

UNIVERSITY OF TWENTE

MASTER THESIS

Machine Learning for Ground Cover and Hot Target Analysis in RGB and Satellite Imagery

Author:
G.L.J. PINGEN¹

Graduation committee:
R.B.N. ALY¹
M.H.T. DE BOER^{2,4}
R. ZURITA-MILLA^{1,3}
G. ENGLEBIENNE¹

¹University of
Twente

²Netherlands
Organization for
Applied
Scientific
Research (TNO)

³Faculty of
Geo-Information
Science and
Earth
Observation
(ITC)

⁴Radboud
University
Nijmegen

*Thesis submitted in fulfillment of the requirements
for the degree of Master of Science*

Faculty of Electrical Engineering, Mathematics and Computer
Science

October 26, 2016

UNIVERSITY
OF TWENTE.

TNO



UNIVERSITY OF TWENTE

Abstract

Master of Science

Machine Learning for Ground Cover and Hot Target Analysis in RGB and Satellite Imagery

by G.L.J. PINGEN

Ground cover, commonly used in agronomic studies, can be used in smart agriculture applications to perform effective automatic treatment of crops, for instance by finding an optimal distribution of water or pesticide administration. This is especially helpful for farms located in third world countries, that have little access to high-tech machinery and a lot to gain in terms of crop yield.

Existing methods of ground cover analysis rely on converting the original RGB features to a colour index. However, since these methods only use the information of a single pixel, they fail to pick up overarching features such as the curvature of a leaf. Machine learning methods could improve on conventional methods, since they are able to learn by example.

In a similar fashion, existing methods of hot target (e.g. wildfires) detection transform the original multispectral feature space using simple logic functions, and encounter similar problems. Machine learning could benefit classification of this type of imagery in like manner.

The aim of this research was to investigate how machine learning can be used to perform ground cover analysis of RGB smartphone photography, and hot target detection in multispectral satellite imagery. Our main focus was on the effectiveness of machine learning methods SVM, MLP, RF, and DNN, compared to large array of existing methods in these domains. We experimented with a large number of optimizations to these machine learning methods to obtain state-of-the-art performance.

We found that for ground cover analysis, machine learning improves on existing methods for both segmentation and estimation. Regarding segmentation of smartphone RGB images, we observed that our DNN implementation outperforms all other existing methods on our scaled dataset, especially when trained on each plant type specifically. Regarding estimation, we see that machine learning methods outperform all other conventional methods. Our DNN implementation shows comparable results to pure regression methods, whilst maintaining the ability to provide accurate segmentation maps. For hot target detection we found that while machine learning methods can greatly improve on current methods, our DNN implementation is not able to produce the same results. Much better results are achieved using an SVM or MLP implementation. Although we mainly see very positive effects of the machine learning methods presented in this research, one disadvantage of learning-based approaches is the necessity of large quantities of accurately-labelled ground truth data, which is not always available in these domains.

Acknowledgements

There are many people that have contributed to this thesis, and my experience at TNO for the last year, to whom I must be thankful.

First I must thank my advisor Maaïke de Boer, for her full support, guidance, and enthusiasm throughout many meetings and emails during my internship and thesis work. Not only did she push me to take a step further in my research, she also introduced me to many interesting people and new academic opportunities. I am grateful to Maaïke for teaching me not just about the computer vision domain, but also about exploring new possibilities.

Second, I would like to thank Raul Zurita-Milla for his incredible assistance, expertise and encouragement. When I first pitched my idea to Raul he was immediately excited, which in turn gave me the confidence to start working on this project. Many of his ideas are weaved into the lines of this work, and I can say assuredly that without him this thesis would not have been possible.

I have to thank Robin Aly for his sage counselling and scientific insight. Robin has helped me countless times over the last year with his stunning ability to dissect a problem and ask the most fundamental questions. It was a pleasure being able to work with, and learn from him.

I would also like to extend my thanks to Gwenn Englebienne, who, on quite short notice, was able to be part of my graduation committee. His support means a lot to me.

For the astounding work of the team that did all the preliminary work for our ground cover analysis, I would like to extend my special thanks to the team in Bangladesh: Urs C. Schulthess, Golam M. Rokon, Khairul Islam, and Md Atikuzzamman. I will continue to follow their success, and hope to see the fruits of their labour in person some day.

I must also thank Sam W. Murphy and Zhenghong Yu for their kindness in sharing their data and research with us, on hot target detection and ground cover analysis respectively.

For the inspiring atmosphere, their friendship, and laughs along the way, I want to thank the people at TNO - in particular the intern group at Intelligent Imaging and at New Babylon.

Finally, I wish to thank Rosanne, Hanny, and Leo, who always believe in me far more than I do, for their amazing support.

Contents

Abstract	iii
Acknowledgements	v
1 Introduction	1
1.1 Ground cover analysis	1
1.1.1 Related work	2
1.1.1.1 Colour indices	3
1.2 Hot target detection	4
1.2.1 Related work	4
1.2.1.1 Remote sensing	5
1.2.1.2 Learning-based remote sensing	7
1.3 Machine learning	8
1.3.0.1 Support Vector Machines	8
1.3.0.2 Random Forest	9
1.3.0.3 Neural networks	9
1.4 Research questions	11
1.4.1 Ground cover analysis	12
1.4.2 Hot target detection	12
1.5 Hypotheses	13
1.5.1 Ground cover analysis	13
1.5.2 Hot target detection	13
2 Data	15
2.1 Ground cover analysis	15
2.2 Hot target detection	17
3 Method	19
3.1 Ground cover analysis	19
3.1.1 Colour indices	19
3.1.2 AP-HI	20
3.1.3 SVM	20
3.1.4 Random Forest	21
3.1.5 Deep neural network	22
3.1.5.1 U-net	22
3.2 Hot target detection	23
3.2.1 HOTMAP	23
3.2.2 SVM	25
3.2.3 Deep neural network	25
3.3 Evaluation	25

4 Results	27
4.1 Ground cover analysis	27
4.1.1 Segmentation	27
4.1.1.1 Colour indices	30
4.1.1.2 AP-HI	32
4.1.1.3 Deep neural network	32
4.1.2 Estimation	34
4.1.2.1 Colour indices	35
4.1.2.2 SVM	35
4.1.2.3 Random Forest	37
4.1.2.4 Deep neural network	37
4.2 Hot target detection	38
4.2.1 CA & HOTMAP	38
4.2.2 SVM & MLP	38
4.2.3 Deep neural network	39
5 Discussion	41
5.1 Ground cover analysis	41
5.2 Hot target detection	43
6 Conclusion	45
6.1 Future Work	46
Appendices	47
A DNN Architecture	47
B Ground cover estimates RMSE	51
C Ground cover segmentation accuracy & precision	53
D FIRES dataset scenes and topographical location	55
E Impact on society	57
Bibliography	59

List of Figures

1.1	Segmentation of rice crops. Image taken from Bai et al. [25].	2
1.2	Bandpass wavelengths for the Landsat 8 OLI and TIRS sensors, compared to the Sentinel-2 MSI sensor, and Landsat 7 ETM+ sensor. Image obtained from [16]	6
1.3	Classification of hot pixels in Landsat 8 data in the vicinity of Adelaide, Australia (LC80970842015004LGN00). <i>a)</i> Landsat 8 spectral bands 2 (<i>top left</i>); 5 (<i>top right</i>); 6 (<i>bottom left</i>); and 7 (<i>bottom right</i>). <i>b)</i> False colour RGB image of bands 2, 3, and 4. <i>c)</i> Binary image of hot pixel classification output by the algorithm proposed by Murphy et al. [81]. <i>d)</i> Hot pixels marked in red superimposed on the original false colour RGB image.	7
1.4	Visualization of the AlexNet architecture, showing a dual-GPU setup that each process a different part of the input image. Image obtained from [66].	10
1.5	Visualization of features in a convolutional neural network. Activation maps of high scoring layers (right side) are shown for random inputs (left side). Images obtained from [122]	11
2.1	The three types of crops present in this dataset. Mung bean (left); Maize (middle); and Wheat (right). Notice the common obstacles for good segmentation: Shade casting, hard lighting, overlapping leaves, and residu on the ground.	15
2.2	Frequency of FCOVER value occurence per plant type.	16
2.3	Sample data of vegetation RGB imagery taken using a smartphone, and hand-annotated ground truth data using CAN-EYE software [3].	17
2.4	Example data of the FIRES dataset by Murphy et al. [81] used in this research. Imagery was taken in the vicinity of Alaska, USA (LC80690182014140LGN00). Shown are OLI bands 1 to 9 (excluding Panchromatic band 8), TIRS bands 10 and 11, a False Colour (FC) composite image, and the binary Ground Truth (GT) labelling.	18
3.1	Sample data downsampled into smaller colour ranges, and upsampled again for visualization purposes. Colour ranges from top left to bottom right: 1, 2, 3, 4, 8, 16, 32, 64, 128, 256 (original).	21
3.2	U-net architecture. The blue boxes represent multi-channel feature maps. The number of channels is denoted at the top of the boxes, and the shape data is denoted at the bottom. The white boxes in the upsampling pathway correspond to the copied feature maps. Arrows denote various layer operations (see legend). Image taken from Ronneberger et al. [94].	24

4.1	RMSE and standard deviation of scaled images for ground cover estimates for all segmentation methods.	30
4.2	Precision and accuracy scores for methods of ground cover segmentation of scaled images. Full resolution image available in Appendix C.	30
4.3	Histogram of the intensity levels of the NDI index for an original Mung bean image and downscaled version, and corresponding segmentation maps (Found Otsu threshold in red).	31
4.4	Comparison of cropped original image (left), extracted ground truth (middle), and resulting DNN segmentation map (right).	32
4.5	Comparison of RMSE for different smartphone types.	34
4.6	RMSE and standard deviation of scaled images for ground cover estimates for all regression methods.	35
4.7	Actual FCOVER values plotted against the predicted value for colour indices methods. Warmer colours indicate higher discrepancies.	36
4.8	Effect of tweaking, in order, kernel type k , error term penalty C , maximum distance between actual and predicted value at which no penalty is given ϵ , tolerance for stopping t , and kernel coefficient γ , on mean error.	36
4.9	Effect of colour range size on mean error.	37
4.10	Crossplot of pixel values for band 7 versus pixel values other bands, showing seperability. Red data points are hot targets, blue data points are non-hot targets.	39
4.11	Examples of DNN segmentations of augmented satellite imagery. Shown are the original OLI band 7 image (left), ground truth hot target annotation (middle), and DNN labelling (right).	39
B.1	RMSE of original images for ground cover estimates for all segmentation methods.	51
B.2	RMSE of scaled images for ground cover estimates for all segmentation methods.	51
B.3	RMSE of scaled images for ground cover estimates for all regression methods.	52
D.1	Locations of the scenes in the FIRES dataset. Image taken from [81].	55

List of Tables

2.1	Statistics of CROPS dataset.	16
2.2	Statistics of CROPS_PHONES dataset.	17
3.1	Mathematical notation of different colour indices.	20
4.1	Example segmentations of Maize, Mung bean, and Wheat crops for the given segmentation algorithms (scaled images).	29
4.2	RMSE results (standard deviation) of a single net versus the scores of the three separately trained networks.	33
4.3	The effect on RMSE (standard deviation) of varying ways to apply dropout.	34
4.4	Mean errors (standard deviations) of an RF classifier for varying colour ranges (CR).	37
4.5	Hot target detection by method.	38
4.6	Accuracy and precision scores of the SVM when removing all information of a certain band.	40
D.1	List of scenes used in the FIRES dataset.	56

List of Abbreviations

ANN	Artificial Neural Network
CNN	Convolutional Neural Network
DNN	Deep Neural Network
FCOVER	Fraction of Ground Cover
ML	Machine Learning
MLP	Multi Layer Perceptron
NLP	Natural Language Processing
OLI	Operational Land Imager
RBM	Restricted Boltzmann Machine
RF	Random Forest
RMSE	Root Mean Squared Error
SVM	Support Vector Machine
TIRS	Thermal Infrared Sensor
TOA	Top Of Atmosphere
VQA	Visual Question Answering

Chapter 1

Introduction

In this work, we evaluate the effectiveness and applicability of machine learning methods in the domains of ground cover analysis and hot target detection. These two seemingly quite different domains come together in this treatise through the shared benefit of machine learning, which we hope to present here. The duality of ground cover analysis and hot target detection will be apparent throughout this thesis. In the following sections we first introduce both domains, after which we will give a broad overview of the machine learning field. We conclude this chapter with a more elaborate exposition of our problem description and corresponding research questions and hypotheses.

1.1 Ground cover analysis

Ground cover (also referred to as FCOVER), the percentage of ground covered by vegetation in a specified area, is a commonly used metric in agronomic studies, for example to estimate crop yield. Ground cover has been shown to be correlated with biomass yield (measured by weighing dried biomass) by Baret and Guyot [26]. The estimation of ground cover can be used in smart agriculture applications to perform effective automatic treatment of crops, for instance by finding an optimal distribution of water or pesticide administration [56]. This is especially helpful for farms located in third world countries, that have a lot to gain in terms of crop yield. Farmers in developing regions such as the Bangladesh delta struggle to cultivate their land during the dry season due to soil moisture levels in these times, which are often lacking. Ground water is often salty or briny, which is unfit for raising crop plants. Farmers need to use water pumps to be able to use sweet surface water from rivers and channels, but the pumps are expensive. Therefore, irrigation needs to be optimized. Effective use of irrigation systems depends largely on the current ground cover and soil water balance. Accurate ground cover analysis could therefore help in developing an advisory system for irrigation scheduling, thereby increasing crop yields [103].

Ground cover analysis, including estimation (the percentage of an area covered by vegetation), classification (instance-based labelling of vegetation), and segmentation (demarcation and extraction of regions of vegetation), is most often done using image-based methods (though other methods such as thermal analysis also exist [77]). A wide range of computer vision and image processing techniques are applied to monitor and study agricultural changes using both ground level RGB cameras and multi-spectral air- and space-borne radiometers. Successful ground level image-based ground cover analysis relies on overcoming a number of common obstacles found in agrarian imagery. Lighting conditions can be extremely varied

over time. Images obtained during sunny weather can result in different ground cover estimations compared to images of the same field obtained with overcast, or rainy weather [56]. This is due to specular reflection in sunny conditions, or differences in refraction when a canopy is showered in raindrops. Another obstacle is the formation of shadows of the canopy onto itself. Parts of the vegetation that are cloaked in the shadow of leaves or other objects may be classified as soil instead. Especially for large crops that have a high number of leaves, this problem can severely affect classification and segmentation accuracy. Finally, the soil area may be littered with objects that do not fall into either of the two categories, soil or vegetation. Residual waste from other nearby vegetation or animals may influence ground cover analysis, as would small stones, or pieces of wood. [78] [56]. An example of crop segmentation is shown in Figure 1.1. An example of adverse lighting effects and overlapping in our dataset can be found in Figure 2.1.

We have listed the main issues facing effective image based ground cover estimation. In the following section, we will elaborate upon the methods currently applied to overcome these challenges.

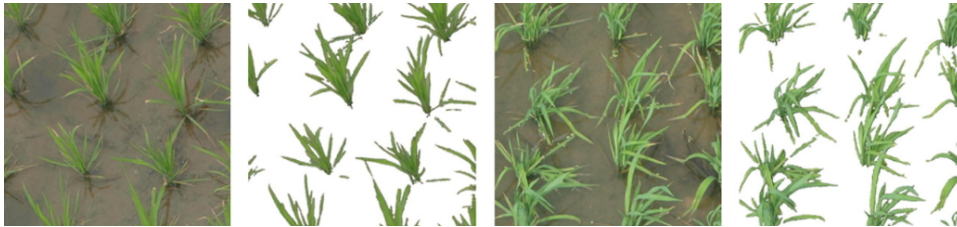


FIGURE 1.1: Segmentation of rice crops. Image taken from Bai et al. [25].

1.1.1 Related work

Ground cover analysis and crop segmentation has been an active field of research for decades [37]. We will mainly consider the segmentation problem here, because it envelops the classification problem to a degree (if a segmentation map can be generated, pixel-wise classification has either been done already, or trivial to perform), and can be used to perform estimation as well (though segmentation is not a requirement for cover estimation, since we can transform it into a regression problem). From this point onwards we will refer to the combined area of *ground cover segmentation* and *ground cover estimation* as *ground cover analysis*. Ground cover segmentation is essentially a two class problem. Pixels in an image belong either to a *vegetation* class, or to a *soil/ground* class. Hamuda et al. [56] provide an extensive research on image based vegetation segmentation techniques, broadly categorizing these techniques as colour index based, threshold based, and machine learning based. We will evaluate existing methods keeping to this categorization, but grouping threshold based methods with colour index based methods because the former is mainly performed using colour indices as well.

1.1.1.1 Colour indices

To a human observer, the most noticeable distinction between soil and vegetation is the colour of these categories. Whilst soil is for the most part brown or grey, vegetation is often mainly green or yellow. This difference in colour is also used in crop segmentation techniques. Most conventional cameras used in ground level field photography will generate RGB images. One problem with RGB images is that they are dynamic, meaning that the colour of soil varies throughout an image. The soil may be less moist in one place and very moist in another, affecting its RGB values. The same holds for different shades of green on leaves. To give more prominence to the soil/vegetation colour difference this RGB colour space is converted to another colour space, for example by attributing different weightings to the R, B, and G values, resulting in a colour index. In extreme cases, this may directly lead to a binary image that can be used for segmentation. These indices are often generated using either expert knowledge, or machine assisted methods such as fuzzy classifiers [56].

Examples of alternative colour spaces are the Normalized Difference Index (NDI), and Excess Green Index (ExG) proposed by Woebbecke et al. [119] [118]; The Excess Red Index (ExR), and Excess Green minus Excess Red Index (ExGR) proposed by Meyer et al. [73] [74]; The Modified Excess Green Index (MExG) proposed by Burgos-Artizzu et al. [30]; The Colour Index of Vegetation Extraction (CIVE) proposed by Kataoke et al. [61]; The Normalized Green-Red Difference Index (NGRDI) proposed by Hunt et al. [60]; And the Vegetative Index (VEG) proposed by Hague et al. [55]. Weighted combinations of these indices were also researched by Guijarro et al. (COM1) [53] and Guerrero et al. (COM2) [52]. Patrignani et al. [88] developed a tool called CANOPEO (CANO), using a colour index based on the Excess Green (ExG) index. An overview of the formulation of these indices is presented in Table 3.1.

Colour index based methods of vegetation segmentation have the advantage that they are simple, fast and computationally light, but struggle to perform accurately when lighting conditions are bad (i.e. in overcast or sunny weather, or with shadows). This is apparent when one observes the large number of different indices used throughout literature for various purposes. The indices found in Table 3.1 also show such a variation in performance based on the type of images they are applied to [56].

Usually, these colour indices are using in combination with a set threshold to generate a binary image that can be used for segmentation. The Otsu method [85] is one of the most common approaches to thresholding. It calculates the optimal value to separate two classes by maximizing the inter-class variance (based on a foreground/background histogram). Equation 1.1 shows the formula of the weighted sum of variances (or intra-class variance) of the two classes that need to be separated. Otsu's method then searches for a threshold that minimizes this intra-class variance (and in doing so maximizes inter-class variance). The class probabilities are calculated from the image histograms. Other techniques include local dynamic thresholding [91], hysteresis thresholding [72], and homogeneity thresholding [46] (deriving a local threshold by calculating local homogeneity from converted greyscale intensity images). Dynamic thresholds increase an algorithms complexity (and therefore influence execution time and memory usage), but often provide better results compared to using only simple colour

index based methods with fixed thresholds.

$$\sigma_w^2(t) = w_0(t) \cdot \sigma_0^2(t) + w_1(t) \cdot \sigma_1^2(t)$$

with

$$w_i(t) = \text{Probability of the class separated by threshold } t$$

$$\sigma_i^2(t) = \text{Variance of the class}$$
(1.1)

1.2 Hot target detection

Hot targets such as wildfires and volcanoes have a devastating impact on the planet, our infrastructure, and our personal health. Wildfires across the globe result in a yearly additional $3.5 \cdot 10^{15}$ g to the existing atmospheric carbon emission [98] [113]. These emission numbers vary per type of fire. If the aim is to decrease emission numbers, grassland, savannah, and deforestation fires are the prime targets for hot target detection. Fire emission time series analysis conducted by van der Werf et al. [114] shows that the largest contributors to carbon emissions were grassland and savannah fires (44%), with tropical deforestation and degradation fires (20%), woodland fires (16%), and forest fires (15%) being big factors as well. Most fires in remote areas and at higher elevation are caused by lightning, while we see an inverse effect near urban conglomerations [108] [63]. Wildfire detection is increasingly important, due to the globally increasing prevalence of wildfires. Westerling et al. show that wildfires have increased in both frequency and duration in the US since the 1980's [117]. We also see a global increase in yearly burned area, though there is quite some spatial variability in these assessments [42].

However, the risk and hazards of wildfires are not restricted to just atmospheric carbon emission. Humans, wildlife, and vegetation are severely affected as well. Injury and death resulting from exposure to heat and smoke inhalation, and trauma due to the loss of structural integrity are just two of the most prominent examples of the adverse effects of wildfires for humans. Other examples include indirect exposure to chemicals released during a fire, through water or soil contamination [44] [105] [70]. The consequences of the destructive nature of wildfires detailed above take years to restore, and can cost billions: the US Forest Service estimates the cost of fire suppression to increase to nearly USD 1.8 billion by 2025 [15]. That is leaving out other costs such as ecological and infrastructural reconstruction, and medical aid.

Clearly, effective detection, suppression, and prevention are necessary to combat wildfires. Detecting hot targets early and monitoring their progress is important to successfully manage and control these situations.

1.2.1 Related work

Detection and suppression of both urban and rural wildfires has been important for many civilizations throughout history. In ancient Rome, the *Vigiles* was a proto-firefighting brigade created by Nero (though they could not prevent the Great Fire of Rome), and similar watch forces were maintained in Europe through the ages [109]. More modern approaches to fire detecting came in the 20th century, when fire lookout towers, infrared camera's,

and smoke detectors were employed. These approaches are referred to as *remote sensing*, which is the collecting and interpreting of information about the environment and earth's surface without making physical contact [6]. Fire analysis using satellite imagery or aerial data, also part of remote sensing, was first performed in the 1980's [22]. Beside local sensor networks and regular fire lookout towers, it is one of the main practices for fire detection used today.

