Figure 5.7.

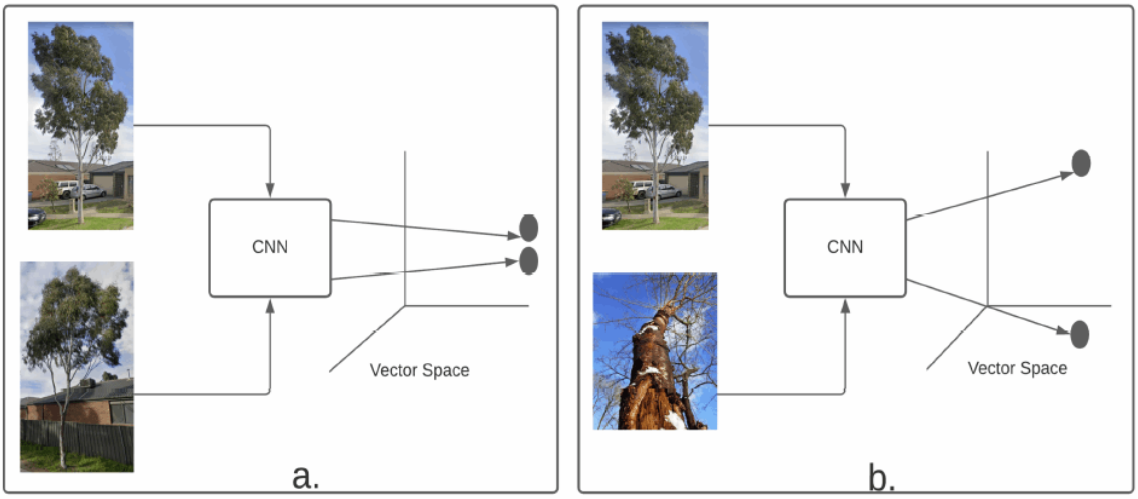


Figure 5.7: Contrastive loss function examples of a.) Positive (similar) and b.) Negative (different), images embedded into a vector space.

5.3.5.2 Mapping to Binary Function

Siamese network takes an input of a pair of images, and the output is a similarity score. The similarity score will be 1 if both images belong to the same class, and it will be 0 if both input images are from different classes.

5.3.6 Geo-location Identification

Our proposed DL-based automatic mapping method for eucalyptus tree from GSV includes three main steps as shown in Figure 5.8. They are the following.

1. Detect eucalyptus tree in the GSV images using a trained DL network.

2. Calculate the azimuth, (Azimuth is an angular measurement in a spherical coordinate system. The vector from an observer (origin) to a point of interest is projected perpendicularly onto a reference plane; the angle between the projected vector and a reference vector on the reference plane is called the azimuth [197]) from each viewpoint to the detected eucalyptus tree based on the known azimuth angles of the GSV images, relative to their view point locations, and the horizontal positions of the target in the images as shown in Figure 5.8 (2) using the mean value of two X values of the bounding box. For Example; suppose a detected eucalyptus tree has a bounding box that is centered on column 228 in a GSV image that is centered at 0° azimuth relative to the image viewpoint. Each GSV image contains 640 columns and spans a 90° horizontal field-of-view; thus, each pixel spans 0.14. The center of the eucalyptus tree is 130 pixels to the right of the image center (at column 320) and so has an azimuth of 18.2° relative to the image viewpoint.

3. The final step is to estimate the target locations based on the azimuths calculated from the second step as presented in Figure 5.8 (3).

5.3.6.1 LOB Measurement Method

The bounding boxes of detected eucalyptus trees, which result from the implementation of odometry from monocular vision of GSV images, are the outputs of eucalyptus

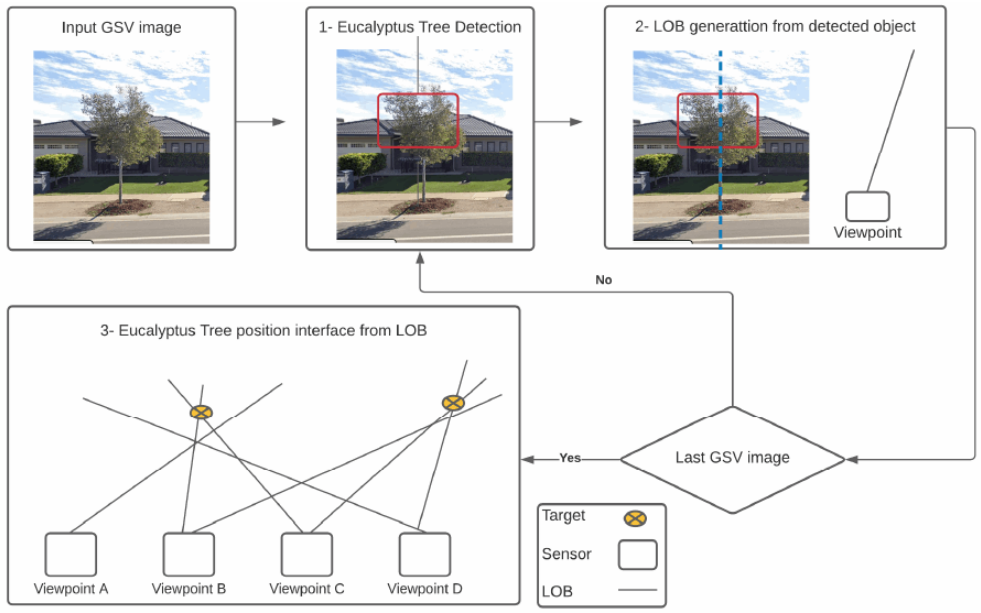


Figure 5.8: The process of using deep learning to map eucalyptus trees from GSV.

tree detection in GSV images using Siamese CNN, as shown in Figure 5.9. As a result, estimating eucalyptus tree positions in pure GSV images is a multiple-source localization issue based on passive angle measurements that has been extensively studied [198], [199]. One of three major multiple-source localization methods is the LOB-based method [200]. Since detected eucalyptus tree are not signal sources like propagating signal sources whose signal intensity can be calculated, a LOB calculation was used to estimate the position of a target eucalyptus tree shown in Figure 5.9. Other methods (such as synchronization and power transit) necessitate more stringent requirements for a LOB calculation. Azimuths from multiple image viewpoints to a given eucalyptus tree enable the eucalyptus tree position to be triangulated in LOB localization presented in Figure 5.9. Since the LOB move through the target, the intersection of several LOB is ideally the exact location of the target as can be seen in Figure 5.9.

When the LOB calculation is used in a dense emitter setting, however, many ghost nodes (i.e., false targets) appear, as shown in our study for estimating eucalyptus

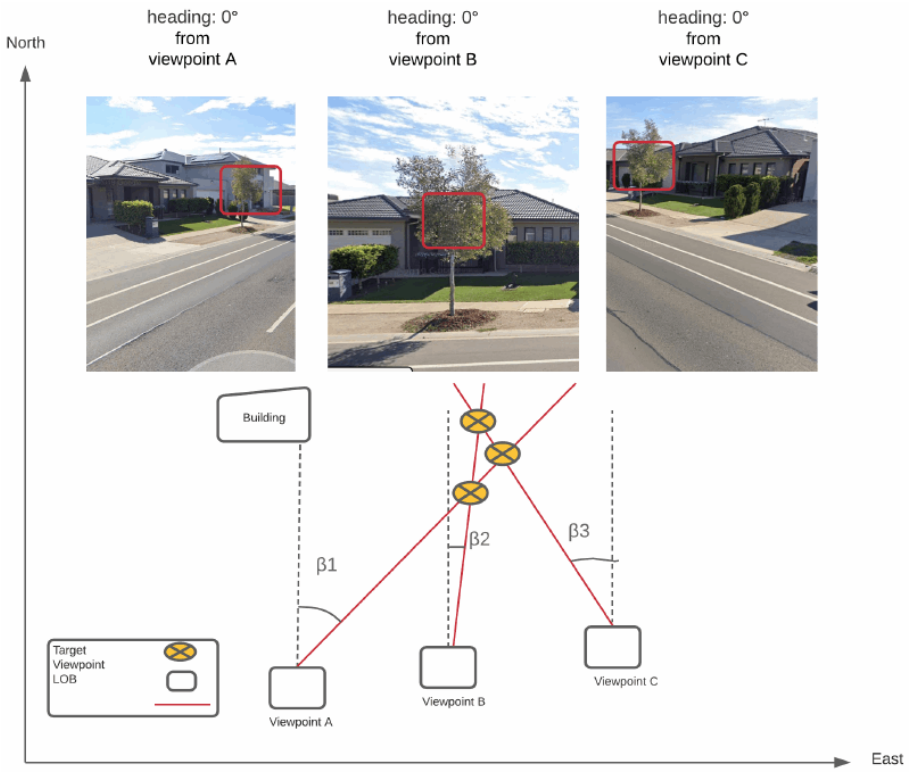


Figure 5.9: An example of using bearing measurements to determine a target position from three different locations using a sensor

tree locations in GSV images [201] as shown in Figure 5.11.

As a result, a modified brute-force-based three-station cross position algorithm was used to reduce the ghost node problem of multiple-source localization using LOB measurement as shown in Figure 5.10; source localization from viewpoints A, B, and C, based on two assumptions:

1. Targets and sensors are in the xy plane, and

2. All LOB measurements are of equal precision [202].

The LOB measurement method shown in Figure 5.11 consists of the following steps:

1. Find the closest neighboring viewpoints for a given viewpoint; we tested the

Figure 5.10: Bounding boxes of labelled eucalyptus tree in 4 GSV images (a-d)

algorithm's performance using 2 to 8 of the closest neighboring viewpoints (i.e., the corresponding number of views is 3 to 9);

2. Measure the angles between each pair of LOBs from all viewpoints [203];

3. Check whether there are positive associations among LOBs (set at 50 m length) from current viewpoint and its neighboring viewpoints.

4. Repeat the process from step 1 to step 3 for every intersection point.

To be more precise, a positive association among LOB is produced by three positive detections from any three views within an angle threshold (β) [202]. As a result, assuming constant detection rates, the number of predicted eucalyptus trees increases as the number of views increases, based on the likelihood of combination. For example, suppose the total number of eucalyptus trees estimation possibilities is t(t ε N); if the detection rate remains constant, the likelihood of a positive

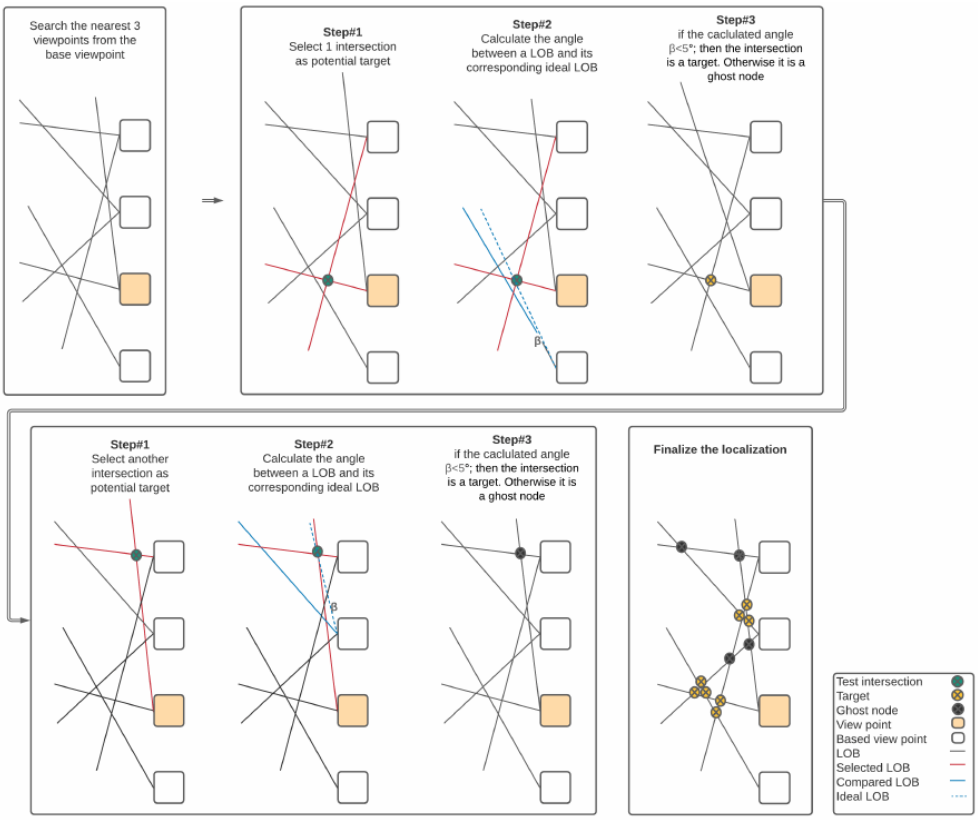


Figure 5.11: An example of how to use the brute-force-based three-station cross position algorithm to remove ghost nodes from four views with a 5° angle threshold.

association with seven views (i.e., C(7, 3)/t) is greater than the probability of positive association with four views. (i.e., C(4, 3)/t). In order to perform cross-validation in this analysis, the closest perspectives were chosen. A list of the nearest neighbouring perspectives (2, 3, 4, 5, 6, 7 and 8 viewpoints; that is 3, 4, 5, 6, 7, 8 and 9 views) and angle thresholds (1°, 2° and 3°) is used for testing to determine whether there is a positive correlation and which threshold functions better. Because of the span of the LOB and the interval between GSV acquisitions, only nine views were chosen for research (10 m). Eight perspectives are on a line on one side of the present perspective in the extreme case of 9 views. For the intersection of two 50-meter LOB, 80 meters is almost the maximum distance needed.

5.3.6.2 Multiple LOB Intersection Points Aggregation

If we use a modified brute-force-based three-station cross-location algorithm, the result will be more than one LOB intersection point, and all these are possible targets for each eucalyptus tree. In order to overcome this situation, we can further apply a geospatial algorithm, i.e., spatial aggregation (Spatial Aggregation calculates statistics in areas where an input layer overlaps a boundary layer [204]) to determine where a eucalyptus tree can be found. The primary purpose of this geospatial aggregation algorithm is to provide a central location (expected correct target) within a range of 10m (this 10m distance is given to the geospatial algorithm to apply aggregation on) of LOB intersection points. There are three main steps of this geospatial aggregation algorithm, as shown in Figure 5.12.

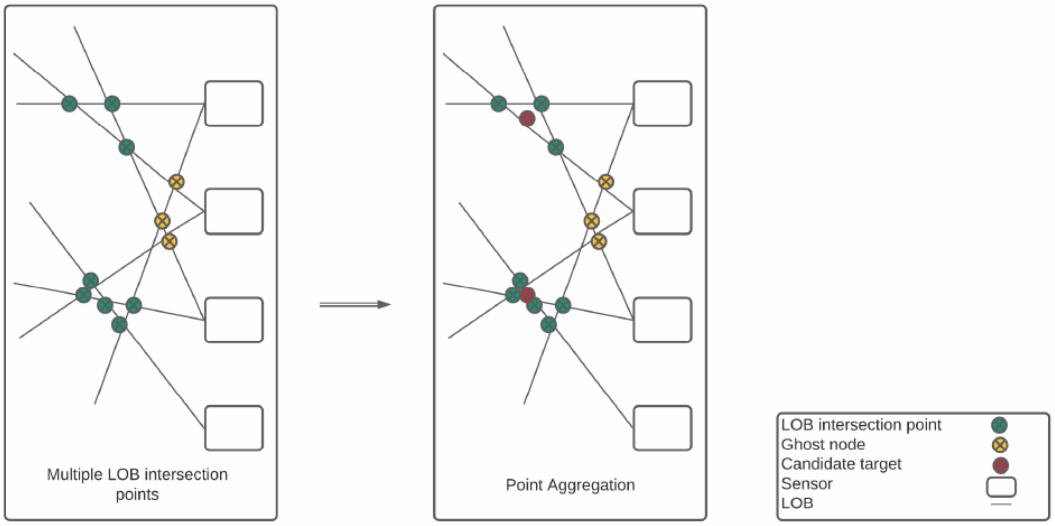


Figure 5.12: An example of aggregating multiple LOB intersection points.

1. Compute the Euclidean distance matrix between all LOB intersection points.

2. The Euclidean distances between LOB intersection points are used to cluster LOB intersection points.

3. Determine the centroid of each intersection point cluster.

5.3.6.3 Spatial Aggregation and Calculation of Points

Aggregation is the process of combining several objects with similar characteristics into a single entity, resulting in a less detailed layer than the original data [205]. Aggregation, like any other type of generalization, removes some information (both spatial and attribute) but simplifies things for the consumer who is more interested in the unit as a whole rather than each individual component within it [205]. Spatial aggregation can be applied on Line, Points or Area; however, the calculation method is slightly different when calculating points. For line and area features, average statistics are determined using a weighted mean. Only the point features inside the input boundary are used to summarise point layers. As a result, no equations are weighted. The following equation is used to calculate weighted mean [204].

$$\bar{x}_w = \frac{\sum_{i=1}^{N} w_i.x_i}{\sum_{i=1}^{N} w_i} \tag{5.3}$$

Where N = number of observations, xi = observations and Wi = weights.

It must be ensured that all data from the same database link is stored in the same spatial reference system while performing spatial aggregation or spatial filtering [204].

5.4 Experiments and Results

5.4.1 Experiments

We implemented our experiments in Keras [206] backend TensorFlow [207]. Typically, any state-of-the-art architecture may be used as a backbone to extract the features. We performed our experiments with VGG-16 [208], AlexNet [209] and

ResNet-34 [210] to explore how effective the backbone network is in extracting features. Siamese network consists of two sister/twin CNNs as both are two copies of the same network. They share the same parameters and network weights were initialized. The initial learning rate was set at 0.001 with an optimizer Stochastic Gradient Descent (SGD) [211], dropout was set to 0.5 and momentum 0.9. We used L2 Regularization to avoid over-fitting in the network [146]. All input images were resized into 100×100 before feeding into two identical networks in the Siamese network. The two input images of eucalyptus trees (X1 and X2) are passed through the networks and then through a fully connected layer to generate a feature vector for each (X1) and (X2)). We added a dense layer with ReLU activation and then finally an output layer with SoftMax activation.

5.4.2 System Configuration

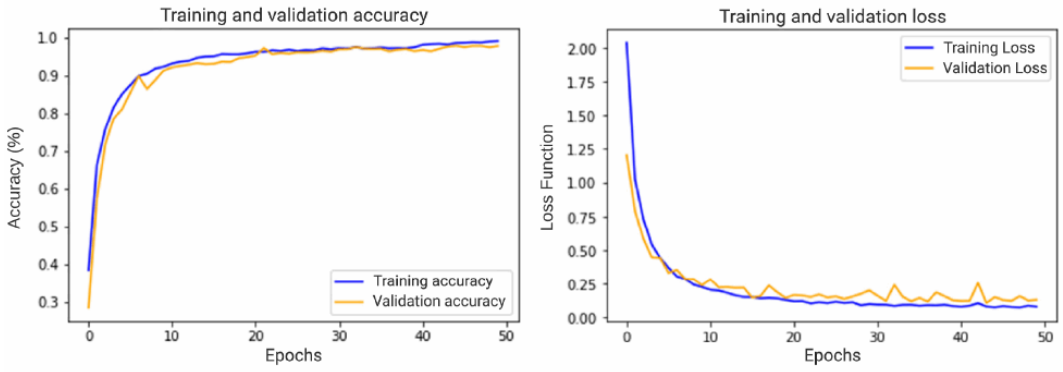
All our experiments were performed on Intel Core i7-9700K CPU @ 3.60GHz (8 cores and 8 threads), 32 GB RAM, NVidia Titan RTX 24GB VRAM GPU. For development and implementation of methodology, we used Python 3.8 and Keras-2.2 with Tensorflow-2.2.0 backend as the deep learning framework.

5.4.3 Approach

The entire dataset was split into 70% training, 10% validation and 20% test set. We applied various data augmentation techniques on the images and resized all images into 100×100 before feeding it into the Siamese network. The weights were initialized to avoid the layer activation from disappearing during the forward passage through a deep neural network [212]. We also used early stopping with a patience of 50 epochs.

5.4.4 Results

We used various networks such as VGG-16 [208], ResNet-34 [210], and AlexNet [209] in our experiments. While performing experiments, first, we froze a few layers in the backbone network and trained the network on the remaining layers that we added. The obtained results from the experiments with various networks were not satisfactory, i.e., 85.33%, 82.67% and 79.89%, respectively. The achieved results from the frozen layers were not satisfactory, so we unfroze all the layers and again performed the experiments to extract features for eucalyptus trees input images. This time the results were 93.2%, 90.43% and 86.26%, respectively. In each experiment, a total of 50 epochs were conducted, where each epoch is the number of iterations. Finally, the Siamese network was trained at a batch size of 32 and stopped training on epoch-50 as shown in Figure 5.13.



(a) Training and Validation Accuracy **(b)** Training and Validation Loss

Figure 5.13: Validating the object detection model learning.

The initial experiments with VGG-16, ResNet-34, and AlexNet demonstrated that VGG-16 consistently produced the best results in our scenario, so we used it as the backbone for all of our experiments.The resulting features of VGG-16 experiments are transferred to the decision network to identify whether or not two input images are similar. A sample output is shown in Figure 5.14.

Figure 5.14: Examples of identifying and classification of healthy and unhealthy eucalyptus trees from GSV and ground truth images.

There are many ways of performance measurement that are used to evaluate the performance of neural networks. They include precision, recall, accuracy, and f1-score. The precision tells us about the correct predictions made out of false-positive while recall tells us about the correct predictions made out of false negatives. The accuracy is the number of correct predictions out of both false positives and false negatives. We calculated all the performance measures for our trained model using formulas listed in Eq (4), (5), (6), and (7) from the confusion matrix.

$$Precision = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5.4}$$

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5.5}$$

$$Accuracy = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}} \tag{5.6}$$

$$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{5.7}$$

Where TP is true positives, TN is true negatives, FP is false positives and FN is

false negatives. Here the TP and TN are the correct predictions while the FP and FN are the wrong predictions made by our model. After computing values from the confusion matrix, the results are shown in Table 5.1.

Table 5.1: Classification / Model Performance Report

Evaluation Metrics	**Value in %**
Precision	93.38%
Recall	92.98%
Accuracy	93.2%
F1-Score	92.17%

5.4.4.1 Location Estimation Accuracy Evaluation

The location estimation accuracy of the eucalyptus tree is shown in Table 5.2 as a percentage of the number of predicted eucalyptus tree positions within the buffer zones of a reference eucalyptus tree. To assess the effects of the number of views, the angle threshold, and the distance to the middle of a chosen road, we considered seven views (i.e., 3, 4, 5, 6, 7, 8, 9), three angle thresholds (i.e., 1°, 2°, and 3°), and three distance thresholds to the centre of a selected road (i.e., 3m, 4m, and 5m) to determine the impacts of the number of views, the angle threshold, and the distance threshold to the centre of a selected road. Around half of the estimated eucalyptus tree locations were within the 6m buffer zone of their reference locations using the method we tested, and up to 79% of the estimated locations were within the 10m buffer zone of their reference locations using the method we tested. However, about 12% of the approximate eucalyptus tree positions were inside the 2m reference position buffer zone.

Table 5.2 reveals that using more views and higher angle thresholds resulted in a more approximate eucalyptus tree in the modified brute-force-based three-station

cross-location algorithm, which is due to the increased relaxation of the modified brute-force-based three-station cross-location algorithm. Meanwhile, because relaxation allows more ghost nodes to be estimated eucalyptus trees, more estimated eucalyptus trees may result in lower accuracy (see Table ref mytable).

Table 5.2 shows that when comparing the results of other numbers of views, the average percentage of predicted eucalyptus tree positions being inside all buffer zones of reference eucalyptus trees for the results of 8 views is the highest (47.80%). Using greater distance to the centre of selected road thresholds, on the other hand, resulted in less approximate eucalyptus trees. Since the optical GSV imagery was the only data source used to perform the localization, the precision of the position estimation for the eucalyptus tree is fair, and the estimated data is helpful.

It is worth noting that GSV image distortion, terrain relief, GSV position accuracy, or limitations in the process we used may have caused location mismatches in some cases due to the ground positions of eucalyptus trees varying from the orthographic predicted locations estimated from GSV images. For areas where GSV imagery is available and a eucalyptus tree distribution map with a 10 m accuracy is appropriate, our proposed approach has a lot of promise. When a given eucalyptus tree was not identified in at least three GSV images out of a certain number of views, our method failed to estimate the eucalyptus tree's location. Three is the minimum number of images needed to triangulate a position and remove ghost nodes (as can be seen in Figure 5.11). This explains why the number of projected eucalyptus trees rises in tandem with the number of views (see Table 5.2).

5.5 Discussion

Eucalyptus trees are evergreen, however, an early sign that shows it is unhealthy if it turns brown, either partially or completely. Various signs / aspects can be spotted in unhealthy eucalyptus trees such as one of the most apparent is the loss or decrease of leaf growth in all or parts of the tree. Other symptoms include the bark of the tree becoming brittle and peeling off, or the trunk of the tree becoming sponge-like or brittle. A tree may have bare branches, i.e., without leaves, in any season can be a sign of dead tree or branches that are loose and weak could indicate a dead or dying tree. Weak joints of eucalyptus tree can be dangerous, as it means branches can come loose during bad weather [213]. If the whole eucalyptus is dead, it can be left untouched for a period of maximum two years; however, after this, it becomes unsafe and needs to be removed.

Some of the common diseases in eucalyptus [214] trees are shown in Figure 5.15 a.), b.) and c.). It is critical to identify such unhealthy trees in order to improve the urban eucalyptus tree's health and environment.

a. Canker disease that infects the bark and then goes inside of the tree,

b. Phytophthora disease goes directly under the bark by discolored leaves and dark brown wood, and

c. The heart disease damages the tree from inside and outside.

Numerous approaches are studied in the current literature with regards to trees and their health in urban areas. Shendryk et al., [155] worked on the trunks of eucalyptus trees, as well as their complex shapes. They used Euclidean distance clustering for individual tree trunk detection. Up to 67 per cent of trees with diameters greater than or equal to 13 cm were successfully identified using their technique.

