Geo-information Science and Remote Sensing

Thesis Report GIRS-2019-xx

BUILDING AN OPEN SOURCE SENSING SYSTEM FOR CONTINUOUS FOREST CANOPY MONITORING

Sam Vellekoop

September 26, 2019

WAGENINGEN
UNIVERSITY & RESEARCH

Building an Open Source Sensing System for Continuous Forest Canopy Monitoring

Sam Vellekoop

Registration number 94 01 16 864 050

Supervisors:

Benjamin Brede

Joao Valente

A thesis submitted in partial fulfilment of the degree of Master of Science

at Wageningen University and Research Centre,

The Netherlands.

September 26, 2019

Wageningen, The Netherlands

Thesis code number: GRS-804xx

Thesis Report: GIRS-2019-xx

Wageningen University and Research Centre

Laboratory of Geo-Information Science and Remote Sensing

Acknowledgements

Foremost, I would like to thank my supervisors Benjamin Brede and Joao Valente for their inspiration, support, feedback, dedication and perhaps most of all their patience. I'm very thankful to have worked on this thesis topic and although at times I've lost my interest in it, the ideas of Joao and Benjamin helped spark it again. Although their ideas did not always match with each other, the combination of different opinions helped guide this thesis in the right direction.

Furthermore, I would like to thank Jan Clevers and Lammert Kooistra for giving me the opportunity to work as a student assistant for Advanced Earth Observation and Remote Sensing and GIS Integration, allowing me to focus my thoughts on something else for a while. I also very much enjoyed the excursion/vacation to Groenlo.

I would like to thank my family -Jacq(ueline), Rokus, Loek, Stella and Balou- for their constant support and for making my weekends in The Hague still feel like home.

Finally, I would like to thank Marjanne for always being there for me and giving me the essential "schop onder de kont" to finalize this thesis.

Abstract

Sam Vellekoop

Building an Open Source Sensing System for Continuous Forest Canopy Monitoring

Monitoring forest canopy characteristics is important for keeping track of forest ecosystem health and is invaluable in forest management. Leaf area index (LAI) is one of the most used indicators for the biophysical plant canopy characteristics in a wide range of research domains and is an important indicator for canopy water interception, radiation extinction, evapotranspiration and biomass production. Large-scale monitoring of LAI is generally done using air- or spaceborne sensors. However, these sensors require extensive calibration and validation using *in situ* LAI estimates. Conventional methods for *in situ* LAI measurement are generally either labour-intensive or require expensive equipment, making them less suited for monitoring LAI for a larger area for an extended period. This is were automated sensors could provide an opportunity. The goal of this research was to design, build and test an automated sensing system for continuous canopy monitoring using open source hardware and software. Using a bare minimum of materials, includng a Raspberry Pi and camera with fisheye lens, the PiLAI was developed, a low-cost, automatic sensing system for LAI. The PiLAI was tested against terrestrial laser scanning (TLS) and conventional digital hemispherical photography (DHP) in two forests in the Netherlands during autumn and spring. With a Pearson correlation of up to 0.647 between the PiLAI and TLS and up to 0.851 between PiLAI and DHP, the PiLAI shows potential as automated alternative for conventional methods.

Chapter 1 introduces the topic, explains the problem and describes the research objective and questions. Chapter 2 elaborates on the background of existing *in situ* LAI determination methods as well as open source hardware and software. The study areas used for this thesis are described in chapter 3. Chapter 4 is split in two parts. The first part covers the development and a description of the PiLAI sensor system and the second part covers the materials and methods used to test the PiLAI. Results are found in chapter 5 and are discussed in-depth in chapter 6. Final conclusions are found in chapter 7.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