In the following sections we will give a brief overview of current relevant research in automatic hot target detection using satellite imagery. We will also expand on relevant research in machine learning, and its role in hot target detection.

1.2.1.1 Remote sensing

Remote sensing approaches such as wireless sensor networks are growing in terms of research and implementation [21]. Wireless sensor networks have been shown to be effective in detecting and forecasting forest fires in real-time, as opposed to satellite-imagery-based methods that have low spatial and temporal resolution. Yu et al. [120], for example, propose a neural network paradigm for wireless sensor networks in which sensor nodes collect data (i.e. temperature, humidity, smoke) that gets sent to a cluster node that - together with other cluster nodes - processes the data using a neural network. The neural network takes the input data and produces a weather index (likelihood for the current weather to cause a fire) and reports it to a manager node, which in turn produces a fire danger rate. Though no comparison with imagery-based methods is made, their neural network approach is more efficient than other in-network processing methods.

Most existing data-driven approaches of fire detection however, are satellite-imagery-based [50] [101] [35] [39] [48]. The moderate-resolution imaging spectroradiometer (MODIS) was launched into orbit aboard the Terra (1999) and Aqua (2002) satellites respectively. Combined, Terra MODIS and Aqua MODIS can map the Earth's surface in 1 to 2 days, obtaining data from 36 spectral bands. These bands come in 3 spatial resolutions: 2 bands at 250m/px, 5 bands at 500m/px and 29 bands at 1km/px [10]. MODIS produces global fire products every day using the original detection algorithm by Kaufman et al. [62] and currently the improved contextual algorithm proposed by Giglio et al. [50] [49]. This implementation of the MODIS fire detection algorithm relies - besides pre-/post-processing steps like cloud masking and sun-glint rejection - on manually selected thresholds for *top of atmosphere* (TOA) reflectance/radiance (see Equations 1.2, 1.3 and 2.1), though improvements are still being actively researched [51]. Vilar et al. [115] compare fire activity as reported by the MODIS algorithm to official government reports in the Mediterranean Europe (EUMED) region, and show that burnt area prediction coincides in more than 90% of the cases with these reports.

$$\rho\lambda' = M_\rho Q_{cal} + A_\rho$$

with

$\rho\lambda'$ = TOA planetary reflectance, without correction for solar angle

M_ρ = Band-specific multiplicative rescaling factor

A_ρ = Band-specific additive rescaling factor

Q_{cal} = Quantized and calibrated standard product pixel values (1.2)

$$L\lambda = M_L Q_{cal} + A_L$$

with

$L\lambda$ = TOA spectral radiance (Watts/($m^2 * \text{srad} * \mu m$))

M_L = Band-specific multiplicative rescaling factor

A_L = Band-specific additive rescaling factor

Q_{cal} = Quantized and calibrated standard product pixel values (1.3)

Fire detection algorithms based on data acquired by the Visual Infrared Imaging Radiometer Suite (VIIRS), an imager with a higher spatial resolution than MODIS, also rely on these manually selected thresholds. VIIRS launched in 2011 aboard the Suomi-NPP satellite, and obtains spectral imaging data from 21 bands (16 at 750m/px and 5 at 375m/px). A second VIIRS is expected to launch aboard the JPSS-1 in 2017 on the same orbit as the first VIIRS. VIIRS fire products are generated using a stripped down version of the MODIS algorithm, where C4 (The 4th iteration of the algorithm) is used for the 750m/px product, and C6 for both 750m/px and 375m/px products [100] [20] [13] [12]. These fire products are used for further analysis on, for example burned area mapping [82], modelling of freight traffic [104], or combustion source characterization [123].

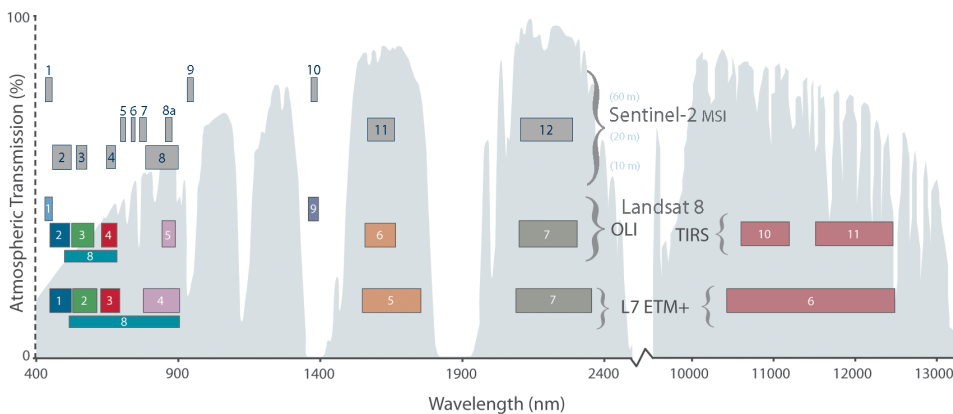


FIGURE 1.2: Bandpass wavelengths for the Landsat 8 OLI and TIRS sensors, compared to the Sentinel-2 MSI sensor, and Landsat 7 ETM+ sensor. Image obtained from [16]

Even higher resolution imagery is provided by the Landsat 8 satellite. Landsat is the longest running satellite imagery program, running since 1972. The latest satellite, Landsat 8, was launched in 2013 and provides

images with a spatial resolution of 15 to 100 m/px and a temporal resolution of 10-16 days. Landsat 8's Operational Land Imager (OLI) can acquire data in 9 spectral bands with 30m/px spatial resolution, while the Thermal Infrared Sensor (TIRS) collects 2 spectral bands with 100m/px spatial resolution. Figure 1.2 shows the bandpass wavelengths for the Landsat 8 sensors [8]. High spatial resolution imaging data acquired by the Landsat satellites have been used for a range of topics, including hot target detection such as volcanism [83] [43] [87] and fires [79] [81] [101]. Due to open access policies, the full Landsat archive has been made publicly available by NASA/USGS. The European Space Agency (ESA) has also adopted similar policies, providing access to data obtained through their Copernicus program (including the latest Sentinel-2 missions), as well as Japan and their ASTER mission. However, even Sentinel-2 data ranging up to 10m/px is insufficient to chart the small fields in the Bangladesh delta, exposing the need for non-satellite based methods as well.

Fusion of different imaging data has also been used to overcome the spatial and temporal limitations of each imager. Boschetti et al. [29] have used MODIS-Landsat fusion to identify burned area with MODIS active fire detection at a 30m/px spatial resolution. Similar fusion has also been applied for gap filling, and radiometric normalization [96]. Murphy et al. [81] present a novel global hot target detection algorithm with high detection rates (80%) and low false positive rates (<10%) that incorporates both Landsat 8 and Sentinel-2 data. An visualization of an implementation of their novel daytime detection algorithm can be found in Figure 1.3. Fusion of Landsat and MERIS imagery has been performed by Zurita-Milla et al. by applying unmixing-based data fusion to combine Landsat's spatial resolution with MERIS' spectral resolution [125].



FIGURE 1.3: Classification of hot pixels in Landsat 8 data in the vicinity of Adelaide, Australia (LC80970842015004LGN00). *a)* Landsat 8 spectral bands 2 (*top left*); 5 (*top right*); 6 (*bottom left*); and 7 (*bottom right*). *b)* False colour RGB image of bands 2, 3, and 4. *c)* Binary image of hot pixel classification output by the algorithm proposed by Murphy et al. [81]. *d)* Hot pixels marked in red superimposed on the original false colour RGB image.

1.2.1.2 Learning-based remote sensing

There have been a number of approaches to remote sensing that rely on machine learning. Petropoulos et al. [90] have investigated the use of SVMs with Landsat data to perform burned area mapping with high accuracy. Persello et al. [89] use SVMs to classify urban areas in an active and semi-supervised learning context, in which the SVM is fed training data, annotated by a human expert, that is expected to be most effective for training. A

full review of the use of SVMs in remote sensing is provided by Mountrakis et al. [80].

In the domain of remote sensing, some research on DNNs has been done. Le et al. [67] show that deep learning architectures can be used to classify RS multispectral data on a dataset with 9 classes ranging from *Water* to *Trees* to *Asphalt*. Basu et al. [27] also use a DNN approach to classify satellite imagery in land cover classes, and obtain impressive results on the SAT-4 and SAT-6 datasets, outperforming other deep methods. Pakhale & Gupta [86] compare an ANN-based method of land pixel classification to an SVM method, and find that the ANN-based method outperforms the SVM-based method (average accuracy of 82.5% and 75.00% respectively on 5 different classes). Castelluccio et al. [32], and Hu et al. [59] utilize a CNN to classify different types of land (i.e. *forest*, *buildings*, *beach*) with great success.

To the best of our knowledge, no DNN-based approach has been published for hot target detection. However, this type of multispectral data lends itself well for DNN analysis, as we have argued in previous sections.

1.3 Machine learning

Machine learning (ML), programming computers to learn from *experience* [99], has been applied in a vast range of data-driven research areas for classification, clustering, segmentation, and regression. Kotsiantis et al. [65] provide a review of supervised ML methods, a category of ML approaches by which a model is trained using labelled example data. The model is iteratively corrected, according to the labels corresponding to the input data, through which a function that maps the input data to the desired output data emerges. Machine learning, including Bayesian-, RF-, SVM-, and neural network-based approaches, has been proven to be very successful in the medical [28], bioinformatics [75], and computer vision domain [95] [38] [102] [92].

1.3.0.1 Support Vector Machines

Support vector machines (SVMs) are a type of supervised learning paradigms used to perform classification, segmentation, and regression. As described above, SVMs utilise labelled training data to classify new data. SVMs have also been used in combination with image analysis. Zhu et al. [124] take an SVM-based approach to classify imagery data from the Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER, a 15-band imager aboard Terra) into 6 different classes of ground type, obtaining an accuracy rate of 89.9% on average. Tellaeché et al. [112] use SVMs to identify detrimental weeds in between healthy crops to decide if an area needs to be sprayed with pesticide. Although machine learning based methods show good accuracy, they require training on large labelled datasets to reach effective classification/segmentation. These datasets are not always available, and may be costly to create. Mitra et al. [76] use SVMs in combination with active learning to classify different types of land cover. They try to overcome the problem of having only a small set of labelled data by training on a small set initially and refining the classification by querying for the most ambiguous data point at each subsequent set (allowing it to train

very effectively). Guo et al. [54] use a decision tree model to overcome the problem of shadows and specular reflection, as described in the previous sections. Yu et al. [121] researched a well performing crop segmentation algorithm, *AP-HI*, also dealing with dynamic lighting conditions and other environmental elements. The algorithm combines hue intensity analysis with affinity propagation clustering (as proposed by Frey and Dueck [45]). Although AP-HI has excellent performance and can deal with shadow areas quite well, it still struggles on regions that are very brightly illuminated, especially on larger canopies.

1.3.0.2 Random Forest

Another type of ML method is Random Forest (RF). It can be applied for classification, regression, and segmentation, and perform well when the task is a multi-class problem. RF constructs a large number of decision trees, the nodes of which are initialised with random weights, and restrictions are placed on the decision parameters. This is why RF is called an ensemble learning method. The data is then run through all the decision trees, after which each decision tree gives its own classification. *Boosting*, the application of random bias in the decision trees can also be used to good effect, so that the trees initially favour a certain class more. An aggregate function such as the mean or maximum value can then be taken to end up with a final classification result. Rodriguez-Galiano et al. [93] have assessed the effectiveness of RF classifiers in land cover classification, and shown that RF can give reasonable results for 14 types of ground cover.

1.3.0.3 Neural networks

More recently, deep learning methods - a branch of artificial neural networks (ANNs) - have gained increasing interest in the machine learning community [68], and in mainstream media due to the success of AlphaGo [106] (The first AI to ever beat a professional human Go player) and IBM's Watson/DeepQA [41] (having participated in a televised episode of *Jeopardy!* against two of the most successful contestants, and won). ANNs are a type of model inspired by its biological counterpart, hence the name. They are structured as a network of nodes and directed weighted edges, and incorporate an activation function similar to the natural neuron. The input for a given node can consist of one or many connections to other nodes. The input from all input nodes is combined using a transfer function (i.e. summation), and tested against an activation function containing a set threshold. When the threshold is reached, the node will *fire*, after which its input is available to succeeding nodes. Eventually the network will calculate an output value (or multiple, depending on the number of output nodes), based on its input, weights, transfer functions and activation functions. *Backpropagation* [97] is then used to allow the model to learn. When the network has calculated its output value, it can be right or wrong with a certain error margin, which can be defined as an error function. By backwardly propagating this error into the network, it can adjust its weights to provide a more accurate output value in the next iteration. This is done using *gradient descent* [33], an optimization algorithm that minimizes the error function stepwise by iterating over the training set (a faster stochastic version using a single data point can also be used, but can provide suboptimal

results). ANNs can be built using many layers, allowing it to learn complex relationships between input data and evaluation data.

There are many types of neural networks: Feedforward networks such as autoencoders, and restricted Boltzmann machines (RBMs); Convolutional networks such as AlexNet [111] or R-CNN [92] that have layers that convolve kernels with the image; Recurrent networks such as LSTM [47] (often used in time series analysis); and Recursive networks (recursive autoencoders, for example). Often, a combination of these networks is used, as is the case in convolutional networks that combine fully connected autoencoders, or stack RBMs.

Deep neural networks (DNNs) are a subclass of ANNs that incorporates a multitude of hidden layers, allowing it to learn more complex functions. They are often called *Deep Learning* methods. DNNs show state-of-the-art results in various domains including natural language processing (NLP) [34], speech recognition [58], and computer vision tasks like concept-detection [107] and visual question answering (VQA) [23]. An example of a DNN architecture (AlexNet [66]) can be found in Figure 1.4.

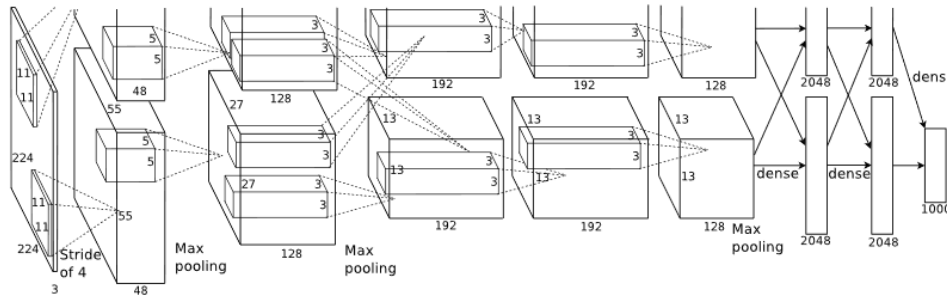


FIGURE 1.4: Visualization of the AlexNet architecture, showing a dual-GPU setup that each process a different part of the input image. Image obtained from [66].

The prime component of a convolutional neural network (CNN) is the convolutional layer. This convolutional layer consists of a kernel (smaller than the original input layer) that is convolved with the image, computing dot products. The main advantage of a convolutional layer is that they are able to learn both spatial patterns across multiple input pixels and patterns between pixel channels (R,G,B for example), instead of only the latter. Often, many of these layers are sequenced together, intermixed with activation functions. Activation functions define the output of a neuron given its input. A simple example of this is the rectified linear unit (ReLU) activation function ($f(x) = \max(0, x)$), but more complex functions are used as well (SoftPlus, Gaussian). This results in an activation map per kernel. A feature visualization of convolutional networks is shown in Figure 1.5. Fully (or dense) connected layers contain neurons that are connected to all input neurons (as is the case with normal neural networks). Pooling layers, such as (soft-)max pooling, can be used a well to downsample and reduce the size of feature representations. Dropout layers, layers that remove nodes based on a certain probability distribution, have also been shown to improve performance because they can prevent overfitting (especially in fully connected networks). An example of the full CNN architecture of AlexNet [66] can be found in Figure 1.4. Typical CNN architectures employ a number of *convolution* and *pooling* layers initially, followed by a number

of *fully connected* layers and finally a *pooling* layer, although recent advances by ResNet [57] and GoogleNet [110] question this prototype.

One of the main disadvantages that neural networks have is that training them requires a lot of labelled data, which may not always be readily available. In addition, localization is not the prime strength of neural networks, therefore accurate pixel-wise segmentation is a challenge. To overcome this problem, Ronneberger et al. propose an effective network architecture (U-net) that relies on data augmentation, that allows pixel-wise state-of-the-art classification [94]. U-net consists of a contracting, down-sampling pathway that incorporates a number of convolutional layers, and an expanding, upsampling pathway that allows for localization. A more exhaustive analysis of U-net is given in chapter 3.

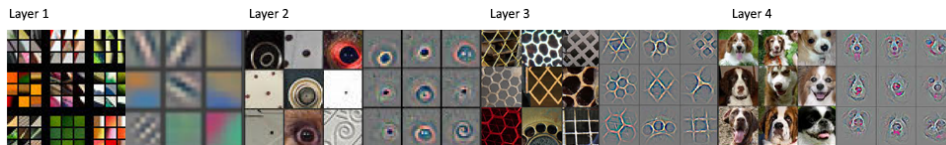


FIGURE 1.5: Visualization of features in a convolutional neural network. Activation maps of high scoring layers (right side) are shown for random inputs (left side). Images obtained from [122]

Although DNNs are used for semantic segmentation (for example by Long et al. [71] and Badrinarayanan et al. [24]), not much research has been done on ground cover segmentation using a deep learning approach. Li et al. [69] use denoising autoencoders to segment RGB images of cotton fields, achieving state-of-the-art performance. To the best of our knowledge, no other publicized research has been performed to perform ground cover segmentation using deep neural networks.

Deep learning based approaches to image segmentation have other interesting applications beside ground cover segmentation. Trained networks may be applicable to other domains with little adjustment. That means we can train a model on, for example, biomedical data such as images of tissue, and use the same model to do segmentation in other domains. One such domain is the field of hot target detection, where deep learning also has the potential to obtain impressive results.

1.4 Research questions

As we have described in the previous sections, the classification of RGB imagery into ground cover categories and the classification of multispectral imagery into hot target categories are related. Both problems can be described as a pixel-wise multi-class classification problem, that can benefit from taking into consideration information outside of that pixel. In both these domains, current methods do not leverage that information. Therefore, we aim to investigate the use of machine learning, in particular SVM, RF, MLP, and DNNs (CNNs in particular), for the application of hot target classification.

1.4.1 Ground cover analysis

Since agricultural tracts of land are small in the Bangladesh delta, open satellite data is not sufficient to chart these fields due to their spatial resolution. Even Sentinel-2 data, ranging up to 10m/px is still insufficient to map these plots of land. Alternatively, ground cover classification can be done with RGB imagery using smartphones. Although this method does not provide multispectral data, it does allow the monitoring of small crop fields, and additionally circumvents the problem of cloud cover due to the camera location.

Ground cover classification using RGB imagery must deal with diverse illumination conditions caused by the sun, rain, clouds, and the vegetation itself. Images taken using smartphones will pick up not only various changes in soil hydration, but also remnants of other plants. Algorithms have been proposed that estimate ground cover with RGB imagery using thresholded colour space analysis, but either require too much calibration for practical use [40] [56], or have not been tested for these conditions [31] [116]. Deep learning based approaches are hard to find in literature, though they have the potential to give state-of-the-art results. We therefore believe this research can make a valuable contribution to the field of ground cover analysis.

1.4.2 Hot target detection

Much research has been done in the domain of satellite image classification. Machine learning methods such as SVMs and DNNs have been used to classify pixels into different land cover classes with great success. However, none of this research is applied to hot target detection. Existing global hot target detection systems have mostly relied on data with low spatial resolution sensors (MODIS: 1 km; VIIRS: 750/375m), but are insufficient to detect small active hot targets due to a tendency for background radiation to dwarf the signal. Higher spatial resolution imagery is available from the Landsat program, with recent advancements in the Landsat 8 series pushing spatial resolution to 30m. This enables improved detection of small fires, and refined mapping of large fires.

Recent studies have shown that small wildfires can be detected in Landsat 8 imagery by using Top of Atmosphere (TOA) reflectance (see Equation 1.2) in bands 5, 6, and 7 (central wavelengths of 0.87, 1.61 and 2.2 μm , respectively) of the Operational Land Imager (OLI) sensor, combined with hand-selected thresholds. Although detection rate is high (>85%), false alarms are also frequent, especially in urban areas (where accurate detection might be most important).

The approaches that are taken in current state-of-the-art research on hot target detection use carefully crafted logic functions and manually selected thresholds. Considering the impressive results that DNN-based methods obtain in related computer vision tasks such as land cover classification and event detection, we believe research on the effective use of DNNs in hot target detection is warranted.

The aim of this research will be to investigate how machine learning can be used to perform ground cover classification of RGB smartphone photography for crop fields, and hot target detection in multispectral satellite

imagery. We propose the following research questions:

How can machine learning be applied to improve ground cover analysis?

1. Do the machine learning methods proposed in this work, namely SVM, RF, MLP, and DNN, improve ground cover segmentation accuracy and precision over existing methods of ground cover segmentation?
2. Do the machine learning methods proposed in this work, namely SVM, RF, MLP, and DNN, improve ground cover estimation over existing methods of ground cover estimation?

The following research questions are also considered, specific to the hot target detection domain.

How can machine learning be applied to improve hot target detection?

1. Do the machine learning methods proposed in this work, namely SVM, RF, MLP, and DNN, improve accuracy and precision of hot target detection over existing methods of hot target detection?

1.5 Hypotheses

In order to evaluate the methods proposed in this work, several hypotheses must be formed. The following subsections will briefly go into more detail on the hypotheses about method performance.

1.5.1 Ground cover analysis

If the methods proposed in this work are effective, they should outperform existing methods of ground cover analysis.

1. **Hypothesis 1.** Accuracy and precision are higher using the methods proposed in this work than existing methods of ground cover segmentation.
2. **Hypothesis 2.** Accuracy is higher using the methods proposed in this work than existing methods of ground cover estimation.

1.5.2 Hot target detection

If the methods proposed here can be applied effectively to hot target detection, we should see an improvement here as well.