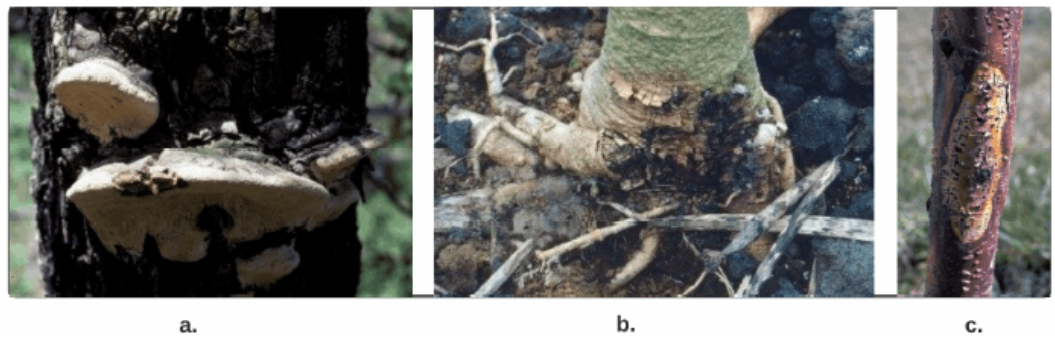


Figure 5.15: Some common diseases a.) Heart rot and b.) Phytophthora, and c.) Canker.

Milto Miltiadou et al., [154] presented a new way to detect dead eucalyptus camaldulensis with the introduction of DASOS (feature vector extraction). To do so, they attempted to research the odds of dead trees being detected using Voxel-based full-waveform (FW) LiDAR without tree delineation. It has been discovered that it is possible to determine tree health without outlining the trees, but since this is a new area of research, there are still many improvements to be made. Xiao et al., [172] presented that the trees were examined using remote sensing data and GIS techniques to examine their health. Trees had their conditions analysed in relation to physiognomy on two scales: the tree itself and in terms of pixels. A pixel-by-pixel analysis was performed in which each tree pixel within the tree crown was classified as either healthy or unhealthy based on values of vegetation index. A quantitative raster-based analysis was conducted on all of the trees, where they used the tree health index, which is a quantitative value that describes the number of healthy pixels compared to the total tree pixels on the crown. Classifying the tree as healthy if the index was greater than 70% of the overall index indicated that a random sample of 1,186 trees was used to verify the accuracy of the tree data. When viewed at the whole tree level, approximately 86% of campus trees were found to be healthy and approximately 88% of mapping accuracy.

In contrast to the above-discussed literature, we propose a deep learning-based network, Siamese convolutional neural network (SCNN), combining a modified brute-force-base line-of-bearing (LOB) algorithm to classify Eucalyptus trees as healthy or unhealthy and to find their geo-location from the GSV and ground truth imagery. Our proposed method successfully achieved an average accuracy of 93.2% in identifying healthy and unhealthy trees and their geo-location. For training and validation of SCNN, a dataset of approximately 4,500 images was used.

The main purpose of using Google imagery is that Google imagery is available publicly online and no privately man laboured efforts are required in order to capture the images. Secondly, using of sentinel imagery would be an expensive option and time consuming solution, as the sentinel's imagery requires longer time period to get the images of specific location and needs to subscribe to pay for getting service; i.e., not publicly available. The sentinel imagery is also protected by copyrights. Therefore, in this work, we used GSV and ground truth image for getting better result and overcome the some of the challenges as discribed in the introduction section. It is worth mentioning that "the satellite data on Google Maps is typically between 1 to 3 years old". According to the Google Earth and other sources, data updates usually about once a month, but they may not show real-time images. Google Earth gathers data from various satellite and aerial photography sources, and it can take months to process, compare and set up the data before it appears on a map. However, in some circumstances, Google Maps are updated in real-time to mark major events and to provide assistance in emergency situations. For example, it updated imagery for the 2012 London Olympic Games just before the Opening Ceremony, and it provided updated satellite crisis maps to help aid teams assess damage and target locations in need of help shortly after the Nepal earthquake in April 2015 [215], [216].

The primary purpose of using Google imagery is that Google imagery is available

publicly online, and no privately manual laboured efforts are required to capture the images. Secondly, using sentinel imagery would be an expensive and time-consuming solution. The sentinel's imagery requires a longer period of time to obtain images of a specific location and a subscription to pay for service, i.e., it is not publicly available for free. Copyrights also protect sentinel imagery. Therefore, in this work, we used GSV and ground truth images to get better results and overcome some of the challenges described in the introduction section. It is worth mentioning that "the satellite data on Google Maps is typically between 1 to 3 years old". According to Google Earth and other sources, data updates are usually about once a month, but they may not show real-time images. Google Earth gathers data from various satellite and aerial photography sources, and it can take months to process, compare and set up the data before it appears on a map. However, in some circumstances, Google Maps are updated in real-time to mark significant events and provide assistance in emergencies. For example, it updated imagery for the 2012 London Olympic Games just before the Opening Ceremony, and it provided updated satellite crisis maps to help aid teams assess the damage and target locations in need of help shortly after the Nepal earthquake in April 2015 [215], [216].

5.6 Conclusion, Limitations and Future Directions

Identifying various healthy and unhealthy eucalyptus trees using traditional and manual methods is time-consuming and labor-intensive. This study is primarily an exploratory one that employs a DL-based method for identification, classification, and geolocation estimation. In this study, we present a Siamese CNN (SCNN) architecture trained to identify and classify healthy and unhealthy eucalyptus trees and their geographical location. The SCNN uses the contrastive loss function to calculate its similarity score from two input images (one for each CNN). With the large number of GSV images available online, the method could be a useful tool for

automatically mapping healthy and unhealthy eucalyptus trees, as well as mapping their geo-location on metropolitan streets and roads. Although the model correctly identifies the eucalyptus tree's health status and position, there are certainly worth mentioning limitations to consider. Firstly, it is still challenging to map up-to-date GSV images with geographical location information because the changing nature of imagery is rapid. Secondly, to achieve reasonable accuracy for geo-location with the DL, a large amount of training data is needed. Thirdly, when eucalyptus trees have a big lean, the LOB method requires more attention; this is due to terrain and GSV's visual distortion without compensation. Finally, the method suggested for automatic tree geo-location recognition can be useful, and in future study, it might be used to detect and classify other objects along the roadside.

Table 5.2: Based on 1039 reference trees, the accuracy assessment of estimating position of eucalyptus trees.

Number of views	Threshold of angle (°)	Threshold of distance to center of selected road (m)	Percentage of the number of estimated locations of eucalyptus tree being within a certain buffer zone of reference eucalyptus tree (%)										Number of estimated eucalyptus tree
			<1m	<2m	<3m	<4m	<5m	<6m	<7m	<8m	<9m	<10m	
3	1	3	1.75	8.04	22.38	35.66	46.85	54.9	62.59	68.18	74.83	80.07	286
	1	4	1.83	8.42	23.44	36.63	47.62	55.68	64.47	70.7	76.92	82.42	273
	1	5	1.92	7.69	23.08	37.31	49.23	57.69	67.31	72.69	79.23	85	260
	2	3	1.71	8.05	19.51	30.98	44.88	53.41	58.78	64.63	72.93	77.8	410
	2	4	1.75	8.27	19.8	31.08	45.36	53.88	60.15	66.92	74.94	79.95	399
	2	5	1.85	8.71	20.32	32.72	47.76	56.73	63.59	69.66	77.31	82.32	379
	3	3	2.37	8.84	20.47	31.9	43.32	50.86	57.54	64.01	71.12	75.86	464
	3	4	2.68	8.95	20.36	31.77	43.18	51.23	58.17	65.55	72.93	77.18	447
	3	5	2.56	9.3	20	32.79	44.88	53.02	60.23	66.98	74.65	79.53	430
	1	3	2	7.56	19.56	30.89	40.67	51.33	58	63.11	71.11	76	450
	1	4	1.87	7.26	20.37	30.91	42.62	53.86	60.89	67.45	74.71	79.16	427
	1	5	2.02	7.83	20.96	32.32	44.44	55.81	63.38	70.45	77.78	83.33	396

Table 5.2 continued from previous page

Number of views	Threshold of angle (°)	Threshold of distance to center of selected road (m)	Percentage of the number of estimated locations of eucalyptus tree being within a certain buffer zone of reference eucalyptus tree (%)										Number of estimated eucalyptus tree
			<1m	<2m	<3m	<4m	<5m	<6m	<7m	<8m	<9m	<10m	
5	2	3	1.31	7.86	17.84	29.13	39.44	48.12	54.99	62.52	67.76	72.83	611
	2	4	1.2	7.72	18.01	29.5	41.68	51.29	58.32	65.87	71.7	76.33	583
	2	5	1.09	8.56	18.58	30.78	43.53	53.37	60.47	68.12	74.32	79.05	549
	3	3	1.35	7.77	16.89	29	38.42	46.79	55.31	62.78	68.76	74.14	669
	3	4	1.39	8.19	18.08	30.45	40.96	49.15	57.5	65.84	71.56	76.82	647
	3	5	1.47	9.61	19.06	30.94	42.51	50.98	59.45	68.4	74.43	78.99	614
	1	3	2.2	10.8	22.34	35.35	45.6	51.83	57.51	64.29	71.98	77.47	546
	1	4	2.46	11	22.35	36.36	48.48	55.3	61.93	68.94	75.19	79.17	528
	1	5	2.63	11.3	22.63	37.37	49.7	56.97	63.84	71.92	77.78	83.23	495
	2	3	2.31	11	19.62	30.3	42.14	48.77	57.58	64.36	71.28	75.47	693
	2	4	2.53	10.3	19.2	32.44	45.83	53.27	61.61	68.3	73.66	77.98	672
	2	5	2.52	10.2	19.69	32.6	47.72	55.28	63.15	70.08	77.17	81.57	635

Table 5.2 continued from previous page

Number of views	Threshold of angle (°)	Threshold of distance to center of selected road (m)	Percentage of the number of estimated locations of eucalyptus tree being within a certain buffer zone of reference eucalyptus tree (%)										Number of estimated eucalyptus tree
			<1m	<2m	<3m	<4m	<5m	<6m	<7m	<8m	<9m	<10m	
	3	3	2.37	10.4	19.05	29.17	39.55	47.83	56.37	63.34	69.51	74.24	761
	3	4	2.84	10.4	19.08	30.72	42.63	51.42	60.35	66.98	72.26	76.73	739
	3	5	3.01	11.1	20.23	32.28	43.9	53.52	62.41	70.01	76.33	80.63	697
6	1	3	2.5	12.2	23.21	36.06	46.41	52.92	60.27	67.78	73.12	78.46	599
	1	4	2.87	12.2	23.99	37.67	48.14	54.56	63.18	70.78	74.32	78.55	592
	1	5	2.5	12.9	25.04	38.64	49.91	57.07	65.47	73.7	78	82.47	559
	2	3	2.43	10.8	21.62	33.92	44.73	52.3	59.86	65.14	71.76	78.24	740
	2	4	2.46	10.3	21.61	34.75	47.74	56.5	64.71	70.86	74.69	79.07	731
	2	5	2.47	11.2	23.11	36.63	50.44	59.88	67.3	73.4	78.05	82.27	688
	3	3	2.22	9.75	20.49	32.47	41.23	50.49	57.9	63.21	70.62	75.43	810
	3	4	2.63	10.1	21.13	33.88	44.5	55.38	62.88	68.63	73.25	76.63	800
	3	5	2.76	10.8	22.97	34.78	46.19	57.09	64.44	70.47	76.12	79.4	762

Table 5.2 continued from previous page

Number of views	Threshold of angle (°)	Threshold of distance to center of selected road (m)	Percentage of the number of estimated locations of eucalyptus tree being within a certain buffer zone of reference eucalyptus tree (%)									Number of estimated eucalyptus tree
			<1m	<2m	<3m	<4m	<5m	<6m	<7m	<8m	<9m	<10m	
7	1	3	2.7	12.4	24.01	38.31	49.92	56.6	63.28	68.68	74.72	79.81	629
	1	4	2.91	12.4	25.36	41.03	52.67	58.16	65.59	72.54	77.71	82.23	619
	1	5	2.74	13.2	27.05	43.15	55.65	60.96	69.01	75.34	80.65	84.93	584
	2	3	2.2	9.95	21.71	34.24	44.44	52.07	59.56	65.37	70.8	76.36	774
	2	4	2.61	9.52	23.21	36.64	47.72	56.19	64.15	70.01	73.14	77.84	767
	2	5	2.37	10.6	23.29	38.35	51.05	59.14	67.78	73.08	77.82	82.01	717
	3	3	2.75	10.2	22.04	33.53	43.95	52.22	59.28	66.35	72.1	75.57	835
	3	4	3.15	10.2	23.12	35.23	46.25	55.33	62.47	68.77	72.88	76.15	826
	3	5	3.31	11.5	23.92	36.01	48.6	56.87	65.14	70.99	75.95	79.64	786
	1	3	3.08	12.2	25.08	40.92	52.92	58.77	64.77	70.46	70.46	76.77	650
	1	4	3.87	12.9	26.63	43.03	55.42	60.84	66.41	73.68	78.17	82.51	646
	1	5	4.08	13.7	28.22	45.02	58.4	63.62	70.47	77.16	82.54	86.79	613

Table 5.2 continued from previous page

Number of views	Threshold of angle (°)	Threshold of distance to center of selected road (m)	Percentage of the number of estimated locations of eucalyptus tree being within a certain buffer zone of reference eucalyptus tree (%)									Number of estimated eucalyptus tree
			<1m	<2m	<3m	<4m	<5m	<6m	<7m	<8m	<9m	<10m	
9	2	3	3.44	11.3	23.66	36.01	47.2	54.33	61.58	65.52	70.74	75.7	786
	2	4	3.68	11.3	25.51	37.44	50	57.61	64.47	69.16	72.21	76.78	788
	2	5	3.49	12.1	26.71	39.19	52.62	59.46	67.65	72.89	77.18	80.54	745
	3	3	3.05	11.2	23.12	34.62	45.42	52.93	60.09	66.2	71.24	75.7	852
	3	4	3.51	11.2	24.09	35.79	47.6	56.02	61.99	69.36	72.4	76.02	855
	3	5	3.78	13.3	25.61	38.66	49.88	57.44	65.73	71.71	76.1	79.63	820
	1	3	2.67	11.7	24.67	41.46	52.15	58.99	65.53	71.92	76.37	82.91	673
	1	4	2.85	12	26.39	42.73	54.72	61.62	66.87	73.01	77.21	82.01	667
	1	5	3.14	13.8	28.57	45.84	58.87	64.52	70.8	76.77	81.16	85.71	637
	2	3	3.18	11.3	22.03	36.72	47.37	54.59	61.57	65.97	70.26	75.64	817
	2	4	3.04	12.2	23.45	37.3	48.97	57.23	62.33	67.8	71.08	75.7	823
	2	5	3.47	12.5	25.06	40.36	51.8	59.13	66.07	71.47	75.45	78.92	778

Table 5.2 continued from previous page

Number of views	Threshold of angle (°)	Threshold of distance to center of selected road (m)	Percentage of the number of estimated locations of eucalyptus tree being within a certain buffer zone of reference eucalyptus tree (%)										Number of estimated eucalyptus tree
			<1m	<2m	<3m	<4m	<5m	<6m	<7m	<8m	<9m	<10m	
	3	3	2.95	11.4	22.05	36.36	47.16	54.77	60.91	65.8	70.11	75.34	880
	3	4	2.83	12.2	23.42	36.88	49.21	56.56	62.44	67.99	71.15	75.23	884
	3	5	3.4	12.9	24.74	39.98	51.11	58.15	64.95	71.04	75.38	78.55	853

Chapter 6

A Multiview Semantic Vegetation Index for Robust Estimation of Urban Vegetation Cover

As mentioned in list of publications, this chapter with same title was published as an original research paper in the Remote Sensing. (2022), 14(1), 22814(1), 228, an official journal of MDPI, doi: https://doi.org/10.3390/rs14010228, in a special issue: Advances in Object and Activity Detection in Remote Sensing Imagery. The contents are the same, with the exception of certain layout adjustments to ensure consistency in the presentation across the thesis.

Abstract

In the contemporary era, urban vegetation growth is vital for creating sustainable and livable cities since it directly helps people's health and well-being. Estimating vegetation cover and biomass is commonly done by calculating various vegetation indexes for automated urban vegetation management and monitoring. However, most of these indexes fail to capture robust estimation of vegetation cover due to inherent focus on colour attributes with limited viewpoint and ignoring seasoning variations.

This article proposed a novel vegetation index to address this weakness robust to the colour, viewpoint, and seasonal variations. Moreover, it can be applied directly to RGB images. This Multiview Semantic Vegetation Index (MSVI) is based on deep semantic segmentation and Multiview field coverage and can be integrated into any vegetation management platform. This index has been tested on Google Street View (GSV) imagery of Wyndham Council, Melbourne, Australia. The experiments and training achieved 89.4% and 92.4% overall pixel accuracy from FCN and U-Net, respectively. Thus, the MSVI can be a helpful instrument for analysing urban forestry and vegetation biomass since it provides an accurate and reliable objective method for assessing the plant cover at the street level.

6.1 Introduction

The changing land use patterns and population growth have had a significant impact on the vegetation composition in the world [217, 218, 219] which is essential for better living conditions of city dwellers. As indicated by Wolf, K.L. [220], a city's vegetation cover (i.e. street woods, lawns, etc.) has long been acknowledged as a key component of urban landscape planning. According to Appleyard [221], the instrumental role of street vegetation is to absorb airborne pollutants through carbon sequestration and oxygen production, to mitigate noise pollution in urban heat islands [222], and reduce storm waters [223, 224]. In addition, the life of vegetation generally raises the aesthetic evaluation of people in urban settings [225, 226]. For this purpose, it is critical to document changes in vegetation so that land management professionals may work to improve the urban environment. Furthermore, changes in the type of land cover (such as building developments) have been found to have a strong correlation with the changes of vegetation in the urban environment.

Moreover, changes in an urban environment are generally very important such as food consumption, energy, water, and land used by urban residents has a significant impact on the environment. Therefore, automated detection of vegetation cover is often done through calculations of various vegetation indexes [227] that hold important information regarding vegetation cover of a particular location. In past, various algorithms were employed for the calculation of vegetation index using various image modalities. However, existing approaches have highly focused on spectral analysis and color variations. For instance, Normalized Difference Vegetation Index (NDVI) tends to amplify atmospheric noise in the Near Infrared Reflectance (NIR) and Red bands and becomes very sensitive to background variation. Therefore, it does not work well for RGB images for street-level vegetation analysis. Remote sensing data collected from above by sensors (aircraft, space) misses the glimpse of urban flora at street level. Thus, profile views of urban greenery from the road level are insufficiently assessed, even though green indices derived from remotely sensed image data might help quantify urban greenery. There is a distinction between vegetation view through ground experience and the view captured by remote sensing systems [228]. Li et al. [2, 229], discovered that people had unequal access to distinct types of urban greenery (street vegetation, private yard total vegetation, private yard trees and shrubs, and urban parks), providing the groundwork for subsequent research into urban greenery inequity.

On the other hand, RGB based vegetation indexes are prone to wrong estimations due to reliance on green colour and ineffectiveness to capture seasonal variations. Rencai et al. [3] utilise the green view index (GVI) as a quantitative indicator to determine how much greenery can be seen by pedestrians and then apply an image segmentation algorithm to figure out how much greenery can be seen by pedestrians in street view images. Zhang et al. [230] used an extensive street view image data set,

as well as a horizontal green view index (HGVI) to calculate the quantity of greenery visible from the street in their research. Long et al. [231] analysed 245 Chinese cities, calculating the GVI values of their central regions and comparing them to the overall GVI conditions of the respective cities. As a result, they discovered that more affluent and well-run cities have longer and greener streets. Several visual qualities of streets such as salient region saturation, visual entropy, a green view index, and a sky-openness index were measured by Cheng et al. [232].

Kendal et al. [233] used colour threshold for extraction of vegetation index. The technique proved to be promising, but only using colour features for segmentation is not an efficient model as any clutter information in the image can match the vegetation colour. Further, in recent years Bawden et al. and Kattenborn et al. [234, 235] used convolutional neural network (CNN) for two studies: In the first approach, they used a CNN-based approach to train data acquired from unmanned aerial vehicle (UAV)-based high-resolution RGB imagery visual interpretation, a fine-grained map for two species of vegetation. While in the second approach, they mapped species of trees or plants cover in different vegetation UAV RGB imagery. However, these approaches suffer due to reliance on colour and specific image features and are unable to handle large variations in vegetation characteristics.

Recent advancements in deep learning have introduced a new level of accuracy in identifying objects of interest through semantic segmentation. Jonathan et al. [236] introduced a fully convolutional neural network (FCN), and Dvornik et al. [237] proposed BlitzNet for object segmentation. Yi et al. [238] constructed an instance aware based semantic segmentation model, which utilized the advantages of FCN for segmentation and classification. As a result of the development, the model was capable of simultaneously recognizing and segmenting the object instances. Liang-Chieh et al. [239] applied fully convolutional neural networks (FCN) to a multi-scale

input image in order to achieve the required results.

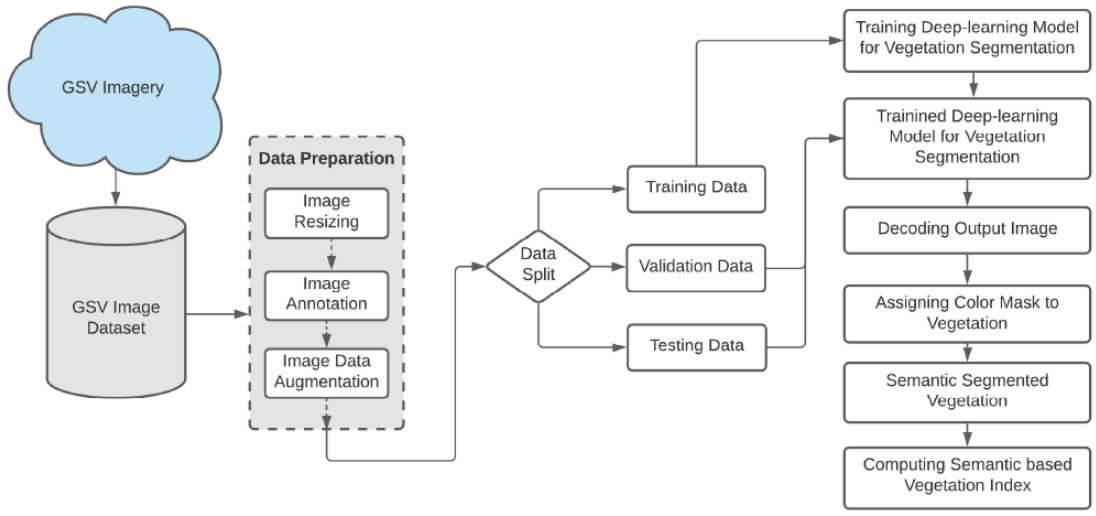


Figure 6.1: A data flow diagram for the MSVI, which highlights the process of calculating the proposed vegetation index.

Motivated with the success of deep semantic segmentation, the conducted research proposes a semantic vegetation index (SVI) for RGB images with robustness against colour changes and seasonal variations. To deal with the limitation of single image coverage, its extension called multi-view semantic vegetation index (MSVI) is also introduced that can estimate vegetation cover from multiple views. The overall framework of this study is presented graphically in Figure 6.1.

The Contribution: According to the literature, semantic vegetation index (SVI) is one of the first approaches to integrate deep semantic segmentation into the process of vegetation index estimation. Although there are a variety of vegetation indexes in the literature, they are limited to a specific image modality and colour feature, or they overlook essential flora semantic information. It makes them more susceptible to noise, resulting in erroneous estimation. The proposed index is robust to colour and seasonal variations and works for any imaging modality. Furthermore, it can be extended to multiple views to expand exposure and reliable calculation. The

segmentation approach is not claimed to have made a contribution in this study. Nonetheless, it compares many ways to determine which are the most appropriate for this aim.

The rest of the paper is organized as follows: Section 2 explains the material and methods taken into account, Section 3 presents detailed information regarding the experiments and results achieved by the proposed methodology, Section 4 presents the comparative analysis with the previous work , section 5 presents a detailed discussion of the proposed work, while Section 6 is the conclusion section of this paper.

6.2 Materials and Methods

6.2.1 Study area

Figure 6.2: The research area in Victoria, Australia, which was chosen for this study. a.) Victoria (Australia), b.) Wyndham City Council, Victoria, Australia, c.) One sample site and d.) a sample street view from a sample site.

Figure 6.2 shows the municipal council of Wyndham (VIC, Australia),as the

selected area for this study. It lies on the western outskirts of Melbourne (VIC, Australia) and covers an area of 542 km^2. According to the 2019 census, its estimated population is 270,478. Wyndham is the third fastest-growing council in the state of Victoria. The population of Wyndham is diverse, and the community development projects suggest that by 2031 more than 330,000 people are expected to come and live. Wyndham is home to 16 suburbs (Cocoroc, Eynesbury, Hoppers Crossing, Laverton North, Laverton RAAF, Little River, Mambourin, Mount Cottrell, Point Cook, Quandong, Tarneit, Truganina, Williams Landing, Manor Lakes, Quandong and Werribee South). The City Council of Wyndham is committed to improving residents' environment and livelihoods. Every year, thousands of new trees and vegetation are planted in this commitment to increase Wyndham's tree canopy cover through the street tree planting program [240].

Figure 6.3: A sample panorama image of a selected study site from Google street view imagery.

6.2.2 Input Dataset / Google Street View Image Collection

In this research work, Google street view images (GSV) [241] is used for the multiview semantic vegetation index (MSVI) estimation. A sample GSV image of a Wyndham Council in Melbourne, Victoria, is shown in Figure 6.3. The GSV panorama view is identical to the real-world view. The process of producing a 360° GSV panorama is

to sequentially capture horizontal X-number (X=6) images and vertical Y-numbers (Y=3) images of the camera [242]. The GSV Image API (Google) [241], together with the position and travelling direction of the GSV car, can be used to obtain every accessible GSV image in an HTTP URL form, for example (https://maps.googleapis.com/maps/api/streetview?parameters). The static GSV image as shown in Figure 6.4, can be retrieved for every point where the GSV is available by establishing URL parameters supplied via a specific HTTP request utilising the GSV Image API **(Google)** [241].