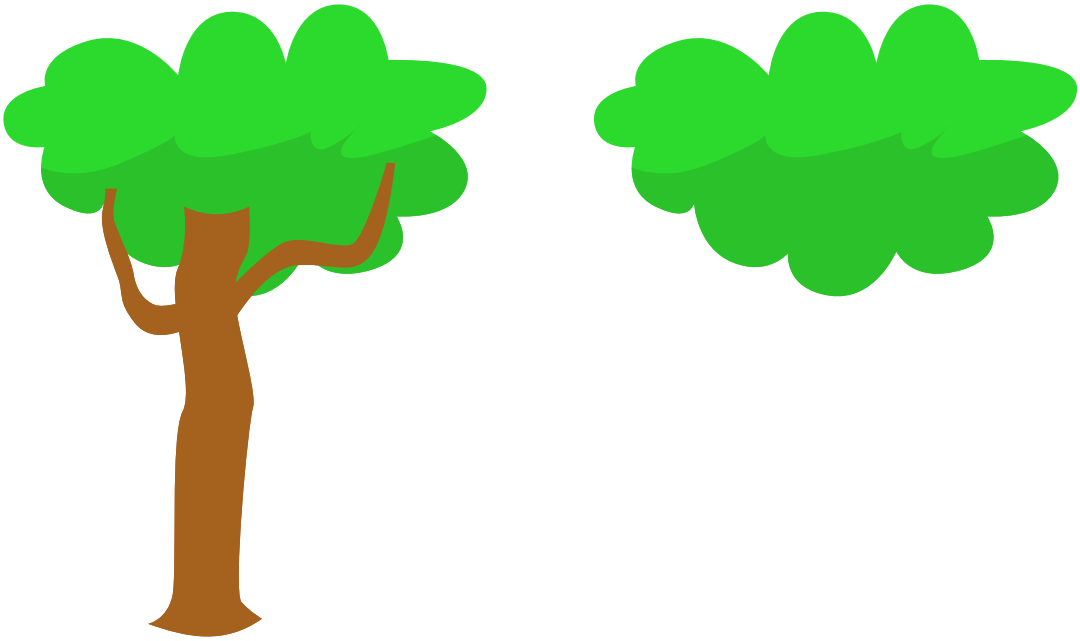
1.1 Motivation

Forest ecosystems provide essential ecological and economical services. Both nearby communities and wildlife benefit from these services and are dependent on them. Forest ecosystems provide services on a larger scale as well, serving as carbon sinks and playing an important role in nutrient cycles. Efficient nutrient cycling is important for ecosystem health and thus for the services the forest ecosystem provides. One of the main driving factors behind nutrient cycling is the canopy of the forest, influencing below-canopy temperature and moisture as well as providing litter for nutrient recycling (Prescott, 2002). Monitoring forest canopy characteristics is therefor important for keeping track of forest ecosystem health and invaluable in forest management.

Leaf area index (LAI) is one of the most used indicators for the biophysical plant canopy characteristics in a wide range of research domains (Fang, Ye, et al., 2018). LAI is an important indicator for canopy water interception, radiation extinction, evapotranspiration and biomass production. As such, LAI is frequently used as input parameter for biomass production models. LAI is defined as the total one-sided area of leaf tissue per unit of ground surface area (Watson, 1947). An index that is often used in tandem with the LAI is the plant area index (PAI), which includes the surface area of other above-ground vegetation as well. The PAI is generally calculated instead of the LAI when no division can be made between leaves and other above-ground vegetation. Since PAI and LAI are closely related and LAI is existing literature generally only mentions LAI, both LAI and PAI will henceforth be referred to as LAI.

Although the use of LAI is widespread, measuring LAI can be difficult, especially due to its spatial and temporal variability (Breda, 2003). Large-scale estimates of LAI can be made using air- and spaceborne remote sensing. The potential of these methods to monitor LAI on national, continental or global scale has been shown in a variety of studies (J. M. Chen, Pavlic, et al., 2002; Colombo et al., 2003; Tillack et al., 2014; Zhang et al., 2014; Alonzo et al., 2015). However, due to the complex nature of air- and spaceborne remote sensing, the quality of these methods depend on adequate *in situ* validation data (Calders, Origo, et al., 2018). This validation data preferably covers varying types of forests and has a high temporal resolution.

Conventional methods for *in situ* LAI measurement are generally either labour-intensive or require expensive equipment, making them less suited for monitoring LAI for a larger area for an extended period. This problem increases when a higher temporal resolution is required or the area is difficult to access. A low-cost, automated sensing system could provide a solution to this problem. Automation of the measuring process reduces the need for (skilled) labour, reducing monitoring costs. By keeping the costs of the sensing system low, a vast network of sensors can be used to cover larger areas.



(A) PAI (B) LAI

FIGURE 1.1: Comparison between PAI (all above-ground biomass) and LAI (leaf area)

1.2 Research goals

The primary goal of this research was to design, build and test an automated sensing system for continuous canopy monitoring using open source hardware and software. Secondary goals were to assess the potential and limitations of open source sensing systems for niche applications.

In order to achieve the mentioned research objectives, the following research questions have been formulated:

RQ 1 How can an automated sensor for LAI measurements be build using open source hardware?

RQ 2 How should the software environment for the sensor be set up?

RQ 3 How does the accuracy of the designed sensor compare to that of a more consolidated system?

RQ 4 What are the possibilities and limitations of the designed system?

Chapter 2

Background

This chapter covers a short description of conventional *in situ* or terrestrial methods for measuring LAI as well as an introduction to open source and its relevance to the development of scientific tools.

2.1 Terrestrial methods for LAI measurement

A wide range of terrestrial methods exist for LAI determination. Breda (2003) and Jonckheere, Fleck, et al. (2004) describe and review the most consolidated of these methods. The methods can be divided in direct methods, which measure the LAI directly, and indirect methods, which measure one or more variables (proxies) from which LAI can be derived.