1. **Hypothesis 3.** Accuracy and precision of hot target detection is higher using the methods proposed in this work than existing methods of hot target detection.

In the following chapter, we will go into further detail on the data and methodology applied to verify these hypotheses.

Chapter 2

Data

We have applied machine learning for image segmentation in the ground cover domain as well as the hot target domain. Therefore we must investigate the data of each of these categories. The following sections describe the data used for both of these domains.

2.1 Ground cover analysis

To train and evaluate our machine learning methods for ground cover analysis, we use the extensive labelled dataset produced in the framework of two projects by NWO through an Applied Research Fund (ARF) [14], and the STARS project [17], a research consortium consisting of the Bangladesh Institute of ICT in Development (BIID) [2]; the International Maize and Wheat Improvement Center (CIMMYT) [4]; and the Geo-Information Science and Earth Observation (ITC) [7]. The STARS project is funded by the Bill and Melinda Gates Foundation. The NWO projects are funded by the Dutch government.



FIGURE 2.1: The three types of crops present in this dataset. Mung bean (left); Maize (middle); and Wheat (right). Notice the common obstacles for good segmentation: Shade casting, hard lighting, overlapping leaves, and residu on the ground.

An 8GB dataset consisting of 2564 images of varying quality and dimensions (around 2000x1500 pixel JPEGs) was hand-annotated by the team in Bangladesh using CAN-EYE imaging software [3]. This dataset is the basis of our ground cover research. Using the CAN-EYE software, the Bangladesh team extracted canopy structure characteristics, such a Leaf Area Index (LAI), and Vegetation cover fraction (FCOVER). Various plant species, including Wheat, Maize, and Mung bean, are photographed (see Figure 2.1 for examples of these crop types). Per plot, around 9 to 10 images are taken in a square formation. Some small overlap is present in these images, but these do not cause any problems for our proposed method since

all 9 images are annotated. This is the ground-truth data to which compare our method. The original dataset consisted of 2801 images, and but was reduced to the number mentioned previously, due to the pruning of corrupted and identical images. After pruning, preprocessing was applied to normalize the RGB images. Some images in the dataset were taken in portrait mode, while others were taken in landscape orientation. To ensure easier processing, we converted all images in the dataset to landscape mode by rotating the subset taken in portrait orientation. Figure 2.3 shows a sample of this dataset. Further statistics can be found in Table 2.1. From here on out, we will refer to this dataset as CROPS.

Type	Number of samples	Fraction
Maize	925	36.076%
Wheat	684	26.677%
Mung bean	955	37.246%
All	2564	100.0%

TABLE 2.1: Statistics of CROPS dataset.

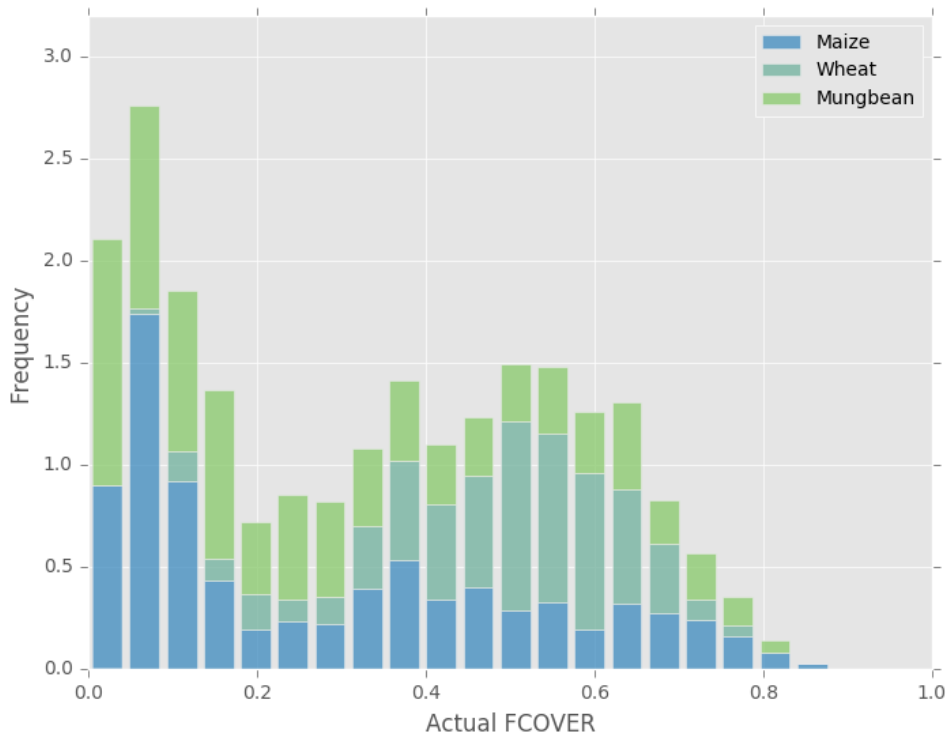


FIGURE 2.2: Frequency of FCOVER value occurrence per plant type.

An additional smaller dataset was provided that contains more complex images of plants with more crop residual, and with phone type annotations. This dataset, which we will refer to as CROPS.PHONES, contains 239 images of variable plant types. Table 2.2 lists the smartphone types present in this dataset, and the number of images belonging to these

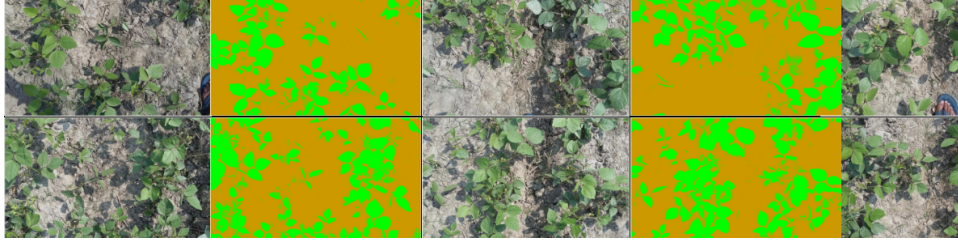


FIGURE 2.3: Sample data of vegetation RGB imagery taken using a smartphone, and hand-annotated ground truth data using CAN-EYE software [3].

types. The CROPS_PHONES dataset will be used to determine which type of smartphone is best suited to take photos to be used for ground cover analysis. The CROPS dataset was created using a single type of Samsung smartphone, while this dataset was created using a range of cheaper smartphones. The goal of the analysis on the CROPS_PHONES dataset is to evaluate the impact of the quality of the camera of a large number of smartphones.

Smartphone type	Number of samples	Fraction
Prime	43	17.992%
Hawei	13	5.439%
Lava	45	18.828%
Nikon	25	10.460%
Okapia	35	14.644%
Symphony	37	15.481%
Tab	41	17.155%
All	239	100%

TABLE 2.2: Statistics of CROPS_PHONES dataset.

2.2 Hot target detection

For hot target detection we use multispectral Landsat 8 imaging data obtained from open-access portals, such as Amazon AWS [9]. Data from these portals can be used in combination with search portals such as the USGS Earth Explorer [5] and news sources to easily locate and obtain imaging data from large and small wildfires. We use all multispectral data as input for our machine learning methods, except OLI band 8, since the spectral information in this panchromatic band is already captured by bands 2, 3, and 4 (see also Figure 1.2). Although the Near Infrared (NIR) and Short-wave Infrared (SWIR) bands are commonly used in hot target detection, the other bands, including the Thermal Infrared Sensor (TIRS) bands, can provide additional information. The machine learning methods may pick up on more subtle relations between irradiative energy output on different wavelengths.

TOA reflectances 1.2 (corrected for solar angle using Equation 2.1 as described by [19]), and TOA radiance 1.3 from all used bands will be used as input for our network, including an 8-neighbourhood window. For the 11-band Landsat 8 scenes, excluding OLI band 8, this would result in an input layer of 10x3x3 neurons.

$$\rho\lambda = \frac{\rho\lambda'}{\cos(\theta_{SZ})} = \frac{\rho\lambda'}{\cos(\theta_{SE})}$$

with

$$\rho\lambda = \text{TOA planetary reflectance} \quad (2.1)$$

$$\theta_{SE} = \text{Local sun elevation angle}$$

$$\theta_{SZ} = \text{Local solar zenith angle}$$

However, to be able to use our machine learning models, we need annotated data. Meaning that beside Landsat or Sentinel scenes, we require the exact positions of hot target pixels in these scenes to be able to allow the network to learn which pixels are hot targets. The size of academic literature on hot target detection algorithms is relatively small, and annotated datasets hard to come by. This makes evaluation of any new method quite labour intensive and error-prone. We augment our data to simulate hot target pixels to generate training data for our network, by extracting obvious hot target pixels from Landsat scenes and incorporating these features in a new dataset. This method however, might cause the machine learning methods to overfit, and we lose the generalization property of machine learning. We have therefore reached out to the authors of [81] with the request of sharing their evaluation set consisting of 45 Landsat 8 OLI scenes, manually annotated by human analysis. They have kindly allowed us to use this dataset for our research. In this dataset, there are five Landsat 8 scenes per geographic region, including radiance and TOA reflectance maps, and pixel coordinates of hot targets. We use this dataset as an evaluation benchmark, together with the simulation dataset. From here on out, we will refer to this dataset as FIRES. An example scene of the FIRES dataset is provided in Figure 2.4.

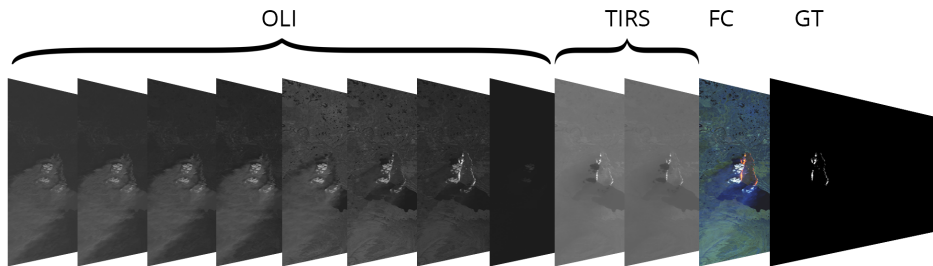


FIGURE 2.4: Example data of the FIRES dataset by Murphy et al. [81] used in this research. Imagery was taken in the vicinity of Alaska, USA (LC80690182014140LGN00). Shown are OLI bands 1 to 9 (excluding Panchromatic band 8), TIRS bands 10 and 11, a False Colour (FC) composite image, and the binary Ground Truth (GT) labelling.

Chapter 3

Method

In the following sections we will expatiate on the various methods presented in the previous chapter and describe how they are implemented and applied.

3.1 Ground cover analysis

First, we will discuss the methods applied for ground cover analysis. These range from colour indices to machine learning methods. Some methods can also be applied for segmentation, from which we can then infer a ground cover estimate. Other methods are suited for ground cover estimation only, for example the Support Vector Regression algorithm. It is important to realise that this direct way of estimating ground cover fraction avoids errors spawning from suboptimal segmentation. This noise, in essence, is not a factor. Although one can argue that estimation without segmentation is an easier task, this is not always true. Thresholded colour indices methods are simply not able to do estimation without first doing segmentation. On the other hand, regression algorithms are not suited for segmentation. Both methods of analysis have their own difficulties and require different ways of approaching them.

3.1.1 Colour indices

To obtain a score for the colour indices methods, we implement all algorithms listed in section 1.1.1.1, namely NDI, ExG, ExR, ExGR, MExG, CIVE, VEG, COM1, COM2, and CANO(PEO), and apply them to our CROPS dataset. Since these algorithms are basic colour space transformations, they require a threshold to do obtain a binary segmentation map. For ExGR the threshold is set at 0, in accordance with literature. For VEG and MExG, we use the mean value of the resulting image as threshold [53]. For the other algorithms, we apply Otsu thresholding [85]. Otsu thresholding is based on a foreground/background histogram and calculates the ideal value to separate the plant and non-plant classes by maximizing the inter-class variance. For ExR and CIVE, due to their formulation, the Otsu threshold [85] is taken as an upper bound. For the others, NDI, ExG, COM1, and COM2, this threshold is a lower bound. CANO's formulation already ensures the generation of a binary map. We leave the P_1 , P_2 , and P_3 parameters to their default values as reported in the CANOPEO paper [88].

Applying these methods results in a binary segmentation map of the input image. From these binary images, we can determine the estimation and segmentation errors. The estimation error is determined by calculating the percentage of plant pixels in the binary map, and comparing this to

Index	Formula
NDI	$128 * ((\frac{G-R}{G+R}) + 1)$
ExG	$2g - r - b$, with r , g , and b the percentage of R, G, and B pixels respectively (normalized).
ExR	$1.3R - G$
ExGR	$ExG - ExR$
MExG	$1.262G - 0.844R - 0.311B$
CIVE	$0.441R - 0.811G + 0.385B + 18.78745$
VEG	$\frac{G}{R^a \cdot B^{1-a}}$ with $a = 0.667$
COM1	$ExG + CIVE + ExGR + VEG$
COM2	$0.36ExG + 0.47CIVE + 0.17VEG$
CANO	$R/G < P_1$ and $B/G < P_2$ and $2G - R - B > P_3$ with $P_1 = P_2 = 0.95$, and $P_3 = 20$

TABLE 3.1: Mathematical notation of different colour indices.

the ground truth FCOVER. The segmentation error can be determined by evaluation of the True Positives, True Negatives, False Positives, and False Negatives (c.f. sections 3.4, and 3.5).

3.1.2 AP-HI

We also use the method proposed by Yu et al. [121] for evaluation, due to their high reported performance. After reaching out to the researchers, they were kind enough to share their MATLAB implementation [1]. The pipeline was easily adapted to be able to run on our CROPS dataset. The AP-HI algorithm produces a binary segmentation image, which can be evaluated in the same way as the colour indices methods described in the previous section.

3.1.3 SVM

Using a regular SVM to do pixel-wise classification is difficult due to the large size of our dataset. However, we can implement a Support Vector Regression (SVR) algorithm to do regression using our ground truth FCOVER values. One obstacle we find is that our input images are quite large. If we opted to use all information in the 8-bit RGB images, we would obtain a feature vector of over 9 million features (an R, G, and B value for each pixel). This is too large for our SVR, and unnecessary to boot. Clearly, we need something better to be able to compare our method to an SVM implementation.

To deal with the large size of our feature vector, we instead use the percentage of each R, G, and B value in the 8-bit image. Initially, this increases the feature vector length to 256^3 , but we apply some additional restrictions. To be able to predict the amount of crops, we do not need the precision of

256 values per colour. We can reduce the feature vector space by converting the initial 8-bit colour space to a smaller one. For example, we can cast each R, G, and B value to a value between 1 and 4. This reduces our feature vector length to a manageable 4^3 . An example of such a conversion is shown in Figure 3.1. In our experiments, we use a colour range of 4. This number was determined by an exhaustive search over an array of colour ranges ranging from 1 to 256. We show the results of these experiments in section 4.1.2.2 of our Results chapter. As described in the example above, this results in a feature vector of 4^3 features.

Obviously, detail is lost when doing these kinds of conversions. This can be a good effect since it can decrease noise in the image, but if we take a colour range that is too narrow we may see detrimental effects. We run additional tests to highlight the effect of the size of the colour range in this colour range conversion.

We initialize our classifier with a radial basis function (RBF) kernel. The SVMs hyperparameters were experimentally tuned, the results of which can be found in section 4.1.2.2. 10-fold cross validation is performed to ensure optimal use of training data, and to reduce the variance of our predicted FCOVER values. The dataset is randomized, after which it is partitioned into 10 equal parts. 1 part is used as the evaluation set, and the other 9 parts are used together as training set. This process is repeated 10 times, each time using a different partition as evaluation set. The resulting mean errors are averaged to obtain a Mean Average Error.

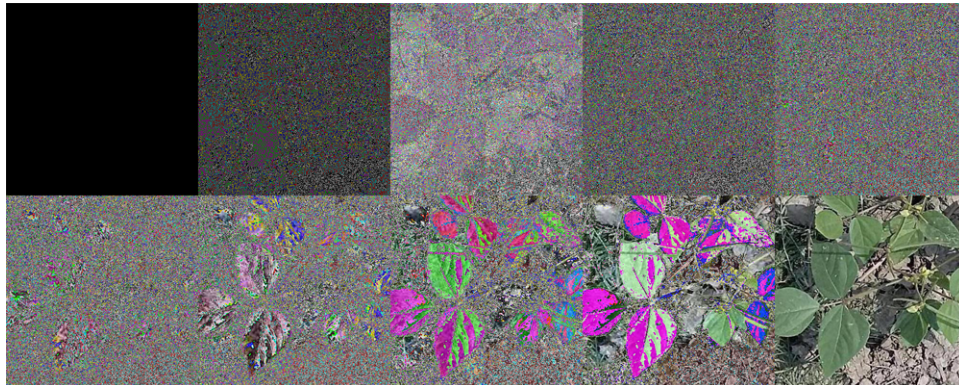


FIGURE 3.1: Sample data downsampled into smaller colour ranges, and upscaled again for visualization purposes. Colour ranges from top left to bottom right: 1, 2, 3, 4, 8, 16, 32, 64, 128, 256 (original).

3.1.4 Random Forest

We also implement a Random Forest classifier to use as benchmark for our deep neural network method. Similar to our SVM method, we convert our feature vector by calculating a colour fraction and decrease the colour range to 4. In accordance with literature [84], we generate 100 decision trees per cross-fold iteration. In preliminary experiments we saw diminishing returns when using a larger number of trees. Bootstrapping is used to reduce variance. No restrictions are set on the number of leaf nodes, the decision tree depth, or the maximum number of features. 10-fold cross validation is applied here as well, in the same manner as with the SVM method. We also inspect the influence of the colour range size when using an RF classifier.

3.1.5 Deep neural network

Convolutional neural networks have shown great promise in classifying image data using annotated training data [111] [92]. This is why we research the use of this type of deep neural network architecture. U-net [94] is a convolutional architecture that obtains state-of-the-art results (better than, for example, sliding window conditional random field CNNs) on biomedical segmentation and allows for pixel-wise classification. We use this architecture as the basis of our experiments, and must therefore go into further detail on its inner workings in this section. We also implement a more simple multilayer perceptron (MLP) network architecture, and run it on our converted feature vectors, as described in section 3.1.3. The side effect of this conversion is that we are not able to do segmentation, but are forced to treat ground cover estimation purely as a regression problem. We vary our the number of layers and nodes in smaller experiments to obtain the most optimal parameters for this problem. We employ stochastic gradient based weight optimization as presented by Kingma et al. [64], due to it's capacity for large datasets, computational efficiency, and low memory requirements.

3.1.5.1 U-net

The U-net architecture consists of two pathways. First, a contracting, down-sampling pathway is used, that includes 2 unpadded convolutions (using ReLu as activation function) followed by a max pooling operation with a stride of 2 for downsampling. Then, an expanding, upsampling pathway is used, that includes an up-convolution followed by crop and merger step, followed by 2 convolutions (again using ReLu as activation function). The upsampling pathway uses the high resolution feature maps generated in the contracting pathway for localization in combination with the upsampled output. A full overview of the U-net architecture can be found in Figure 3.2. We use an adapted version of the MXNet [11] implementation provided by de Wit [18]. Our network architecture consists of 23 convolutional layers. The final step is a 1 by 1 convolution and Otsu thresholding to produce a binary segmentation map. In our experiments we reduce the number of features to 10 in the first convolutional layer, and increase the amount further down the pipeline (up to a 3x3x80 convolution). We employ Logistic Regression as our loss function. The complete DNN architecture used in our experiments can be found in Appendix A.

Weight initialization is very important for DNNs, especially in deeper networks with a large number of layers. To ensure that each layer is used in the output, and not starved, weights should be calibrated so that the feature maps have roughly unit variance [94]. In an architecture such as U-net, this is done through random weight initialization by drawing from a Gaussian distribution with $\mu = 0$ and $\sigma = \sqrt{\frac{2}{N}}$ with $N = \text{Number of incoming nodes}$ for a neuron.

To deal with the high-resolution data, we split the input into 256x256 pixel tiles, and train/test the network on these samples. The input for the network therefore consists of an 8-bit 3-channel 256x256 pixel RGB image. The output will be a single channel 256x256 pixel binary map. Since we cannot divide the original image into tiles of this shape perfectly, some overlap is apparent. It can be argued that the overlap will introduce some overfitting, but this is overfitting is marginal. The samples are augmented

by performing 5 independent semi-random elastic transformations on the original image, and 4 rotations. This boosts the size of our training set, and decreases variance. This results in a total set of 5,101,076 input images, to be split into training and validation data using K-fold cross validation.

We also train the net using scaled images where no transformations are performed. This has the added benefit of negating any imprecise annotations by downsampling the errors together with the image, while the FCOVER remains the same by definition. By downsampling the original images using a nearest neighbour resampling filter, and applying the same transformation to our ground truth images, we obtain input images of 256x194x3 pixels. We verified that the FCOVER does not change significantly upon downsampling, and found that the mean difference in FCOVER between the original and scaled down version is around 0.0009512. This corresponds to a 0.9512 percentage point difference, which we deem acceptable for our experiments, since we use the scaled down FCOVER in these cases.

We will vary the networks hyperparameters, such as number of layers, and number of features, as part of our experiment. Starting with the architecture described above, we will adjust the hyperparameters for maximum performance. Since the literature on the subject of tweaking deep learning architectures does not provide an optimal strategy for this, we vary the parameters until we see diminishing returns. In similar manner we also evaluate the effect of Dropout at different positions, since this has been shown to increase performance in DNNs (see Section 1.3.0.3).

In addition to a single network for all plant types in our CROPS dataset, we will also evaluate the DNNs performance by training it on each type individually. This will result in three networks - a Wheat, Maize, and Mung bean segmentation network. We believe this might give a performance boost.

Finally, we will evaluate the DNN on our CROPS_PHONES dataset, to assert which smartphone is best suited to create phones for our method of ground cover analysis. We will scale the dataset as described above in the same way and use our default DNN architecture.

3.2 Hot target detection

In this chapter we also discuss the methods applied for hot target detection. Since we have already outlined the current state of hot target detection research in previous sections, we will only briefly touch on their implementation in the following sections. Then we will go into more detail on the machine learning methods proposed in this work.