Figure 6.4: A static image of a research site taken from Google Street View imagery.

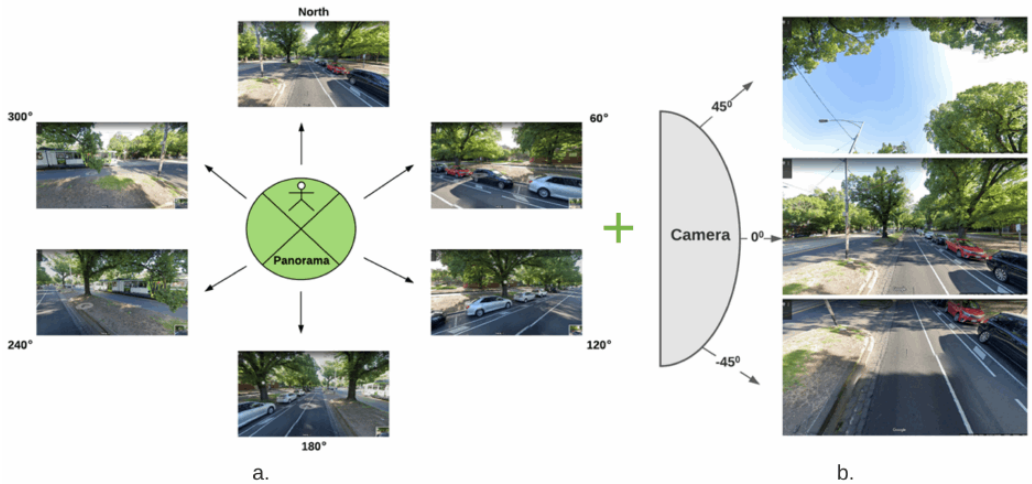


Figure 6.5: a.) Sample of images taken from pedestrian view in six different angles and b.) From pedestrian view, three images taken from three vertical angles (45°, 0°, -45°).

The GSV images for each sample site in six directions were collected as illustrated in Figure 6.5 (a), and in three vertical angles to determine the green areas visible to pedestrians as presented in Figure 6.5 (b). 0°, 60°, 120°, 180°, 240°, and 300° were set as the heading parameters whereas 45°, 0°, and -45° as pitch parameters. As a result, a total of eighteen images are captured for a specific location, ensuring that no vegetation area is left out of the index calculation. A Python programming language script is executed on all the GSV images to read and downloaded from each example site by automatically parsing the GSV URL.

6.2.3 Deep Semantic Segmentation

The act of grouping sections of an image in such a way that each pixel in a group correlates to the object class represented by the group as a whole can be defined as semantic segmentation for images in this manner [243, 244]. The object classes in the current work correspond to trees and green vegetation terrain. Images can be segmented by allocating each pixel of an input image to a label class object, which is referred to as semantic image labelling [245]. Image segmentation is also known as semantic image labelling. This method often combines image segmentation with object identification techniques to produce a final result. Various deep learning-based segmentation models, such as FCN [246], DeepLabv3+ [247], Mask R-CNN [248], are being developed for use in a variety of applications and environments. For the purpose of semantic vegetation segmentation and to calculate the vegetation index from GSV imagery in this research work, FCN [236], and U-Net [249] semantic segmentation models are used. Their selection was based on their high precision and excellent performance in medical imaging area. The results of the experiments demonstrate that deep learning-based segmentation models are effective at segmenting vegetation images using semantic attributes.

6.2.3.1 Fully Convolutional Network (FCN)

Fully Convolutional Network (FCN) [236] uses locally connected layers, such as up-sampling, pooling, and convolution, to achieve segmentation. The architecture does not include any dense layers in order to reduce the amount of time it takes to compute and the number of parameters it requires. A segmentation map uses two paths to obtain output: the first is a down-sampling road, which is used to collect semantic/contextual information, and the second is an up-sampling path, which is used to recover spatial features. The architecture of FCN is depicted in Figure 6.6. Fully convolutional network architecture (FCN) was presented by Long et al. [236] for robust segmentation by adopting fully convolutional layers in place of the last fully linked layers. This approach allows the network to generate a dense pixel-wise prediction as a result of the advancement. The combination of up-sampled outputs with high-resolution activation maps results in improved localisation performance, which is then passed to the convolution layers to produce the correct output. The performance of FCN motivated to employ it as an important component of the proposed approach.

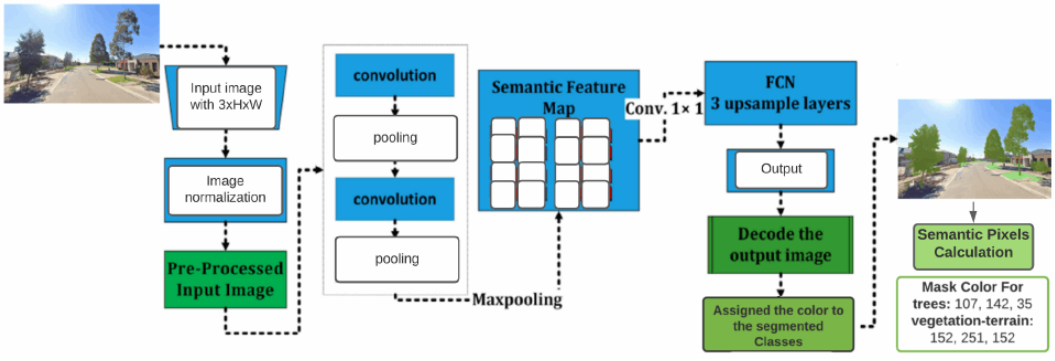


Figure 6.6: The architecture of fully convolution network (FCN) showing network processes. The masks for trees and vegetation are shown as RGB color codes.

6.2.3.2 U-Net

The second model employed in this work is U-Net [249], which has a similar encoder-decoder architecture to that of FCN but has two significant traits that distinguish it from the former. Since U-Net is symmetric, it bypasses the connections between the up-and down-sampling paths, which is useful when employed as a concatenation operator. Using the colour variable, models assign a colour to an item after they have been trained. The U-Net network (Figure 7.5) is built on an encoder-decoder architecture [249]. The encoder consists of a stack of convolutions and max-pooling layers that work together. The decoder is a symmetric expanding path that up-samples the feature maps with the use of learnable deconvolution filters, which can be learned. The major innovation brought about by this network is the way in which the so-called skip connections are utilised. To be more specific, they enable the concatenation of the output of the transposed convolution layers with the corresponding feature maps from the encoder stage during the convolution stage [250]. The main objective of this step is to get all the fine characteristics that were learned throughout the contracting stages in order to restore the spatial resolution of the original input image [249].

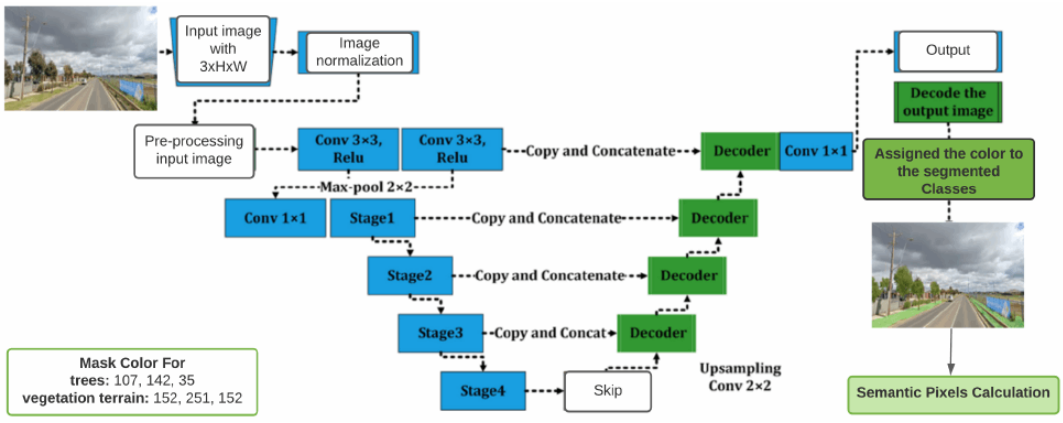


Figure 6.7: The architecture of U-Net showing network processes. The masks for trees and vegetation terrain are shown as RGB color codes.

According to standard practice, in the U-Net approach, the input image is initially processed by an encoder path, which is comprised of convolutional and pooling layers that degrade the spatial resolution of the input image. It is then followed by a decoder path that restores the original spatial imagery resolution by adopting up-sampling layers followed by convolutional layers, which is a technique known as "up-convolution". Apart from that, the network makes use of so-called skip connections, which connect the output of the relevant layers in the encoder path to the inputs of the decoder path by adding them to the inputs of the decoder path. Whereas FCN allows pixel-wise classification performed for segmentation where features from initial convolutional layers are upsampled to develop deconvolution layers. These deconvolution layers develop the same size image, which is segmented on the basis of learnt features. Fine-tuning was performed to allow the network to learn efficient features of the vegetation region.

6.2.4 Vegetation Index Calculation from RGB Images

Various approaches are adopted in the literature for vegetation index calculation. Some of those are listed in section 2.4.1. However, most of them used either colour, threshold or green area segmentation that might lead to promising results. To achieve robustness in vegetation index calculation, a semantic approach based on the unique colour for each class of plants is proposed in this article. RGB colour codes (107, 142, 35 and 152, 251, 152) for trees and vegetation terrain were assigned, respectively. After segmentation of the vegetation (trees and vegetation terrain), respective masks are applied to calculate accurate vegetation index as discussed in section 2.4.2. For a better understanding of the RGB colour space, 3D data distribution in the RGB domain in Figure 6.8 is shown.

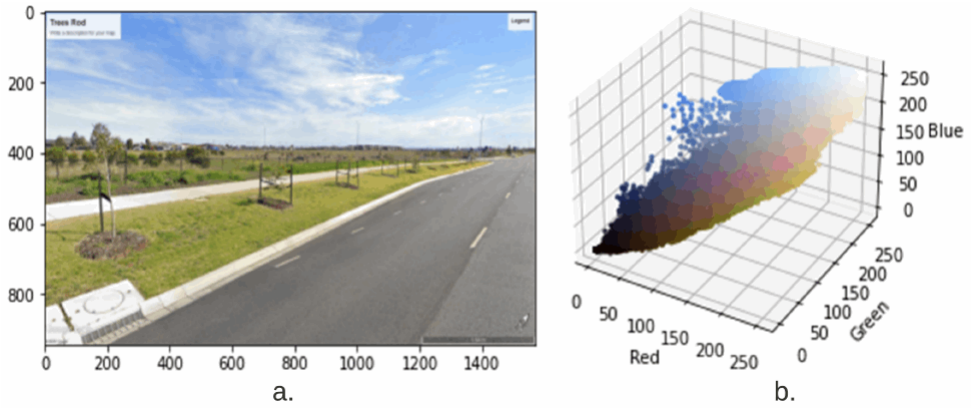


Figure 6.8: A sample image is presented in 3D color spaces for better understanding of data distribution. a.) sample image, b.) data distribution in RGB color space. As data in different color channels is tightly correlated, it provides inherent difficulties to differentiate colour and semantic information in RGB domain.

6.2.4.1 Green View Index (GVI)

Mohamed et al. [251], explored extracting green vegetation from remotely sensed multispectral images. It has been identified that both, i.e., near-infrared and red bands, are being utilised quite often for vegetation detection. One of the primary reasons is that on red bands, the vegetation shows less absorption, and on infrared, they show great reflection. However, GSV images cover only the blue, red, green, and near-infrared bands. It was established by Yang et al. [228] that the GVI value was affected by two factors: the size of a tree's crown and the distance between the camera and the subject. A non-supervised classification methodology was used by Li et al. [2] to extract green vegetation from GSV images, which was justified by the fact that a significant number of GSV images were not available in the near-infrared band. According to their findings, green vegetation is significantly less reflective in red and blue bands. The red bands, on the other hand, are extremely reflective. As a result of this phenomenon, they developed extracting green vegetation from GSV images based on the natural hues of the images. There are a number of steps

involved in the workflow.

Step - 1: First of all, the subtraction of red band from green band generates Diff 1, and subtraction of blue band from green band gives Diff 2.

Step - 2: Then the two images Diff 1 and Diff 2 were multiplied to create one Diff image. Normally, the green vegetation has greater reflectance values in the green band than the other two red and blue bands, hence the Diff image has positive green vegetation pixels.

Step - 3: The pixels that has lower values in green band as compared to the red and blue bands, exhibit negative values in the Diff image

Step - 4: As a result, an additional criterion was added stating that pixel values in the green band must be greater than those in the red band.

Usually, multiple spark points existed in the resulting images after the initial classification images utilising the pixel-based classification approach as described in the steps above (steps 1 - 4) were obtained [252]. The spectral vegetation variation has led to classifying individual pixels differently from their surrounding areas, leading to sparks in the classed image.

$$Green\ \ View\ \ Index = \frac{\text{Number of green pixels segmented}}{\text{Total number of pixels in an image}} \tag{6.1}$$

The above equation gives information regarding the available greenery in the image. Yang et al. [228] proposed the *Green View Index* (GVI), which measures the visibility of urban woods in terms of greenery. Its GVI was defined as the relationship between the total green space and four image(s) taken at the intersection of the road and the sum of the four images taken at the intersection as shown in the following equation:

$$Green\ View\ Index = \frac{\sum_{i=1}^{4} Area_{g\_i}}{\sum_{i=1}^{4} Area_{t\_i}} * 100\% \tag{6.2}$$

Here the $Area_{g\_i}$ present the green pixels of the images taken in the direction of ith out of the four images taken in the (north, east, south, and west) directions. $Area_{t\_i}$ represents the total number of pixels in the image in the direction of ith. According to Li et al. [2], in this scenario, some surrounding vegetation may be missed from the calculation of the GVI since only four images cannot be seen in the fields of vision from pedestrian view. Therefore they modified the equation (2) as below:

$$Green\ View\ Index = \frac{\sum_{i=1}^{6}\sum_{j=1}^{3} Area_{g\_ij}}{\sum_{i=1}^{6}\sum_{j=1}^{3} Area_{t\_ij}} * 100\% \tag{6.3}$$

Where $Area_{g\_ij}$ denotes the number of green pixels in one of these images, which were taken in six directions with three vertical view angles for each sample site and then averaged over all six directions. As a result, $Area_{t\_ij}$ represents the total amount of pixels included within each one of the eighteen GSV images.

6.2.4.2 The Proposed Semantic Vegetation Index (SVI)

For robust calculation of the vegetation index of each sample location on the road or street, the approach of semantic pixels (SP) is used, which is based on the unique colour pixels assigned to vegetation's specific class (Vegetation terrain and trees), and are extracted based on the deep features through the use of a deep neural network. For index calculations, Google street view (GSV) images was used as such dataset is readily available. Therefore, for calculating the vegetation index in this investigation, used a single image to calculate the vegetation index accurately based on the semantic pixels so that to cover all the vegetation area in the image. Hence,

in each sample image, the number of semantic pixels will be determined as SP_a, with the Area being the total semantic pixel numbers in one GSV image. The original equation 1 has been updated and is now referred to as the semantic vegetation index (SVI).

$$SVI = \frac{\sum_{i=1}^{n} SP_{a\_i}}{\sum_{i=1}^{n} Area_{t\_i}} * 100\% \tag{6.4}$$

Where SVI stands for semantic vegetation index, n is the total number of images, SP_{a_i} denotes the amount of semantic pixel area representing greenery in an image and $Area_{t\_i}$ denotes the total amount of pixels in an image.

Similarly, for calculating the multiview semantic vegetation index, a total of six images covering the 360° horizontal environment with three vertical angles of, i.e., 45°, 0°and -45° are used. The process is shown in Figure 6.5 (b) to calculate the vegetation index accurately based on the semantic pixels so that to cover all vegetation area. Hence, in each sample site, the number of semantic pixels will be determined as SP_{a-ij}, with the Area being the total semantic pixel numbers in one of the 18 GSV images. The equation 3 has been modified for utilising semantic pixels for calculating the multiview semantic vegetation index ($MSVI$).

$$MSVI = \frac{\sum_{i=1}^{6}\sum_{j=1}^{3} SP_{a\_ij}}{\sum_{i=1}^{6}\sum_{j=1}^{3} Area_{t\_ij}} * 100\% \tag{6.5}$$

Where $MSVI$ stands for multi-view semantic vegetation index, SP_{a_ij} presents semantic pixels area of vegetation in input images which are taken from different pitch angles (45°, 0° and -45°) vertically as well as six horizontal direction covering 360° area and $Area_{t\_ij}$ represents the sum of pixels in an image from the eighteen images of GSV.

6.3 Results

6.3.1 Preparation and Annotation of Dataset

For the experiments and implementation of the proposed model, first downloaded a total of 3000 Google street view (GSV) images using a python script. The next step was the pre-processing of the dataset so that the images could be used for training and testing phases. For the annotation of the training data, a cloud-based tool known as "Apeer", a ZEISS initiative [253] has been used. Image annotation generates labels that serve as the basis for machine learning training. Machine learning accuracy is determined by the amount of training data as well as the accuracy of annotations. The process of Annotation is summarised in Figure 6.9.

6.3.2 Experimental Environment Configuration:

For the experiments and results, the hardware and software resources used are listed in the Table 7.1.

Table 6.1: Configuration of experimental environment

Item Name	Parameter
Central processing unit (CPU)	Intel i7 9700k
Operating system	MS Windows 10
Operating volatile memory	32GB RAM
Graphic processing unit (GPU)	Nvidia Titan RTX
Development environment configuration	Python 3.8 + TensorFlow 2.5 + CUDA 11.2 + cuDNN V8.1.0 + Visual Studio 2019

Figure 6.9: The process of data annotation shown in this figure. a.) A data annotation cloud based platform known as "Apeer", b.) sample image for annotation, c) After completion of annotation and d.) Area zoomed for annotation in c.) and pointed with arrow.

6.3.3 Training of Deep Semantic Segmentation Models

The complete data set was split up into three distinct sections: training, validation, and testing sets, each comprising 80%, 15%, and 5% of the total, respectively. Before starting the training, hyperparameters were set to avoid the overfitting and underfitting issues of the model. The hyper-parameters used for the training of semantic segmentation model are: batch size kept as 16, learning rate as 0.0001, loss function as categorical cross-entropy, number of iteration/epochs as 200, NMS threshold as 0.45 and an optimizer as stochastic gradient descent "SGD". The training loss, validation loss, training accuracy and validation accuracy curve graphs are presented

in Figures 6.10 a.) and b.) and 6.11 a.) and b.) for FCN Model and U-Net Model, respectively. The accuracy curve for the U-Net beats the accuracy curve for the FCN, as shown in the graph in Figure 6.11.

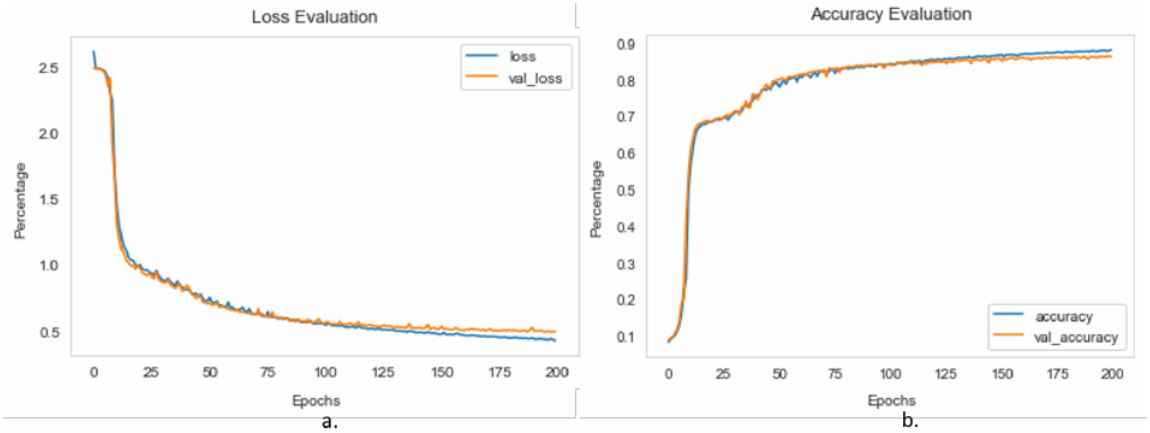


Figure 6.10: FCN segmentation model trend graphs for (a.) training and validation loss. & (b.) training and validation accuracy.

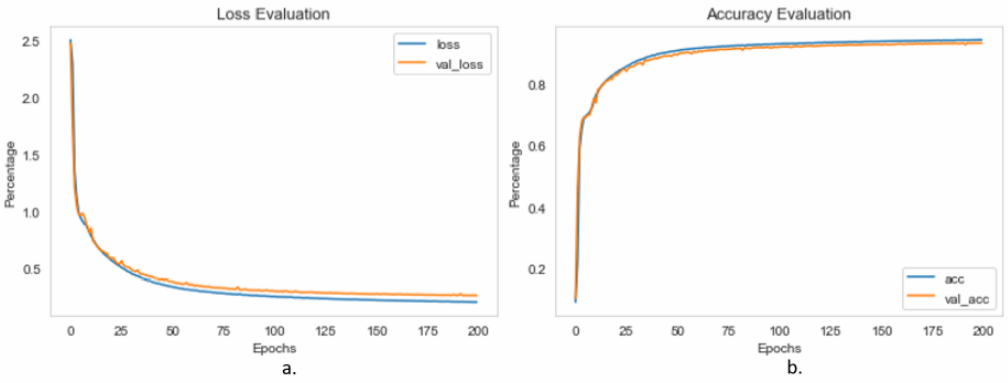


Figure 6.11: The U-Net segmentation model trend graphs for (a.) training, validation loss. & (b.) training and validation accuracy.

Some of the sample results using FCN and U-Net segmentation models are shown in Figure 6.12 and vegetation index values are computed using Equation (4). The vegetation index values calculated from FCN for the test input images are 43%, 30% and 32%. While vegetation index values calculated from U-Net for the test input images are 41.4%, 33% and 37%. The results show that the U-Net segmentation

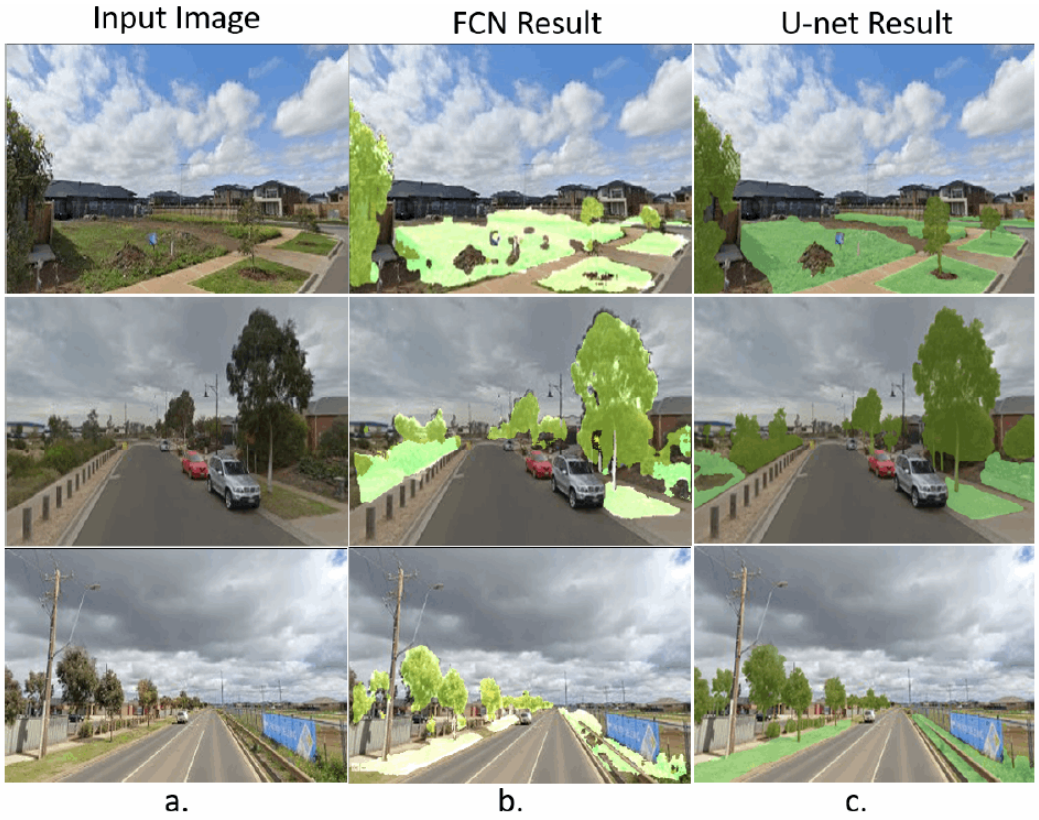


Figure 6.12: Segmentation and extraction of vegetation results from test input images. a.) presenting input images, b.) is the results generated using FCN and c.) presenting results generated using U-Net model.

model is giving comparatively more accurate and promising results than the FCN segmentation model. The ground truth results are computed manually for comparing the results after masking manually and then calculating the pixel values of the vegetation, using Adobe Photoshop application software. The computed results are in percentage, as evident from Equation (4). Thus, on the basis of the ground truth data, U-Net vegetation index results are quite promising and are more close to the ground truth results.

6.3.4 Performance Evaluation of Semantic Segmentation Networks

The performance of the semantic segmentation technique is evaluated using the metrics of precision, recall, F1-score, pixel accuracy (PA), intersection over union (IoU), and mean intersection over union (mIoU). Figure 6.13 shows the results of FCN and U-Net.