2.1.1 Direct methods

The most consolidated methods of direct LAI determination are destructive sampling, the model tree method and litter traps. Of these, only the latter is non-destructive. Destructive sampling is done by harvesting the trees on a sampling plot and determining the LAI through either planimetric or gravimetric approaches. The planimetric approach measures the area of the individual leaves using scanning planimeters (*e.g.* LI-3000C, LI-3100C (LICOR, 2016)), which use a scanner over which the leaves are drawn, or video image analysis (*e.g.* WinDIAS 3 (Delta-T Devices Ltd, 2017)), which uses a digital camera and computer vision to separate the leaves from the background. The area of individual leaves is then used to calculate the LAI. The gravimetric approach uses the correlation of dry leaf weight and LAI to determine LAI. This requires the leaf mass per area (LMA) to be determined beforehand. Once the LMA has been determined (with the planimetric approach), the dried leaves can be weighed to determine LAI. This makes the process significantly less labour-intensive than the planimetric approach when used in relatively homogeneous tree stands, since once the LMA has been determined, the leaves do not have to be measured individually anymore. However, the LMA can vary even within tree species, requiring the LMA to be determined multiple times when used in less homogeneous tree stands (Jonckheere, Fleck, et al., 2004).

For the model tree method a small number of trees that are representative of the forest stand are chosen, destroying less vegetation than destructive sampling and making the method less labour-intensive. Using this method, heterogeneity in the forest stand is not captured as well as using the full destructive sampling method.

Litter traps are a non-destructive, direct method for LAI determination. The traps consist of a box in which falling leaves are caught. LAI is then determined using the gravimetric approach. This method can be up-scaled easily and generally yields accurate LAI estimates. As this method only yields one LAI value a year, it cannot

be used in year-round forest canopy monitoring. Furthermore, the method does not work for tree stands that contain evergreens, as the weight of the needles cannot be correlated to the LAI of the current or previous year (J. M. Chen, Rich, et al., 1997).

Direct methods of measurement have generally been regarded as the most accurate, however, due to the labour-intensity of the methods, only a limited number of measurements can be taken, making the methods unsuitable for monitoring. Up-scaling can form a problem as well, especially in heterogeneous forests (Jonckheere, Fleck, et al., 2004).

2.1.2 Indirect methods

Indirect methods of LAI determination can be subdivided in contact methods and non-contact (optical) methods. Contact methods are allometric techniques and inclined point quadrats. Non-contact or optical methods include laser point quadrats, gap size distribution (TRAC (J. M. Chen and Cihlar, 1995)) analysis and gap fraction analysis (Jonckheere, Fleck, et al., 2004).

Allometric techniques use measured dimensions of trees to determine the LAI. This relatively simple technique (in idea, getting the correct measurements can sometimes be difficult) can be used with just the circumference and height of the tree or with an extensive series of measurements. Allometric techniques are generally labour-intensive and hard to up-scale in a forest with a variety of tree species. Furthermore, as only the woody parts of the tree are measured to determine the LAI, seasonality and tree health are not taken into account, making these techniques unsuited for LAI monitoring.

For the inclined point quadrats method, a long needle is used to penetrate the canopy at known zenith angle (θ), counting the number of contacts the needle makes with leaves (Wilson, 1960). LAI (L) can be calculated as:

$$L = \frac{N_\theta}{K_\theta}$$

where N_θ is the number of contacts at zenith angle θ and K_θ is the extinction coefficient at zenith angle θ. Accurate estimates of LAI require a substantial number of measurements, making the method labour-intensive. Furthermore, inclined point quadrats are increasingly hard to use with higher canopies, making the method unsuited for determining LAI in high canopy forests (J. M. Chen, Rich, et al., 1997; Jonckheere, Fleck, et al., 2004).

The laser point quadrats method follows the principles of inclined point quadrats, substituting the long needle with a laser. This makes the method more practical in its use, especially for high canopy forests. The substitution of a physical needle with a laser significantly decreases the time and labour required for the method as well. Studies with terrestrial laser scanners (TLS) have shown this is a robust method for measuring forest canopies (Seidel, Fleck, and Leuschner, 2012; Calders, Armston, et al., 2014; Woodgate et al., 2015; Calders, Schenkels, et al., 2015). A downside of this method is the high cost of the TLS.

Optical terrestrial methods for LAI determination are primarily based on analysis of gap fraction and gap size distribution. This analysis can be done with a variety of instruments. The Tracing Radiation and Architecture of Canopies (TRAC) has been succesfully used in different studies for determining LAI using gap size distribution (J. M. Chen and Cihlar, 1995; J. M. Chen, Pavlic, et al., 2002). Gap fraction analysis can be done with both widely used commercial products (*e.g.* LI-COR LAI-2200C) and (digital) hemispherical photography (DHP) (Fang, W. Li, et al., 2014; Fang, Ye, et al., 2018; Woodgate et al., 2015; Pearse, Watt, and Morgenroth, 2016). LAI determination through gap fraction analysis requires segmentation of the canopy images

using a thresholding method, after which the gap fraction can be determined for each annulus (ring around the image centre representing the same view zenith angle), from which the LAI can be derived (Jonckheere, Nackaerts, et al., 2005). The simple nature of the material needed for DHP, a digital camera and a fish-eye lens, make it a suitable candidate for a low-cost sensing system.