3.2.1 HOTMAP

HOTMAP, the algorithm proposed by Murphy et al. [81], depends on a number of parameters, mainly the α and β parameters. They can be described as boolean mapping functions, taking a multispectral image (converted to TOA reflectances using Equation 1.2) as input and outputting binary maps. We show their formulas in Equations 3.1 and 3.2.

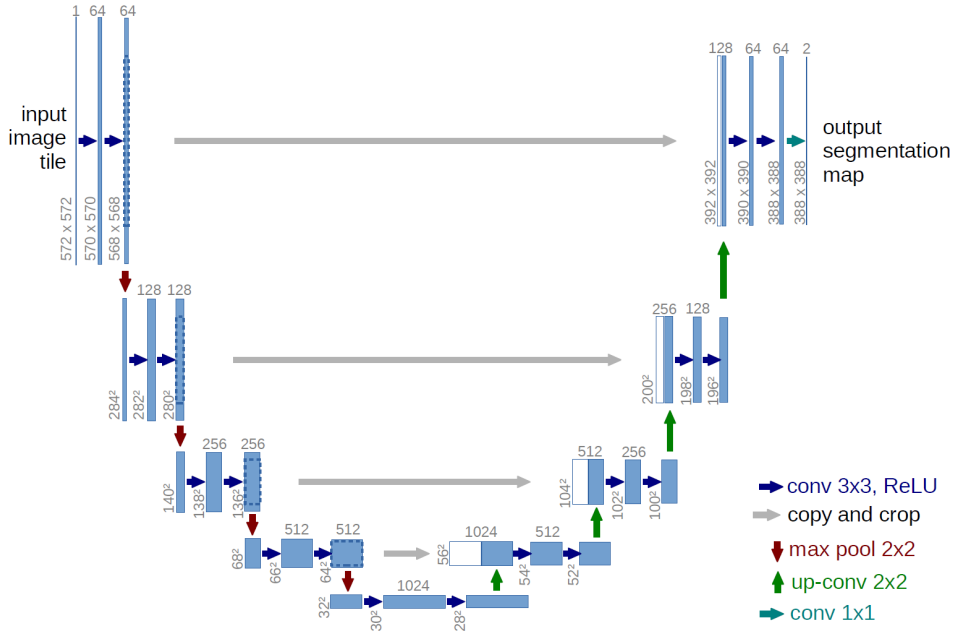


FIGURE 3.2: U-net architecture. The blue boxes represent multi-channel feature maps. The number of channels is denoted at the top of the boxes, and the shape data is denoted at the bottom. The white boxes in the upsampling pathway correspond to the copied feature maps. Arrows denote various layer operations (see legend). Image taken from Ronneberger et al. [94].

$$\alpha = \frac{\rho_7}{\rho_6} \geq 1.4 \text{ AND } \frac{\rho_7}{\rho_5} \geq 1.4 \text{ AND } \rho_7 \geq 0.15 \quad (3.1)$$

with

ρ_n = Top of Atmosphere reflectance in band n

$$\beta = \left(\frac{\rho_6}{\rho_5} \geq 2.0 \text{ AND } \rho_6 \geq 0.5 \right) \text{ OR } S_6 \text{ OR } S_7$$

with

ρ_n = Top of Atmosphere reflectance in band n

S_n = Top of Atmosphere reflectance is saturated in band n

(3.2)

The α and β maps are merged using a simple logical OR operator (α OR β), after which a clustering algorithm generates clusters of hot target candidates. Clusters that do not contain at least a single α pixel are discarded. Remaining clusters are labelled as hot targets.

The α parameter is designed specifically to produce as little false alarms as possible, but to detect at least a single pixel in a fire cluster. Its precision means that it can sometimes miss hot target clusters (especially on saturation of band 7 with very hot targets), which is why the β parameter is used in combination with the α parameter. The β parameter is used to detect the particularly hot targets, for example in cases of saturation of bands 6 and 7. The β map will have good accuracy, but can also produce a large number of false alarms. When used together, the parameters provide effective hot target detection.

We have implemented the algorithm to verify its effectiveness on the dataset. We will compare our results to the results reported in [81] considering we use the FIRES dataset as well.

3.2.2 SVM

We also implement an support vector approach for hot target classification. We initialize our classifier with a radial basis function kernel, with kernel coefficient $\gamma = \frac{1}{n}$ with n =Number of features, in accordance with our feature vector size. The other hyperparameters are tweaked as part of our experiment. Each pixel is used as a data point, consisting of 10 features, being the combined OLI and TIRS channels. The binary labels are used for ground truth validation. To ensure a balanced dataset, we use all hot target data points, and use the same amount of non-hot target data points. The rest of the non-hot target data points are discarded. We randomize the data before we apply pruning. This results in a total size of 43546 data points (with 21773 hot target pixels). Using 10-fold cross-validation we utilize the entire dataset, each fold being employed as validation set once.

3.2.3 Deep neural network

We use the same architecture reported in the previous section, but adapt it to be applicable for multispectral images. Instead of a 3-channel 8-bit RGB input, we use 10-channel 16-bit multispectral images. The output (and concurrently validation data) remains the same, a binary segmentation map the size of the original input image. We again employ data augmentation (rotations, and elastic transformations) to generate more training data, and so allow the network to encounter a broader range of data.

We also train an MLP using the feature vectors described in section 3.2.2, with *Adam* [64] stochastic gradient based weight initialization. The MLP's hyperparameters are tweaked as part of the experiments.

3.3 Evaluation

Root mean squared error, as described in Equation 3.3, will be used as a measure of ground cover estimation. Additionally, we determine accuracy (defined by Equation 3.4), precision (defined by Equation 3.5), and algorithm classification run-time. Omission (missed ground cover or hot target pixels) and commission (false alarm) errors are often reported numbers in relevant literature. We use accuracy and precision to easily compare our method to state-of-the-art results [56] [81] [101] of ground cover and hot target segmentation.

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{N}}$$

with

$$RMSE = \text{Root Mean Squared Error} \quad (3.3)$$

N = Number of predictions

y_i = Actual value for data point i

\hat{y}_i = Predicted value for data point i

These numbers are obtained through comparison with our labelled dataset. The evaluation set by Murphy et al. [81] contains hand-annotated hot target data. It is hard to verify by hand which pixels are hot and which are not, meaning that every evaluation dataset annotated by human experts contains an uncertainty factor. Therefore we must note that these figures are only relative to the evaluation as performed by a human expert. This complication can partly be alleviated by creating a simulation dataset, as is our aim in this work. Murphy et al. [81] also report numbers of *associated* false alarms, which are the false alarm pixels that are part of a detected hot target cluster (compared to false alarms that are not inside a cluster). This also helps to distinguish between truly missed hot targets, and targets missed (possibly) due to the subjectivity in annotating the evaluation dataset.

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

with

$$TP = \text{Number of true positives} \quad (3.4)$$

$$TN = \text{Number of true negatives}$$

$$FP = \text{Number of false positives}$$

$$FN = \text{Number of false negatives}$$

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

with

$$TP = \text{Number of true positives} \quad (3.5)$$

$$FP = \text{Number of false positives}$$

We compare our ground cover method with the colour indices methods mentioned in section 1.1.1.1, the AP-HI method by Yu et al. [121], an SVM (SVR) implementation, and a Random Forest implementation. K-fold cross validation is performed where our data allows it to effectively use the dataset for both training and evaluation. Wherever we can, we will distinguish between different crop types: Maize, Wheat, and Mung bean. Our hot target detection method is compared to the findings of Murphy et al. (HOTMAP) [81] and Giglio et al. (CA) [49].

In this chapter we have presented the type of data we use in our experiments, the methods that are currently being used to perform ground cover analysis and hot target detection, the methods we will implement as part of our research, and the metrics we employ to evaluate these methods. The following chapter is dedicated to the results of this evaluation.

Chapter 4

Results



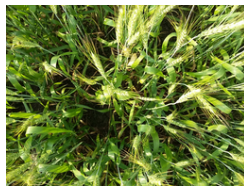

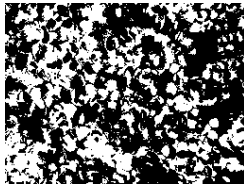
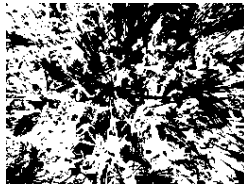
In this chapter, we present the results of our work. First, we will report the results of our ground cover analysis experiments. Starting with ground cover segmentation, we will continue with ground cover estimation, since methods that are able to do segmentation can also be used for estimation. Then, we will present the results of our hot target detection analysis using deep neural networks.

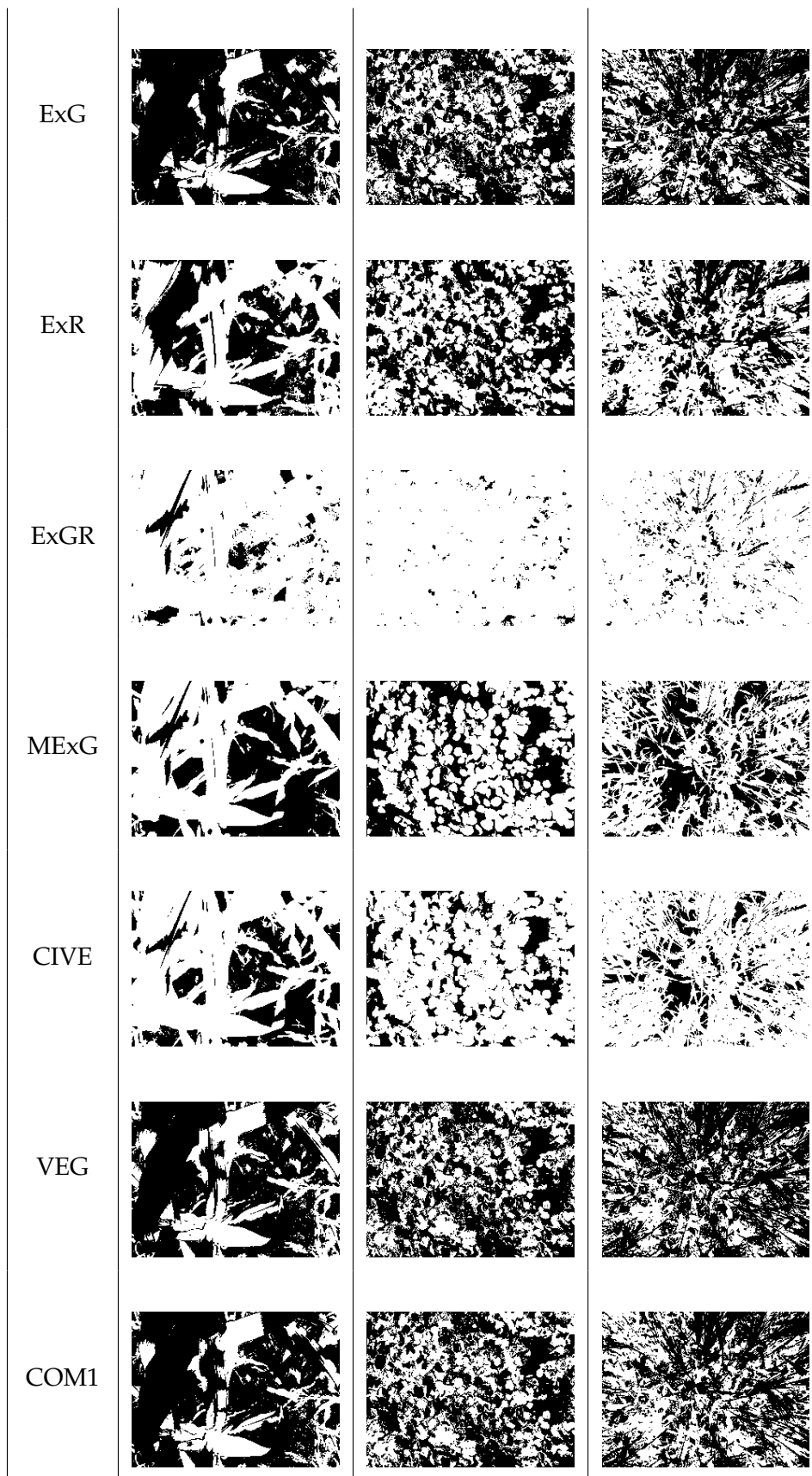
4.1 Ground cover analysis

We have implemented a number of different methods to be used as benchmark for our machine learning methods. For some of these methods, we have conducted additional experiments to gauge the effects of certain parameters on the resulting segmentation or estimation. We set out with the exposition of these results, and reserve the findings of our primary experiment - the deep learning method - to latter sections (4.1.1.3).

4.1.1 Segmentation

There are a number of ground cover analysis algorithms that can be used to generate a binary segmentation map. The colour indices methods coupled with a threshold, the AP-HI algorithm, and our deep neural network, all have this capacity. We present examples of their segmentations for the three types of crops in Table 4.1, and will discuss their individual advantages and disadvantages in the following sections.

Method	Maize	Mung bean	Wheat
Original			
NDI			



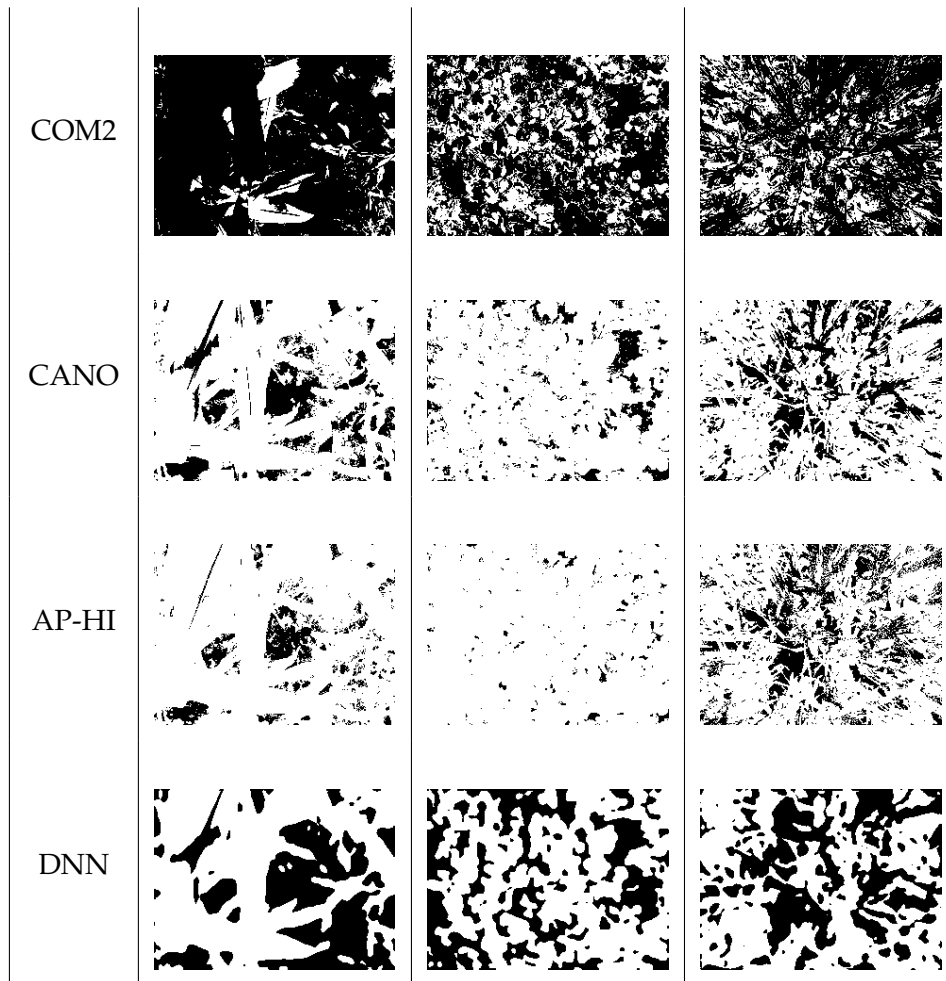


TABLE 4.1: Example segmentations of Maize, Mung bean, and Wheat crops for the given segmentation algorithms (scaled images).

We evaluate the segmentations using the methods described in Equations 3.4 and 3.5, using RMSE, and precision and accuracy scores. Overall, Maize seems to be the easiest crop type to segment using these methods. Figure 4.2 shows the algorithms' precision plotted against their accuracy, split on crop type. The algorithms are able to score highest on Maize images, and lowest on Wheat images. We see that DNN outperforms all conventional methods, although there are instances where another method has a slight edge on the DNN - for example COM2 shows a minor improvement in accuracy for Wheat images. We see that in most cases, RMSE scores correlate with accuracy and precision scores. However, this is not always the case. For instance, we can observe that COM2 has quite a high precision score for Maize images, but the corresponding RMSE is contrastingly relatively high. This discrepancy is due to COM2 underestimating Maize images quite a lot (which, in turn, is caused by the index not being able to recognize very light leaf areas). Since COM2 does not classify many pixels as plant pixels, it does not generate many false positives either.

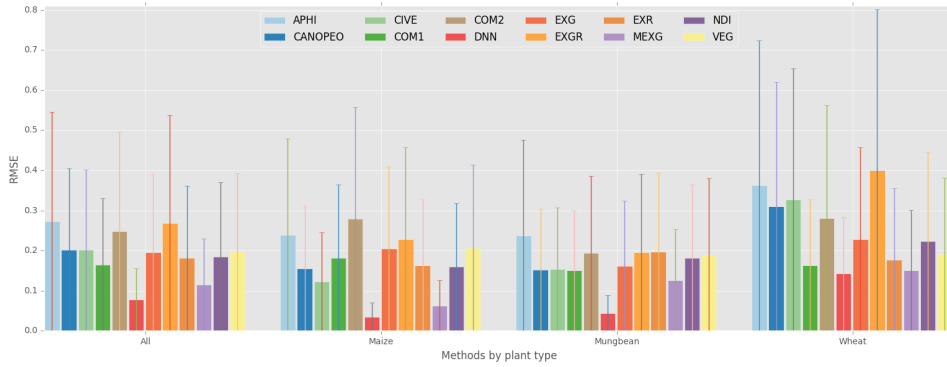


FIGURE 4.1: RMSE and standard deviation of scaled images for ground cover estimates for all segmentation methods.

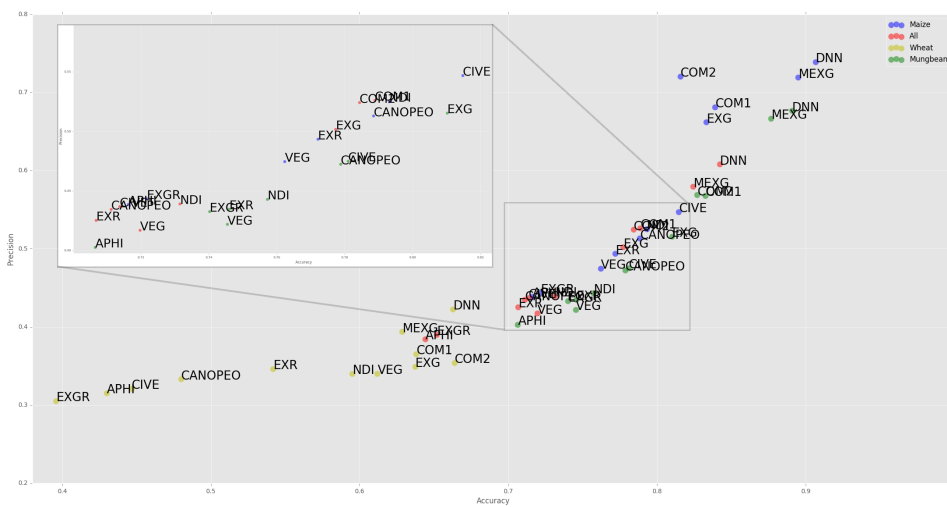


FIGURE 4.2: Precision and accuracy scores for methods of ground cover segmentation of scaled images. Full resolution image available in Appendix C.

4.1.1.1 Colour indices

The colour indices methods we implemented provide varying segmentation results. Depending on the method, we see the detrimental effects of changing lighting in the leaves, clutter on the ground, and shadows cast by other leaves. First we will consider the scaled images, because we this provides us with a fairer comparison with the DNN method. We see MExG performing best overall. ExG, NDI, VEG, COM1, and COM2 have the biggest trouble with hard lighting changes, as can be seen in the examples above. The other indices, ExR, ExGR, MExG, CIVE, and CANO, overcome this problem. However, this feature also makes them tend to overestimate the crop area, as can be seen in the Wheat examples most prominently due to the density of the crops in this type of vegetation. Their ability to pick up light surfaces has the downside of not being able to distinguish between darker surfaces adequately.

Striking is that the colour index that performs best is surprisingly simple. The MExG index is a weighted linear combination of the R, B, and G planes. We find that MExG only really struggles when dealing with very

hard lighting, and near-white surfaces. An example of this can be found in the Maize column in Table 4.1, in the area on the large leaf on the left side of the image. We see the error rate increase in images of Wheat especially, since the spikelets are highly reflective. The high blue values in white pixels cause problems for this index. Red values are relatively high even in green leaves, which is why MExG penalizes them strongly. Since colour indices methods do not take into account other information than the R, G, and B values of a single pixel, they cannot infer that white pixels with high RGB values actually belong to a leaf element in the image, rather than background noise.

The colour indices that perform the worst, most notably ExGR, drastically underestimate the vegetation area, due to their inability to deal with big changes in leaf colour when the light conditions are extreme. On the other hand, ExGR sometimes labels shadows as being part of the plant, resulting in an overestimation of the FCOVER.

We must also review the results of the segmentations of our original images. When looking at Table B.1 we see results that mimic the results for the scaled images, with a few notable exceptions. COM2, ExR, and NDI each show a much higher RMSE value. This is peculiar, given the fact that these indices do not rely on other information beside the RGB values for a given pixel (disregarding normalization). The index itself therefore, should not change by a large margin. After further (visual) inspection, we find that this is indeed the case. The only factor affecting the resulting segmentation map is the thresholding step. We see that, indeed, Otsu thresholding does not provide an optimal threshold value for most of the segmentation maps generated by these indices. This is due to the intensity level histogram for these indices for the original image being more skewed, as can be seen in Figure 4.3.