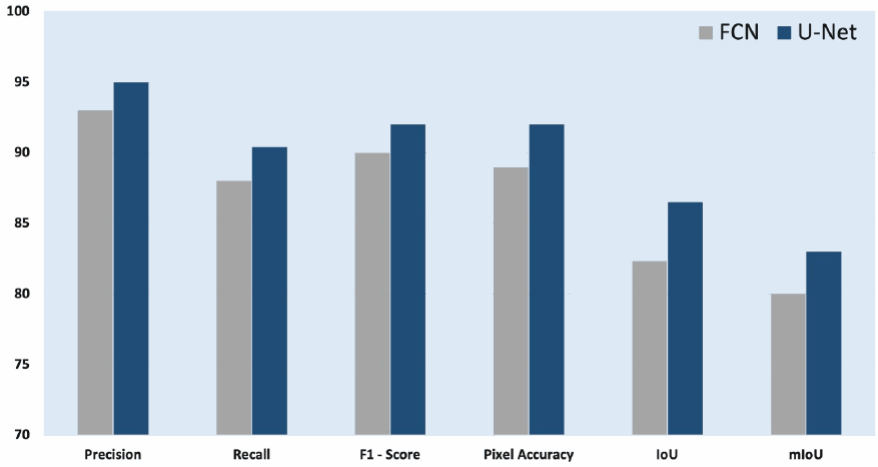


Figure 6.13: Performance Evaluation of FCN and U-Net segmentation models

The accuracy of object contour segmentation is measured using the PA method, while the accuracy of an object detector on a particular dataset is measured using the IoU metric. mIoU is the average of IoU and is defined to show the overall enhancement of semantic segmentation accuracy.

6.3.4.1 Precision, Recall and F1-Score

FCN and U-Net segmentation models were compared in terms of precision, recall, and F-measure. The results of the comparison are shown in the Table 7.2.

Precision is defined as the relationship between the number of accurately segmented vegetation pixels and the total number of pixels segregated as a vegetation

region by the technique. The recall is the ratio between the number of successfully segmented vegetation pixels and the total number of vegetation pixels in the labelled image.

$$Precision = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{6.6}$$

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6.7}$$

The equation of F1-score is shown below,

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{6.8}$$

6.3.4.2 Pixel accuracy (PA)

In the evaluation of segmentation models, the pixel accuracy metric is the most commonly employed. It is defined as the accuracy of the pixel-wise prediction, given as;

$$PA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k} \sum_{j=0}^{k} p_{ij}} \tag{6.9}$$

Where k represents the total number of pixels in a test image, and pii is used to present the true positive predicted pixels as of class i, while pij is presenting the ground class i pixels as the number of pixels of class j.

6.3.4.3 Intersection Over Union (IoU)

Intersection over Union (IoU) is also known as the Jaccard Index [254], and it is a typically used assessment statistic for segmentation models that is used to calculate their overall performance. As shown below, it is commonly defined as the ratio of

intersection and union areas between the projected segmentation map and ground truth.

$$IoU = \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}} \tag{6.10}$$

Where k indicates the total number of classes, p_{ii} represents the number of true positives, and p_{ij} and p_{ji} represent the number of false positives and false negatives, respectively.

6.3.4.4 Mean-IoU (mIoU)

mIoU is yet another matrix that is commonly used in segmentation models. It is calculated as the average value of all IoU label classes taken as a whole. This type of report is commonly used to summarise the performance of segmentation models, given as;

$$mIoU = \frac{1}{k+1} \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}} \tag{6.11}$$

Where k indicates the total number of classes, p_{ii} represents the number of true positives, and p_{ij} and p_{ji} represent the number of false positives and false negatives, respectively.

Table 6.2: Performance evaluation results

Segmentation Model	Precision	Recall	F1-Score	Pixel Accuracy	IoU	mIoU
FCN	93.2	87.3	90.1	89.4	82.3	80
U-Net	95	90.8	92.3	92.4	86.5	83

Table 7.2 shows the results achieved by different segmentation models used for vegetation index calculation on the basis of semantic pixels in an image. The U-Net

model showed really promising results.

6.4 Comparative Analysis

Extraction of green vegetation from street view images is a difficult task because of a variety of factors, including the presence of shadows and spectral confusion between vegetation and other artificial green features (green walls, windows, shadows, and signboards etc.). Two studies are most relevant to this research: Yang et al. [228] used four GSV images in their work. As a result, Li et al. [2] modified the Green View Index (GVI) calculation, and they subsequently conducted a case study assessment of street vegetation using GSV images in the East Village of Manhattan District, New York City. They assert that the modified GVI may be a relatively objective measure of street-level greenery and that the use of GSV in conjunction with the modified GVI may be particularly effective in directing urban landscape planning and management practices.

For the purpose of comparison with literature, sample images containing green vegetation, as well as green walls, signboards, and décor, were segmented and extracted for the vegetation index calculation. Sample images segmentation results based on Li et al. [2] and Rencai et al. [3] are presented in Figure 6.14 and Table 6.3.

Table 6.3: Comparison Table for vegetation segmentation and their vegetation index calculation using various vegetation extraction and index calculation approaches.

S.No.	GVI [2]	GVI [3]	SVI [ours]
1	57.50%	55.91%	47.55%
2	46.62%	43.12%	35.44%
3	52.68%	51.25%	40.33%
4	43.08%	40.55%	27.42%

From the results, it can be seen that the results of segmentation also included other green objects as vegetation because both the studies are principally based on

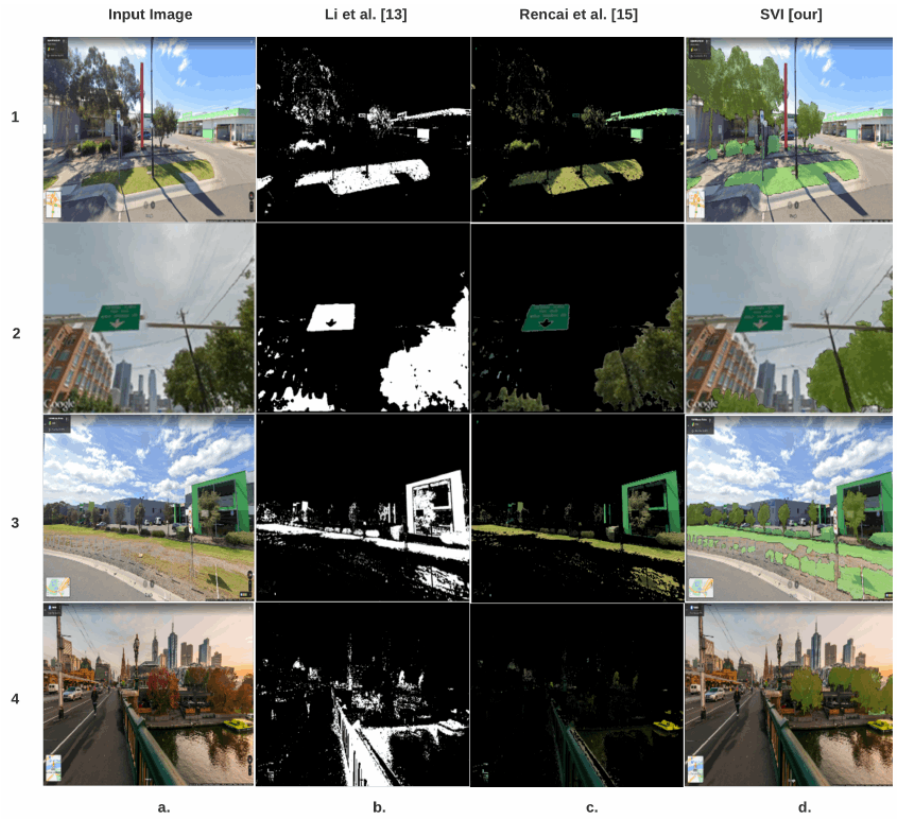


Figure 6.14: a.) Sample of images and their segmentation (vegetation extraction) using different approaches a.) input images, b.) Li et al. [2], c.) Rencai et al. [3] and SVI [proposed].

green colour. Both of the studies have mentioned this drawback in their studies and results. Thus yielding inaccurate vegetation index because of the other green colour objects inclusion. Hence the vegetation index calculated values are higher as compared to our results. However, this study results included vegetation only while ignoring other green colour objects for calculating index because it is based on semantic segmentation, thus giving an accurate vegetation index value.

Multi-view semantic vegetation index calculation for panoramic images taken at different angles horizontally (a.) and with varying angles of pitch vertically (b.) as shown in Figure 6.5 and the respective calculated vegetation index values are presented in Table 6.4.

Table 6.4: Comparative analysis of vegetation index calculation through various approaches

Li et al. [2]	**Rencai et al. [3]**	**MSVI [Proposed]**
63.40%	62.9%	56.19%

In the table, it is clear that results of Li et al. [2], and Rencai et al. [3] are quite similar as both studies rely on green colour; hence there are chances that during the calculation of vegetation index, most of the time include other objects of green colour as mentioned before in the sample segmentation result shown in the Table 6.3. Therefore, the results are inaccurate, and the vegetation index percentage indicated is larger than ours because both comparison studies employed the green area index, and the tram in the image was also used to compute green colour in those studies shown in Figure 6.5. On the other hand, the proposed model extracted only vegetation index. The input image on the second row in the Figure 6.14 is taken from Li et al. [2] paper just only for the comparison purpose. Where exactly, they mentioned that their algorithm is based on green colour, thus include another green object during the calculation of the green view index.

6.5 Discussion

Based on the research study, the semantic segmentation leads to accurate index calculation. The publicly available GSV imagery of the urban areas used to quantify street greenery, i.e. SVI of the urban streets. As GSV are publicly freely available and can be used in machine learning/computer vision in an efficient way to perform multiple activities automatically. The SVI can be utilized as useful information/data for a better assessment of urban greenery by considering people's envisioned vegetation on a street scale for urban planners and others. To assess the greenery of

street vegetation, GSV images captured from the ground should be similar to those of pedestrians.

A single vertical point of view is insufficient to express correctly the surrounding vegetation index that pedestrians may observe; two vertical points of view are required. Therefore, the multiview semantic vegetation index (MSVI) is employed for six GSV images in this experiment to calculate the vegetation index, each spanning a 360° horizontal and three vertical angles of 45°, 0°and -45°, to calculate the vegetation index appropriately on the basis of the semantic pixels.

According to the findings of this study, GSV images are qualified for assessing street greenery, and the modified GVI may be a more objective measurement of street-level greenery. The multiview semantic vegetation index (MSVI) took advantage of the characteristics of GSV images, used 18 GSV images taken from different viewing angles, making the index more efficient for evaluating street greenery in urban areas. Because it measures the amount of visible urban greenery on the ground, the SVI formula is simpler to understand for the general public. As a result, it can give a monitoring tool to analyze gains or losses in urban vegetation. It may serve to help urban planners select the sites, sizes and varieties of greenery for best effect in the planning stage of an urban greening program. It, therefore, seems to be a promising instrument, not a mere gadget for users, for future urban planning and urban environmental management.

The strength of SVI lies in its robustness to colour variations and viewpoint contraints. The limitation of the approach is its reliance on captured viewpoints and attributes of the captured image, like its zoom level and image quality. Therefore, if SVI is utilised for long-term vegetation monitoring, it is proposed that proper dataset normalisation and image registration scale or affine invariant [4] be used before SVI estimation.

6.6 Conclusions

This research paper, proposes a robust vegetation index based on semantic segmentation called a multiview semantic vegetation index (MSVI). The Google street view (GSV) imagery dataset is used for calculating and indexing the vegetation cover of an urban area of the Wyndham city council in Melbourne, Australia. The MSVI is based on the deep features learned from a deep neural network to calculate the vegetation index of each sample location in the urban area. For vegetation segmentation, different deep learning-based semantic segmentation models, such as FCN and U-Net, were tried. Using the GSV data set, both segmentation models were trained and tested to improve their overall performance.

The proposed method for segmenting urban vegetation areas has yielded promising results. Generally speaking, U-Net shows better results than FCN. FCN and U-Net models achieve Pixel Accuracy of 89.4% and 92.4%, Precision of 93.2% and 95%, IoU of 82.3% and 86.5%, and mIoU of 80% and 83% respectively. The proposed MSVI index measures the broad visible urban greenery on the ground, which can assist urban planners and strategists in better understanding urban green spaces.

It is intended to use this approach in the future for real-time vegetation index calculation using Google panoramic cameras such as Pilot Era 360°, INSTA360 PRO, and INSTA360 PRO2, which will be of great help in the quest for ecological improvement.

Chapter 7

Deep Semantic Vegetation Health Monitoring Platform for Citizen Science Imaging Data

As mentioned in list of publications, this chapter with same title is accepted as an original research paper in the **'PLOS ONE'**, journal. Now it is under production. The contents are the same, with the exception of certain layout adjustments to ensure consistency in the presentation across the thesis.

Abstract

Automated monitoring of vegetation health in a landscape is often attributed to calculating values of various vegetation indexes over a period of time. However, such approaches suffer from an inaccurate estimation of vegetational change due to the over-reliance of index values on vegetation's colour attributes and the availability of multi-spectral bands. One common observation is the sensitivity of colour attributes to seasonal variations and imaging devices, thus leading to false and inaccurate change detection and monitoring. In addition, these are very strong assumptions in a citizen science project. In this article, we build upon our previous work on de-

veloping a Semantic Vegetation Index (SVI) and expand it to introduce a semantic vegetation health monitoring platform to monitor vegetation health in a large landscape. However, unlike our previous work, we use RGB images of the Australian landscape for a quarterly series of images over six years (2015–2020). This Semantic Vegetation Index (SVI) is based on deep semantic segmentation to integrate it with a citizen science project (Fluker Post) for automated environmental monitoring. It has collected thousands of vegetation images shared by various visitors from around 168 different points located in Australian regions over six years. This paper first uses a deep learning-based semantic segmentation model to classify vegetation in repeated photographs. A semantic vegetation index is then calculated and plotted in a time series to reflect seasonal variations and environmental impacts. The results show variational trends of vegetation cover for each year, and the semantic segmentation model performed well in calculating vegetation cover based on semantic pixels (overall accuracy = 97.7%). This work has solved a number of problems related to changes in viewpoint, scale, zoom, and seasonal changes in order to normalise RGB image data collected from different image devices.

7.1 Introduction

The increasing population of the world and the change in land use by them have significantly affected vegetation and landscape composition [218, 219]. It has been discovered that changes in the kind of land cover (such as building developments) have a substantial relationship with changes in vegetation. Calculations of various vegetation indexes are frequently used to automate the identification of vegetation cover and important variations in the environment [227]. Therefore, all of these natural resources require environmental monitoring to be protected and conserved,

and this is especially true for open green areas and public lands. Because vegetation plays a vital role in the improvement of an ecosystem, it enables the climate to have the positive change required for better living. This is being done with the help of land-care agencies, environmental groups, and local governments using drones, UAVs, satellites, and remote sensing.

Vegetation landscapes, by their very nature, are dynamic and constantly changing in response to human use. Unfortunately, not all changes to the landscape are positive ones. In turn, this has negative consequences for farmland and grazing resources, landscape diversity, cultural values, and biological variety [255, 256]. Vegetative landscape monitoring is a good way to make sure that changes in the landscape are going in the right direction.

It is recommended to monitor anything throughout time, which means observing and documenting any changes that occur. Monitoring is all about performing periodic assessments or surveys, collecting outcomes, and then comparing them to determine the efficacy of activities or the development of projects. Management actions might be evaluated based on the feedback they receive via the monitoring team, and it also helps to determine whether natural resources are improving, stabilising, or decreasing. Understanding how and why the land and its vegetation behave over time is essential for land managers to do their jobs effectively.

Researchers have developed a number of different algorithms for calculating the vegetation index from photographs. According to reports, researchers have placed a strong emphasis on remote sensing [257] images because of their numerous advantages, including large area exposure, reliability, and many others. The green area seen in remote sensing photos is used to extract the vegetation region available in certain places [258]. Remote sensing data is collected from above by sensors (aircraft, space) that misses the glimpse of vegetation. Accordingly, ground-level profile

views are inadequate for assessing urban greenery even if green indices generated from remotely sensed image data could assist in quantifying greenery. There's a big difference between what the average person sees on the ground and what remote sensing technologies see [228].

A simple but extremely important way of monitoring vegetation, whether it is remnant vegetation or revegetated sites, is to take a series of images, which is referred to as "photopoint monitoring" or "repeat photography" [44]. Repeat photography is a method where ground-level photographs are taken from exactly the same location at different points in time. In the case of landscapes, the time stamps between the images are usually several years or decades, sometimes even up to a whole century. However, with the technological advances in aerial and satellite remote sensing, ground-based photographs have lost most of their relevance in modern landscape monitoring. A repeat photograph is a photograph that has been purposefully created to reproduce certain characteristics of another, previously taken photograph. The new image often duplicates the location coordinates of the original, presenting the user with the identical scenario for the second time and encouraging them to compare the two images [147]. In the literature, many researchers have used repeat photography for many purposes, like estimating changes in tree lines [**?**], for analysing plant phenology, vegetation cover estimation [**?**, 71, **?**, 73] and many others.

A number of ecological studies used a combination of repeat photographs as well as field measurements to obtain quantitative results [76]. Clark et al., [77] performed point sampling along horizontal transects that were randomly placed through the image to create their results. They manually categorised each image into cover classes, and they developed the concept of image cover as a quantitative metric. Roush et al., [67] calculated the vegetation cover percentage by applying a rectangular grid on top of each photo. Fortin et al., [27] used repeat photographs collected as part of the

Mountain Legacy Project to derive class-specific land cover estimates and compared them to Landsat classifications. There is one thing that all of these studies have in common: the classification step is completed manually, usually by drawing polygons around specified landscape objects and then performing a visual interpretation of the results. Despite the fact that there is a large range of automatic segmentation and classification algorithms for aerial and satellite images, there is no single best solution, but the fact that ground-based images have such an angular viewpoint means that these approaches do not operate in the same manner [26]. Rohde et al., [259] findings are based on a comparison of 100 re-photographed or "matched" historical landscape images. Changes in woody cover at every photo site were analysed and used as a proxy for climatic change in the area. Toda et al., [260], used the Bartlett Experimental Forest data to compare the phenological metrics of leaf area index, plant area index, and their associated transition dates. They gathered digital repeat photography images using two separate methods: "canopy cover" and "phenocam". Zier et al., [41] presented his findings from an investigation of vegetation change in the San Juan Mountains conducted with repeat photography and Hendrick et al., [42] in the Appalachian Mountains. In Australia, the practise of repeat photography has been quite rare in recent years. With the help of repetitive photos, John Pickard [44], demonstrated how the landscape of Australia has changed throughout time. But most of the methods that were talked about involve a lot of manual work that takes a lot of time and does not use automation.

Digital image classification has become increasingly automated with the help of machine learning and deep learning, which has quickly become one of the most popular methodologies [261, 262]. In particular, deep learning eliminates the requirement for the time-consuming and complex feature extraction (such as fractal dimension, local binary patterns, texture features, shape features, colour features, etc.) method

that was previously required. Instead, throughout the CNN training process, the model learns and extracts the necessary information on its own, without the need for human intervention. Deep learning's most major drawback is that it necessitates the use of massive amounts of labelled training data [263]. Convolutional neural networks (CNN) are one of the deep learning architectures that are particularly well suited for image analysis because of their capacity to extract spatial characteristics from images. In convolutional neural networks, the image is fed into the network in its raw form (pixels). The network transforms the image many times. First, the image goes through many convolutional layers. In those convolutional layers, the network learns new and increasingly complex features in its layers. Then the transformed image information goes through the fully connected layers and turns into a classification or prediction. CNNs have been shown to be quite effective in a wide range of applications, including object detection [264], plant segmentation [147], classification [265, 266, 267, 268, 269], plant disease identification and classification [146, 270] and semantic segmentation [271, 272, 236].

Researchers have proposed various algorithms for the calculation of the vegetation index using images. However, it appears that researchers have placed a high priority on remote sensing images due to benefits such as large area exposure, repeatability, and many others [257]. Remote sensed images are used to extract the green area present in them, which represents the vegetation region available in some particular region [258]. Images from remote sensing are taken from above or from the air, which is higher than ground level and captures a larger area. This makes it hard to estimate how much vegetation there is in a certain place. Pictures taken from the ground can help a lot with figuring out how the vegetation is changing, which can be used to help the environment.

The profile view of the site can provide in-depth analysis, as demonstrated by

Yang et al., [228]. They compare profile views of two different forests and demonstrate how depth information can be extracted using this technique. However, the advancement in computer vision has enabled researchers to work on the profile view images and carry out comparisons between two images taken at different times from different angles, which is basically a phenomenon of repeat photography.

Kendal et al., [**?**] used colour thresholding for the extraction of vegetation index. The technique proved to be promising. However, only using colour features for segmentation is not an efficient method as any clutter information in the image can match the vegetation colour. Harbaš et al., [273] used a fully convolutional network (FCN) to detect and segment roadside vegetation for the navigation of autonomous vehicles. Hung et al., [274] used a learned feature approach to classify weeds and non-weeds. The images used were acquired by an unmanned vehicle. Furthermore, in recent years, Bawde et al., [275] proposed an algorithm that classifies forbs and grass. However, researchers did not focus on the change in vegetation index using repeat photography. The major focus remained on the classification of different species, as it is really important to get information about vegetation change so that steps for the improvement of the environment can be taken.

This paper focuses on the calculation of changes and monitoring in vegetation index for which registration of images is performed at an initial level. Registration is carried out because focus of authors is on repeat photography, which carries angle and scale variation. Therefore, transformation is needed to be done for this problem. This has already been done by the authors in their previously published work [4]. A novel deep affine invariant network was proposed for non-rigid image registration of multi-temporal repeated photography. Strong point matching and affine invariance are included in the suggested framework for reliable multi-temporal image registration. Furthermore, a novel approach to semantic segmentation is adopted to efficiently

extract the vegetation region from the image. The vegetation region can then be used for the calculation of the vegetation index in an effective manner. The overall work flow of this proposed study is presented in Fig 7.1.

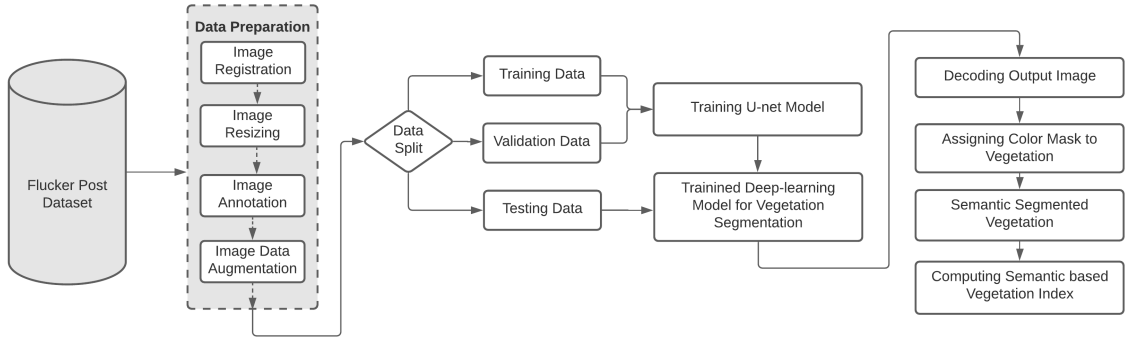


Figure 7.1: The general data flow diagram shows how the proposed system for checking the health of plants would work.

The rest of the paper is organised as follows: Section 2 is about the dataset taken into account; section 3 is detailed information regarding the proposed algorithm and methodology; section 4 discusses the results achieved by the proposed methodology; and section 5 is the conclusion section of this paper.

7.2 Materials and Methods

7.2.1 Taking Repeat Photographs

It has become increasingly vital to be able to work with large numbers of images and annotate them for specific purposes. The images may come from a variety of sources, including different users, the media, artists, or even video surveillance. The images used and how they are assembled can distinguish the content and audience relevance of a work [276]. Repeat photography is more than simply taking a second picture at the same moment in time when conducting scientific research.

In order to avoid perspective shifts, these images must be taken from the same place every time. This is more important than the physical characteristics [276] shown in Fig 7.2. It needs careful planning and preparation, as well as close attention to the physical environment around it.

Figure 7.2: An example of repeat photography, showing images taken of a site at different times.

Lighting, weather, and seasonal changes should be as similar as possible in order to produce visually identical image pairs. However, weather and lighting circumstances are difficult to recreate in practise due to the fact that field activities must be planned in advance, finances are limited, and deadlines are frequently constrained in nature [277].

7.2.2 Study Area / Fluker post project dataset

For the proposed technique, we used the Flukerpost dataset [80]. The Fluker Post dataset is an initiative taken by the Victorian Government and Victoria University to maintain a record of the environmental conditions of different sites throughout

Australia. The dataset has been put together and is ready for people to contribute. There are around one hundred and sixty-eight (168) Fluker post point sites. At a Fluker Post point, there is no camera installed. Instead, visitors passing by from fixed photo-points use the Fluker Post app on their own phones to snap a photo of the sight in front of them. This simple method of taking pictures over and over again is a good way to manage natural resources over the long term.

The Fluker Post has been running effectively so far, collecting useful imagery data about vegetation, parks, watersheds, rivers, streams, etc. An example of a repeated image collected from 2015 to 2020 quarterly of one study site is presented in the Fig 7.3.