Recently automatic methods for canopy monitoring have been developed. Lecerf et al. (2010) and Brede, Gastellu-Etchegorry, et al. (2018) have shown the potential of the PASTIS-57, a system that uses transmittance sensors to derive the LAI. Arduino and Raspberry Pi boards have gained in popularity and have been used to sense the environment (Ferdoush and X. Li, 2014; Kirby, L. Chapman, and V. Chapman, 2018). X. Li et al. (2015) have designed an automatic LAI sensor for crop LAI.

2.2 Open source

A problem with existing *in situ* methods is the dominance of closed source or proprietary standards. Proprietary software, in contrast to open source software, hides its internal architecture, the code. Without access to the code, users depend on the developer of the software to fix errors and add features. In some cases errors might not be found at all and, without the user being aware of it, influence the data collection or analysis. This creates a 'black box' where the process by which the output is obtained is unknown to the user. Opening the 'black box' by using open source software can be beneficial to science as it gives insight in the processes that deliver the output, errors can be found and, with enough expertise, fixed by the users themselves and inherent limitations of the software are known to the user. This can improve the reliability of measurements and data analysis.

Open source software is not necessarily user- or community-driven. Requirements for open source software include free distribution of the software including the source code and permission for others to modify and redistribute the software[1]. Following these requirements ensures that the software can be inspected and modified by others. Third parties can add new features to the software, removing the need to wait for the developer of (proprietary) software to approve of and work on a new feature request. This can improve software interoperability as well. Although proprietary software packages from a single developer is generally well integrated with each other, interoperability with software from third parties can be problematic. Since the source code is not known to the other party, it is difficult to find a solution that functions as a bridge for the software packages.

[1] https://opensource.org/osd

Chapter 3

Study area

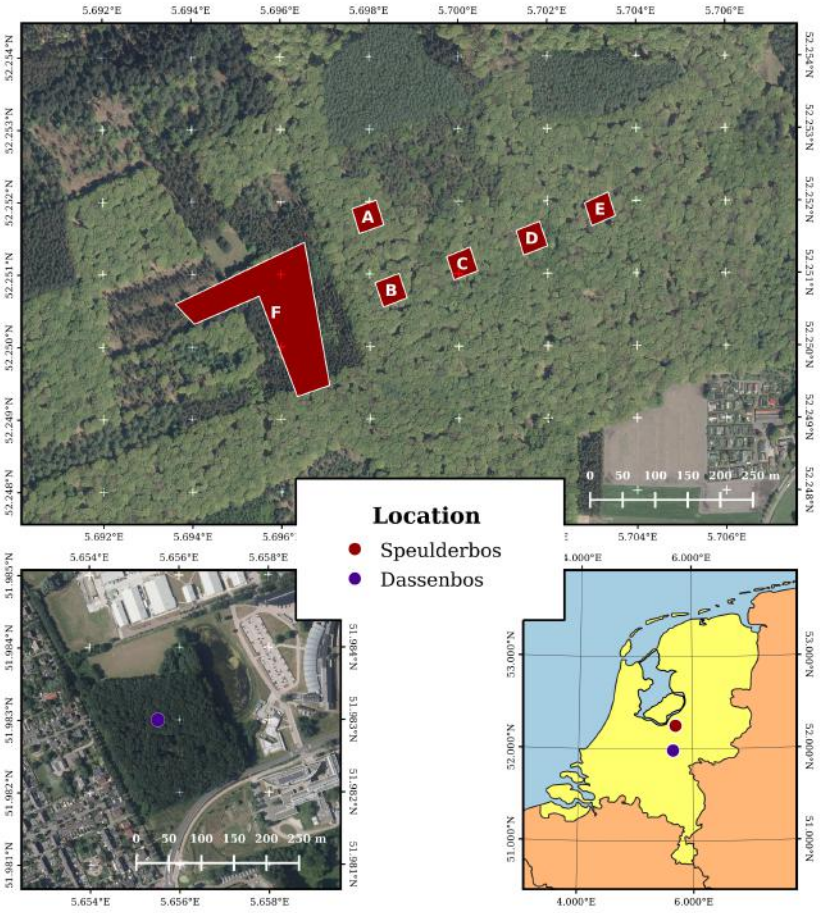


FIGURE 3.1: Study areas Speulderbos (upper) and Dassenbos (lower, left).

Dassenbos and Speulderbos (Figure 3.1) were chosen as study areas for this research.

3.1 Dassenbos

The Dassenbos study area is a small forest plot (*circa* 3.4 ha) located in Wageningen, the Netherlands at 51.983° N, 5.656° E and is regularly used as study area by researchers from Wageningen University and Research. The forest consists mainly of Oak (*Quercus* spp.) and Birch (*Betula* spp.). The plot used for this research consisted of *circa* 85% Oak and 15% Birch (Calders, Schenkels, et al., 2015).