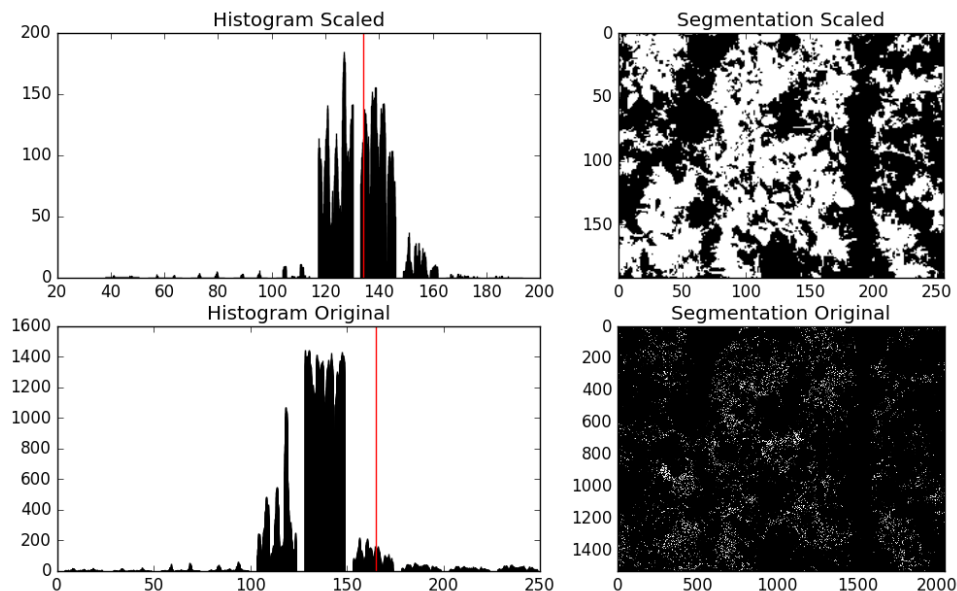


FIGURE 4.3: Histogram of the intensity levels of the NDI index for an original Mung bean image and downscaled version, and corresponding segmentation maps (Found Otsu threshold in red).

4.1.1.2 AP-HI

The AP-HI method as proposed by Yu et al. [121] performs relatively well for the original images, but not quite as well as, for example, MExG or the DNN. Especially when segmenting images of Wheat plants, it lacks the power to identify the yellowish, lighter spikelets on the top of the plant as being part of the plant. The AP-HI method, like some of the colour indices, struggles to appropriately classify strong highlights, albeit to a much lesser degree.

The resulting segmentations of the scaled images paint a slightly different picture. AP-HI performs somewhat worse on our scaled CROPS dataset, as it does on the original set. We believe that due to the decrease in pixel availability in smaller images, the sample size of the hue intensity lookup table decreases as well. In addition, the affinity propagation clustering algorithm has less data to work with, leading to non-optimal results.

4.1.1.3 Deep neural network

We will look at the results of our convolutional neural network using the original images and the scaled images, as we did with the other segmentation methods. Since the DNN cannot handle very high resolution input images, we split our original input images into separate blocks of 256x256 pixels (and augmented our data using a number of transformations), as described in Section 3.1.5. These results are displayed next to the other segmentation methods in Figure B.1. The DNN did not perform well here, when considering RMSE. A closer look at the DNNs segmentations compared to the original image and ground truth is provided in Figure 4.4.



FIGURE 4.4: Comparison of cropped original image (left), extracted ground truth (middle), and resulting DNN segmentation map (right).

What is apparent is that our ground truth is not always the best. We touched on this briefly in previous sections. Therefore, the comparison we make with the ground truth has its disadvantages. We will elaborate on this issue more extensively in Section 5. However, the suboptimal labelling of our ground truth images is not the only source of misclassification. In many cases we observe that the DNN overestimates the plant region, which could be caused by a combination of the input images being rather small, and the large variety in lighting conditions in our CROPS dataset. We do observe the potential of the DNN to overcome intra-leaf lighting changes. We did not train separate networks for each plant type for the original image, but this could potentially lead to improvements in DNN segmentation.

When we investigate the resulting segmentation map of our scaled images, we see that the convolutional neural network provides segmentations that overcome the problems that hinder conventional colour indices methods. Comparing the DNN to the other methods we clearly see that DNN outperforms conventional colour indices methods, and the AP-HI method. Not only is it the best performing segmentation method overall, we also see top performance for each individual plant type. Especially for Maize and Mung bean we can observe exceptional performance. The DNN obtains similar performance to the well performing colour indices methods for the Wheat type.

The DNN is able to handle extreme changes in lighting conditions, more so than conventional methods. This is due to the added information of other pixels surrounding the target pixel. The convolutional layer helps in learning these gradual and abrupt hue and intensity differences in a leaf. This does lead to rounded edges in most segmentation maps, as can be seen in Table 4.1. In Mung bean images, this is an advantage since the leaves are relatively round themselves. In Maize, but especially in Wheat, this leads to misclassification of a large number of pixels, but the DNN is still able to outperform the other methods.

When looking at the precision and accuracy of the DNN scaled segmentations, we again see that it outperforms all methods. Per individual plant type, there is only one other method that obtains better accuracy: COM2 on Wheat images. The combination of good accuracy and precision, and a low mean error provide solid evidence for our hypothesis.

We have trained a array of different network setups. The DNN results shown in Figure 4.1 show the most optimal performance of our method, which corresponds to three separately trained networks for each plant type. Initially, we trained a single network on our CROPS dataset, which resulted in the scores listed in Table 4.2. The largest improvement in RMSE is visible in Maize imagery (16.029%), followed by Wheat (13.914%). There is a 10.121% improvement in Mung bean imagery.

Network	All	Maize	Mung bean	Wheat
Single	0.0901 (0.0175)	0.0418 (0.0113)	0.0494 (0.0046)	0.1653 (0.0240)
Typewise	0.0779 (0.0193)	0.0351 (0.0053)	0.0444 (0.0040)	0.1423 (0.0351)

TABLE 4.2: RMSE results (standard deviation) of a single net versus the scores of the three separately trained networks.

As part of our experiments, we varied the number of features for our convolutional layers. We observed a minimum in our error rates at a 10-channel feature map in the first layer. Adding more layers did not increase performance significantly. Dropout was tested at several locations, since it is difficult to gauge where they are most effective, and due to the dropout layer affecting the entire network (not simply the layer that succeeds it). Adding dropout layers directly after the copying and merger of the feature maps from the downsampling pathway showed good results, but the

best results were obtained when adding dropout layers directly after each upsampling convolutional layer. We show the results for various dropout layer positions in Table 4.3. The full architecture can be found in Appendix A.

Dropout type	All	Maize	Mung bean	Wheat
de Wit [18]	0.0779 (0.0193)	0.0351 (0.0053)	0.0444 (0.0040)	0.1423 (0.0351)
Only in up-pathway	0.0798 (0.0157)	0.0393 (0.0080)	0.0566 (0.0049)	0.1379 (0.0262)
Alternating in up-pathway	0.0894 (0.0171)	0.0623 (0.0064)	0.0784 (0.0073)	0.1316 (0.0310)
Additional in down-pathway	0.0923 (0.0145)	0.0677 (0.0067)	0.0885 (0.0093)	0.1251 (0.0248)
After up-convolution	0.0702 (0.0105)	0.0360 (0.0072)	0.0578 (0.0076)	0.1144 (0.0142)

TABLE 4.3: The effect on RMSE (standard deviation) of varying ways to apply dropout.

In other experiments, we evaluated the segmentation of more complex images. The images in our CROPS.PHONES dataset were made with a number of different smartphones (and tablets), each with its own resolution and image quality. We show the RMSE for all phone types in Figure 4.5. We will discuss these results further in the next section.

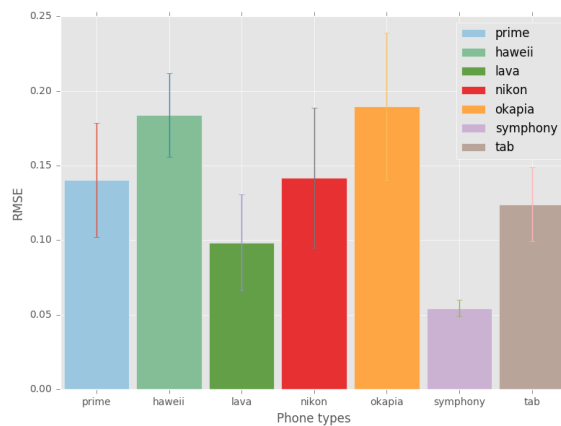


FIGURE 4.5: Comparison of RMSE for different smartphone types.

4.1.2 Estimation

We have explained how segmentation algorithms can also provide an estimation of ground cover. A number of algorithms are able to provide an estimate of the percentage of vegetation in our images without creating a binary map first. They are machine learning methods that are trained on the ground truth cover fraction. Together with the segmentation methods, we

show their estimates here. Figure 4.6 shows the RMSE for each regression method, and the DNN (on scaled images).

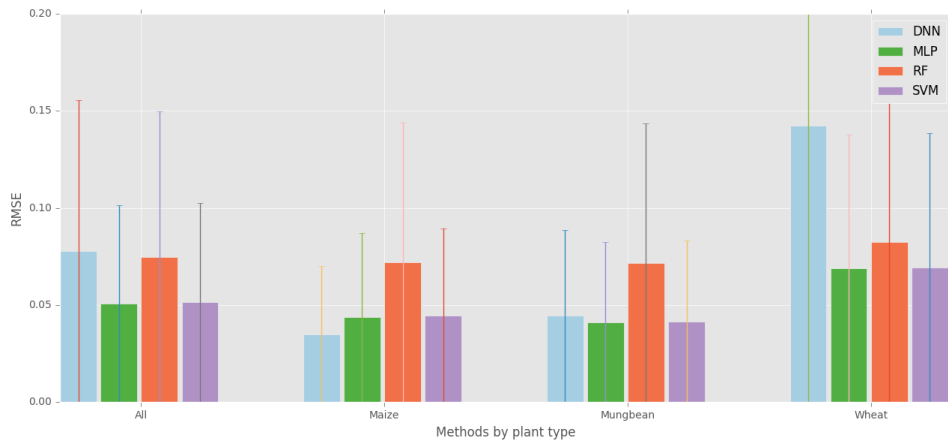


FIGURE 4.6: RMSE and standard deviation of scaled images for ground cover estimates for all regression methods.

4.1.2.1 Colour indices

To further investigate the estimation properties of the colour indices, we have plotted the predicted FCOVER values against the actual FCOVER values for each method. This investigation was done on the original dataset. These results are shown in Figure 4.7. This figure immediately shows the under- or overestimation of the FCOVER for each method. For example, CIVE does quite well overall, but consistently overestimates the FCOVER. ExG, on the other hand, shows a general underestimation. This information gives an indication of how to adjust the colour indices to generate better segmentation maps. A notable phenomenon in these figures is the presence of a conglomeration of data points in either the top or the bottom of the graph. This is a side effect of the threshold failing to create a good split between plant and background. If the threshold is set too high so that the leaves do not get classified correctly, most of the image will be labelled as background - resulting in the severe underestimation of FCOVER. Vice versa, if the threshold is set too low, most of the image will be classified as plant, leading to a drastic overestimation of the FCOVER. Visual inspection learns us that in most of these cases the Otsu algorithm fails to find a good threshold because the colour index mislabels many pixel instances, which skews the colour histogram on which Otsu acts.

4.1.2.2 SVM

The SVM obtains just a slightly higher RMSE than the MLP, and overall gives very accurate estimations. We experimented with a large number of variations of the SVM's parameters to obtain the most optimal results. We varied each parameter while fixing the other parameters to their default (RBF kernel, $\gamma = \frac{1}{64}$, $C = 1.0$, $\epsilon = 1e - 1$, $t = 1e - 3$). The results of these additional experiments are shown in Figure 4.8. A linear kernel gave the best results disregarding other parameters, but a radial basis function kernel combined with an epsilon value (the maximum distance between the actual value and the predicted value at which no penalty is associated

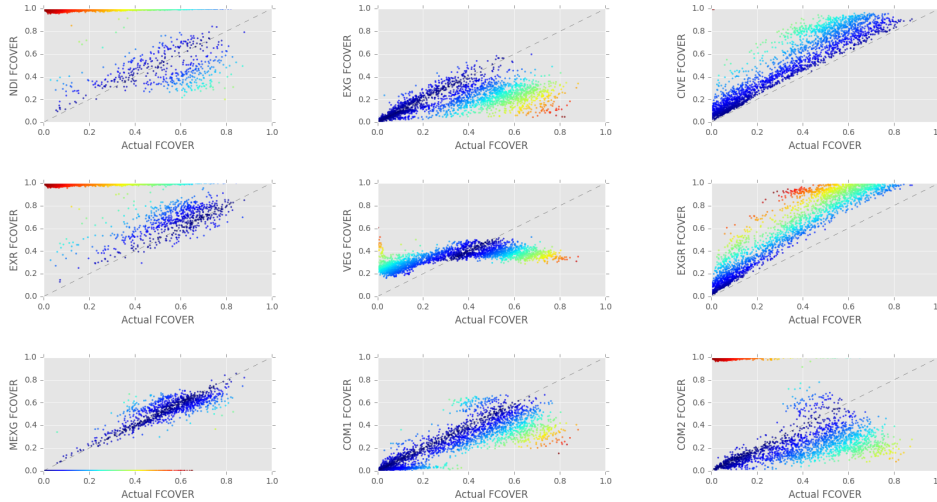


FIGURE 4.7: Actual FCOVER values plotted against the predicted value for colour indices methods. Warmer colours indicate higher discrepancies.

in the loss function) of around $\epsilon = 1e - 1$, an error term penalty of $C = 1.0$, a kernel coefficient of $\gamma = 0.6952$, and a tolerance for stopping of $t = 1e - 3$ gave the best overall results.

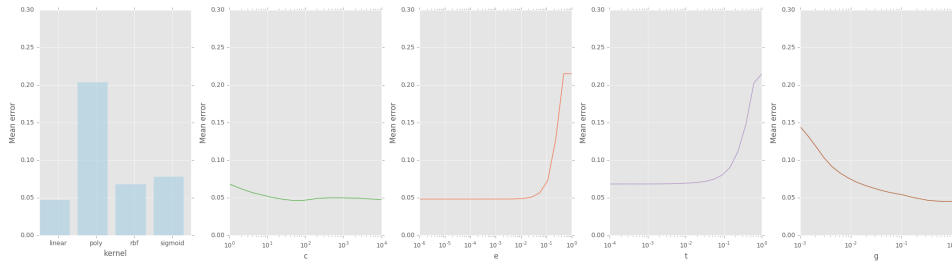


FIGURE 4.8: Effect of tweaking, in order, kernel type k , error term penalty C , maximum distance between actual and predicted value at which no penalty is given ϵ , tolerance for stopping t , and kernel coefficient γ , on mean error.

We also ran additional experiments to discover the effect of the colour range size in the conversion step on the error rate. We ran a simple SVR implementation (RBF kernel, $\gamma = \frac{1}{64}$, $C = 1.0$, $\epsilon = 1e - 1$, $t = 1e - 3$) on a range of variations of colour ranges of our dataset. Figure 4.9 describes the outcome of these experiments. We can observe a clear minimum around a colour range of 4 to 8. We believe this is because of heavy loss of information in smaller colour ranges, and the gradual introduction of more noise in larger colour ranges. We have used a colour range of 8 in our other experiments, because it gave the best overall results.

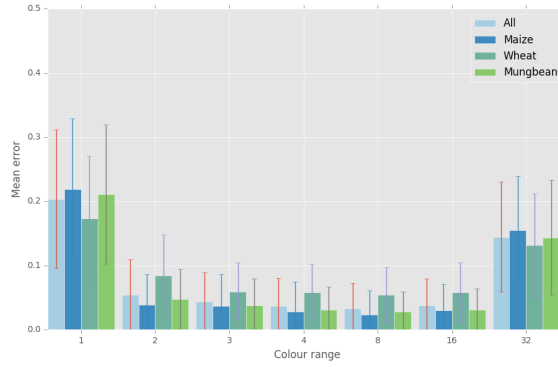


FIGURE 4.9: Effect of colour range size on mean error.

4.1.2.3 Random Forest

We ran the same tests on our RF implementation, sticking to 100 generated trees in our random forest. In initial tests with lower numbers we observed diminishing returns around this number. A minimum mean error of $\mu = 0.0468$, $\sigma = 0.0557$ was found at a colour range (CR) of 4. A full overview is given in Table 4.4.

CR	All	Maize	Mung Bean	Wheat
1	0.2627 (0.2053)	0.2175 (0.2098)	0.2169 (0.1995)	0.3879 (0.1466)
2	0.0529 (0.0589)	0.0429 (0.0552)	0.0491 (0.0566)	0.0716 (0.0627)
3	0.0473 (0.0532)	0.0388 (0.0504)	0.0463 (0.0529)	0.0604 (0.0547)
4	0.0468 (0.0557)	0.0413 (0.0575)	0.0422 (0.0507)	0.0607 (0.0573)
8	0.0479 (0.0540)	0.0420 (0.5188)	0.0453 (0.0536)	0.0597 (0.0557)
16	0.0529 (0.0619)	0.0484 (0.0654)	0.0501 (0.0611)	0.0628 (0.0573)

TABLE 4.4: Mean errors (standard deviations) of an RF classifier for varying colour ranges (CR).

4.1.2.4 Deep neural network

We have already indirectly touched on the estimation results of the DNN in the previous section where we discussed the segmentation results. All regression methods outperform the DNN. However, the difference in performance is small. In fact, the DNN provides better estimation results for Maize type vegetation, and is on par with the SVM and MLP for estimation of images containing Mung bean plants. The regression methods do significantly outperform the DNN method on Wheat imagery, however.

It is hard to gauge exactly why the regression methods do so much better on Wheat images, since we converted the original image to a percentage based feature vector. We know that our DNN underperforms on Wheat image because it is prone to generate rounded, thick edges, quite the opposite of Wheat leaves. Methods that do direct regression on the FCOVER value do not encounter this problem, which might help to explain the discrepancy in scores. We will further discuss this in Chapter 5.

4.2 Hot target detection

To benchmark our deep neural network implementation of hot target detection we implemented the CA, and HOTMAP algorithms proposed in [101] and [81]. In addition, we implement an SVM and a simple MLP neural network. We have optimized the parameters of these implementations in the same manner described in Section 4.1.2.2, the full results of which we will also show in the following sections. We will start with an overview of the accuracy and precision scores as noted in section 3.3, and continue with a more expansive view of our results - most notably the machine learning results.

Method	Accuracy	Precision
CA [49]	0.9999	0.2397
HOTMAP [81]	0.9999	0.6766
SVM	0.9847	0.9954
MLP	0.9844	0.9951
DNN	0.9797	0.0015

TABLE 4.5: Hot target detection by method.

4.2.1 CA & HOTMAP

The contextual algorithm (CA) of Giglio et al. [49], and the HOTMAP algorithm both provide excellent accuracy. However, this good result is primarily due to the large number of non-hot target pixels in the satellite imagery. We get a clearer picture of how these algorithms perform when looking at the precision scores, the fraction of true hot targets over all marked hot targets (also called positive detection rate in Giglio et al. [49]). In line with Murphy et al. [81], we observe that CA results in many false positives, while HOTMAP performs much better. We believe this difference is caused by incorporating extra information from additional wavelengths.

4.2.2 SVM & MLP

The SVM and MLP were both trained on a dataset consisting of 50% hot target pixels and 50% non-hot target pixels (selected randomly from the complete FIRES dataset). We obtain exceptional precision scores, even with 10-fold cross validation. Accuracy scores are slightly lower than CA and HOTMAP, but this is due to the difference in dataset size. We ran CA and HOTMAP on the full images, the majority of which were non-hot target pixels, while the SVM and MLP saw only 50% non-hot target pixels. When we use the trained SVM to predict on full images, we achieve similar accuracy results.

To determine the effectiveness of these methods we first determined the data separability in n -dimensional feature space, by generating crossplots of the pixel values in every wavelength band, and labelling them with our ground truth (hot target or non-hot target). An example of these crossplots, namely for OLI band 7, is shown in Figure 4.10. We can see that there exists

a near linear separability when comparing the values of OLI band 6 versus that of OLI band 7.

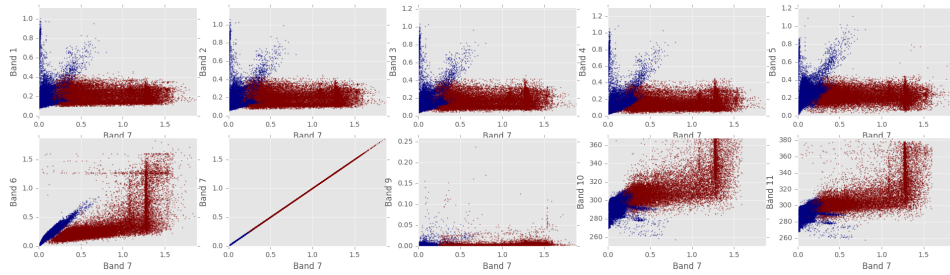


FIGURE 4.10: Crossplot of pixel values for band 7 versus pixel values other bands, showing separability. Red data points are hot targets, blue data points are non-hot targets.

We have also looked at the effect of our features (being the 10 OLI and TIRS channels) on the performance of the SVM, by removing these features from our dataset. These results are reported in Table 4.6. In line with the results of our visual inspection of separability, we observe that removing the information in OLI band 7 results in the most severe performance drop.

4.2.3 Deep neural network

The DNN fails to perform accurately, as the results in Table 4.5 clearly show. When we investigate individual segmentations, a batch of which is displayed in Figure 4.11, we see that not only does the DNN not pick up the hot targets, it also tends to label other structures such as cloud formations as hot targets.