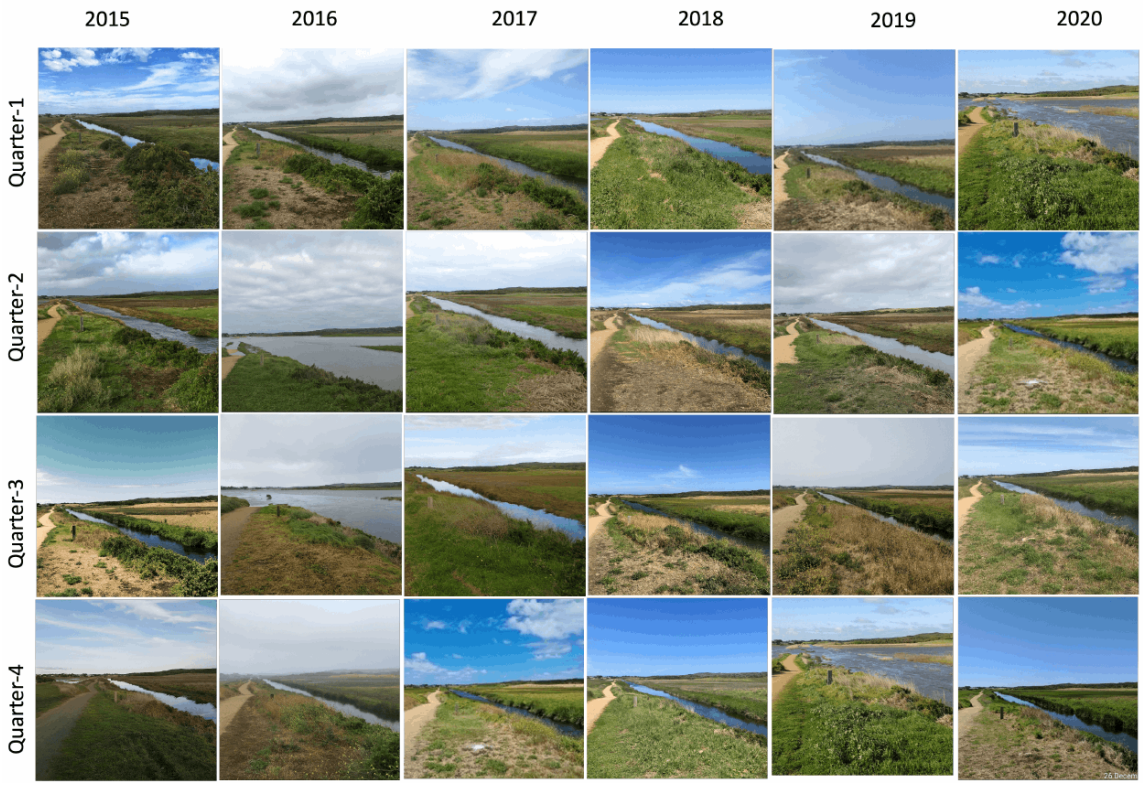


Figure 7.3: Sample images of the Warrnambool Region site, one of the Fluker post points of collection, taken quarterly over several years between 2015 and 2020.

This vast amount of photographic data is currently being manually analysed in

order to point out areas where state agencies like Parks Victoria might be able to help. But because of the large volume of data, this traditional approach is less successful, and more automation is needed to improve and speed up data processing. Manual image analysis is significantly more time-consuming and inefficient when dealing with repeat photography data. In total, more than 4000 photographs have been collected for this project, all of which have been sorted into albums according to various areas throughout Australia. This project's website address is https://www.flukerpost.com/, and its interface is shown in Fig 7.4. The issue, however, is that the images were captured using a variety of camera sensors and from various vantage points. Photographs are taken at random times of the day and seasons. All of these things make it hard to do, and as a result, comparing and extracting useful information like vegetation change detection has become very challenging.

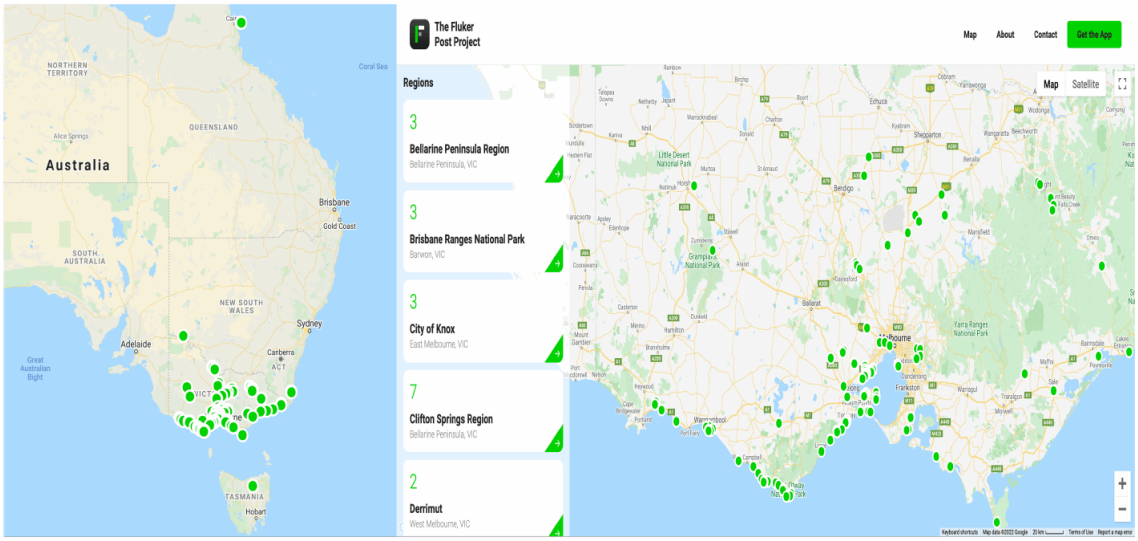


Figure 7.4: Fluker Post Project Location and Details: a.) The green circles indicate the locations of posts installed across Australia. b.) Details about each post's location, as well as the number of images saved for a specific site.

7.2.3 Vegetation Segmentation

The advancement of convolutional neural networks (CNNs) has achieved many milestones in the field of computer vision. They have given state of the art results for many research problems as the algorithm doesn't require extraction of features separately and machine learning processes separately. It performs both tasks in the network and provides promising results. Networks take the image's complex features and use them to identify any object in the image.

Our major focus is to calculate and then monitor the vegetation index of a location over different time series. For our process to be completed, we need to classify the vegetation region and non-vegetation region in the input image. We have developed a pixel-wise classification algorithm so that segmentation of vegetation can be done with the greatest efficiency.

7.2.4 U-Net

The U-Net model [249], which has an encoder-decoder architecture, as shown in Fig 7.5. was used in this research. The fact that U-Net is symmetric means that it does not have to deal with the connections between the up and down-sampling paths, which is advantageous when used as a concatenation operator. After they have been trained based on the colour variable in the dataset, models assign a colour to an object.

Typically, in the U-Net approach, the input image is first processed by an encoder path, which is composed of convolutional and pooling layers that degrade the spatial resolution of the input image, according to conventional practice. It is then followed by a decoder path that restores the original spatial imagery resolution by adopting up-sampling layers followed by convolutional layers, which is a technique known

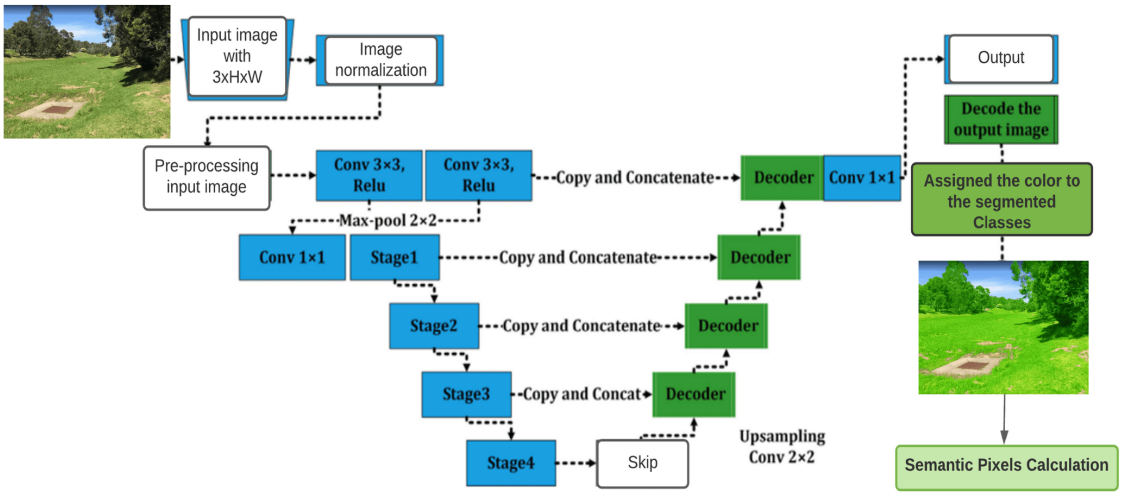


Figure 7.5: The architecture of U-Net shows network processes.

as "up-convolution". Apart from that, the network makes use of so-called skip connections, which connect the output of the relevant layers in the encoder path to the inputs of the decoder path by adding them to the inputs of the decoder path.

7.2.5 Vegetation Index Calculation From RGB Images

There are a variety of methods for calculating and monitoring the vegetation index. However, the majority of them employed either colour, threshold, or green area segmentation, which could lead to some encouraging outcomes. As a result, we propose a semantic-based approach for calculating and monitoring a robust vegetation index based on the distinctive colour of certain classes using a color-based approach. In the proposed work, RGB colour codes (107, 142, 35) were assigned to trees and vegetation terrain. After the trees and vegetation terrain have been separated, the correct masks are used to calculate the vegetation index.

7.2.6 The Proposed Semantic Vegetation Index (SVI)

For robust calculation and monitoring of the vegetation index of each sample location over different time stamps, we used the approach of semantic pixels (SP) calculation, based on the unique colour pixels assigned to a specific class and extracted based on the deep features through the use of a deep neural network.

For vegetation index calculation and monitoring, we used the Fluker Post project image dataset, as the data is available with a time series. We observed that one image of a location is not sufficient for index monitoring. Therefore, for calculating the vegetation index in this investigation, we used the repeat photography technique. We selected four images, quarterly, per year between 2015 and 2020 for each site, out of 168 total sites. Some of the sample images of a specific site (Warrnambool Region, VIC) are shown in Fig 7.3. In each sample photo, the number of semantic pixels will be counted, and the area will be equal to the total number of semantic pixels in one of the three photos of a certain place.

Consequently, for the purpose of this study, a single image was used to precisely compute the vegetation index based on the semantic pixels in order to cover the whole vegetation area seen in the image, and the vegetation index was calculated using the semantic pixels approach. The number of semantic pixels in each sample image will be computed as SP_a with respect to the total pixels ($Area_t$) of an image. The semantic vegetation index (SVI) is calculated with the following equation:

$$SVI = \frac{\sum_{i=1}^{n} SP_{a\_i}}{\sum_{i=1}^{n} Area_{t\_i}} * 100\% \tag{7.1}$$

Where SVI stands for Semantic Vegetation Index, SP_{a_i} presents semantic pixels area in an image and $Area_{t\_i}$ represents the sum of pixels in an image of a specific location/site.

7.3 Experiments and Results

7.3.1 Data preprocessing & preparation

The data set consists of around 3500 images taken at different times over several years. Therefore, it is essential to preprocess and prepare the data for model training in order to achieve better outcomes. We registered the images, performed data augmentation, labelled the data, and then separated the data into training, validation, and testing data (80%, 15% and 5% respectively) throughout the preprocessing and data preparation processes.

7.3.1.1 Image Registration

Image registration is one of the most critical processes in this process. Image registration allows the system to compare two images that have undergone a similar alteration to one another. The image that is being transformed is referred to as the sensed image, and the image that is being altered in relation to it is referred to as the reference image. Image registration tries to eliminate the geometric position inconsistency between two photographs, resulting in the same image coordinates reflecting the same objects on both dates. To properly handle and analyse multiple images, it is necessary to perform image registration first. The vegetation images must be re-registered in the same dimensions as the images in the dataset because the images in the dataset were taken at various times, seasons, and locations, so their dimensions are different. To properly handle and analyse multiple photos, it is necessary to perform image registration first [104]. It's a technique for combining photos (two or more) obtained at various times, from various vantage points, and with various sensors to create a composite image. The image registration technique

is adapted from the authors' (our) [4] previously published work. In that paper, the authors proposed a new deep affine invariant network for non-rigid image registration of multi-temporal repeat photography. Robust point matching and affine invariance are also part of the proposed framework for robust multi-temporal image registration. An example of image registration in the form of a checkerboard is presented in the Fig 7.6.

In this study, the images taken quarterly over a year of various site locations were used and resized to 256 x 256 pixels after the image registration process. Because the images are taken from different angles and sensors, it was necessary to use an image registration process to normalise the data for the neural network training and get better results.

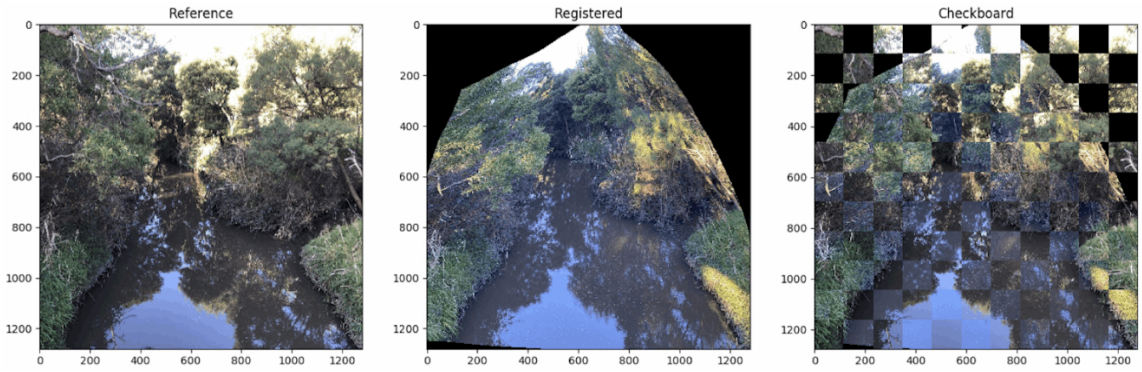


Figure 7.6: An example of image registration is applied to an input and sensed image.

7.3.1.2 Data augmentation

A large number of images are used to train a deep neural network model to achieve highly precise prediction and accuracy. In our case, some of the Fluker post sites had fewer images than others. Because of this, the technique of data augmentation was used on the sites with fewer images. The process of data augmentation [278] provided us with new images based on our existing images. Different augmenta-

tion techniques like blurriness, rotation, flipping (horizontal and vertical), zooming, translation, and the addition of noise were applied accordingly. An illustration of different augmentation techniques is shown in Fig 7.7. By using this method, the number of images in our dataset grew, which is important for getting more accurate results after the training stage of a CNN.

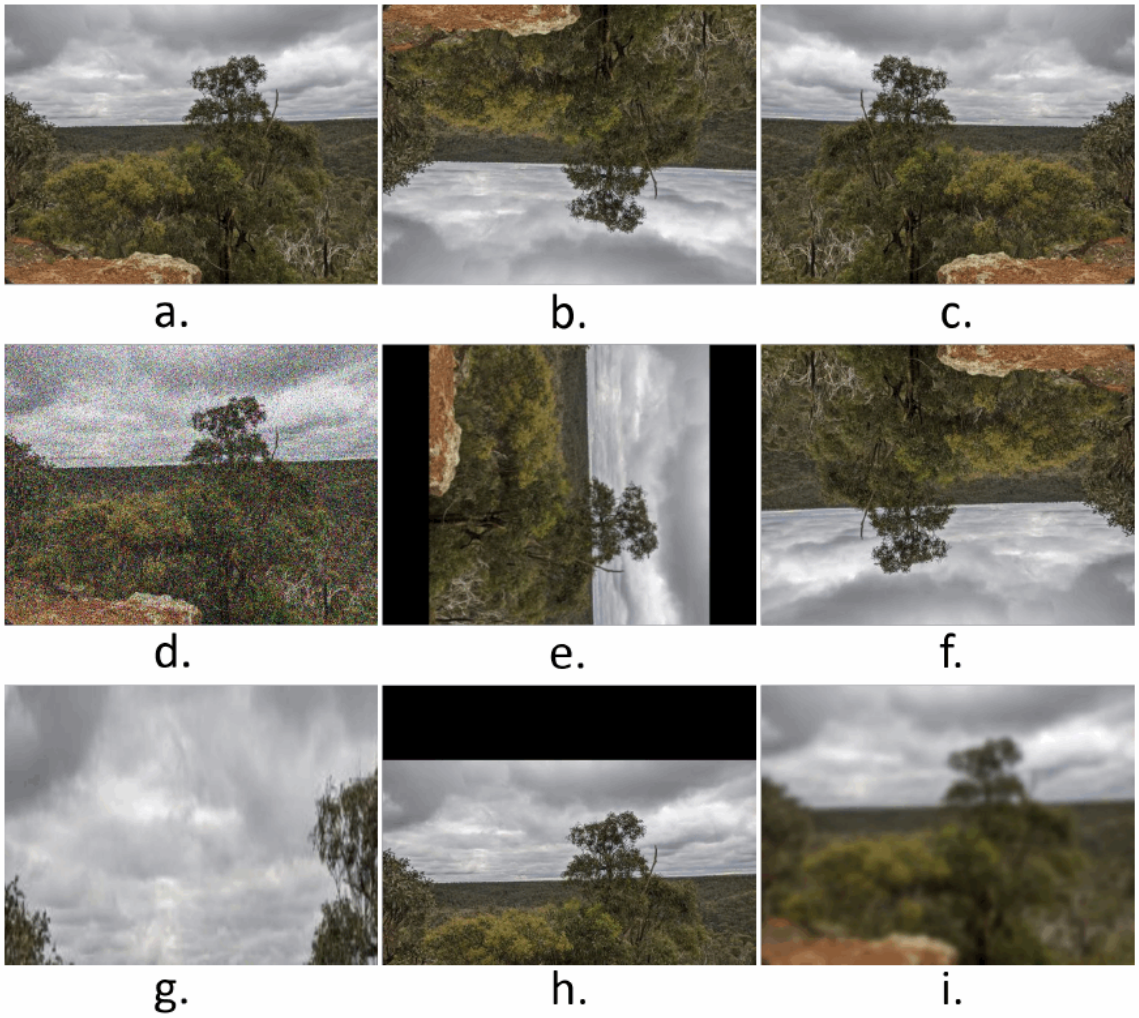


Figure 7.7: Different data augmentation technique applied include: (a). Original Image, (b). Vertical Flip, (c). Horizontal Flip, (d) Random Gaussian Noise, (e). 90 degree rotation, (f). 180 degree rotation, (g). Random zoom, (h). Translation, and (i). Blur

7.3.1.3 Data Labelling

The light intensity of the dataset images varies since the images were taken at different times and with different cameras. Therefore, preprocessing of the dataset images is required before the registration procedure to ensure that the images are usable for the training and testing process. The training dataset was annotated with a cloud-based program called "Apeer" [253], which is available for free as part of a ZEISS initiative. Image annotation generates labels that serve as the basis for machine learning training. The amount of training data as well as the correctness of annotations are both important factors in determining machine learning accuracy. Fig 7.8. presents a high-level overview of the annotation process. The region of interest (ROI) is the labelled area of an image slice, which is usually only a small portion of the image. The ROI mask is inserted into the CNN with the image as a binary map, with pixels belonging to the ROI set to one and pixels belonging to the background set to zero.

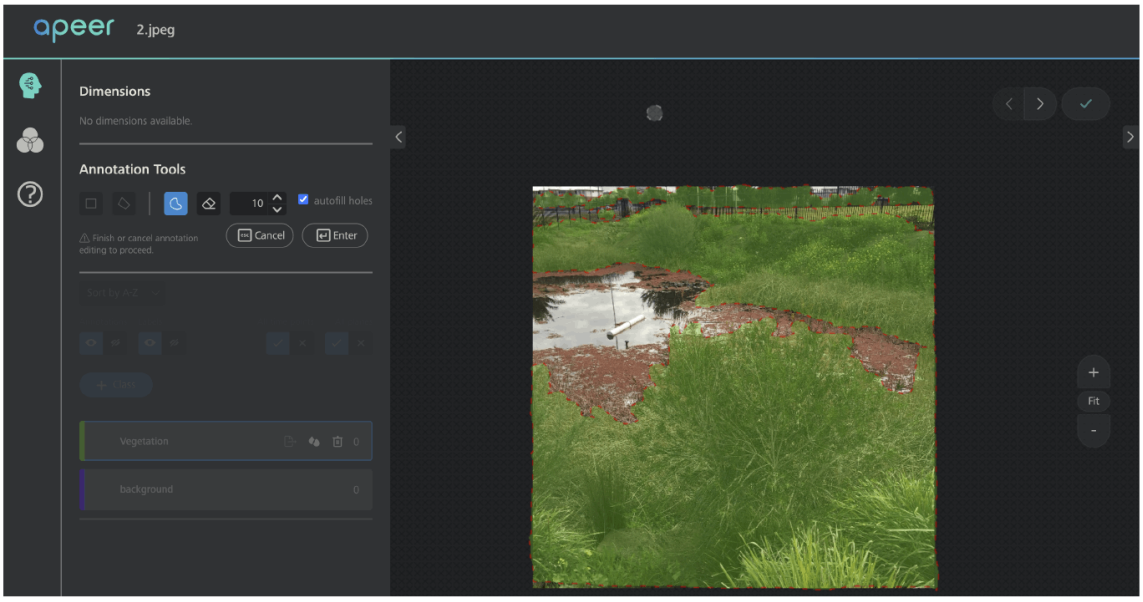


Figure 7.8: Apeer, an annotation tool's interface, and a sample annotated image.

7.3.2 Network model training

The entire dataset was divided into three parts: training, validation, and testing sets, each comprising 80%, 15%, and 5% of the total, respectively. Before starting the training, hyperparameters were set to avoid the overfitting and underfitting issues of the model. The hyper-parameters are set as: batch size kept at 16, learning rate as 0.0001, loss function as categorical cross-entropy, number of iterations/epochs as 200, NMS threshold as 0.45, and an optimizer as Stochastic gradient descent (SGD). The loss function ensures that the neural network optimises itself by reducing the amount of error it generates during the training process. The training loss indicates how well the model optimises the training data, while the validation loss indicates how well the model fits new data. Non-Maximum Suppression (NMS) is a technique used in numerous computer vision tasks. It is a class of algorithms to select one entity (e.g., bounding boxes) out of many overlapping entities. Most of the time, the criteria are some kind of probability number and a way to measure overlap, such as "intersection over union". The training loss, validation loss, training accuracy, and validation accuracy curve graphs are presented in Fig 7.9 a.) Training and Validation Loss; b.) Accuracy of U-Net Model Training and Validation.

The Table 7.1 lists the hardware and software resources used in the experiments and results.

7.3.3 Model Performance Evaluation

Both the training and validation sets were used to calculate the accuracy and loss of the model. The prediction accuracy on individual images was calculated using the 175 images from the test set (= 5%), which had been separated from the total number of samples before training. The confusion matrix is made up of pixel numbers

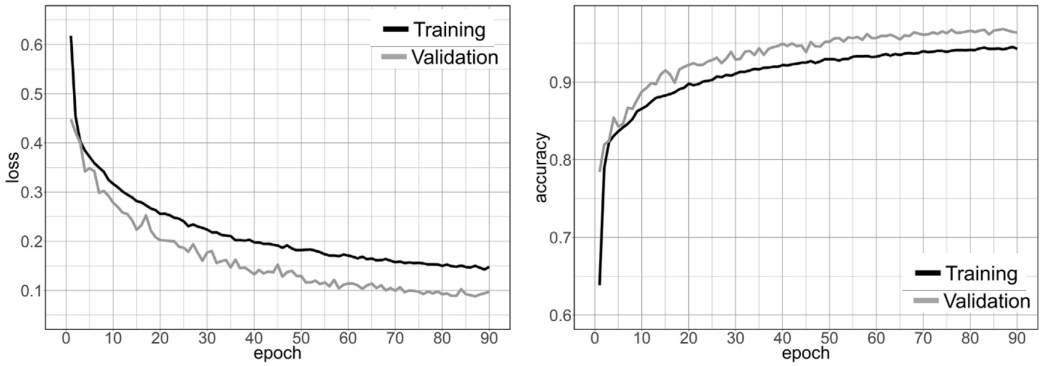


Figure 7.9: Over 90 epochs, the learning process for loss (on the left) and model accuracy (on the right) are shown. If dropout is used on the training data, the accuracy of the training data and the validation data will be different.

Table 7.1: **The details of the configuration of the experimental environment.**

Item Name	Parameter
Central processing unit (CPU)	Intel i7 9700k
Operating system	MS Windows 10
Operating volatile memory	32GB RAM
Graphic processing unit (GPU)	Nvidia Titan RTX
Development environment configuration	Python 3.8 + TensorFlow 2.5 + CUDA 11.2 + cuDNN V8.1.0 + Visual Studio 2019

representing true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). As for accuracy metrics, we use $Precision$, $Recall$, $F1-score$, and $Overall\ Accuracy$ (OA). These metrics are calculated as follows:

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{7.2}$$

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{7.3}$$

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{7.4}$$

$$Overall\ Accuracy\ (OA) = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{7.5}$$

The learning process stopped after 90 epochs when the learning curve converged and loss values stopped decreasing. Fig 7.9. depicts the improvement in loss and accuracy during the training process. It is a fact, that when dropout and data augmentation are applied exclusively to training data, the validation accuracy often exceeds the training accuracy. The model achieved a maximum accuracy of 96.6% (loss = 0.11), based on the validation set. Additionally, we examined the model's performance on a second test dataset consisting of n = 175 unique images that were not utilised in the model selection procedure. The model achieved an accuracy of 97.4% (loss = 0.07) on these individual images. After testing the model, overall accuracy varied significantly amongst images, ranging from 74.1% to 96.6%.

A comparison was also performed through two CNN architectures: Fully convolutional network (FCN) and U-Net, while keeping the same technical environments as mentioned in Table 7.1. After conducting experiments, the following results, as mentioned in Table 7.2, were achieved. A more comprehensive comparison has already been performed with the current literature studied in the author's (our) previous published work [271].

Table 7.2: Comparative analysis of FCN and U-Net results.

Segmentation Model	Precision	Recall	F1-Score	Pixel Accuracy	IoU	mIoU
FCN	94.2	85.3	91.1	90.4	83.3	81
U-Net	96	91.8	92.3	93.4	87.5	84

The Fig 7.10. shows some of the segmentation results from the randomly selected test images. They are quite promising results.

Figure 7.10: Some sample segmentation results for the randomly selected test input images.

7.4 Discussion

While evaluating the trained U-Net model's performance in classifying vegetation in repeated landscape photographs, an over-all accuracy (OA) of 97.4% on individual images was achieved. In contrast, several studies for similar problems, such as Zhang et al., [279] used a spatial contextual superpixel model to achieve an accuracy of 79.8% on the class "tree" in real roadside images. For trees in ground images, Byeon et al., [280] used an LSTM Re-current Neural Network (RNN) to get a class accuracy

of 64.2%. In a similar manner, Shuai et al., [281] paired a CNN with a directed acyclic graphic RNN (DAG-RNN) for scene identification and achieved an accuracy of 82.5% for the tree-class.