3.2 Speulderbos

Speulderbos is a *circa* 3300 ha large forest plot located in the Netherlands. The study area used is located at 52.251° N, 5.699° E, close to Garderen. The study area consists mainly of Beech (*Fagus* spp.) at plots A, B, C, D and E as well as Giant Fir (*Abies grandis*), Norway Spruce (*Picea abies*) and Douglas Fir (*Pseudotsuga menziesii*) at plot F (Table 3.1). The Speulderbos study area is used as reference site for biophysical forest characteristics and is monitored using a variety of of terrestrial, airborne and spaceborne sensors (Brede, Bartholomeus, et al., 2016; Brede, Gastellu-Etchegorry, et al., 2018).

Table 3.1: Speulderbos measurement plots.

Plot	Subplot	Description
A	b01	*Fagus* spp.
A	b02	*Fagus* spp.
A	b03	*Fagus* spp.
A	b04	*Fagus* spp.
A	a05	*Fagus* spp.
B	b01	*Fagus* spp.
B	b02	*Fagus* spp.
B	b03	*Fagus* spp.
B	b04	*Fagus* spp.
B	a05	*Fagus* spp.
C	b01	*Fagus* spp.
C	b02	*Fagus* spp.
C	b03	*Fagus* spp.
C	b04	*Fagus* spp.
C	a05	*Fagus* spp.
D	b01	*Fagus* spp.
D	b02	*Fagus* spp.
D	b03	*Fagus* spp.
D	b04	*Fagus* spp.
D	a05	*Fagus* spp.
E	b01	*Fagus* spp.
E	b02	*Fagus* spp.
E	b03	*Fagus* spp.
E	b04	*Fagus* spp.
E	a05	*Fagus* spp.
F	f01	*Fagus* spp. and *Abies grandis* (close to road)
F	f02	*Abies grandis* (rows)
F	f03	*Abies grandis* (rows)
F	f04	*Fagus* spp. and *Abies grandis*
F	f05	*Abies grandis* (dense)
F	f06	*Abies grandis* (dense)
F	f07	*Abies grandis* (close to road)
F	f08	*Fagus* spp. and *Picea abies*
F	f09	*Picea abies*
F	f10	*Picea abies*
F	f11	*Picea abies*
F	f12	*Pseudotsuga menziesii*
F	f13	*Pseudotsuga menziesii*
F	f14	*Pseudotsuga menziesii*
F	f15	*Pseudotsuga menziesii*
F	f16	*Pseudotsuga menziesii*

Chapter 4

Methodology

4.1 PiLAI

4.1.1 Hardware

The core of the PiLAI sensor system was designed to consist of only a power supply, a Raspberry Pi as main processing unit and a camera module with fish-eye lens as the actual sensor. The system was enclosed in a weatherproof box, with the camera module covered by a transparent dome. One additional module that was used was a voltage regulator to convert the 12 V from the battery to a usable 5 V for the Raspberry Pi. The PiLAI is shown closed and opened in Figures 4.1 and 4.2 respectively. Materials for the system components can be found in Table 4.1.

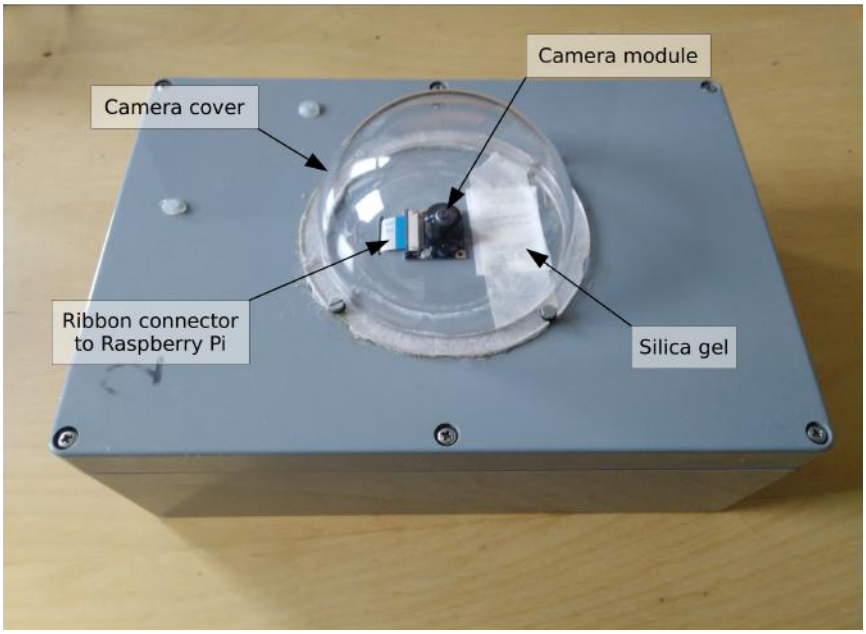


FIGURE 4.1: PiLAI sensor system when closed.