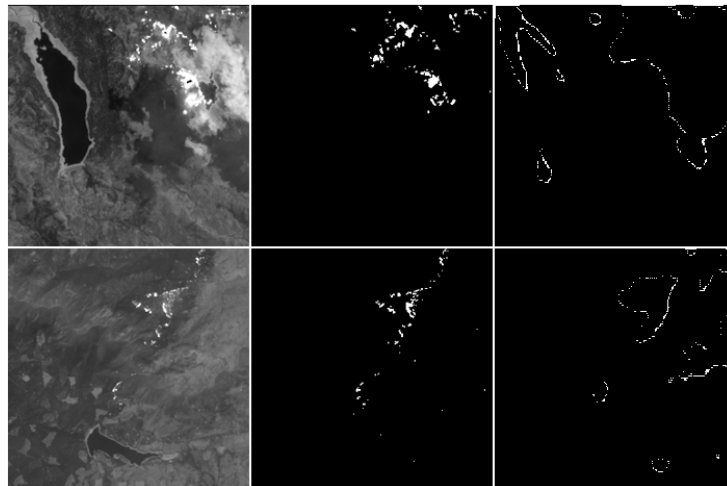


FIGURE 4.11: Examples of DNN segmentations of augmented satellite imagery. Shown are the original OLI band 7 image (left), ground truth hot target annotation (middle), and DNN labelling (right).

This detrimental effect could be caused by the lack of large swaths of hot targets. The hot targets in our satellite imagery are usually only a couple of pixels large, while our DNN architecture works better for larger targets, as

Removed band	Accuracy	Precision
1	0.9848	0.9955
2	0.9850	0.9955
3	0.9853	0.9955
4	0.9846	0.9951
5	0.9844	0.9954
6	0.9754	0.9906
7	0.8941	0.9347
9	0.9851	0.9952
10	0.9877	0.9953
11	0.9870	0.9941

TABLE 4.6: Accuracy and precision scores of the SVM when removing all information of a certain band.

is evident by our ground cover analysis results. We have tried a number of changes in architecture, similar to those reported in previous ground cover analysis sections, but found no major improvement.

Chapter 5

Discussion

We will now discuss the results described in the previous chapter. First, we will consider the problem of ground cover analysis, after which we will cover our results for hot target detection. We will evaluate our methods, note the shortcomings of our experiments, and finally, address other avenues for future research.

5.1 Ground cover analysis

Accurate ground cover analysis has to overcome a number of obstacles, such as dealing with various lighting differences, shadows, rubble, et cetera, as we have discussed in previous sections. We have implemented current methods that are based on transforming the RGB colour space to another index, and compared them to methods that use machine learning. We find that colour indices are simple to understand, which makes them easily tweakable. Since most variables can be directly linked to the RGB values, it is immediately apparent how to best change the index to better suit the data. Their simplicity also ensures their fast runtime, and their ability to handle large images is very advantageous since most modern camera setups produce high resolution images. On top of that, most of these colour indices methods do not require training.

Considering all these advantages, one would expect colour indices to be a top choice for ground cover analysis. With a little tweaking, for some problems, they certainly are. Keeping in mind that there are large differences between colour indices (as shown in Figure 4.1, for example), we see that some indices (MExG) can obtain great RMSE scores for certain plant types. However, there are also large downsides to colour indices methods. Because the indices can only use the information stored in a single pixel, the RGB values, they fail to deal with, for example, the downward curvature of some leaves casting shadows. Some colour indices try to make up for this by incorporating the average values calculated in the entire image, or setting lower thresholds to compensate for the index' specificity.

This is where the DNN outperforms conventional methods. When considering estimation results, we see that the convolutional layers allow the DNN to deal with intra-leaf changes in lighting. Shadows and hard lighting have little effect on the resulting segmentation map. We see that the DNN outperforms all other segmentation methods by a large margin. Even when looking at pure regression methods that cannot perform segmentation (MLP, RF, SVM), we see that the DNN can obtain comparable results, and for some plant types even surpass those regression methods (see Figure 4.6). Our DNN architecture has many advantages, the main one being

its robustness to lighting changes, the one thing colour indices methods really struggle with. Especially when we train our networks on each plant type specifically, we see a big increase in performance (see Table 4.2). This does require an extra preliminary step of labelling the data with the plant types. When we look at the results of more complex plant types we can see that the strength of the DNN lies in rounded, smooth, segmentations. For images of more weedy, grass-like plants, or lean plant types with a lot of stalks, we observe a degradation in performance (often caused by oversegmentation). This may be fixed by decreasing the number of features in the network architecture.

We must note that our architecture also has a number of disadvantages. First of all, neural networks, convolutional neural networks in particular, are heavily reliant on properly labelled data. This means that a lot of preliminary work and preprocessing steps are involved with using this type of method for ground cover analysis. In addition, the learning phase is quite slow, especially compared to the colour indices methods that were able to handle high resolution input. Training time was around an hour on an 8-core i7-4700MQ with an NVIDIA Quadro K1100M. This means that for regression tasks, it is probably preferable to use an SVM-based approach, considering the algorithms speed and very accurate estimation results. Second, the architecture we used was not able to handle very large input images. The original image resolution of around 2000x1500px was too large for our network c.q. machine, and we were forced to scale down our input images. Initially, we split our original images into smaller blocks and used those as input images, however, we found that our ground truth data was not always optimal. We must also address that while we get very good results on networks trained separately on each crop type, this will have implications for the operational use of our methods, and introduce additional costs. A final disadvantage of DNNs is that while they can achieve good performance, it is hard to observe why they perform so well. Methods of visualizing layer activation are currently being researched however to try and overcome this problem.

This brings us to the limitations of our experiments. When we performed visual inspection on our original ground truth, we found that there were many small errors. This was an artefact of doing annotation with the CANEYE software, but difficult to resolve. We expect that the non-optimal ground truth labelling is the reason that the DNN did not perform accurately on the original splitted images. By scaling the original image, and with it the ground truth, we solved many small errors through nearest-neighbour resampling. This leads to a better ground truth, and much better results. We expect that with more accurate ground truth, the DNN is also able to obtain better results on the original (splitted) images. In addition, we expect that with a better machine, larger size input images can also be processed.

When we review our hypotheses on ground cover analysis, we must answer positively to both the hypotheses of segmentation, and estimation. The methods proposed in this work outperform existing methods of ground cover analysis, both in RMSE, as well as accuracy and precision. Out of our machine learning methods, the SVM, RF, MLP, and DNN, our DNN architecture shows the best segmentation performance while also being able to provide accurate ground cover estimation. In line with our hypotheses,

machine learning is an effective tool that can be applied to perform more accurate ground cover analysis than conventional methods.

5.2 Hot target detection

We will now evaluate our results of hot target detection. The existing algorithms of Giglio et al. [49] and Murphy et al. [81] both have excellent accuracy. This is due to the combination of the algorithms effective ability to discern non-hot target pixels (under-classification) and the large number of non-hot target pixels in our FIRES dataset. When we consider precision however, the fraction of accurately labelled hot targets out of all labelled hot targets, we see that there is definitely room for improvement. The existing algorithms were able to be run on the original input image, and somewhat resemble the colour indices methods of ground cover analysis. Their advantages, much like the colour indices methods, are the speed of execution, simplicity, and ability to handle large input. Their disadvantages are immediately apparent in the precision scores listed in Table 4.5.

When we look at our machine learning methods, we see that these methods can obtain a much higher precision rate. When we investigate the feature space, we can see why. There is a relatively clean segregation between hot target pixels and non-hot target pixels, even when only considering just two features (as can be seen in Figure 4.10). The SVM and MLP are able to find this as well and generate an accurate decision boundary *cq.* distribution of weights. A downside of these methods is that they require some time to generate full resolution segmentation maps. The other machine learning methods we implemented, our DNN architecture, fails to perform. We observe very low precision results, due to the network's inability to generate effective feature maps. We believe these low scores can be explained by the shape of our ground truth data. The fires present in our dataset usually only contain a few pixels. In some cases, there are a few hundred (for example, the imagery in Figure 4.11), but spread out and sparse. The architecture we used generates smooth, round, segmentation maps, which we showed earlier as well. While we tried various alternative architectures, with a lower number of features, and convolutional layers, to decrease the daubing effect visible in the segmentation maps, we did not observe any major increase in performance. An additional limitation was our small dataset. As mentioned previously, convolutional neural networks require large datasets to perform accurately, and even with augmented data our FIRES set was not very sizeable. These shortcomings, in combination with the innate training time of DNNs, make them unfit, in this form, for accurate hot target detection. We do believe other DNN architectures might be able to generate better segmentation maps, although it is hard to rival SVM-based approaches.

When we consider our hypothesis on hot target detection, we cannot answer it positively in its entirety. While machine learning methods such as SVM and MLP outperform conventional methods, our DNN drastically underperformed. To summarize, we believe that machine learning can provide very large gains in effective hot target detection, but DNNs are in its current state, too slow, and too inaccurate to outperform other methods.

Chapter 6

Conclusion

In this thesis we researched the applicability of machine learning in the domains of ground cover analysis and hot target detection. Our main focus was on the effectiveness of deep neural networks in these areas. Existing methods in both ground cover analysis and hot target detection rely on converting the original feature space (be it RGB or multispectral) to a more suitable one. However, since these methods only use the information of a single pixel per classification, they often fail to pick up overarching features such as the curvature of a leaf casting a shadow. We evaluated whether machine learning methods could improve on the results of conventional methods, since they are able to learn, and overcome these obstacles, from examples. Support Vector Machine (SVM), Multilayer Perceptron (MLP), Random Forest (RF), and Deep Neural Network (DNN) methods were implemented and compared to existing colour indices methods, and other methods found in relevant literature (AP-HI, CA, HOTMAP).

We found that for ground cover analysis, machine learning improves on existing methods for both segmentation and estimation. Regarding segmentation, we observed that our DNN implementation outperforms all other existing methods on our scaled dataset, especially when trained on each plant type specifically. On our original dataset the DNN does not perform as well, due to suboptimal ground truth labelling. Regarding estimation, we see that machine learning methods outperform all other conventional methods. SVM, RF, and MLP, all do regression on our converted colour range dataset, but cannot provide segmentation maps. Our DNN implementation, which can provide segmentations, shows comparable results.

For hot target detection we found that machine learning methods can greatly improve on current methods. We see that our SVM and MLP implementations have a significantly higher precision rate than existing methods. Our DNN implementation is not able to produce the same results. Although we mainly see very positive effects of the machine learning methods presented in this research, one disadvantage of learning-based approaches is the necessity of large quantities of accurately-labelled ground truth data, which is not always available in these domains.

Concluding, we have presented a number of machine learning algorithms that outperform existing methods applicable to the ground cover analysis and hot target detection domains. Although these learning-based methods have their own shortcomings, they far outweigh their advantages compared to conventional methods in most problem domains.

6.1 Future Work

We have seen that our machine learning methods have its advantages, but also some shortcomings. In addition, we have described a number of shortcoming in our experiments. In this section we will provide a number of ideas and pointers for future research in this area.

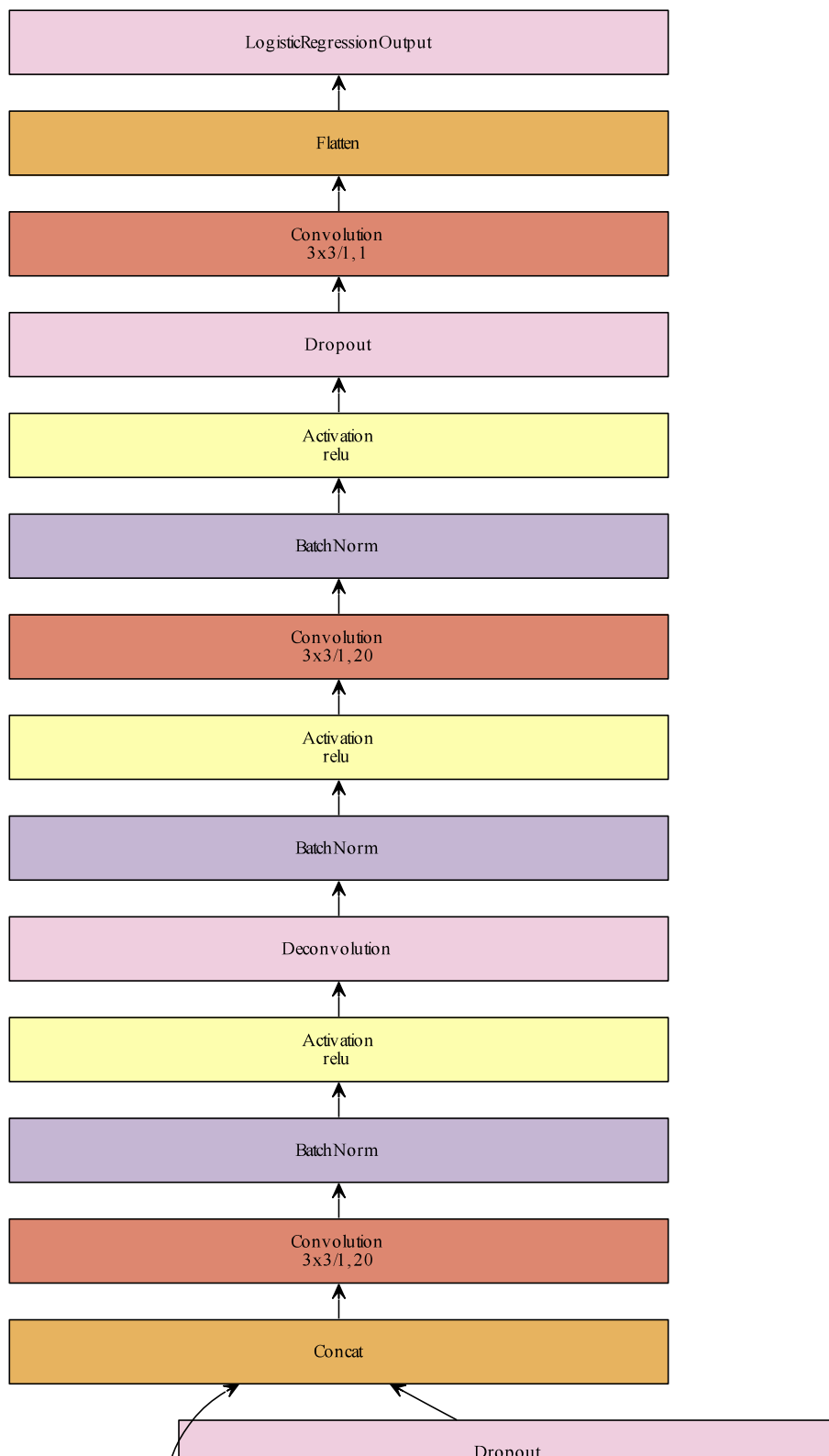
We have experimented with different types of DNN architectures for ground cover analysis, all based on U-net researched by Ronneberger et al. [94] since it provided the best pixel-wise segmentation results to date. However, many other network architectures exist. In preliminary experiments we have shown that networks trained separately for each plant type perform better than a single network trained on the entire CROPS set. Each plant type has its own characteristics, which could be captured better by different networks. Further research could be done to investigate the best architecture for specific plant types, or even stages of growth. Taking this idea one step further, one could imagine training a generative network to determine the optimal network architecture for a specific stage. For example, work in the domain of architecture-learning by Denil et al. [36] shows that by learning only a small number of weights, over 95% of the other weights can be predicted without affecting prediction performance.

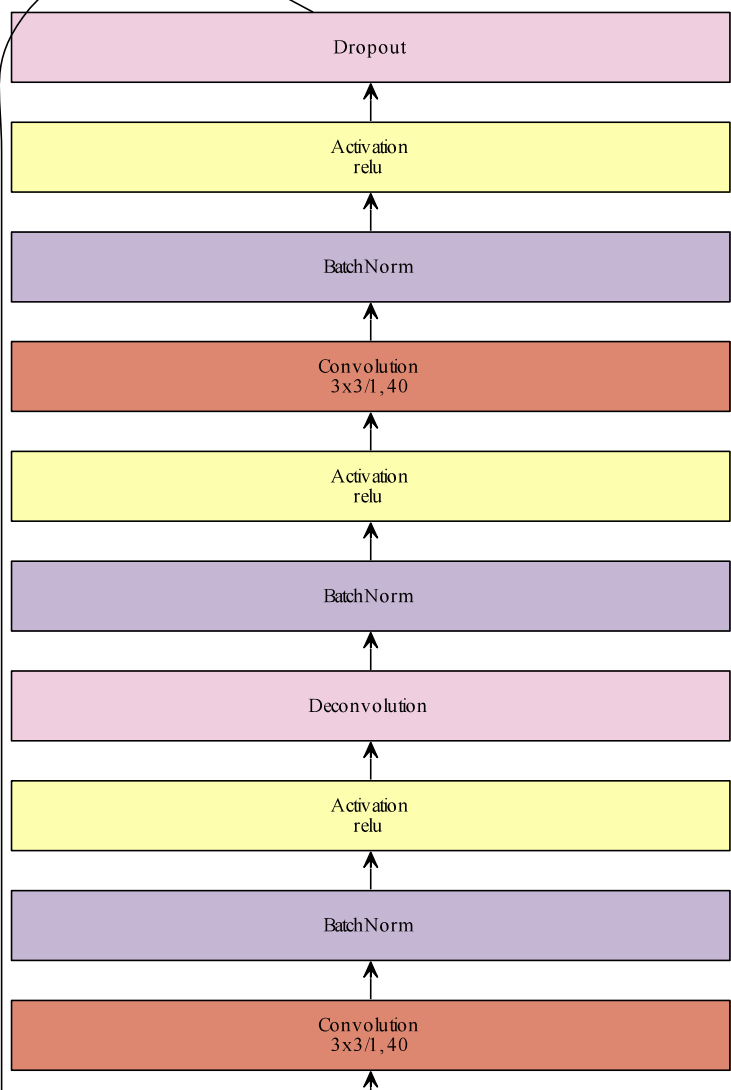
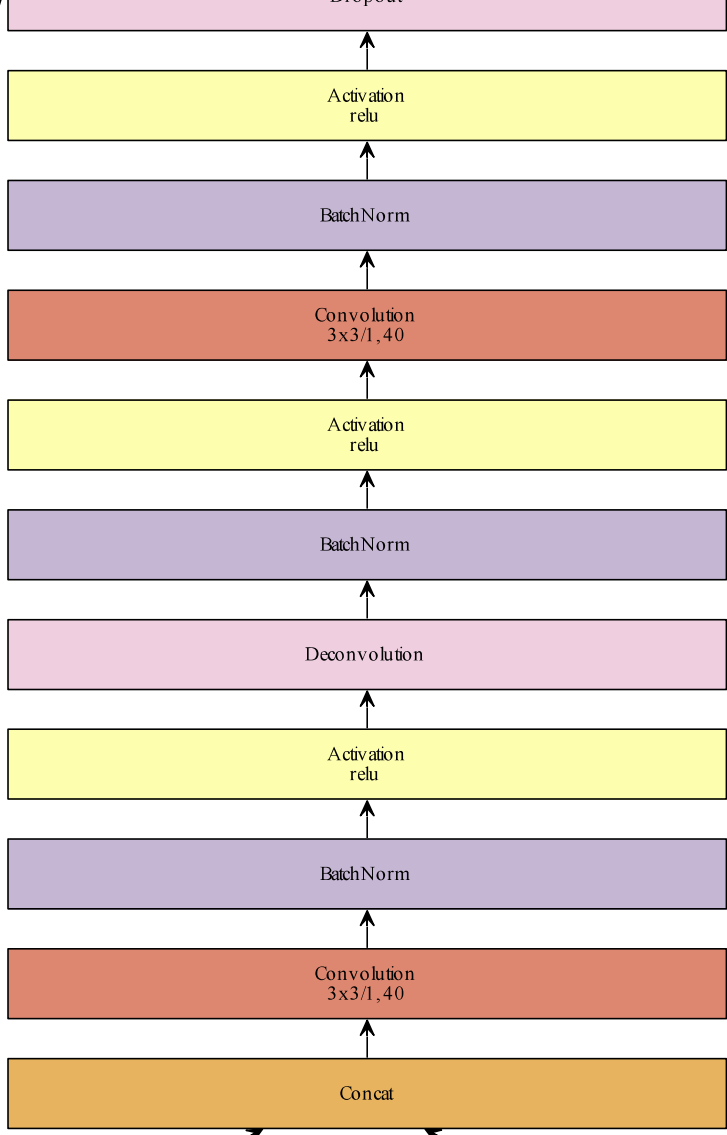
With regards to regression, an avenue of research would be to extend the feature space created by the colour conversion with pixel-specific features. Currently, we only use information on the fraction of a certain colour for our SVM implementation. However, this information could also be useful for segmentation methods, including our DNN.