The quality of the image was recognised as a significant influencing element. It is possible that the wide range of image content, resolution, scale, and illumination used in a specific job has an impact on the classification and identification accuracy. The advancements in digital camera technology have resulted in a significant improvement in image quality over time. As a result, older RGB images are often of poorer quality than current RGB images, which has a negative impact on the efficacy of the classification and detection methods. Clark et al., [77] encounter the same image quality issues when attempting to measure vegetation changes between repeat images using transect point sampling.

When Skovsen et al., [282] attempted to differentiate clover from grasses and weeds using fuzzy images, they found a greater rate of misclassification because of the quality of the images.

Segmentation and computing the index values were done for all the Fluker Post sites. However, only a few of them (Youyung Park, the Warrnambool region, Knox, and the Kororoit Creek site) are shown in the Fig 7.11. From the results, it is observed that those Fluker post sites where the visitors frequently go have a large number of images for each month of the year, while some sites have fewer images. Also, there were only a couple of images for some sites. The above facts may impact the results in terms of trends. To overcome the above issue, an average index value was computed to show the trend of a specific site. For example, in Fig 7.11, the semantic vegetation index values computed are the average index values for each quarter of a year.

Seasonal variation affecting the vegetation health: According to the Bu-

Figure 7.11: Figures depicting the average semantic vegetation index calculated quarterly from 2015 to 2020 for a.) Youyung Park, b.) Warrnambool region, c.) City of Knox, and d.) Kororoit Creek site.

reau of Meteorology, Australia, there are several environmental factor that seriously affect the vegetation health [283]. They are:

- Low rainfalls
- Extremely dry season
- Consecutive periods of dry or cold weather
- Early and long-lasting devastating bushfires

Low rainfalls: Australia had a very wet winter and spring in 2016. In 2017, things dried up a lot. Most of the interior of southeastern Australia didn't get much rain in 2017, 2018, and 2019. In some places, like western Victoria and western Queensland, there was more dry weather than other parts of the country had in these years. Only a slight recovery followed a very dry and cool season in October

and December 2017 and 2018. For several years, there was record-low rainfall. This year's cool season was very dry, and it did not end until the end of the year. From January 2017 to December 2019, the Murray–Darling Basin and New South Wales have had the driest three years on record, as shown in Fig 7.12. Other areas that have not been getting enough rain for a long time include eastern Victoria, eastern and northern Tasmania, eastern South Australia, except for the southeast and some parts of the southwest, and Western Australia.

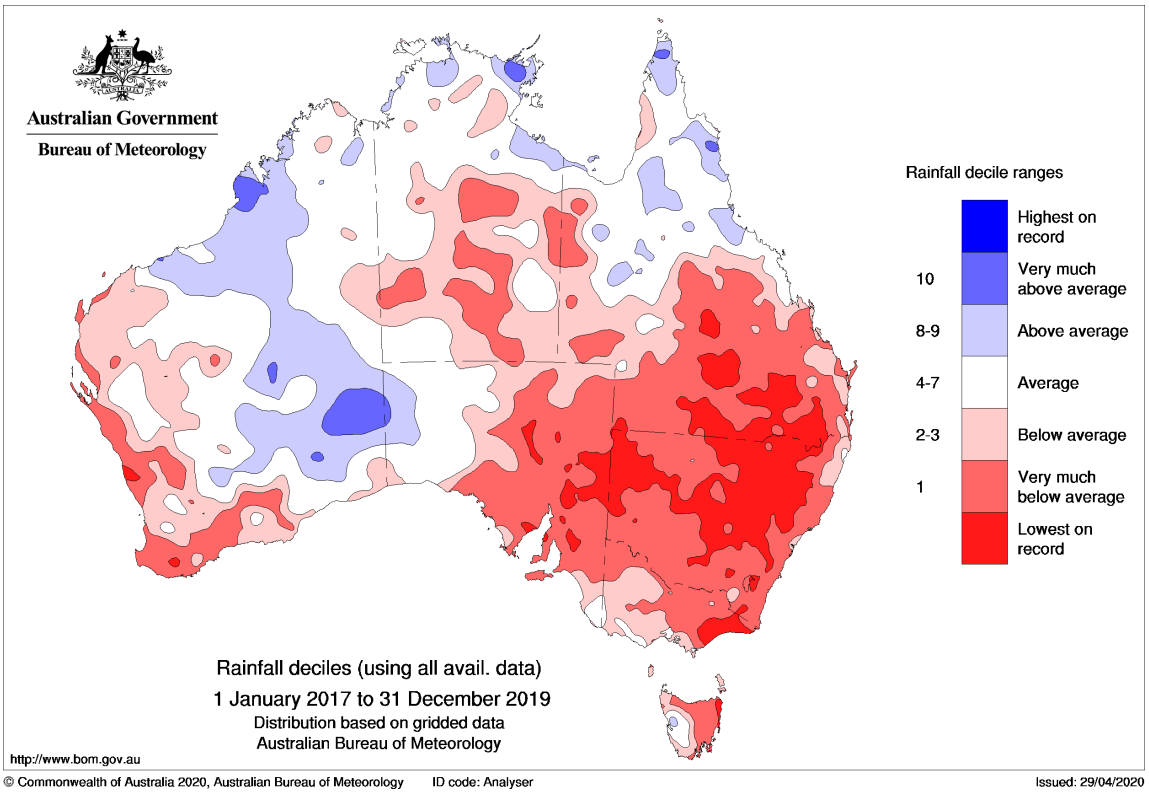


Figure 7.12: Australian rainfall deciles for the combined three-year April–September periods of 2017, 2018, and 2019. (based on all years since 1900).

Extremely dry season: Extremely dry conditions, especially in the northern half of New South Wales, have had the worst droughts over long periods of time.This is in stark contrast to what occurred during the Millennium drought, when the southern basin had the worst droughts and the north had the best. Two other places

with a long-term lack of rain, namely Gippsland in eastern Victoria and eastern Tasmania, were two others. In Gippsland, the most severe deficits were found. 2019 was marked as the third year in a row that the area did not receive enough rain. While it wasn't as dry as in 2017 and 2018, the lack of rain kept building up over time. This led to multi-year deficits. The east coast of Tasmania also saw a lot less rain than usual during this time.

Consecutive dry, cool seasons: These three years (2017-2019) didn't get enough rain, but they were terrible in the fall and winter. From 2017 to 2019, it was very dry in a lot of New South Wales from April to September each year. It was the same in Queensland, south of the Tropic of Capricorn, where the weather was the same. People in New South Wales and the Murray–Darling Basin didn't get much rain in April and September.

Early and long-lasting devastating bushfires: As measured by the Forest Fire Danger Index (FFDI), which is a common way to measure fire weather conditions, spring 2019 saw the highest level of fire weather danger across the whole country. Record high FFDI values were found in all states and territories. The hot weather made things even more dangerous for fires in December 2019 and early January 2020.

As can be seen from the above trends in Fig 7.13, during the years 2017–2019, the environmental factors mentioned above significantly affected the vegetation in most of the Australian regions and territories. Therefore, it can be seen that the average semantic vegetation index calculated has low values due to environmental factors. However, beginning in 2020, those areas were given extra care with the help of citizens and the government to help the plants grow back and save the biodiversity.

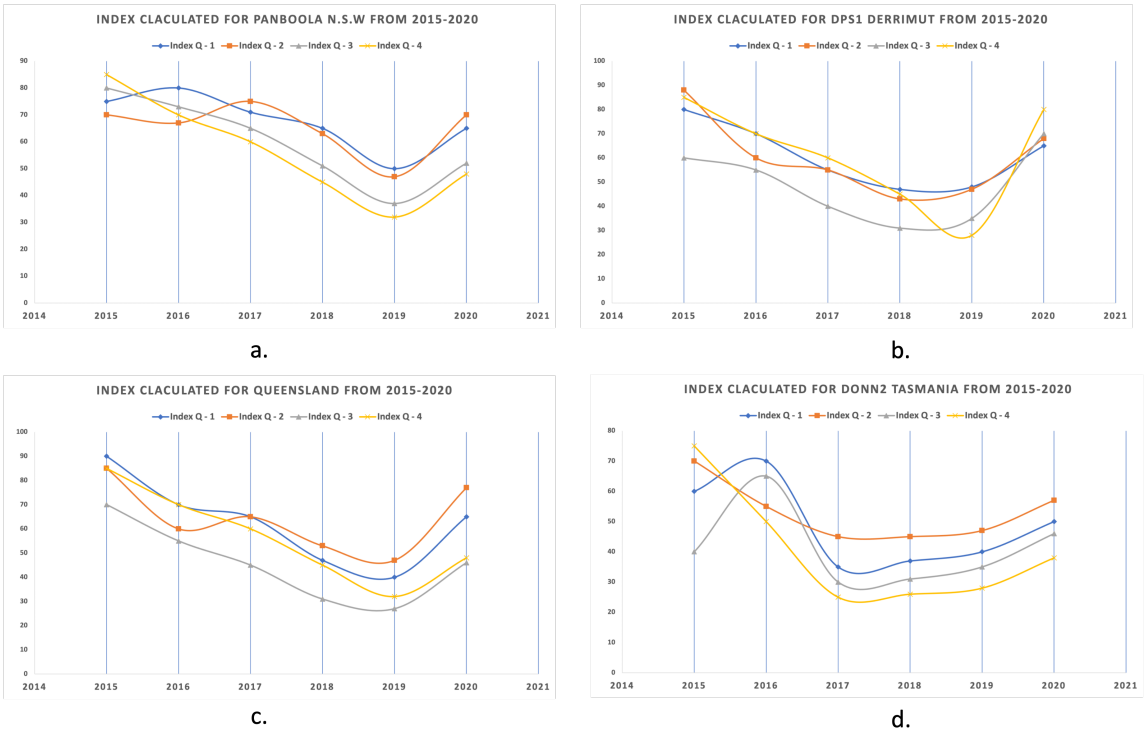


Figure 7.13: Trends of vegetation with respect to environmental factors from 2015-2020 quarterly for a.) Panboola (NSW), b.) Derrimut (VIC), c.) Queensland and d.) Donn2 (Tasmania).

7.5 Conclusion

This research article proposes automatic vegetation health monitoring using repetitive photography. This study tried to address challenges related to citizen science data processing. For this purpose, we normalise data collected from various visitors who visited Fluker post point sites in Australia and use it to estimate the vegetation cover change from images taken quarterly over six years in various specific locations throughout Australia. In general, using repeat photography in vegetation monitoring brings significant value to other quantitative data retrieved from remote sensing and field measurements. Furthermore, visitor-acquired photographs can raise awareness about landscape and vegetation change among policymakers and the general public and provide clear feedback on the effects of land management. It is observed that a few significant factors can impact the performance of the automatic approach, including image quality, shadow cast, and varying scale. Deep characteristics learned

from a deep neural network are used to make sure that the vegetation index for each sample location is accurate. The proposed method for segmenting vegetative areas has produced encouraging results. Based on the results achieved from the semantic segmentation model, the trends are plotted. Those plotted trends revealed that some of the Fluker post site vegetation increased gradually at some locations, the vegetation treads remained almost the same, showing no significant increase or decrease, while some of the study sites showed a dramatic decline in vegetation due to floods and harsh weather. Thus, those results present the increase or decrease of vegetation at specific site locations, which can be beneficial information for agriculture management officials and the research community on a wide range of research issues. It provides a robust platform to handle citizen science data for automated community service projects. The images in the Fluker Post Project are the collection of images acquired by visitors and citizens over different times of the year with their handheld cameras and mobile devices, in other words, through various sensors. However, there are a few challenges to be kept in mind. If the same pictures were taken with the UAV, it would be important to make sure that only authorised people with licences and basic knowledge of taking pictures were doing it. Secondly, UAV equipment is expensive, so cost would be allocated for that. The same idea could be used to look at changes in vegetation in a city using Google Street View (GSV) imagery, as long as there are enough images for that city.

Chapter 8

Conclusions And Future Work Recommendations

This chapter outlines the project's primary outcomes and advantages before making various recommendations for future research based on identified research gaps. Despite the fact that each chapter's conclusion is delivered at the end, the overall key findings are presented in this chapter. Finally, based on the findings of this study, it makes some recommendations for future research directions.

8.1 Conclusions:

Through a series of published works, this thesis has its core focus on the automation of vegetation segmentation and health monitoring using deep learning. Using deep convolutional neural networks (CNNs), this thesis contributes novel techniques for multiview vegetation segmentation, robust calculation of vegetation indexes, and real-time vegetation health assessment. This chapter summarises its contributions, discusses the research findings, and suggests future research directions.

In today's world, the population has increased rapidly with the advancement of technology, which has created a need for and importance of environmental moni-

toring. Humans have affected the environment to such an extent that the time has come to pay our attention to this issue. One of the most important components of environmental monitoring is observing the information regarding vegetation, which is essential to predict at an early stage regarding ongoing trends. Even though urban landscapes are getting bigger and more complicated, planners and experts are becoming more aware of how vegetation can help solve many of the problems that come with urbanization.

Vegetation management and vegetation index calculation pose a number of challenges due to their multidisciplinary nature, such as automated detection, health assessment, and monitoring of vegetation and trees by integrating techniques from computer vision, machine learning, and remote sensing. As computer vision and machine learning have become more advanced, researchers are now able to come up with new algorithms that will allow an automated system to monitor the environment.

In this thesis, studies were conducted using a range of remote sensing datasets to examine the following critical areas of vegetation management:

- Automatic vegetation detection techniques provide new methods for vegetation segmentation and indexing in repeat photography.
- DeepLens Classification and Detection Model (DCDM) for real-time health assessment of plant leaves based on Amazon SageMaker.
- The classification of healthy and unhealthy trees and the identification of their geo-location.
- Multiview Semantic Vegetation Index (MSVI) model for sematic segmentation based vegetation index calculation.
- Semantic vegetation index (SVI) calculation and application for time series-

based vegetation health monitoring.

To demonstrate the applicability of remote sensing for vegetation management, this thesis has provided a pragmatic framework for vegetation monitoring in urban environments. In the introduction of this thesis, four fundamental questions of vegetation monitoring were identified, including:

- How Multi-temporal imagery such Fluker post project dataset can be normalised to use it for convolution neural network training?
- How are vegetation indexes extracted using deep CNNs from street level imagery such as the Google Street View image dataset?
- How to utilise the ordinal information contained in image labels to derive a semantic vegetation index?
- How to build a health monitoring system using deep learning to quantify the vegetation's health in terms of an index?

Methodologies addressing each of the research questions were presented through a set of published papers, and their contributions are summarised as follows.

Chapter 2 of this thesis is based on data preparation techniques and a paper presented at the Pacific-Rim Symposium on Image and Video Technology (PSIVT), 2019, published in Lecture Notes in Computer Science, vol. 11854. Springer, Cham. The main focus of this paper is how multi-temporal image registration can be simultaneously made accurate with deep convolutional networks and robust against affine transformations. In this paper, we investigated the integration of citizen science and deep learning-based image registration to facilitate automated image analytics for environmental monitoring. We proposed a novel deep affine invariant network for non-rigid image registration of multi-temporal repeat photography. It is described

in this chapter that direct automation of large image collections is not suitable for any image analysis and registration due to variations in imaging taken by multi-sensors, i.e., variations in viewpoints, scales, luminosity, and camera characteristics. This multi-temporal image data is not registered. Thus, robust multi-temporal image registration is urgently required. We also integrated robust point matching and affine invariance into our framework for robust multi-temporal image registration. The experimental results indicated that the proposed approach delivered higher quality performance than the existing techniques. This work would open up new research directions to achieve a fully automated environmental monitoring system.

Related publication:

- *Multi-temporal registration of environmental imagery using affine invariant convolutional features.*

  Khan, A., Ulhaq, A., & Robinson, R. W. (2019, November). Multi-temporal registration of environmental imagery using affine invariant convolutional features. In Pacific-Rim Symposium on Image and Video Technology (pp. 269-280). Springer, Cham. doi: https://doi.org/10.1007/978-3-030-34879-3_21.

Chapter 3 of this thesis presents a paper, published at Statistics for Data Science and Policy Analysis. Springer, Singapore. In this chapter, a novel approach towards segmentation is proposed, which works on a machine learning-based algorithm for vegetation index calculation. The proposed algorithm registers the image so that comparison can be carried out in an accurate manner using a single framework for all the images. The registration algorithm aligns the new image with the already present previous image by performing a transformation. The registration process is followed by a segmentation process that segments out the vegetation region from the

image.

The proposed algorithm showed promising results, with an F-measure of 85.36%. The segmentation result leads us to an easy calculation of the vegetation index. which can be used to create a vegetation record for a specific site.

Related publication:

- *Detection of vegetation in environmental repeat photography: a new algorithmic approach in data science.*

  Khan, A., Ulhaq, A., Robinson, R., & Rehman, M. U. (2020). Detection of vegetation in environmental repeat photography: a new algorithmic approach in data science. In Statistics for Data Science and Policy Analysis (pp. 145-157). Springer, Singapore. doi: https://doi.org/10.1007/978-981-15-1735-8_11.

Chapter 4 is about plant health assessment. This chapter is composed of a research paper that was published in the PLOS ONE journal on December 17, 2020. This research paper describes the real-time health assessment of small plants. For implementing automatic detection and classification of a plant's health, Amazon SageMaker, a cloud-based environment, is used for training and testing of a model. The proposed model, known as the DeepLens Classification and Detection Model (DCDM), to identify and classify various fruits and vegetables' leaf diseases is based on Deep Convolutional Neural Network (DCNN) [96]. After completion of training DCDM, it was deployed on the Internet of Things (IoT) device known as AWS DeepLens to make it a scalable and efficient real-time classification and identification model.

To train the DCDM deep learning model, forty thousand images were used, and then it was evaluated on ten thousand images. It takes an average of 0.349s to test

an image for disease diagnosis and classification using AWS DeepLens, providing the consumer with disease information in less than a second. It obtained an average accuracy rate of 98.78% on test images. The findings are the first step towards a system based on an AWS DeepLens camera for plant disease diagnosis. Moreover, in this chapter, I also extracted feature maps [89] of an input image after passing through the CNN model and applied filters to visualise the activations through the CNN layers [101].

Related publication:

- *Real-time Plant Health assessment via Implementing Cloud-based Scalable Transfer Learning on AWS DeepLens.*

  Khan, A., Nawaz, U., Ulhaq, A., & Robinson, R. W. (2020). Real-time plant health assessment via implementing cloud-based scalable transfer learning on AWS DeepLens. Plos one, 15(12), e0243243. doi: https://doi.org/10.1371/journal.pone.0243243.

Chapter 5 presents a siamese convolutional neural network for health assessment and is described via a research article, Remote Sensing 13, no. 11: 2194. This paper proposes a deep learning-based network, the Siamese convolutional neural network (SCNN), combined with a modified brute-force-base line-of-bearing (LOB) algorithm that evaluates the health of eucalyptus trees as healthy or unhealthy and identifies their geo-location in real-time from Google Street View (GSV) and ground truth images. The reason behind this work was that street trees are an essential feature of urban or metropolitan areas, although relatively ignored. One such tree, eucalyptus, is a valuable asset for communities in urban areas (Australia).

The evaluation of tree health conditions is highly critical for biodiversity, forest management, global environmental monitoring and carbon dynamics. Unhealthy

tree features are identifiable and can be used to build a detection and classification model using deep learning to intelligently diagnose eucalyptus in healthy and unsanitary/dead trees. Detection and recognition of eucalyptus tree health presents a challenging task since many trees have few pixels across their input images, and some trees are also overshadowed by other trees and cannot be found due to weather conditions or lighting. To address these challenges and achieve high accuracy and precise prediction, a large amount of labelled training data for feature extraction of healthy and unhealthy class features is required. For this, we used GSV imagery, and ground truth images were taken from different viewpoints and a variety of places at different times.

Related publication:

- *Health Assessment of Eucalyptus Trees Using Siamese Network from Google Street and Ground Truth Images.*

  Khan, A., Asim, W., Ulhaq, A., Ghazi, B., & Robinson, R. W. (2021). Health Assessment of Eucalyptus Trees Using Siamese Network from Google Street and Ground Truth Images. Remote Sensing, 13(11), 2194. doi: https://doi.org/10.3390/rs13112194.

Chapter 6 contains a research article published in Remote Sens. 2022, 14, 228. This article proposed a novel vegetation index, the Multiview Semantic Vegetation Index (MSVI), that is robust to colour and seasonal variations and works for any imaging modality. MSVI is based on deep semantic segmentation and Multiview field coverage and can be integrated into any vegetation management platform. The MSVI is based on the deep features learned from a deep neural network to calculate the vegetation index of each sample location in the urban area. The Google Street View (GSV) imagery dataset is used for calculating and indexing the vegetation. A

single vertical point of view is insufficient to accurately express the surrounding vegetation index that pedestrians may observe; two vertical points of view are required. Therefore, the multiview semantic vegetation index (MSVI) is employed for six GSV images in this experiment to calculate the vegetation index, each spanning a 360° horizontal and three vertical angles of 45°, 0°and -45°, to calculate the vegetation index appropriately on the basis of the semantic pixels.

The Multiview semantic vegetation index (MSVI) took advantage of the characteristics of GSV images and used 18 GSV images taken from different viewing angles, making the index more efficient for evaluating street greenery in urban areas. As a result, it can provide a monitoring tool to analyse gains or losses in urban vegetation. During the experiments and training phase, FCN and U-Net achieved overall pixel accuracy of 89.4 percent and 92.4 percent, respectively. Thus, the MSVI can be a helpful instrument for analysing urban forestry and vegetation biomass since it provides an accurate and reliable objective method for assessing the plant cover at street level.

Related publication:

- *A Multiview Semantic Vegetation Index for Robust Estimation of Urban Vegetation Cover.*

  Khan, A., Asim, W., Ulhaq, A., & Robinson, R. W. (2022). A Multiview Semantic Vegetation Index for Robust Estimation of Urban Vegetation Cover. Remote Sensing, 14(1), 228. doi: https://doi.org/10.3390/rs14010228.

Chapter 7 of this thesis represents the health monitoring platform for citizen science data. In this paper, I build upon my previous work on the development of a Semantic Vegetation Index (SVI), mentioned in Chapter 6, and expand it to introduce a sematic vegetation health monitoring platform to monitor vegetation

health in a large landscape. This Semantic Vegetation Index (SVI) is based on deep semantic segmentation to integrate it into a citizen science project for automated environmental monitoring. In this paper, a deep learning-based semantic segmentation model is first used to classify vegetation in repeated photographs. A semantic vegetation index is then calculated and plotted in a time series to reflect seasonal variations and environmental impacts. In this work, we have addressed several challenges related to viewpoint variations, scale viand zoom related image variations, and seasonal variations to normalise RGB imaging data collected from diverse image devices. It is a robust platform that can handle citizen science data for automated community service projects.

It is observed that a few significant factors can impact the performance of the automatic approach, including image quality, shadow cast, and varying scale. The proposed method for segmenting vegetative areas has produced encouraging results. Based on the results achieved from the semantic segmentation model, the trends are plotted. Those plotted trends revealed that some of the Fluker post site vegetation increased gradually at some locations, the vegetation treads remained almost the same, showing no significant increase or decrease, while some of the study sites showed a dramatic decline in vegetation due to floods and harsh weather. Thus, those results present the increase or decrease of vegetation at specific site locations, which can be beneficial information for agriculture management officials and the research community on a wide range of research issues (Overall accuracy = 97.7%).

Related publication:

- *A Deep Semantic Vegetation Health Monitoring Platform For Citizen Science Imaging Data.*

  Khan, A.; Asim, W.; Ulhaq, A.; Robinson, R.W. "A Deep Semantic Vegeta-

tion Health Monitoring Platform For Citizen Science Imaging Data."
Under production with PLOS ONE Journal..

8.2 Future Research Recommendations:

This study has demonstrated the critical function that data acquired from remote sensing can play in index calculation and monitoring vegetation in urban contexts at various scales. However, the studies were conducted within certain boundaries, prompting further investigation. Hence, there are a few more areas where future research could be concentrated.

- In future work, to investigate the semantic segmentation of various tree types in thermal and LiDAR imagery, a new state-of-the-art algorithm could be further developed, for example, to aid in the classification of tree species and the accurate estimation of crown radius (CR) and crown base heights (CBH), which would aid in the evaluation of wood quality.

- Automating species identification and conservation, on the other hand, requires expertise in species identification, which can only be acquired through extensive training and experience. Field researchers, land managers, educators, government officials, and the general public would all benefit significantly from easily accessible, up-to-date tools that automate the process of species identification.

- Technological advancements such as imaging spectroscopy and LIDAR, the increasing free availability of satellite image time series, and online sharing of ecological data through crowd sourcing and open access datasets will enable further development in this field of mapping and monitoring vegetation at

various scales in the future.

- Conservation drones or unmanned drones may be effective in vegetation condition monitoring since they can correlate with the timing of site-based assessments, but their coverage is limited due to flying duration constraints. Such advancements could lead to a better understanding of vegetation's complex spatial patterns and processes, which is important for evidence-based natural resource management and measuring vegetative condition at various scales.

Bibliography

[1] Aws deeplens - developer guide. https://docs.aws.amazon.com/deeplens/latest/dg/deeplens-dg.pdf#what-is-deeplens. (Accessed on 08/10/2020).

[2] Xiaojiang Li, Chuanrong Zhang, Weidong Li, Robert Ricard, Qingyan Meng, and Weixing Zhang. Assessing street-level urban greenery using google street view and a modified green view index. *Urban Forestry & Urban Greening*, 14(3):675–685, 2015.

[3] Rencai Dong, Yonglin Zhang, and Jingzhu Zhao. How green are the streets within the sixth ring road of beijing? an analysis based on tencent street view pictures and the green view index. *International journal of environmental research and public health*, 15(7):1367, 2018.

[4] Asim Khan, Anwaar Ulhaq, and Randall W Robinson. Multi-temporal registration of environmental imagery using affine invariant convolutional features. In *Pacific-Rim Symposium on Image and Video Technology*, pages 269–280. Springer, 2019.

[5] Muhammad Sharif, Muhammad Attique Khan, Zahid Iqbal, Muhammad Faisal Azam, M Ikram Ullah Lali, and Muhammad Younus Javed. Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection. *Computers and electronics in agriculture*, 150:220–234, 2018.

[6] Justine Boulent, Samuel Foucher, Jérôme Théau, and Pierre-Luc St-Charles. Convolutional neural networks for the automatic identification of plant diseases. *Frontiers in plant science*, 10, 2019.