A Raspberry Pi 2B was chosen as the main processing unit as it has relatively high processing power and enough random access memory (RAM) for image acquisition and processing, runs primarily on open source software and is well supported by the community. This low-cost single-board computer is frequently used in (open source) projects and has proven to be a versatile tool. Although not consisting of open hardware, most specifications and mechanical drawings are publicly available and the board runs primarily open-source software. True open hardware alternatives, *e.g.* Arduino, are generally too low-powered to be used in Mega-pixel image acquisition and processing.

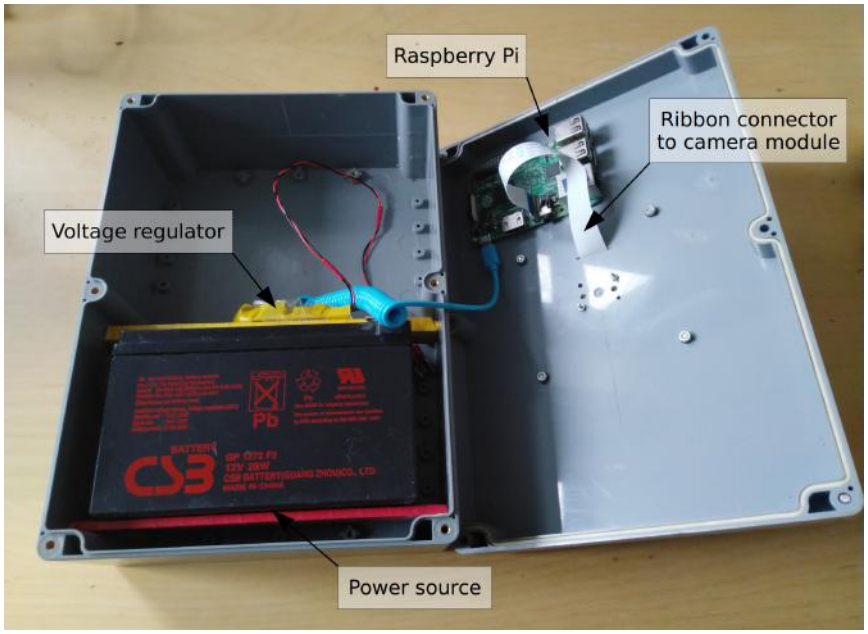


FIGURE 4.2: PiLAI sensor system when opened.

TABLE 4.1: List of materials for the PiLAI.

Component	Product	Purchased from / provided by
Processing unit	Raspberry Pi 2B	reichelt.com (33 €)
Camera module	Joy-IT 5MP Wide-Angle Camera	reichelt.com (37 €)
Power source	12 V, 7 Ah lead-fleece battery	Wageningen University
Voltage regulator	LM2596 3A DC-DC converter	Wageningen University
Storage unit	Transcend MicroSDHC-Card 32 GB	reichelt.com (7 €)
Casing	RND 265 * 165 * 95 mm ABS box	reichelt.com (13 €)
Camera cover	10 cm acrylic dome	aliexpress.com (11 €)

The camera module, produced by Joy-IT, consisted of an Omnivision OV5647 CMOS sensor and a 200° fish-eye lens with equidistant projection. Contrary to the specified requirements, the camera module does not consist of open source hardware. This is due to a lack of open source options for Mega-pixel camera sensors. Although the hardware itself is not open source, the driver software does largely consist of open source software. This camera module has been chosen due to its open source interface, fish-eye lens with large field of view and equidistant projection and relatively low cost. The open source interface gives virtually complete control over the camera settings (*e.g.* shutter speed, ISO value), making it a preferred choice over the OpenCV interface which is more generic and complex and works only with supported cameras (OpenCV, 2018).

The sensor system was enclosed in a 265 * 165 * 95 mm ABS box with an Ingress Protection Marking of 65 (IP 65) and resistant to temperatures between -20 and 80 °C, making it weatherproof under most circumstances. The camera module was mounted on the lid of the box, covered by a clear acrylic dome with a diameter of 10 cm.

4.1.2 Software

The Raspberry Pi was operated by Raspbian (Stretch, version 2018-06-27), a Debian-based Linux distribution specifically designed for ARM processor architectures and

the officially supported operating system for the Raspberry Pi. Python 2.7.9 was used for designing the software on the PiLAI.

A Python image acquisition script, responsible for setting the camera parameters, capturing images, testing image for over-exposure and saving the image to a file, was created using the opencv-python, numpy and picamera packages. This `lai_camera.py` is called upon by `timer.py`, which continuously tests whether the capture interval has passed. Once the capture interval has passed, `timer.py` instructs `lai_camera.py` to open the camera device and automatically sets the shutter time. If the queried shutter time exceeds 100 ms, the ISO value is increased by 100 and shutter time is limited to 100 ms. This was done to ensure a short shutter time to prevent errors from trees swaying in the wind. After the shutter time and ISO are set an image is taken. This image is tested for under-/overexposure by `lai_camera.py` using the `check_histogram()` function. This function returns the percentage of saturated pixels, pixels with a digital number (DN) of 255. If this percentage is greater than 0.2%, the image is classified as overexposed and the shutter time is reduced by 20% before capturing a new image and testing it. If instead no saturated pixels are found, the image is classified as underexposed and shutter time is increased by 50% before capturing a new image and testing it. A maximum of 10 tries is allowed before storing the image as 3-channel PNG file. If no under-/overexposure is found, the image is stored immediately.