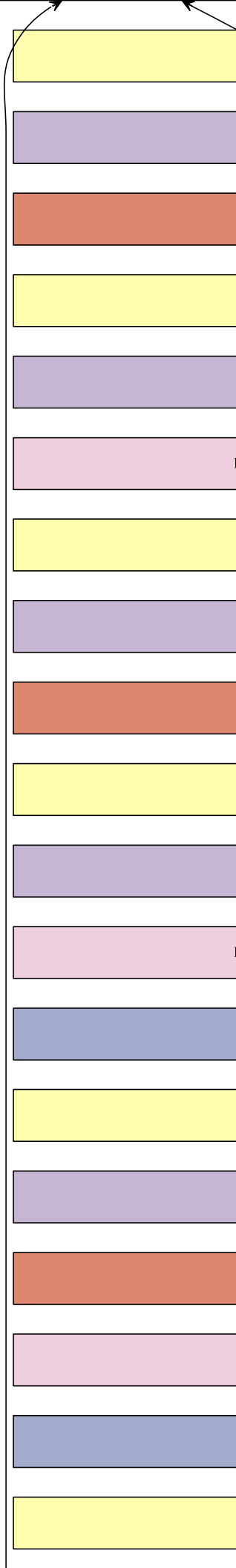
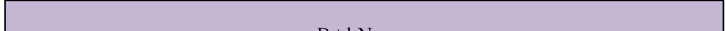
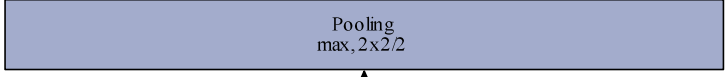
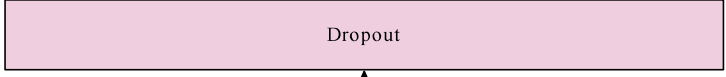
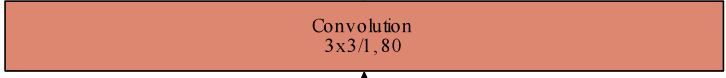
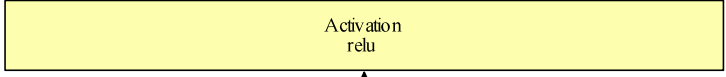
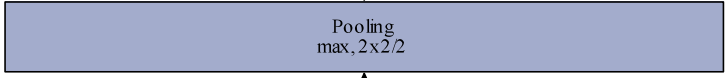
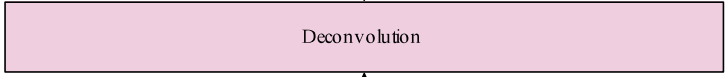
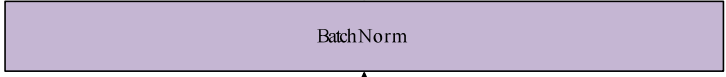
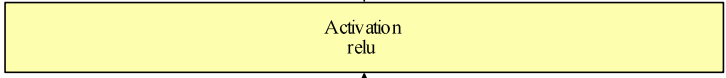
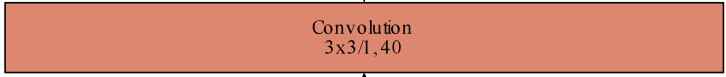
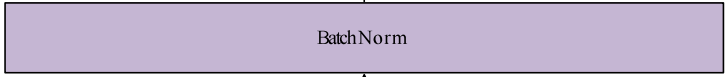
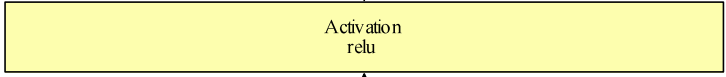
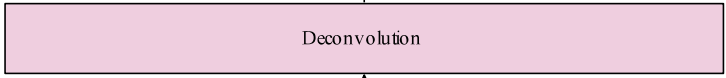
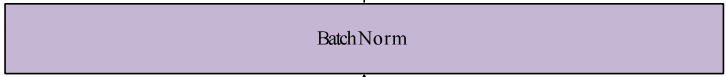
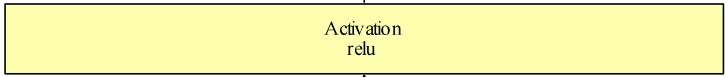
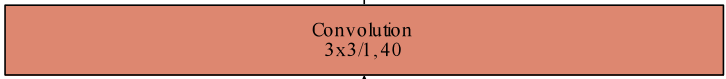
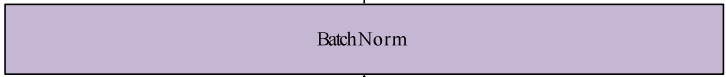
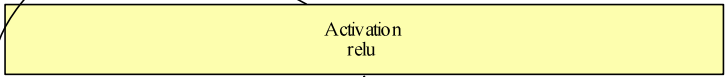
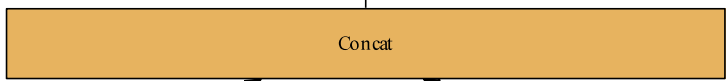
Regarding our hot target detection research, we propose to experiment with different DNN architectures. Our current architecture was clearly insufficient to generate correct segmentation maps, however, we believe that the information of neighbouring pixels can be helpful in detecting hot targets. Alternatively, we expect that DNNs can be useful in detecting smoke patches (indicating commencing, low visibility fires). We also advise to evaluate the SVM and MLP methods on a larger dataset, building on the FIRES dataset. We found that wildfire data is very hard to come by, and working towards an extensive historical wildfire dataset would greatly aid research in this area.

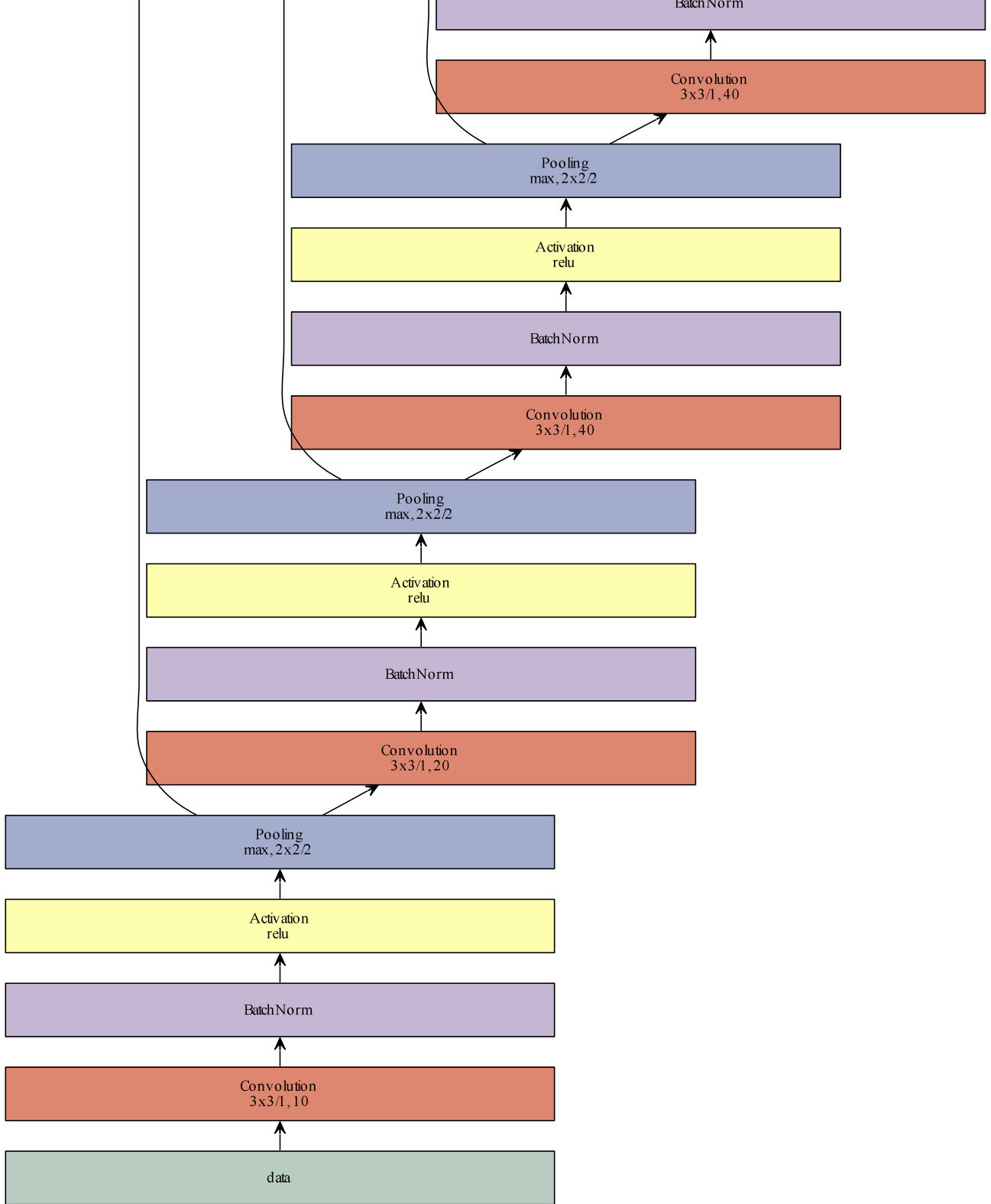
Appendix A

DNN Architecture









Appendix B

Ground cover estimates RMSE

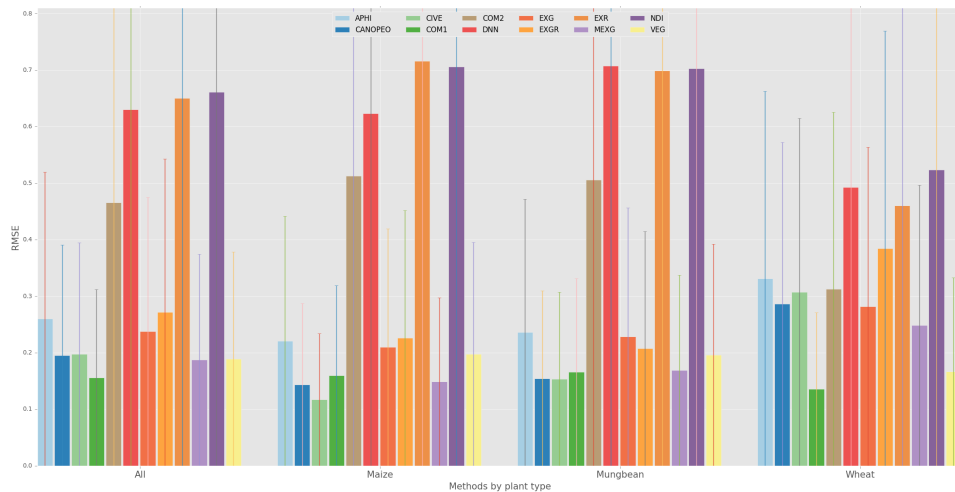


FIGURE B.1: RMSE of original images for ground cover estimates for all segmentation methods.

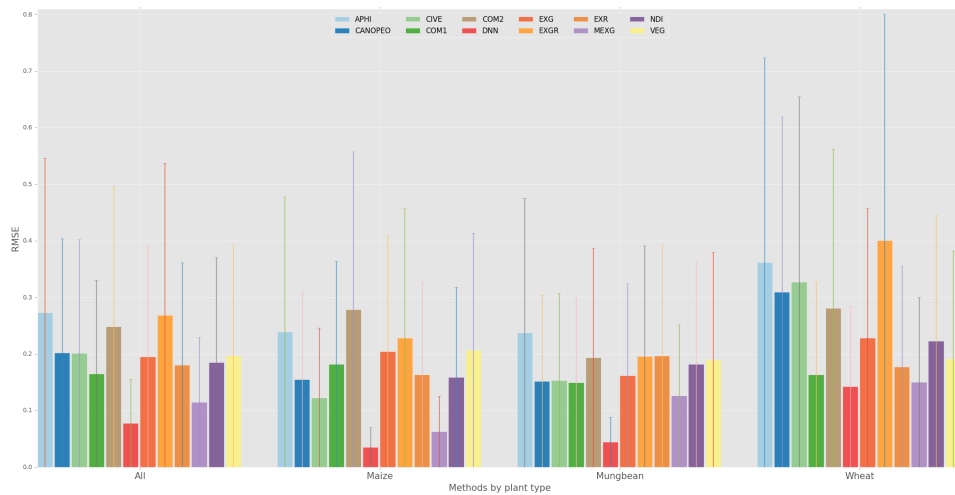


FIGURE B.2: RMSE of scaled images for ground cover estimates for all segmentation methods.

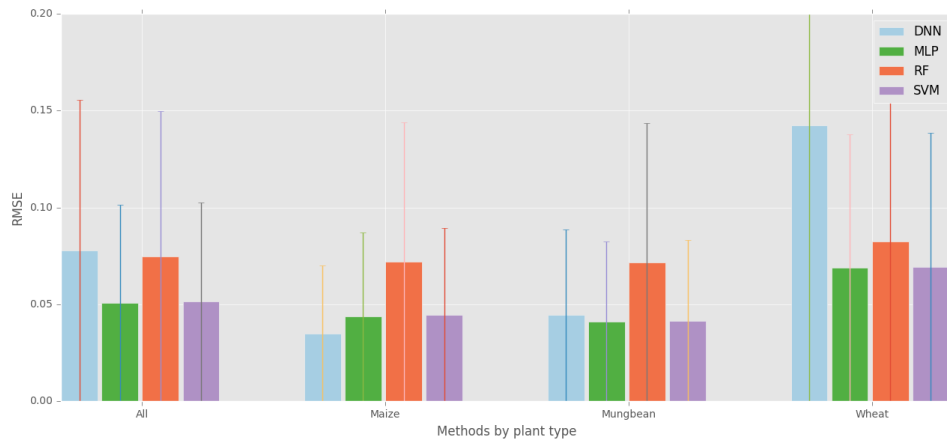
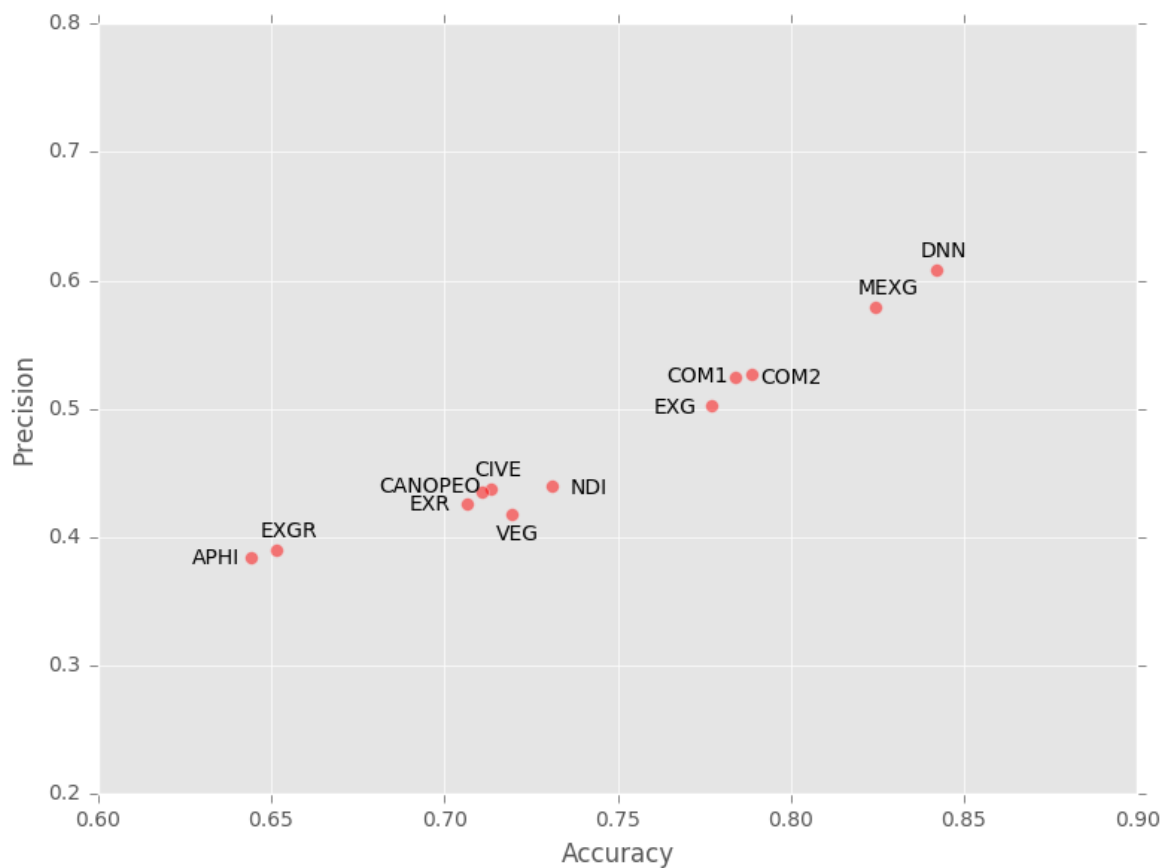
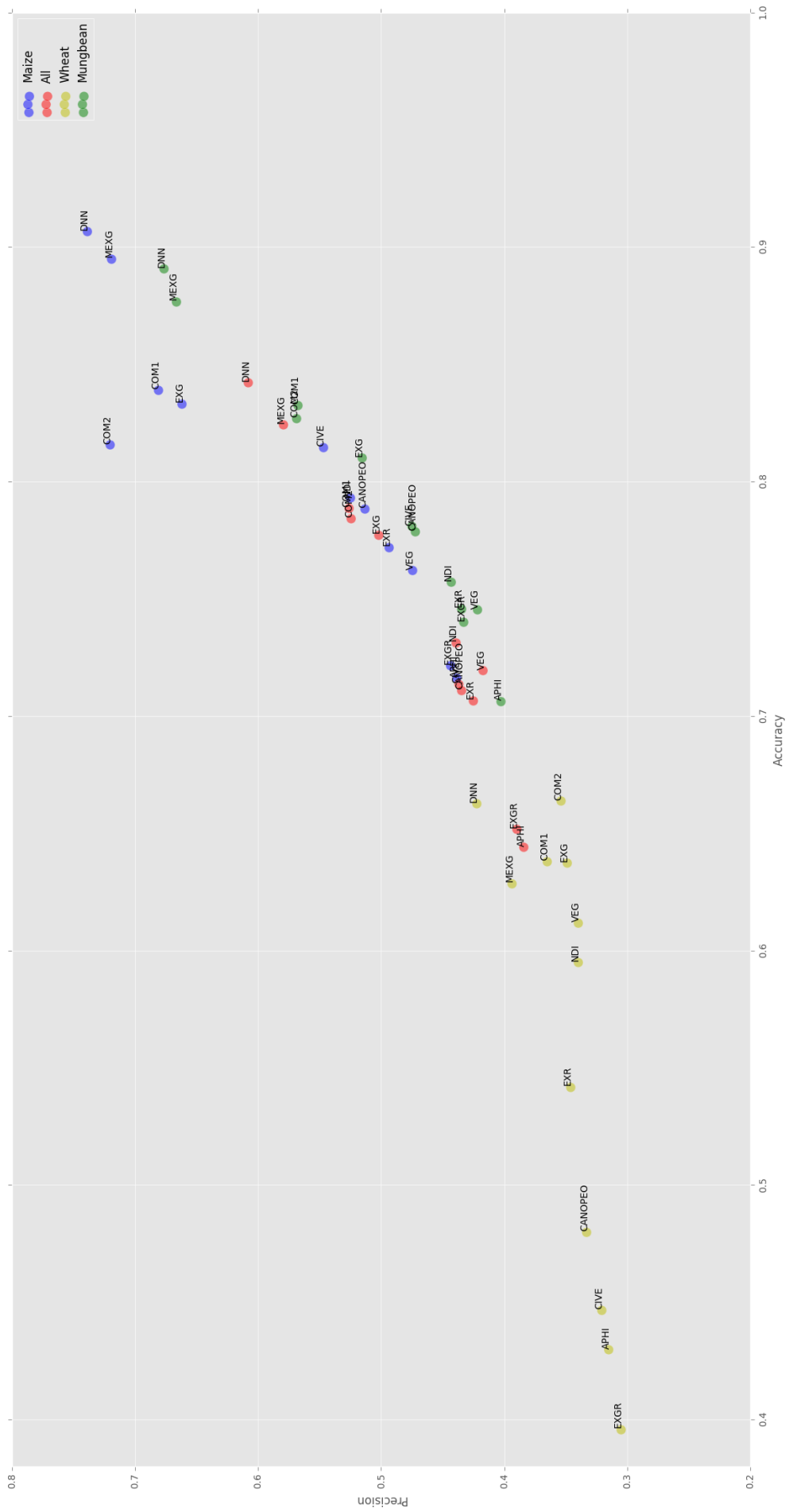


FIGURE B.3: RMSE of scaled images for ground cover estimates for all regression methods.

Appendix C

Ground cover segmentation accuracy & precision





Appendix D

FIRES dataset scenes and topographical location



FIGURE D.1: Locations of the scenes in the FIRES dataset.
Image taken from [81].

ID	Product	Region
1	LC81680812014210LGN00	Africa
2	LC81710592015010LGN00	Africa
3	LC81790682014207LGN00	Africa
4	LC81920542015029LGN00	Africa
5	LC82000542015005LGN00	Africa
6	LC80950862014019LGN00	Australia
7	LC80900892015018LGN00	Australia
8	LC80970842015004LGN00	Australia
9	LC81040692014210LGN00	Australia
10	LC81090762015024LGN00	Australia
11	LC81790272014207LGN00	Europe
12	LC81880322014238LGN00	Europe
13	LC82020342014256LGN00	Europe

14	LC81890342014229LGN00	Europe
15	LC81940182014216LGN00	Europe
16	LC81340412015023LGN00	India
17	LC81340432015023LGN00	India
18	LC81360412015021LGN00	India
19	LC81440422014090LGN00	India
20	LC81440472015029LGN00	India
21	LC80290442015023LGN01	Mexico
22	LC80290452015023LGN01	Mexico
23	LC80290462015023LGN01	Mexico
24	LC80290472015023LGN01	Mexico
25	LC80320402015028LGN00	Mexico
26	LC81200222014226LGN00	Russia
27	LC81310142014207LGN00	Russia
28	LC81360202014210LGN00	Russia
29	LC81660252014228LGN00	Russia
30	LC81750282014211LGN00	Russia
31	LC81160482015025LGN00	South East Asia
32	LC81250372014069LGN00	South East Asia
33	LC81260522015031LGN00	South East Asia
34	LC81280482015029LGN00	South East Asia
35	LC81260592014060LGN00	South East Asia
36	LC80040552015024LGN00	South America
37	LC80070572015029LGN00	South America
38	LC82140652015023LGN00	South America
39	LC82200662014206LGN00	South America
40	LC82330862015028LGN00	South America
41	LC80690182014140LGN00	North America
42	LC80290322014084LGN00	North America
43	LC80430332014262LGN00	North America
44	LC80430302014198LGN00	North America
45	LC80170242013173LGN00	North America

TABLE D.1: List of scenes used in the FIRES dataset.

Appendix E

Impact on society

In closing, we present a text reflecting on the relevance of this research in society.

As we have explained in previous sections, it is difficult for poor farmers in second- and third-world countries to develop efficient irrigation systems. Dry seasons and low soil moisture levels make it a struggle to cultivate their land. Water pumps that allow access to sweet surface water are expensive, therefore farmers need to optimize their irrigation systems. Plants require a certain level of irrigation to grow quickest, depending on the plant's stage of development. The estimation of ground cover is a good metric for this since it correlates well with biomass yield. Previous research has shown that accurate ground cover estimation can help in developing an advisory system for irrigation scheduling. In this work we have researched the best approach to use machine learning to perform ground cover analysis, and proposed a number of new methods that outperform conventional methods of ground cover analysis. In doing so, we have made a contribution to the advance of small-scale agriculture in developing regions that have little access to high-tech machinery.

We have also detailed the devastating effects wildfires have on our society. Flora and fauna alike are threatened by this destructive force of nature. The cost of wildfire suppression in the US alone ranges in the billions, and comes at the cost of many lives, devastation of vegetation, and increased atmospheric carbon emission. Fast and accurate wildfire detection and monitoring is therefore incredibly important. While conventional methods can obtain relatively accurate detection results, they report many false positives as well. In this work we have shown that machine learning is an effective tool for detecting wildfires in satellite imagery accurately, and presented methods surpassing existing methods in terms of precision. We hope to have made a valuable contribution to the domain of hot target detection, and the decline of loss of life and destruction of nature due to wildfires.