[7] Konstantinos P Ferentinos. Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture*, 145:311–318, 2018.

[8] Hyeon Park, Jee-Sook Eun, and Se-Han Kim. Image-based disease diagnosing and predicting of the crops through the deep learning mechanism. In *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 129–131. IEEE, 2017.

[9] M. Al-Amin, T. A. Bushra, and M. Nazmul Hoq. Prediction of potato disease from leaves using deep convolution neural network towards a digital agricultural system. In *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, pages 1–5, 2019.

[10] Jinling Zhao, Yan Fang, Guomin Chu, Hao Yan, Lei Hu, and Linsheng Huang. Identification of leaf-scale wheat powdery mildew (blumeria graminis f. sp. tritici) combining hyperspectral imaging and an svm classifier. *Plants*, 9(8):936, 2020.

[11] Donald Michie, David J Spiegelhalter, CC Taylor, et al. Machine learning. *Neural and Statistical Classification*, 13(1994):1–298, 1994.

[12] Marti A. Hearst. Support vector machines. *IEEE Intelligent Systems*, 13(4):18–28, July 1998.

[13] K. Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2015.

[14] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks, 2016.

[15] R. Girshick. Fast r-cnn. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015.

[16] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. *Lecture Notes in Computer Science*, page 21–37, 2016.

[17] Muhammad Hammad Saleem, Johan Potgieter, and Khalid Mahmood Arif. Plant disease detection and classification by deep learning. *Plants*, 8(11):468, 2019.

[18] Jonathan L Ramseur. Epa policies concerning integrated planning and affordability of water infrastructure. *Congressional Research Service*, 2017.

[19] Kaiyan Lin, Jie Chen, Huiping Si, and Junhui Wu. A review on computer vision technologies applied in greenhouse plant stress detection. In *Chinese Conference on Image and Graphics Technologies*, pages 192–200. Springer, 2013.

[20] David G Lowe et al. Object recognition from local scale-invariant features. In *iccv*, volume 99, pages 1150–1157, 1999.

[21] Zhuoqian Yang, Tingting Dan, and Yang Yang. Multi-temporal remote sensing image registration using deep convolutional features. *IEEE Access*, 6:38544–38555, 2018.

[22] Xiaohuan Cao, Jianhuan Yang, Li Wang, Zhong Xue, Qian Wang, and Dinggang Shen. Deep learning based inter-modality image registration supervised by intra-modality similarity. In *International Workshop on Machine Learning in Medical Imaging*, pages 55–63. Springer, 2018.

[23] Aharon Azulay and Yair Weiss. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv preprint arXiv:1805.12177*, 2018.

[24] Bob D de Vos, Floris F Berendsen, Max A Viergever, Hessam Sokooti, Marius Staring, and Ivana Išgum. A deep learning framework for unsupervised affine and deformable image registration. *Medical image analysis*, 52:128–143, 2019.

[25] Thomas Blaschke. Object based image analysis for remote sensing. *ISPRS journal of photogrammetry and remote sensing*, 65(1):2–16, 2010.

[26] Andrew P Tewkesbury, Alexis J Comber, Nicholas J Tate, Alistair Lamb, and Peter F Fisher. A critical synthesis of remotely sensed optical image change detection techniques. *Remote Sensing of Environment*, 160:1–14, 2015.

[27] Julie A Fortin, Jason T Fisher, Jeanine M Rhemtulla, and Eric S Higgs. Estimates of landscape composition from terrestrial oblique photographs suggest homogenization of rocky mountain landscapes over the last century. *Remote Sensing in Ecology and Conservation*, 5(3):224–236, 2019.

[28] Alfredo Huete, Kamel Didan, Willem van Leeuwen, Tomoaki Miura, and Ed Glenn. Modis vegetation indices. In *Land remote sensing and global environmental change*, pages 579–602. Springer, 2010.

[29] Naomi Augar and Martin Fluker. Towards understanding user perceptions of a tourist-based environmental monitoring system: An exploratory case study. *Asia Pacific Journal of Tourism Research*, 20(10):1081–1093, 2015.

[30] Oliver Sonnentag, Koen Hufkens, Cory Teshera-Sterne, Adam M Young, Mark Friedl, Bobby H Braswell, Thomas Milliman, John O'Keefe, and Andrew D Richardson. Digital repeat photography for phenological research in forest ecosystems. *Agricultural and Forest Meteorology*, 152:159–177, 2012.

[31] Graham Whitelaw, Hague Vaughan, Brian Craig, and David Atkinson. Establishing the canadian community monitoring network. *Environmental monitoring and assessment*, 88(1-3):409–418, 2003.

[32] Jules Pretty. Social capital and the collective management of resources. *Science*, 302(5652):1912–1914, 2003.

[33] Anna Lawrence. 'no personal motive?'volunteers, biodiversity, and the false dichotomies of participation. *Ethics Place and Environment*, 9(3):279–298, 2006.

[34] Nuria Castell, Mike Kobernus, Hai-Ying Liu, Philipp Schneider, William Lahoz, Arne J Berre, and Josef Noll. Mobile technologies and services for environmental monitoring: The citi-sense-mob approach. *Urban climate*, 14:370–382, 2015.

[35] Federico Montori, Luca Bedogni, and Luciano Bononi. A collaborative internet of things architecture for smart cities and environmental monitoring. *IEEE Internet of Things Journal*, 5(2):592–605, 2018.

[36] Cathy C Conrad and Krista G Hilchey. A review of citizen science and community-based environmental monitoring: issues and opportunities. *Environmental monitoring and assessment*, 176(1-4):273–291, 2011.

[37] Catherine T Conrad and Tyson Daoust. Community-based monitoring frameworks: Increasing the effectiveness of environmental stewardship. *Environmental management*, 41(3):358–366, 2008.

[38] Barbara A Israel, Amy J Schulz, Chris M Coombe, Edith A Parker, Angela G Reyes, Zachary Rowe, and Richard L Lichtenstein. Community-based participatory research. *Urban Health*, page 272, 2019.

[39] Andy Sharpe and Cathy Conrad. Community based ecological monitoring in nova scotia: challenges and opportunities. *Environmental monitoring and assessment*, 113(1-3):395–409, 2006.

[40] Robert H Webb. *Repeat photography: methods and applications in the natural sciences*. Island Press, 2010.

[41] James L Zier and William L Baker. A century of vegetation change in the san juan mountains, colorado: an analysis using repeat photography. *Forest Ecology and Management*, 228(1-3):251–262, 2006.

[42] Laura E Hendrick and Carolyn A Copenheaver. Using repeat landscape photography to assess vegetation changes in rural communities of the southern appalachian mountains in virginia, usa. *Mountain Research and Development*, 29(1):21–29, 2009.

[43] Julianne Lynch, Efrat Eilam, Martin Fluker, and Naomi Augar. Community-based environmental monitoring goes to school: translations, detours and escapes. *Environmental Education Research*, 23(5):708–721, 2017.

[44] John Pickard. Assessing vegetation change over a century using repeat photography. *Australian Journal of Botany*, 50(4):409–414, 2002.

[45] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al. Recent advances in convolutional neural networks. *Pattern Recognition*, 77:354–377, 2018.

[46] Lisa Gottesfeld Brown. A survey of image registration techniques. *ACM computing surveys (CSUR)*, 24(4):325–376, 1992.

[47] Barbara Zitová and Jan Flusser. Image registration methods: A survey. *Image and Vision Computing*, 21:977–1000, 10 2003.

[48] Manuel Guizar-Sicairos, Samuel T Thurman, and James R Fienup. Efficient subpixel image registration algorithms. *Optics letters*, 33(2):156–158, 2008.

[49] Marius Erdt, Sebastian Steger, and Georgios Sakas. Regmentation: A new view of image segmentation and registration. *Journal of Radiation Oncology Informatics*, 4(1):1–23, 2017.

[50] Ruben Fernandez-Beltran, Filiberto Pla, and Antonio Plaza. Intersensor remote sensing image registration using multispectral semantic embeddings. *IEEE Geoscience and Remote Sensing Letters*, 2019.

[51] 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.

[52] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015.

[53] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: learning local features from images. In *Advances in Neural Information Processing Systems*, pages 6234–6244, 2018.

[54] Yurun Tian, Bin Fan, and Fuchao Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 661–669, 2017.

[55] Simon Beckouche, Sébastien Leprince, Neus Sabater, and François Ayoub. Robust outliers detection in image point matching. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 180–187. IEEE, 2011.

[56] Gang Wang, Qiangqiang Zhou, and Yufei Chen. Robust non-rigid point set registration using spatially constrained gaussian fields. *IEEE Transactions on Image Processing*, 26(4):1759–1769, 2017.

[57] Philippe Thévenaz, Thierry Blu, and Michael Unser. Interpolation revisited [medical images application]. *IEEE Transactions on medical imaging*, 19(7):739–758, 2000.

[58] Andriy Myronenko and Xubo Song. Point set registration: Coherent point drift. *IEEE transactions on pattern analysis and machine intelligence*, 32(12):2262–2275, 2010.

[59] Penelope Figgis. *Australia's National Parks and Protected Areas: Future Directions: a Disscussion Paper*. Australian Committee for IUCN Incorporated, 1999.

[60] Su Zhang, Yang Yang, Kun Yang, Yi Luo, and Sim-Heng Ong. Point set registration with global-local correspondence and transformation estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2669–2677, 2017.

[61] Virginia H Dale, Rebecca A Efroymson, and Keith L Kline. The land use–climate change–energy nexus. *Landscape ecology*, 26(6):755–773, 2011.

[62] Laura De Baan, Rob Alkemade, and Thomas Koellner. Land use impacts on biodiversity in lca: a global approach. *The International Journal of Life Cycle Assessment*, 18(6):1216–1230, 2013.

[63] Mark Klett. Three methods of presenting repeat photographs. *Repeat photography: Methods and applications in the natural sciences. Washington, DC: Island*, pages 32–45, 2010.

[64] Shah F Khan, Ulrich Kamp, and Lewis A Owen. Documenting five years of landsliding after the 2005 kashmir earthquake, using repeat photography. *Geomorphology*, 197:45–55, 2013.

[65] Amaury Frankl, Jan Nyssen, Morgan De Dapper, Mitiku Haile, Paolo Billi, R Neil Munro, Jozef Deckers, and Jean Poesen. Linking long-term gully and river channel dynamics to environmental change using repeat photography (northern ethiopia). *Geomorphology*, 129(3-4):238–251, 2011.

[66] Marco Conedera, Claudio Bozzini, Cristian Scapozza, Lorenza Rè, Ueli Ryter, and Patrik Krebs. Anwendungspotenzial des wsl-monoplotting-tools im naturgefahrenmanagement. *Schweizerische Zeitschrift fur Forstwesen*, 164(7):173–180, 2013.

[67] W Roush, Jeffrey S Munroe, and Daniel B Fagre. Development of a spatial analysis method using ground-based repeat photography to detect changes in the alpine treeline ecotone, glacier national park, montana, usa. *Arctic, Antarctic, and Alpine Research*, 39(2):297–308, 2007.

[68] Rik Van Bogaert, Kristof Haneca, Jan Hoogesteger, Christer Jonasson, Morgan De Dapper, and Terry V Callaghan. A century of tree line changes in sub-arctic sweden shows local and regional variability and only a minor influence of 20th century climate warming. *Journal of Biogeography*, 38(5):907–921, 2011.

[69] B Reimers, CL Griffiths, and MT Hoffman. Repeat photography as a tool for detecting and monitoring historical changes in south african coastal habitats. *African Journal of Marine Science*, 36(3):387–398, 2014.

[70] Tommaso Julitta, Edoardo Cremonese, Mirco Migliavacca, Roberto Colombo, Marta Galvagno, Consolata Siniscalco, Micol Rossini, Francesco Fava, Sergio Cogliati, Umberto Morra di Cella, et al. Using digital camera images to analyse snowmelt and phenology of a subalpine grassland. *Agricultural and Forest Meteorology*, 198:116–125, 2014.

[71] Yunpeng Luo, Tarek S El-Madany, Gianluca Filippa, Xuanlong Ma, Bernhard Ahrens, Arnaud Carrara, Rosario Gonzalez-Cascon, Edoardo Cremonese, Marta Galvagno, Tiana W Hammer, et al. Using near-infrared-enabled digital repeat photography to track structural and physiological phenology in mediterranean tree–grass ecosystems. *Remote Sensing*, 10(8):1293, 2018.

[72] Caitlin E Moore, Tim Brown, Trevor F Keenan, Remko A Duursma, Albert IJM Van Dijk, Jason Beringer, Darius Culvenor, Bradley Evans, Alfredo Huete, Lindsay B Hutley, et al. Reviews and syntheses: Australian vegetation phenology: new insights from satellite remote sensing and digital repeat photography. *Biogeosciences*, 13(17):5085–5102, sep 2016.

[73] Keirith A Snyder, Bryce L Wehan, Gianluca Filippa, Justin L Huntington, Tamzen K Stringham, and Devon K Snyder. Extracting plant phenology metrics in a great basin watershed: Methods and considerations for quantifying phenophases in a cold desert. *Sensors*, 16(11):1948, 2016.

[74] Daniel J Manier and Richard D Laven. Changes in landscape patterns associated with the persistence of aspen (populus tremuloides michx.) on the western slope of the rocky mountains, colorado. *Forest Ecology and Management*, 167(1-3):263–284, 2002.

[75] Jeanine M Rhemtulla, Ronald J Hall, Eric S Higgs, and S Ellen Macdonald. Eighty years of change: vegetation in the montane ecoregion of jasper national park, alberta, canada. *Canadian Journal of Forest Research*, 32(11):2010–2021, 2002.

[76] Mmoto L Masubelele, Michael T Hoffman, and William J Bond. A repeat photograph analysis of long-term vegetation change in semi-arid south africa in response to land use and climate. *Journal of Vegetation Science*, 26(5):1013–1023, 2015.

[77] Patrick E Clark and Stuart P Hardegree. Quantifying vegetation change by point sampling landscape photography time series. *Rangeland ecology & management*, 58(6):588–597, 2005.

[78] Christian A Kull. Historical landscape repeat photography as a tool for land use change research. *Norsk Geografisk Tidsskrift-Norwegian Journal of Geography*, 59(4):253–268, 2005.

[79] Frederick C Hall. *Ground-based photographic monitoring*, volume 503. US Department of Agriculture, Forest Service, Pacific Northwest Research Station, 2001.

[80] Martin Flucker. Fluker post. https://www.flukerpost.com/, JANUARY 2022. (Accessed on 01/29/2022).

[81] 2017 georgia plant disease loss estimates. https://secure.caes.uga.edu/extension/publications/files/pdf/AP%20102-10_1.PDF. (Accessed on 10/20/2020).

[82] Ye Sun, Renfu Lu, Yuzhen Lu, Kang Tu, and Leiqing Pan. Detection of early decay in peaches by structured-illumination reflectance imaging. *Postharvest Biology and Technology*, 151:68–78, 2019.

[83] Mohammed Al-Shawwa and Samy S Abu-Naser. Knowledge based system for apple problems using clips. *International Journal of Academic Engineering Research (IJAER)*, 3(3):1–11, 2019.

[84] Albert Cruz, Yiannis Ampatzidis, Roberto Pierro, Alberto Materazzi, Alessandra Panattoni, Luigi De Bellis, and Andrea Luvisi. Detection of grapevine yellows symptoms in vitis vinifera l. with artificial intelligence. *Computers and Electronics in Agriculture*, 157:63–76, 2019.

[85] G. Belli, Piero Bianco, and M Conti. Grapevine yellows in italy: Past, present and future. *JOURNAL OF PLANT PATHOLOGY*, 92:303–326, 02 2010.

[86] Charmaine Butt, Jagpal Gill, David Chun, and Benson A Babu. Deep learning system to screen coronavirus disease 2019 pneumonia. *Applied Intelligence*, page 1, 2020.

[87] Monzurul Islam, Anh Dinh, Khan Wahid, and Pankaj Bhowmik. Detection of potato diseases using image segmentation and multiclass support vector machine. In *2017 IEEE 30th Canadian conference on electrical and computer engineering (CCECE)*, pages 1–4. IEEE, 2017.

[88] Shiv Ram Dubey and Anand Singh Jalal. Detection and classification of apple fruit diseases using complete local binary patterns. In *Proceedings of the 3rd*

international conference on computer and communication technology, pages 346–351, 2012.

[89] Srdjan Sladojevic, Marko Arsenovic, Andras Anderla, Dubravko Culibrk, and Darko Stefanovic. Deep neural networks based recognition of plant diseases by leaf image classification. *Computational intelligence and neuroscience*, 2016, 2016.

[90] Miaomiao Ji, Lei Zhang, and Qiufeng Wu. Automatic grape leaf diseases identification via unitedmodel based on multiple convolutional neural networks. *Information Processing in Agriculture*, 2019.

[91] Xiaoyue Xie, Yuan Ma, Bin Liu, Jinrong He, Shuqin Li, and Hongyan Wang. A deep-learning-based real-time detector for grape leaf diseases using improved convolutional neural networks. *Frontiers in Plant Science*, 11, 2020.

[92] Shanwen Zhang, Wenzhun Huang, and Chuanlei Zhang. Three-channel convolutional neural networks for vegetable leaf disease recognition. *Cognitive Systems Research*, 53:31–41, 2019.

[93] 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.

[94] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *CoRR*, abs/1511.08458, 2015.

[95] Hui Xu. PlantVillage Disease Classification Challenge - Color Images. March 2018. https://gitlab.com/huix/leaf-disease-plant-village.

[96] Michael A Nielsen. *Neural networks and deep learning*, volume 2018. Determination press San Francisco, CA, 2015.

[97] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[98] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2018.

[99] Forrest N. Iandola, Matthew W. Moskewicz, Khalid Ashraf, Song Han, William J. Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <1mb model size. *CoRR*, abs/1602.07360, 2016.

[100] Joseph Redmon. Darknet: Open source neural networks in c. http://pjreddie.com/darknet/, 2013–2016.

[101] Sharada P Mohanty, David P Hughes, and Marcel Salathé. Using deep learning for image-based plant disease detection. *Frontiers in plant science*, 7:1419, 2016.

[102] Ameet V Joshi. Amazon's machine learning toolkit: Sagemaker. In *Machine Learning and Artificial Intelligence*, pages 233–243. Springer, 2020.

[103] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60, 2019.

[104] Ruitao Feng, Qingyun Du, Xinghua Li, and Huanfeng Shen. Robust registration for remote sensing images by combining and localizing feature-and area-based methods. *ISPRS Journal of Photogrammetry and Remote Sensing*, 151:15–26, 2019.

[105] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283, 2016.

[106] Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, 2019.

[107] Ç F Özgenel and A Gönenç Sorguç. Performance comparison of pretrained convolutional neural networks on crack detection in buildings. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, volume 35, pages 1–8. IAARC Publications, 2018.

[108] Aws deeplens – deep learning enabled video camera for developers - aws. https://aws.amazon.com/deeplens/. (Accessed on 07/19/2020).

[109] Robert Kleinberg, Yuanzhi Li, and Yang Yuan. An alternative view: When does sgd escape local minima? *arXiv preprint arXiv:1802.06175*, 2018.

[110] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[111] Twan Van Laarhoven. L2 regularization versus batch and weight normalization. *arXiv preprint arXiv:1706.05350*, 2017.

[112] SanaUllah Khan, Naveed Islam, Zahoor Jan, Ikram Ud Din, and Joel JP C Rodrigues. A novel deep learning based framework for the detection and classification of breast cancer using transfer learning. *Pattern Recognition Letters*, 125:1–6, 2019.

[113] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In *International conference on artificial neural networks*, pages 270–279. Springer, 2018.

[114] Joanna Jaworek-Korjakowska, Pawel Kleczek, and Marek Gorgon. Melanoma thickness prediction based on convolutional neural network with vgg-19 model transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[115] Maria Galkin, Kashmala Rehman, Benjamin Schornstein, Warren Sunada-Wong, and Harvey Wang. A hygiene monitoring system. 2019.

[116] Create and publish an aws deeplens inference lambda function - aws deeplens. https://docs.aws.amazon.com/deeplens/latest/dg/deeplens-inference-lambda-create.html. (Accessed on 08/10/2020).

[117] Invoke aws lambda functions - amazon connect. https://docs.aws.amazon.com/connect/latest/adminguide/connect-lambda-functions.html. (Accessed on 08/10/2020).

[118] Tianrui Liu, Jun-Jie Huang, Tianhong Dai, Guangyu Ren, and Tania Stathaki. Gated multi-layer convolutional feature extraction network for robust pedestrian detection. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3867–3871. IEEE, 2020.

[119] Vedat Tümen, Ömer Faruk Söylemez, and Burhan Ergen. Facial emotion recognition on a dataset using convolutional neural network. In *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, pages 1–5. IEEE, 2017.

[120] Guotian Xie, Kuiyuan Yang, and Jianhuang Lai. Filter-in-filter: Low cost cnn improvement by sub-filter parameter sharing. *Pattern Recognition*, 91:391–403, 2019.

[121] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[122] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[123] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

[124] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[125] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[126] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[127] Steve Branson, Jan Dirk Wegner, David Hall, Nico Lang, Konrad Schindler, and Pietro Perona. From google maps to a fine-grained catalog of street trees. *ISPRS Journal of Photogrammetry and Remote Sensing*, 135:13–30, 2018.

[128] Jennifer A Salmond, Marc Tadaki, Sotiris Vardoulakis, Katherine Arbuthnott, Andrew Coutts, Matthias Demuzere, Kim N Dirks, Clare Heaviside, Shanon Lim, Helen Macintyre, et al. Health and climate related ecosystem services provided by street trees in the urban environment. *Environmental Health*, 15(1):95–111, 2016.

[129] Pauline Ladiges. The story of our eucalypts - curious. https://www.science.org.au/curious/earth-environment/story-our-eucalypts. (Accessed on 12/23/2020).

[130] Eucalypt forest - department of agriculture. https://www.agriculture.gov.au/abares/forestsaustralia/profiles/eucalypt-2016. (Accessed on 12/23/2020).

[131] Paul Berrang, David F Karnosky, and Brian J Stanton. Environmental factors affecting tree health in new york city. *Journal of arboriculture*, 11(6):185–189, 1985.

[132] Bert M Cregg and Mary Ellen Dix. Tree moisture stress and insect damage in urban areas in relation to heat island effects. *Journal of Arboriculture*, 27(1):8–17, 2001.

[133] Matthew F Winn, Sang-Mook Lee, and Philip A Araman. Urban tree crown health assessment system: a tool for communities and citizen foresters. *Proceedings, Emerging Issues Along Urban-Rural Interfaces II: Linking Land-Use Science and Society. 180-183.*, 2007.

[134] Izabela Czerniawska-Kusza, Grzegorz Kusza, and Mariusz Dużyński. Effect of deicing salts on urban soils and health status of roadside trees in the opole

region. *Environmental Toxicology: An International Journal*, 19(4):296–301, 2004.

[135] Susan D Day and Nina L Bassuk. A review of the effects of soil compaction and amelioration treatments on landscape trees. *Journal of arboriculture*, 20(1):9–17, 1994.

[136] T Doody, I Overton, et al. Environmental management of riparian tree health in the murray-darling basin, australia. *River Basin Management V*, 124:197, 2009.

[137] Nathalie Butt, Laura J Pollock, and Clive A McAlpine. Eucalypts face increasing climate stress. *Ecology and Evolution*, 3(15):5011–5022, 2013.

[138] Davide Chicco. Siamese neural networks: An overview. *Artificial Neural Networks*, pages 73–94, 2020.

[139] About us — wyndham city. https://www.wyndham.vic.gov.au/about-us. (Accessed on 12/22/2020).

[140] Google. Google street view imagery - google search, December 2020. (Accessed on 12/22/2020).

[141] Dragomir Anguelov, Carole Dulong, Daniel Filip, Christian Frueh, Stéphane Lafon, Richard Lyon, Abhijit Ogale, Luc Vincent, and Josh Weaver. Google street view: Capturing the world at street level. *Computer*, 43(6):32–38, 2010.

[142] Xiaojiang Li, Chuanrong Zhang, Weidong Li, Yulia A Kuzovkina, and Daniel Weiner. Who lives in greener neighborhoods? the distribution of street greenery and its association with residents' socioeconomic conditions in hartford, connecticut, usa. *Urban Forestry & Urban Greening*, 14(4):751–759, 2015.

[143] Xiaojiang Li, Chuanrong Zhang, and Weidong Li. Building block level urban land-use information retrieval based on google street view images. *GIScience & Remote Sensing*, 54(6):819–835, 2017.

[144] Weixing Zhang, Weidong Li, Chuanrong Zhang, Dean M Hanink, Xiaojiang Li, and Wenjie Wang. Parcel-based urban land use classification in megacity using airborne lidar, high resolution orthoimagery, and google street view. *Computers, Environment and Urban Systems*, 64:215–228, 2017.

[145] Xiaojiang Li, Carlo Ratti, and Ian Seiferling. Quantifying the shade provision of street trees in urban landscape: A case study in boston, usa, using google street view. *Landscape and Urban Planning*, 169:81–91, 2018.

[146] Asim Khan, Umair Nawaz, Anwaar Ulhaq, and Randall W. Robinson. Real-time plant health assessment via implementing cloud-based scalable transfer learning on aws deeplens. *PLOS ONE*, 15(12):1–23, 12 2020.

[147] Asim Khan, Anwaar Ulhaq, Randall Robinson, and Mobeen Ur Rehman. Detection of vegetation in environmental repeat photography: a new algorithmic approach in data science. In *Statistics for Data Science and Policy Analysis*, pages 145–157. Springer, 2020.

[148] Paulius Tumas, Adam Nowosielski, and Arturas Serackis. Pedestrian detection in severe weather conditions. *IEEE Access*, 8:62775–62784, 2020.

[149] Guanjun Guo, Hanzi Wang, Yan Yan, Jin Zheng, and Bo Li. A fast face detection method via convolutional neural network. *Neurocomputing*, 395:128–137, 2020.

[150] Jialian Wu, Liangchen Song, Tiancai Wang, Qian Zhang, and Junsong Yuan. Forest r-cnn: Large-vocabulary long-tailed object detection and instance segmentation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1570–1578, 2020.