With the data acquisition scripts solely available for the PiLAI itself, the data processing and analysis scripts were designed to run on either the PiLAI or a personal computer. The Python script `image_io.py` is responsible for reading and writing acquired images in a wide variety of formats. With the captured images read, `image_processor.py` is responsible for classifying the images using 3 classes: vegetation, sky and mixed. The script offers 2 separate classification algorithms, iterative selection/K-means clustering (Ridler and Calvard, 1978) and Otsu's algorithm, with the possibility for the user to add new classification algorithms if needed. After classification, `image_io.py` stores the classified image.

Classified images are analysed by `image_analyzer.py`, which creates an image overlay similar to the one shown in Figure 4.3. Gap fraction is calculated for each segment, enabling the calculation of the clumping factor. The clumping factor corrects for the clumping of the canopy when this deviates from a random case (S. G. Leblanc et al., 2005). This can have a significant impact on the calculated LAI, especially for highly clumped canopies. Gap fractions are calculated by taking the mean pixel value of the segment and dividing it by 100. For segments with a gap fraction of 0, a gap fraction of 0.00001 is assigned in order to prevent division by 0 in the calculation of the clumping factor. The clumping factor for each annulus is calculated using the log-average method (Lang, 1986; S. G. Leblanc et al., 2005):

$$\Omega_{log}(\theta) = \frac{\ln[P(\theta)]}{\overline{\ln[P(\theta)]}} \tag{4.1}$$

where $\Omega_{log}(\theta)$ is the clumping factor for an annulus with an average zenith angle θ, $P(\theta)$ is the average gap fraction for zenith angle θ and $\overline{\ln[P(\theta)]}$ is the logarithmic average of all segments.

Using the calculated gap fractions and clumping factors, LAI is calculated using Equation 4.2 (Fernandes et al., 2001):

$$L = -2 \int_0^{\pi/2} \frac{\ln[P(\theta)]}{\Omega_{log}(\theta)} \cos\theta \sin\theta d\theta \tag{4.2}$$

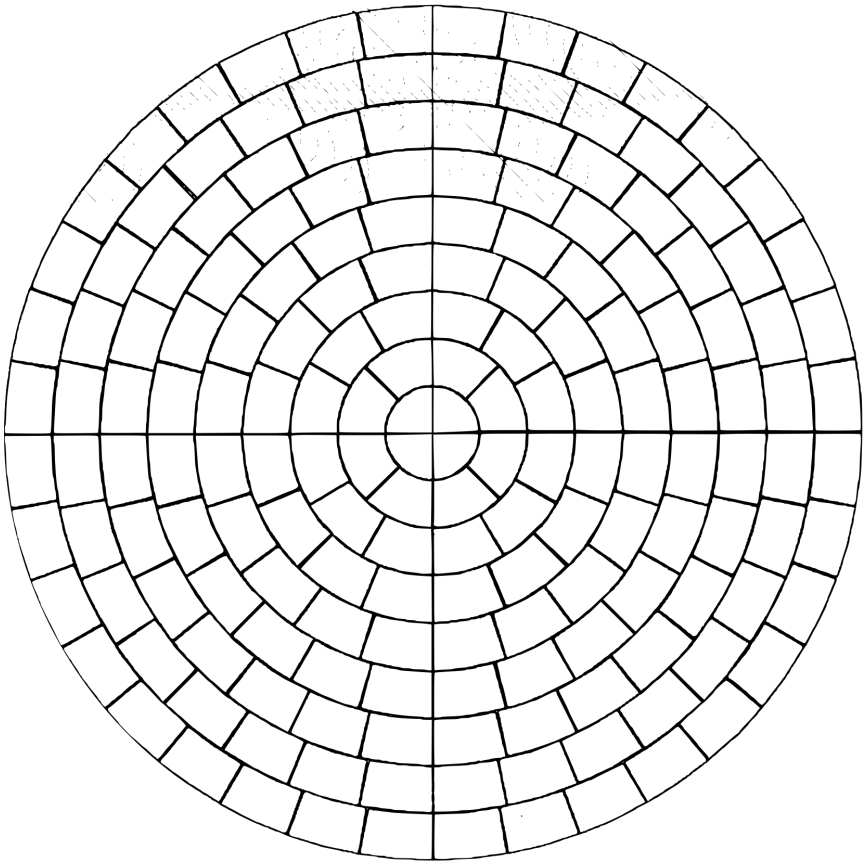


FIGURE 4.3: Overlay for the analysis of the classified images. Annuli are drawn as well as inter-annular segments (Gardingen et al., 1999).