G.L.J. Pingen

Bibliography

- [1] AP-HI - Hao Lu. <https://sites.google.com/site/poppinace/>. Accessed: 2016-05-27.
- [2] Bangladesh institute of ICT in development (BIID). <http://www.biid.org.bd>. Accessed: 2016-05-27.
- [3] CAN-EYE. <http://www6.paca.inra.fr/can-eye>. Accessed: 2016-05-27.
- [4] CIMMYT. international maize and wheat improvement center. <http://www.cimmyt.org>. Accessed: 2016-05-27.
- [5] Earthexplorer. <http://earthexplorer.usgs.gov/>. Accessed: 2016-04-29.
- [6] GIS dictionary - remote sensing. <http://support.esri.com/en/knowledgebase/GISDictionary/term/remote%20sensing>. Accessed: 2016-04-29.
- [7] ITC - faculty of geo-information science and earth observation - university of twente. <https://www.itc.nl>. Accessed: 2016-05-27.
- [8] Landsat. <http://landsat.usgs.gov>. Accessed: 2016-04-29.
- [9] Landsat on AWS. <https://aws.amazon.com/public-data-sets/landsat/>. Accessed: 2016-04-29.
- [10] MODIS web. <http://modis.gsfc.nasa.gov>. Accessed: 2016-04-29.
- [11] MXNet. <https://github.com/dmlc/mxnet>. Accessed: 2016-08-08.
- [12] NASA - polar orbiting missions. <http://npp.gsfc.nasa.gov/viirs.html>. Accessed: 2016-04-29.
- [13] The new VIIRS (and landsat-8) active fire detection algorithms. http://www.redlatif.org/wp-content/uploads/2015/11/Schroeder_VIIRS_Fire.compressed.pdf. Accessed: 2016-04-29.
- [14] Nwo - netherlands organisation for scientific research. <http://www.nwo.nl/en>. Accessed: 2016-09-30.
- [15] The rising cost of fire operations: Effects on the forest services non - fire work. <http://www.fs.fed.us/sites/default/files/2015-Fire-Budget-Report.pdf>. Accessed: 2016-04-29.
- [16] Sentinel-2a launches - our compliments & our complements. <http://landsat.gsfc.nasa.gov/?p=10643>. Accessed: 2016-04-29.

- [17] STARS project - agricultural improvement in emerging economies - STARS project. <http://www.stars-project.org/en>. Accessed: 2016-05-27.
- [18] Third place solution for the second national datascience bowl. <http://juliandewit.github.io/kaggle-ndsb/>. Accessed: 2016-08-08.
- [19] Using the USGS landsat 8 product. http://landsat.usgs.gov/Landsat8_Using_Product.php. Accessed: 2016-04-29.
- [20] VIIRS active fire. <http://viirsfire.geog.umd.edu>. Accessed: 2016-04-29.
- [21] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer networks*, 38(4):393–422, 2002.
- [22] V. G. Ambrosia, S. W. Buechel, J. A. Brass, J. R. Peterson, R. H. Davies, R. J. Kane, and S. Spain. An integration of remote sensing, GIS, and information distribution for wildfire detection and management. *Photogrammetric Engineering and Remote Sensing*, 64:977–986, 1998.
- [23] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433, 2015.
- [24] V. Badrinarayanan, A. Handa, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. *arXiv preprint arXiv:1505.07293*, 2015.
- [25] X. Bai, Z. Cao, Y. Wang, Z. Yu, X. Zhang, and C. Li. Crop segmentation from images by morphology modeling in the cie l* a* b* color space. *Computers and electronics in agriculture*, 99:21–34, 2013.
- [26] F. Baret and G. Guyot. Potentials and limits of vegetation indices for LAI and APAR assessment. *Remote sensing of environment*, 35(2):161–173, 1991.
- [27] S. Basu, S. Ganguly, S. Mukhopadhyay, R. DiBiano, M. Karki, and R. Nemani. DeepSAT—a learning framework for satellite imagery. *arXiv preprint arXiv:1509.03602*, 2015.
- [28] R. Bellazzi and B. Zupan. Predictive data mining in clinical medicine: current issues and guidelines. *International journal of medical informatics*, 77(2):81–97, 2008.
- [29] L. Boschetti, D. P. Roy, C. O. Justice, and M. L. Humber. Modis-landsat fusion for large area 30m burned area mapping. *Remote Sensing of Environment*, 161:27–42, 2015.
- [30] X. P. Burgos-Artizzu, A. Ribeiro, M. Guijarro, and G. Pajares. Real-time image processing for crop/weed discrimination in maize fields. *Computers and Electronics in Agriculture*, 75(2):337–346, 2011.

- [31] J. Casadesús and D. Villegas. Conventional digital cameras as a tool for assessing leaf area index and biomass for cereal breeding. *Journal of integrative plant biology*, 56(1):7–14, 2014.
- [32] M. Castelluccio, G. Poggi, C. Sansone, and L. Verdoliva. Land use classification in remote sensing images by convolutional neural networks. *arXiv preprint arXiv:1508.00092*, 2015.
- [33] A. Cauchy. Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, 25(1847):536–538, 1847.
- [34] R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008.
- [35] D. K. Davies, S. Ilavajhala, M. M. Wong, and C. O. Justice. Fire information for resource management system: archiving and distributing MODIS active fire data. *Geoscience and Remote Sensing, IEEE Transactions on*, 47(1):72–79, 2009.
- [36] M. Denil, B. Shakibi, L. Dinh, N. de Freitas, et al. Predicting parameters in deep learning. In *Advances in Neural Information Processing Systems*, pages 2148–2156, 2013.
- [37] P. Doraiswamy, J. Hatfield, T. Jackson, B. Akhmedov, J. Prueger, and A. Stern. Crop condition and yield simulations using landsat and modis. *Remote sensing of environment*, 92(4):548–559, 2004.
- [38] C.-J. Du and D.-W. Sun. Learning techniques used in computer vision for food quality evaluation: a review. *Journal of food engineering*, 72(1):39–55, 2006.
- [39] E. Dwyer, S. Pinnock, J.-M. Grégoire, and J. Pereira. Global spatial and temporal distribution of vegetation fire as determined from satellite observations. *International Journal of Remote Sensing*, 21(6-7):1289–1302, 2000.
- [40] H. M. Easlon and A. J. Bloom. Easy leaf area: Automated digital image analysis for rapid and accurate measurement of leaf area. *Applications in plant sciences*, 2(7), 2014.
- [41] D. Ferrucci, E. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. A. Kalyanpur, A. Lally, J. W. Murdock, E. Nyberg, J. Prager, et al. Building watson: An overview of the deepqa project. *AI magazine*, 31(3):59–79, 2010.
- [42] M. D. Flannigan, M. A. Krawchuk, W. J. de Groot, B. M. Wotton, and L. M. Gowman. Implications of changing climate for global wildland fire. *International Journal of Wildland Fire*, 18(5):483–507, 2009.
- [43] L. P. Flynn, A. J. Harris, and R. Wright. Improved identification of volcanic features using landsat 7 etm+. *Remote Sensing of Environment*, 78(1):180–193, 2001.

- [44] C. for Disease Control, Prevention, et al. Wildfire-related deaths—texas, march 12–20, 2006. *MMWR: Morbidity and mortality weekly report*, 56(30):757–760, 2007.
- [45] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *science*, 315(5814):972–976, 2007.
- [46] S. Gebhardt, J. Schellberg, R. Lock, and W. Kühbauch. Identification of broad-leaved dock (*rumex obtusifolius* l.) on grassland by means of digital image processing. *Precision Agriculture*, 7(3):165–178, 2006.
- [47] F. A. Gers, J. Schmidhuber, and F. Cummins. Learning to forget: Continual prediction with lstm. *Neural computation*, 12(10):2451–2471, 2000.
- [48] L. Giglio, I. Csiszar, and C. O. Justice. Global distribution and seasonality of active fires as observed with the terra and aqua moderate resolution imaging spectroradiometer (MODIS) sensors. *Journal of Geophysical Research: Biogeosciences*, 111(G2), 2006.
- [49] L. Giglio, I. Csiszar, Á. Restás, J. T. Morissette, W. Schroeder, D. Morton, and C. O. Justice. Active fire detection and characterization with the advanced spaceborne thermal emission and reflection radiometer (aster). *Remote Sensing of Environment*, 112(6):3055–3063, 2008.
- [50] L. Giglio, J. Descloitres, C. O. Justice, and Y. J. Kaufman. An enhanced contextual fire detection algorithm for MODIS. *Remote sensing of environment*, 87(2):273–282, 2003.
- [51] L. Giglio, W. Schroeder, and C. O. Justice. The collection 6 MODIS active fire detection algorithm and fire products. *Remote Sensing of Environment*, 178:31–41, 2016.
- [52] J. M. Guerrero, G. Pajares, M. Montalvo, J. Romeo, and M. Guijarro. Support vector machines for crop/weeds identification in maize fields. *Expert Systems with Applications*, 39(12):11149–11155, 2012.
- [53] M. Guijarro, G. Pajares, I. Riomoros, P. Herrera, X. Burgos-Artizzu, and A. Ribeiro. Automatic segmentation of relevant textures in agricultural images. *Computers and Electronics in Agriculture*, 75(1):75–83, 2011.
- [54] W. Guo, U. K. Rage, and S. Ninomiya. Illumination invariant segmentation of vegetation for time series wheat images based on decision tree model. *Computers and electronics in agriculture*, 96:58–66, 2013.
- [55] T. Hague, N. Tillett, and H. Wheeler. Automated crop and weed monitoring in widely spaced cereals. *Precision Agriculture*, 7(1):21–32, 2006.
- [56] E. Hamuda, M. Glavin, and E. Jones. A survey of image processing techniques for plant extraction and segmentation in the field. *Computers and Electronics in Agriculture*, 125:184–199, 2016.

- [57] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.
- [58] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97, 2012.
- [59] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li. Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, 2015, 2015.
- [60] E. R. Hunt Jr, M. Cavigelli, C. S. Daughtry, J. E. Mcmurtrey III, and C. L. Walthall. Evaluation of digital photography from model aircraft for remote sensing of crop biomass and nitrogen status. *Precision Agriculture*, 6(4):359–378, 2005.
- [61] T. KATAOKA, T. KANEKO, H. OKAMOTO, et al. Crop growth estimation system using machine vision. In *Advanced Intelligent Mechatronics, 2003. AIM 2003. Proceedings. 2003 IEEE/ASME International Conference on*, volume 2, pages b1079–b1083. IEEE, 2003.
- [62] Y. J. Kaufman, C. O. Justice, L. P. Flynn, J. D. Kendall, E. M. Prins, L. Giglio, D. E. Ward, W. P. Menzel, and A. W. Setzer. Potential global fire monitoring from eos-MODIS. *Journal of Geophysical Research: Atmospheres*, 103(D24):32215–32238, 1998.
- [63] J. E. Keeley. Distribution of lightning and man-caused wildfires in california. In *Proceedings of the Symposium on Dynamics and Management of Mediterranean-type Ecosystems. USDA Forest Service General Technical Report PSW-58*, pages 431–437, 1982.
- [64] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [65] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas. Supervised machine learning: A review of classification techniques, 2007.
- [66] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [67] J. H. Le, A. P. Yazdanpanah, E. E. Regentova, and V. Muthukumar. A deep belief network for classifying remotely-sensed hyperspectral data. In *Advances in Visual Computing*, pages 682–692. Springer, 2015.
- [68] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.
- [69] Y. Li, Z. Cao, Y. Xiao, H. Lu, and Y. Zhu. A novel denoising autoencoder assisted segmentation algorithm for cotton field. In *Chinese Automation Congress (CAC), 2015*, pages 588–593. IEEE, 2015.
- [70] M. Lipsett and B. Materna. *Wildfire smoke: a guide for public health officials*. Office of Environmental Health Hazard Assessment, 2008.

- [71] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.
- [72] J. A. Marchant, R. D. Tillett, and R. Brivot. Real-time segmentation of plants and weeds. *Real-time imaging*, 4(4):243–253, 1998.
- [73] G. E. Meyer, T. W. Hindman, and K. Laksmi. Machine vision detection parameters for plant species identification. In *Photonics East (ISAM, VVDC, IEMB)*, pages 327–335. International Society for Optics and Photonics, 1999.
- [74] G. E. Meyer, J. C. Neto, D. D. Jones, and T. W. Hindman. Intensified fuzzy clusters for classifying plant, soil, and residue regions of interest from color images. *Computers and electronics in agriculture*, 42(3):161–180, 2004.
- [75] S. Min, B. Lee, and S. Yoon. Deep learning in bioinformatics. *arXiv preprint arXiv:1603.06430*, 2016.
- [76] P. Mitra, B. U. Shankar, and S. K. Pal. Segmentation of multispectral remote sensing images using active support vector machines. *Pattern recognition letters*, 25(9):1067–1074, 2004.
- [77] M. Möller, V. Alchanatis, Y. Cohen, M. Meron, J. Tsipris, A. Naor, V. Ostrovsky, M. Sprintsin, and S. Cohen. Use of thermal and visible imagery for estimating crop water status of irrigated grapevine. *Journal of experimental botany*, 58(4):827–838, 2007.
- [78] M. Montalvo, J. M. Guerrero, J. Romeo, L. Emmi, M. Guijarro, and G. Pajares. Automatic expert system for weeds/crops identification in images from maize fields. *Expert Systems with Applications*, 40(1):75–82, 2013.
- [79] J. T. Morissette, L. Giglio, I. Csiszar, and C. O. Justice. Validation of the modis active fire product over southern africa with aster data. *International Journal of Remote Sensing*, 26(19):4239–4264, 2005.
- [80] G. Mountrakis, J. Im, and C. Ogole. Support vector machines in remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(3):247–259, 2011.
- [81] S. W. Murphy, C. R. de Souza Filho, R. Wright, G. Sabatino, and R. C. Pabon. HOTMAP: Global hot target detection at moderate spatial resolution. *Remote Sensing of Environment*, 177:78–88, 2016.
- [82] P. Oliva and W. Schroeder. Assessment of viirs 375m active fire detection product for direct burned area mapping. *Remote Sensing of Environment*, 160:144–155, 2015.
- [83] C. Oppenheimer. Lava flow cooling estimated from landsat thematic mapper infrared data: the lonquimay eruption (chile, 1989). *Journal of Geophysical Research: Solid Earth*, 96(B13):21865–21878, 1991.
- [84] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas. How many trees in a random forest? In *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, pages 154–168. Springer, 2012.

- [85] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.
- [86] G. Pakhale and P. Gupta. Comparison of advanced pixel based (ann and svm) and object-oriented classification approaches using landsat-7 etm+ data. *International Journal of Engineering and Technology*, 2(4):245–251, 2010.
- [87] M. Patrick, K. Dean, and J. Dehn. Active mud volcanism observed with landsat 7 etm+. *Journal of Volcanology and Geothermal Research*, 131(3):307–320, 2004.
- [88] A. Patrignani and T. E. Ochsner. Canopeo: A powerful new tool for measuring fractional green canopy cover. *Agronomy Journal*, 107(6):2312–2320, 2015.
- [89] C. Persello and L. Bruzzone. Active and semisupervised learning for the classification of remote sensing images. *Geoscience and Remote Sensing, IEEE Transactions on*, 52(11):6937–6956, 2014.
- [90] G. P. Petropoulos, C. Kontoes, and I. Keramitsoglou. Burnt area delineation from a uni-temporal perspective based on landsat tm imagery classification using support vector machines. *International Journal of Applied Earth Observation and Geoinformation*, 13(1):70–80, 2011.
- [91] J. Reid and S. Searcy. An algorithm for separating guidance information from row crop images. *Transactions of the ASAE*, 31(6):1624–1632, 1988.
- [92] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.
- [93] V. F. Rodriguez-Galiano, B. Ghimire, J. Rogan, M. Chica-Olmo, and J. P. Rigol-Sanchez. An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 67:93–104, 2012.
- [94] O. Ronneberger, P. Fischer, and T. 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.
- [95] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *Computer Vision—ECCV 2006*, pages 430–443. Springer, 2006.
- [96] D. P. Roy, J. Ju, P. Lewis, C. Schaaf, F. Gao, M. Hansen, and E. Lindquist. Multi-temporal modis–landsat data fusion for relative radiometric normalization, gap filling, and prediction of landsat data. *Remote Sensing of Environment*, 112(6):3112–3130, 2008.
- [97] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988.

- [98] S. W. Running. Is global warming causing more, larger wildfires? *SCIENCE-NEW YORK THEN WASHINGTON-*, 313(5789):927, 2006.
- [99] A. L. Samuel. Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 3(3):210–229, 1959.
- [100] W. Schroeder, P. Oliva, L. Giglio, and I. A. Csiszar. The new viirs 375m active fire detection data product: Algorithm description and initial assessment. *Remote Sensing of Environment*, 143:85–96, 2014.
- [101] W. Schroeder, P. Oliva, L. Giglio, B. Quayle, E. Lorenz, and F. Morelli. Active fire detection using landsat-8/OLI data. *Remote Sensing of Environment*, 2015.
- [102] C. Schüldt, I. Laptev, and B. Caputo. Recognizing human actions: a local svm approach. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 32–36. IEEE, 2004.
- [103] U. Schulthess, T. Krupnik, Z. Ahmed, and A. McDonald. Technology targeting for sustainable intensification of crop production in the delta region of bangladesh. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(7):1475, 2015.
- [104] K. Shi, B. Yu, Y. Hu, C. Huang, Y. Chen, Y. Huang, Z. Chen, and J. Wu. Modeling and mapping total freight traffic in china using npp-viirs nighttime light composite data. *GIScience & Remote Sensing*, 52(3):274–289, 2015.
- [105] D. Shusterman, J. Z. Kaplan, and C. Canabarro. Immediate health effects of an urban wildfire. *Western journal of medicine*, 158(2):133, 1993.
- [106] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.
- [107] C. G. Snoek and M. Worring. Concept-based video retrieval. *Foundations and Trends in Information Retrieval*, 2(4):215–322, 2008.
- [108] B. Stocks, J. Mason, J. Todd, E. Bosch, B. Wotton, B. Amiro, M. Flannigan, K. Hirsch, K. Logan, D. Martell, et al. Large forest fires in canada, 1959–1997. *Journal of Geophysical Research: Atmospheres*, 107(D1), 2002.
- [109] P. M. Swan. *The Augustan succession: an historical commentary on Cassius Dio's Roman history, Books 55-56 (9 BC-AD 14)*. An American Philological Association Book, 2004.
- [110] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.
- [111] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

- [112] A. Tellaeche, G. Pajares, X. P. Burgos-Artizzu, and A. Ribeiro. A computer vision approach for weeds identification through support vector machines. *Applied Soft Computing*, 11(1):908–915, 2011.
- [113] G. R. Van Der Werf, J. T. Randerson, G. J. Collatz, L. Giglio, P. S. Kasibhatla, A. F. Arellano, S. C. Olsen, and E. S. Kasischke. Continental-scale partitioning of fire emissions during the 1997 to 2001 el nino/la nina period. *Science*, 303(5654):73–76, 2004.
- [114] G. R. van der Werf, J. T. Randerson, L. Giglio, G. Collatz, M. Mu, P. S. Kasibhatla, D. C. Morton, R. DeFries, Y. v. Jin, and T. T. van Leeuwen. Global fire emissions and the contribution of deforestation, savanna, forest, agricultural, and peat fires (1997–2009). *Atmospheric Chemistry and Physics*, 10(23):11707–11735, 2010.
- [115] L. Vilar, A. Camia, and J. San-Miguel-Ayanz. A comparison of remote sensing products and forest fire statistics for improving fire information in mediterranean europe. *European Journal of Remote Sensing*, 48:345–364, 2015.
- [116] M. Weiss, F. Baret, G. Smith, I. Jonckheere, and P. Coppin. Review of methods for in situ leaf area index (lai) determination: Part ii. estimation of lai, errors and sampling. *Agricultural and forest meteorology*, 121(1):37–53, 2004.
- [117] A. L. Westerling, H. G. Hidalgo, D. R. Cayan, and T. W. Swetnam. Warming and earlier spring increase western us forest wildfire activity. *science*, 313(5789):940–943, 2006.
- [118] D. Woebbecke, G. Meyer, K. Von Bargen, and D. Mortensen. Color indices for weed identification under various soil, residue, and lighting conditions. *Transactions of the ASAE*, 38(1):259–269, 1995.
- [119] D. M. Woebbecke, G. E. Meyer, K. Von Bargen, and D. A. Mortensen. Plant species identification, size, and enumeration using machine vision techniques on near-binary images. In *Applications in Optical Science and Engineering*, pages 208–219. International Society for Optics and Photonics, 1993.
- [120] L. Yu, N. Wang, and X. Meng. Real-time forest fire detection with wireless sensor networks. In *Wireless Communications, Networking and Mobile Computing, 2005. Proceedings. 2005 International Conference on*, volume 2, pages 1214–1217. IEEE, 2005.
- [121] Z. Yu, Z. Cao, X. Wu, X. Bai, Y. Qin, W. Zhuo, Y. Xiao, X. Zhang, and H. Xue. Automatic image-based detection technology for two critical growth stages of maize: Emergence and three-leaf stage. *Agricultural and forest meteorology*, 174:65–84, 2013.
- [122] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *Computer vision—ECCV 2014*, pages 818–833. Springer, 2014.
- [123] M. Zhizhin, C. D. Elvidge, F.-C. Hsu, and K. E. Baugh. Using the short-wave infrared for nocturnal detection of combustion sources in

- viirs data. *Proceedings of the Asia-Pacific Advanced Network*, 35:49–61, 2013.
- [124] G. Zhu and D. G. Blumberg. Classification using aster data and svm algorithms;: The case study of beer sheva, israel. *Remote sensing of Environment*, 80(2):233–240, 2002.
- [125] R. Zurita-Milla, J. G. Clevers, and M. E. Schaepman. Unmixing-based landsat TM and MERIS FR data fusion. *IEEE Geoscience and Remote Sensing Letters*, 5(3):453–457, 2008.