[151] Md Mohaimenul Islam, Hsuan-Chia Yang, Tahmina Nasrin Poly, Wen-Shan Jian, and Yu-Chuan Jack Li. Deep learning algorithms for detection of diabetic retinopathy in retinal fundus photographs: A systematic review and meta-analysis. *Computer Methods and Programs in Biomedicine*, 191:105320, 2020.

[152] Zeng Degui and Yu Fei. Research on the application of big data automatic search and data mining based on remote sensing technology. In *2020 3rd International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pages 122–127. IEEE, 2020.

[153] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018, 2018.

[154] Milto Miltiadou, Neil DF Campbell, Susana Gonzalez Aracil, Tony Brown, and Michael G Grant. Detection of dead standing eucalyptus camaldulensis without tree delineation for managing biodiversity in native australian forest. *International journal of applied earth observation and geoinformation*, 67:135–147, 2018.

[155] I. Shendryk, M. Broich, M. G. Tulbure, and S. V. Alexandrov. Bottom-up delineation of individual trees from full-waveform airborne laser scans in a structurally complex eucalypt forest. *Remote Sensing of Environment*, 173:69–83, 2016.

[156] Agnieszka Kamińska, Maciej Lisiewicz, Krzysztof Stereńczak, Bartłomiej Kraszewski, and Rafał Sadkowski. Species-related single dead tree detection using multi-temporal als data and cir imagery. *Remote Sensing of Environment*, 219:31–43, 2018.

[157] Martin Weinmann, Michael Weinmann, Clément Mallet, and Mathieu Brédif. A classification-segmentation framework for the detection of individual trees in dense mms point cloud data acquired in urban areas. *Remote sensing*, 9(3):277, 2017.

[158] S Briechle, Peter Krzystek, and G Vosselman. Classification of tree species and standing dead trees by fusing uav-based lidar data and multispectral imagery in the 3d deep neural network pointnet++. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 5(2), 2020.

[159] YT Mollaei, A Karamshahi, and SY Erfanifard. Detection of the dry trees result of oak borer beetle attack using worldview-2 satellite and uav imagery an object-oriented approach. *J Remote Sensing & GIS*, 7(232):2, 2018.

[160] W Yao, P Krzystek, and M Heurich. Identifying standing dead trees in forest areas based on 3d single tree detection from full waveform lidar data. *ISPRS Annals of the Protogrammetry, Remote Sensing and Spatial Information Sciences*, 1:7, 2012.

[161] Xiaoling Deng, Yubin Lan, Tiansheng Hong, and Junxi Chen. Citrus greening detection using visible spectrum imaging and c-svc. *Computers and Electronics in Agriculture*, 130:177–183, 2016.

[162] Yubin Lan, Zixiao Huang, Xiaoling Deng, Zihao Zhu, Huasheng Huang, Zheng Zheng, Bizhen Lian, Guoliang Zeng, and Zejing Tong. Comparison of machine learning methods for citrus greening detection on uav multispectral images. *Computers and Electronics in Agriculture*, 171:105234, 2020.

[163] Iurii Shendryk, Mark Broich, Mirela G Tulbure, Andrew McGrath, David Keith, and Sergey V Alexandrov. Mapping individual tree health using full-waveform airborne laser scans and imaging spectroscopy: A case study for a floodplain eucalypt forest. *Remote Sensing of Environment*, 187:202–217, 2016.

[164] Ran Meng, Philip E Dennison, Feng Zhao, Iurii Shendryk, Amanda Rickert, Ryan P Hanavan, Bruce D Cook, and Shawn P Serbin. Mapping canopy defoliation by herbivorous insects at the individual tree level using bi-temporal airborne imaging spectroscopy and lidar measurements. *Remote Sensing of Environment*, 215:170–183, 2018.

[165] Manuel López-López, Rocío Calderón, Victoria González-Dugo, Pablo J Zarco-Tejada, and Elías Fereres. Early detection and quantification of almond red leaf blotch using high-resolution hyperspectral and thermal imagery. *Remote Sensing*, 8(4):276, 2016.

[166] Chloe Barnes, Heiko Balzter, Kirsten Barrett, James Eddy, Sam Milner, and Juan C Suárez. Airborne laser scanning and tree crown fragmentation metrics

for the assessment of phytophthora ramorum infected larch forest stands. *Forest Ecology and Management*, 404:294–305, 2017.

[167] Fabian Ewald Fassnacht, Hooman Latifi, Aniruddha Ghosh, Pawan Kumar Joshi, and Barbara Koch. Assessing the potential of hyperspectral imagery to map bark beetle-induced tree mortality. *Remote Sensing of Environment*, 140:533–548, 2014.

[168] Dengkai Chi, Jeroen Degerickx, Kang Yu, and Ben Somers. Urban tree health classification across tree species by combining airborne laser scanning and imaging spectroscopy. *Remote Sensing*, 12(15):2435, 2020.

[169] Roope Näsi, Eija Honkavaara, Päivi Lyytikäinen-Saarenmaa, Minna Blomqvist, Paula Litkey, Teemu Hakala, Niko Viljanen, Tuula Kantola, Topi Tanhuanpää, and Markus Holopainen. Using uav-based photogrammetry and hyperspectral imaging for mapping bark beetle damage at tree-level. *Remote Sensing*, 7(11):15467–15493, 2015.

[170] Roope Näsi, Eija Honkavaara, Minna Blomqvist, Päivi Lyytikäinen-Saarenmaa, Teemu Hakala, Niko Viljanen, Tuula Kantola, and Markus Holopainen. Remote sensing of bark beetle damage in urban forests at individual tree level using a novel hyperspectral camera from uav and aircraft. *Urban Forestry & Urban Greening*, 30:72–83, 2018.

[171] Jeroen Degerickx, Dar A Roberts, Joe P McFadden, Martin Hermy, and Ben Somers. Urban tree health assessment using airborne hyperspectral and lidar imagery. *International journal of applied earth observation and geoinformation*, 73:26–38, 2018.

[172] Qingfu Xiao and E Gregory McPherson. Tree health mapping with multispectral remote sensing data at uc davis, california. *Urban Ecosystems*, 8(3-4):349–361, 2005.

[173] Grigorijs Goldbergs, Stefan W Maier, Shaun R Levick, and Andrew Edwards. Efficiency of individual tree detection approaches based on light-weight and low-cost uas imagery in australian savannas. *Remote Sensing*, 10(2):161, 2018.

[174] Fabian Ewald Fassnacht, Daniel Mangold, Jannika Schäfer, Markus Immitzer, Teja Kattenborn, Barbara Koch, and Hooman Latifi. Estimating stand density, biomass and tree species from very high resolution stereo-imagery – towards an all-in-one sensor for forestry applications? *Forestry: An International Journal of Forest Research*, 90(5):613–631, 03 2017.

[175] Weijia Li, Conghui He, Haohuan Fu, Juepeng Zheng, Runmin Dong, Maocai Xia, Le Yu, and Wayne Luk. A real-time tree crown detection approach for large-scale remote sensing images on fpgas. *Remote Sensing*, 11(9):1025, 2019.

[176] X. Zhao, S. Zhou, L. Lei, and Z. Deng. Siamese network for object tracking in aerial video. In *2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)*, pages 519–523, 2018.

[177] Manish Chandra, Pratik Shalikrao Patil, Saurav Roy, and Shukrayani Sanjay Redkar. Classification of various plant diseases using deep siamese network.

[178] Victoria Ruiz, Ismael Linares, Angel Sanchez, and Jose F Velez. Off-line handwritten signature verification using compositional synthetic generation of signatures and siamese neural networks. *Neurocomputing*, 374:30–41, 2020.

[179] Dattaraj Rao, Shruti Mittal, and S. Ritika. Siamese neural networks for one-shot detection of railway track switches. 12 2017.

[180] Mohammad Shorfuzzaman and M Shamim Hossain. Metacovid: A siamese neural network framework with contrastive loss for n-shot diagnosis of covid-19 patients. *Pattern Recognition*, page 107700, 2020.

[181] Jane Bromley, James W. Bentz, Leon Bottou, Isabelle Guyon, Yann Lecun, Cliff Moore, Eduard Sackinger, and Roopak Shah. Signature verification using a "siamese" time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 07(04):669–688, 1993.

[182] Bin Wang and Dian Wang. Plant leaves classification: A few-shot learning method based on siamese network. *IEEE Access*, 7:1–1, 10 2019.

[183] Wyndham city suburbs — wyndham city advocacy. https://wyndham-digital.iconagency.com.au/node/10. (Accessed on 12/22/2020).

[184] What is an application programming interface (api)? — ibm. https://www.ibm.com/cloud/learn/api. (Accessed on 12/22/2020).

[185] Tim Berners-Lee, Larry Masinter, Mark McCahill, et al. Uniform resource locators (url). 1994.

[186] Github - robolyst/streetview: Python module for retrieving current and historical photos from google street view. https://github.com/robolyst/streetview. (Accessed on 12/22/2020).

[187] Github-tzutalin/labelimg:labelimg is a graphical image annotation tool and label object bounding boxes in images. https://github.com/tzutalin/labelImg. (Accessed on 12/22/2020).

[188] A friendly introduction to siamese networks — by sean benhur j — towards data science. https://towardsdatascience.com/a-friendly-introduction-to-siamese-networks-85ab17522942. (Accessed on 12/22/2020).

[189] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep cnn denoiser prior for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3929–3938, 2017.

[190] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? In *Advances in neural information processing systems*, pages 2483–2493, 2018.

[191] Yuanzhi Li and Yang Yuan. Convergence analysis of two-layer neural networks with relu activation. In *Advances in neural information processing systems*, pages 597–607, 2017.

[192] Rob A Dunne and Norm A Campbell. On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function. In *Proc. 8th Aust. Conf. on the Neural Networks, Melbourne*, volume 181, page 185. Citeseer, 1997.

[193] Ahmed Samy Nassar, Sébastien Lefèvre, and Jan Dirk Wegner. Multi-view instance matching with learned geometric soft-constraints. *ISPRS International Journal of Geo-Information*, 9(11):687, 2020.

[194] Contrastive loss explained. contrastive loss has been used recently... — by brian williams — towards data science. https://towardsdatascience.com/contrastive-loss-explaned-159f2d4a87ec. (Accessed on 12/22/2020).

[195] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005.

[196] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[197] Azimuth - wikipedia. https://en.wikipedia.org/wiki/Azimuth. (Accessed on 05/01/2021).

[198] M. Gavish and A.J. Weiss. Performance analysis of bearing-only target location algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 28(3):817–828, 1992.

[199] Hongjian Zhang, Zhongliang Jing, and Shiqiang Hu. Localization of multiple emitters based on the sequential phd filter. *Signal Process.*, 90(1):34–43, jan 2010.

[200] Jesse D Reed, Claudio RCM da Silva, and R Michael Buehrer. Multiple-source localization using line-of-bearing measurements: Approaches to the data association problem. In *MILCOM 2008-2008 IEEE Military Communications Conference*, pages 1–7. IEEE, 2008.

[201] Michael T Grabbe, Brandon M Hamschin, and Andrew P Douglas. A measurement correlation algorithm for line-of-bearing geo-location. In *2013 IEEE Aerospace Conference*, pages 1–8. IEEE, 2013.

[202] Kun Tan, Hong Chen, and Xiao-xia Cai. Research into the algorithm of false points elimination in three-station cross location. *Shipboard Electron. Countermeas*, 32:79–81, 2009.

[203] Jesse Reed. *Approaches to multiple source localization and signal classification*. PhD thesis, Virginia Tech, 2009.

[204] Spatial aggregation—arcgis insights — documentation. https://doc.arcgis.com/en/insights/latest/analyze/spatial-aggregation.htm. (Accessed on 05/01/2021).

[205] Aggregation - gis wiki — the gis encyclopedia. http://wiki.gis.com/wiki/index.php/Aggregation#cite_note-1. (Accessed on 05/01/2021).

[206] Nikhil Ketkar. Introduction to keras. In *Deep learning with Python*, pages 97–111. Springer, 2017.

[207] Nikhil Ketkar. Introduction to tensorflow. In *Deep Learning with Python*, pages 159–194. Springer, 2017.

[208] Cesare Alippi, Simone Disabato, and Manuel Roveri. Moving convolutional neural networks to embedded systems: the alexnet and vgg-16 case. In *2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 212–223. IEEE, 2018.

[209] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018.

[210] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. *arXiv preprint arXiv:1602.07261*, 2016.

[211] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn:

Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[212] James Dellinger. Weight initialization in neural networks: A journey from the basics to kaiming — by james dellinger — towards data science. https://towardsdatascience.com/weight-initialization-in-neural-networks-a-journey-from-the-basics-to-kaiming-954fb9b47c79, April 2019. (Accessed on 12/22/2020).

[213] 4 signs that your tree is dying — perth arbor services. https://pertharborservices.com.au/4-signs-your-tree-is-dying-what-to-do/. (Accessed on 12/22/2020).

[214] Common eucalyptus tree problems: Eucalyptus tree diseases. https://www.gardeningknowhow.com/ornamental/trees/eucalyptus/eucalyptus-tree-problems.htm. (Accessed on 12/22/2020).

[215] How often does google maps update satellite images? — techwalla. https://www.techwalla.com/articles/how-often-does-google-maps-update-satellite-images. (Accessed on 05/15/2021).

[216] 9 things to know about google's maps data: Beyond the map — google cloud blog. https://cloud.google.com/blog/products/maps-platform/9-things-know-about-googles-maps-data-beyond-map. (Accessed on 05/18/2021).

[217] Xiao-Peng Song, Matthew C Hansen, Stephen V Stehman, Peter V Potapov, Alexandra Tyukavina, Eric F Vermote, and John R Townshend. Global land change from 1982 to 2016. *Nature*, 560(7720):639–643, 2018.

[218] Matt Edgeworth, Erle C Ellis, Philip Gibbard, Cath Neal, and Michael Ellis. The chronostratigraphic method is unsuitable for determining the start of the anthropocene. *Progress in Physical Geography: Earth and Environment*, 43(3):334–344, 2019.

[219] Thais Michele Rosan, Luiz Aragão, Imma Oliveras, Oliver Phillips, Yadvinder Malhi, Manuel Gloor, and Fabien Wagner. Extensive twenty-first century woody encroachment in south america's savanna. *Geophysical Research Letters*, 06 2019.

[220] Kathleen L Wolf. Business district streetscapes, trees, and consumer response. *Journal of Forestry*, 103(8):396–400, 2005.

[221] Donald Appleyard. Urban trees, urban forests: What do they mean. In *Proceedings of the national urban forestry conference*, pages 138–155, 1979.

[222] David J Nowak, Robert Hoehn, and Daniel E Crane. Oxygen production by urban trees in the united states. *Arboriculture & Urban Forestry. 33 (3): 220-226.*, 33(3), 2007.

[223] Xiao-Ling Chen, Hong-Mei Zhao, Ping-Xiang Li, and Zhi-Yong Yin. Remote sensing image-based analysis of the relationship between urban heat island and land use/cover changes. *Remote sensing of environment*, 104(2):133–146, 2006.

[224] Akio Onishi, Xin Cao, Takanori Ito, Feng Shi, and Hidefumi Imura. Evaluating the potential for urban heat-island mitigation by greening parking lots. *Urban forestry & Urban greening*, 9(4):323–332, 2010.

[225] Morelia Camacho-Cervantes, Jorge E Schondube, Alicia Castillo, and Ian MacGregor-Fors. How do people perceive urban trees? assessing likes and dislikes in relation to the trees of a city. *Urban ecosystems*, 17(3):761–773, 2014.

[226] Shivanand Balram and Suzana Dragićević. Attitudes toward urban green spaces: integrating questionnaire survey and collaborative gis techniques to improve attitude measurements. *Landscape and urban planning*, 71(2-4):147–162, 2005.

[227] Lin Gao, Xiaofei Wang, Brian Alan Johnson, Qingjiu Tian, Yu Wang, Jochem Verrelst, Xihan Mu, and Xingfa Gu. Remote sensing algorithms for estimation of fractional vegetation cover using pure vegetation index values: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 159:364–377, 2020.

[228] Jun Yang, Linsen Zhao, Joe Mcbride, and Peng Gong. Can you see green? assessing the visibility of urban forests in cities. *Landscape and Urban Planning*, 91(2):97–104, 2009.

[229] Xiaojiang Li, Chuanrong Zhang, Weidong Li, and Yulia A Kuzovkina. Environmental inequities in terms of different types of urban greenery in hartford, connecticut. *Urban Forestry & Urban Greening*, 18:163–172, 2016.

[230] Yonglin Zhang and Rencai Dong. Impacts of street-visible greenery on housing prices: Evidence from a hedonic price model and a massive street view image dataset in beijing. *ISPRS International Journal of Geo-Information*, 7(3):104, 2018.

[231] Ying Long and Liu Liu. How green are the streets? an analysis for central areas of chinese cities using tencent street view. *PloS one*, 12(2):e0171110, 2017.

[232] Liang Cheng, Sensen Chu, Wenwen Zong, Shuyi Li, Jie Wu, and Manchun Li. Use of tencent street view imagery for visual perception of streets. *ISPRS International Journal of Geo-Information*, 6(9):265, 2017.

[233] Dave Kendal, Cindy E Hauser, Georgia E Garrard, Sacha Jellinek, Katherine M Giljohann, and Joslin L Moore. Quantifying plant colour and colour difference as perceived by humans using digital images. *PLoS one*, 8(8):e72296, 2013.

[234] Javier Lopatin, Klara Dolos, Teja Kattenborn, and Fabian E Fassnacht. How canopy shadow affects invasive plant species classification in high spatial resolution remote sensing. *Remote Sensing in Ecology and Conservation*, 5(4):302–317, 2019.

[235] Felix Schiefer, Teja Kattenborn, Annett Frick, Julian Frey, Peter Schall, Barbara Koch, and Sebastian Schmidtlein. Mapping forest tree species in high resolution uav-based rgb-imagery by means of convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 170:205–215, 2020.

[236] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[237] Nikita Dvornik, Konstantin Shmelkov, Julien Mairal, and Cordelia Schmid. Blitznet: A real-time deep network for scene understanding. In *Proceedings of the IEEE international conference on computer vision*, pages 4154–4162, 2017.

[238] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. Fully convolutional instance-aware semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2359–2367, 2017.

[239] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016.

[240] Wyndham City Council. Street tree planting — wyndham city. https://www.wyndham.vic.gov.au/treeplanting. (Accessed on 02/23/2020).

[241] Street view static api overview — google developers. https://developers.google.com/maps/documentation/streetview/overview. (Accessed on 07/02/2021).

[242] Victor JD Tsai and Chun-Ting Chang. Three-dimensional positioning from google street view panoramas. *IET Image Processing*, 7(3):229–239, 2013.

[243] Shijie Hao, Yuan Zhou, and Yanrong Guo. A brief survey on semantic segmentation with deep learning. *Neurocomputing*, 406:302–321, 2020.

[244] Jonas Uhrig, Marius Cordts, Uwe Franke, and Thomas Brox. Pixel-level encoding and depth layering for instance-level semantic labeling. In *German Conference on Pattern Recognition*, pages 14–25. Springer, 2016.

[245] Yanming Guo, Yu Liu, Theodoros Georgiou, and Michael S Lew. A review of semantic segmentation using deep neural networks. *International journal of multimedia information retrieval*, 7(2):87–93, 2018.

[246] Xiaolong Liu, Zhidong Deng, and Yuhan Yang. Recent progress in semantic image segmentation. *Artificial Intelligence Review*, 52(2):1089–1106, 2019.

[247] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[248] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[249] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[250] Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, and Jose Garcia-Rodriguez. A review on deep learning techniques applied to semantic segmentation. *arXiv preprint arXiv:1704.06857*, 2017.

[251] Mohamed H Almeer. Vegetation extraction from free google earth images of deserts using a robust bpnn approach in hsv space. *International Journal of Advanced Research in Computer and Communication Engineering*, 1(3):134–140, 2012.

[252] Thomas Blaschke, Stefan Lang, Eric Lorup, Josef Strobl, and Peter Zeil. Object-oriented image processing in an integrated gis/remote sensing environment and perspectives for environmental applications. *Environmental information for planning, politics and the public*, 2:555–570, 2000.

[253] Zeiss. Apeer annotate. https://www.apeer.com/annotate, February 2020. (Accessed on 08/15/2021).

[254] Lieve Hamers et al. Similarity measures in scientometric research: The jaccard index versus salton's cosine formula. *Information Processing and Management*, 25(3):315–18, 1989.

[255] Valerio Amici, Simona Maccherini, Elisa Santi, Dino Torri, Francesca Vergari, and Maurizio Del Monte. Long-term patterns of change in a vanishing cultural landscape: A gis-based assessment. *Ecological Informatics*, 37:38–51, 2017.

[256] Teodoro Lasanta, Estela Nadal-Romero, and José Arnáez. Managing abandoned farmland to control the impact of re-vegetation on the environment. the state of the art in europe. *Environmental Science & Policy*, 52:99–109, 2015.

[257] Kshama Gupta, Pramod Kumar, Subhan Khan Pathan, and Kamesh Prasad Sharma. Urban neighborhood green index–a measure of green spaces in urban areas. *Landscape and urban planning*, 105(3):325–335, 2012.

[258] Sh Faryadi and Sh Taheri. Interconnections of urban green spaces and environmental quality of tehran. 2009.

[259] Richard F. Rohde, M. Timm Hoffman, Ian Durbach, Zander Venter, and Sam Jack. Vegetation and climate change in the pro-namib and namib desert based on repeat photography: Insights into climate trends. *Journal of Arid Environments*, 165:119–131, 2019.

[260] Motomu Toda and Andrew D. Richardson. Estimation of plant area index and phenological transition dates from digital repeat photography and radiometric approaches in a hardwood forest in the northeastern united states. *Agricultural and Forest Meteorology*, 249:457–466, 2018.

[261] Martin Längkvist, Andrey Kiselev, Marjan Alirezaie, and Amy Loutfi. Classification and segmentation of satellite orthoimagery using convolutional neural networks. *Remote Sensing*, 8(4):329, 2016.

[262] Ce Zhang, Isabel Sargent, Xin Pan, Huapeng Li, Andy Gardiner, Jonathon Hare, and Peter M Atkinson. An object-based convolutional neural network (ocnn) for urban land use classification. *Remote sensing of environment*, 216:57–70, 2018.

[263] Esmael Hamuda, Brian Mc Ginley, Martin Glavin, and Edward Jones. Improved image processing-based crop detection using kalman filtering and the hungarian algorithm. *Computers and electronics in agriculture*, 148:37–44, 2018.

[264] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler. Efficient object localization using convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 648–656, 2015.

[265] Boukaye Boubacar Traore, Bernard Kamsu-Foguem, and Fana Tangara. Deep convolution neural network for image recognition. *Ecological Informatics*, 48:257–268, 2018.

[266] Guang Xu, Xuan Zhu, Dongjie Fu, Jinwei Dong, and Xiangming Xiao. Automatic land cover classification of geo-tagged field photos by deep learning. *Environmental Modelling & Software*, 91:127–134, 2017.

[267] Jihen Amara, Bassem Bouaziz, and Alsayed Algergawy. A deep learning-based approach for banana leaf diseases classification. *Datenbanksysteme für Business, Technologie und Web (BTW 2017)-Workshopband*, 2017.

[268] Heng Lu, Xiao Fu, Chao Liu, Long-guo Li, Yu-xin He, and Nai-wen Li. Cultivated land information extraction in uav imagery based on deep convolutional neural network and transfer learning. *Journal of Mountain Science*, 14(4):731–741, 2017.

[269] Dongmei Han, Qigang Liu, and Weiguo Fan. A new image classification method using cnn transfer learning and web data augmentation. *Expert Systems with Applications*, 95:43–56, 2018.

[270] Asim Khan, Warda Asim, Anwaar Ulhaq, Bilal Ghazi, and Randall W. Robinson. Health assessment of eucalyptus trees using siamese network from google street and ground truth images. *Remote Sensing*, 13(11), 2021.

[271] Asim Khan, Warda Asim, Anwaar Ulhaq, and Randall W Robinson. A multi-view semantic vegetation index for robust estimation of urban vegetation cover. *Remote Sensing*, 14(1):228, 2022.

[272] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.

[273] Iva Harbaš, Pavle Prentašić, and Marko Subašić. Detection of roadside vegetation using fully convolutional networks. *Image and Vision Computing*, 74:1–9, 2018.

[274] Calvin Hung, Zhe Xu, and Salah Sukkarieh. Feature learning based approach for weed classification using high resolution aerial images from a digital camera mounted on a uav. *Remote Sensing*, 6(12):12037–12054, 2014.

[275] Owen Bawden, Jason Kulk, Ray Russell, Chris McCool, Andrew English, Feras Dayoub, Chris Lehnert, and Tristan Perez. Robot for weed species plant-specific management. *Journal of Field Robotics*, 34(6):1179–1199, 2017.

[276] Mark Klett. Repeat photography in landscape research. *The Sage handbook of visual research methods*, pages 114–131, 2011.

[277] Oskar Puschmann and Wenche E Dramstad. Documenting landscape change through fixed angle photography. *Agricultural impacts on landscapes*, page 258, 2002.

[278] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

[279] Ligang Zhang, Brijesh Verma, and David Stockwell. Spatial contextual superpixel model for natural roadside vegetation classification. *Pattern Recognition*, 60:444–457, 2016.

[280] Benjamin Klein, Lior Wolf, and Yehuda Afek. A dynamic convolutional layer for short range weather prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4840–4848, 2015.

[281] Bing Shuai, Zhen Zuo, Bing Wang, and Gang Wang. Dag-recurrent neural networks for scene labeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3620–3629, 2016.

[282] Søren Skovsen, Mads Dyrmann, Anders Krogh Mortensen, Kim Arild Steen, Ole Green, Jørgen Eriksen, René Gislum, Rasmus Nyholm Jørgensen, and Henrik Karstoft. Estimation of the botanical composition of clover-grass leys from rgb images using data simulation and fully convolutional neural networks. *Sensors*, 17(12):2930, 2017.

[283] The Bureau of Meteorology: Australia. Previous droughts. http://www.bom.gov.au/climate/drought/knowledge-centre/previous-droughts.shtml, April 2020. (Accessed on 02/08/2020).