4.2 Data Acquisition

Data was acquired in both Dassenbos and Speulderbos. Images were acquired in Dassenbos for the intervals 2018-11-04 till 2018-11-26 (autumn dataset) and 2019-04-13 till 2019-05-15 (spring dataset), as shown in Table 4.2. These images cover a variety of weather types and have been captured at multiple times during the day at intervals of 30 minutes.

Data collection in Speulderbos was done on 2018-12-06, between 12:30 and 14:30 (UTC+1). The sky was completely cloudy (8 oktas), with a relative humidity of 96% and a light drizzle (1.9 mm). Temperatures were between 7.3 and 12.3 °C with and average of 10.9 °C. Wind speeds were generally below 10 m/s[1]. Data was collected at 41 locations within Speulderbos (3.1) using the PiLAI, TLS and DHP. All measurements were taken from a leveled tripod at a height of 1.45 m.

Measurements with the PiLAI were taken by placing the system on the tripod and leaving it there for a number of minutes, with the PiLAI taking an image every 30 seconds. Settings for the PiLAI can be found in Table 4.3.

TLS measurements were done using a RIEGL VZ-400 TLS (Figure 4.4), provided by Wageningen University. The VZ-400 is an active system and uses a wavelength of 1550 nm (near infrared), making the influence of light conditions negligible. Internal sensors record information for the pitch, roll and yaw of the VZ-400 (RIEGL, 2017). For each location a scan using an azimuth angle range of 360° was made while the VZ-400 was is upright position. Due to the limiting angle zenith range (30° to 130°), a second scan was required to capture the full hemisphere. This was done with the VZ-400 in tilted position. Settings for the VZ-400 are summarized in Table 4.4

[1]Source: KNMI

Table 4.2: Dassenbos data acquisition dates.

Date	**Rain (mm / day)**	**Cloud cover (oktas)**	**Humidity (%)**	**Measurements**
2018-11-04	0	4	90	12
2018-11-05	0	8	95	10
2018-11-06	0	8	79	12
2018-11-09	0	7	88	19
2018-11-16	0	2	92	10
2018-11-22	0	8	86	9
2018-11-23	0	6	89	8
2018-11-26	0	8	88	9
2019-04-13	0	3	75	20
2019-04-14	0	4	68	3
2019-04-15	0	3	59	14
2019-04-18	0	5	42	13
2019-04-20	0	0	43	7
2019-04-22	0	4	42	8
2019-04-23	0	6	50	1
2019-04-24	9	7	64	18
2019-04-25	5	7	79	8
2019-04-26	0	7	72	10
2019-04-27	5	8	76	15
2019-04-29	0	6	75	20
2019-05-02	4	7	86	16
2019-05-03	2	8	73	18
2019-05-09	2	7	78	24
2019-05-10	1	8	77	11
2019-05-14	0	4	62	19
2019-05-15	0	4	51	14

Table 4.3: Settings for the PiLAI.

Setting	**Value**
Image width	2592
Image height	1944
Auto white balancing	sunlight
Starting ISO	100
Max shutter time	100 ms
Max percentage saturated pixels	0.2%

FIGURE 4.4: RIEGL VZ-400 TLS.

TABLE 4.4: Settings for the RIEGL VZ-400.

Setting	**Value**
Theta angle start	30.00°
Theta angle stop	130.00°
Theta angle increment	0.06°
Phi angle start (upright)	0.00°
Phi angle stop (upright)	360.00°
Phi angle start (tilted)	230.00°
Phi angle stop (tilted)	490.00°
Phi angle increment	0.06°
Measurement program	High speed

DHP was done using a Nikon D7000 digital SLR camera with a Sigma 4.5 mm F2.8 fish eye lens. Images were captured in sets of three for each location using exposure bracketing, one with limited overexposure, one with limited underexposure and one in between. This was done using varying ISO and shutter time values. Camera settings are summarized in Table 4.5. Images were saved in Nikon Electronic Format (.NEF), Nikon's RAW image file format. This was done to ensure that all data (including acquisition parameters) was saved.

TABLE 4.5: Settings for the Nikon D7000.

Setting	**Value**
Image width	4992 pixels
Image height	3280 pixels
Aperture	F4
Exposure program	Aperture priority
Exposure mode	Auto
Shooting mode	Exposure bracketing

4.3 Data Processing

4.3.1 PiLAI

PiLAI data was processed using Python (version 3.7.4). Captured images were masked to the preferred field of view, converting degrees to pixels using Equation 4.3:

$$r_{pixel} = f * \frac{\theta * \pi}{180} / l_{pixel} \tag{4.3}$$

where r_{pixel} is the radius of the circle in pixels from the focal centre, f is the focal length of the lens, θ is half the preferred field of view and l_{pixel} is the pixel size. The formula assumes a lens with perfect equidistant distortion. Visual analysis of the distortion, using a calibration board (Figure 4.5), showed that this assumption is sufficiently met by the used lens, especially for the area of interest (FOV $< 160°$).

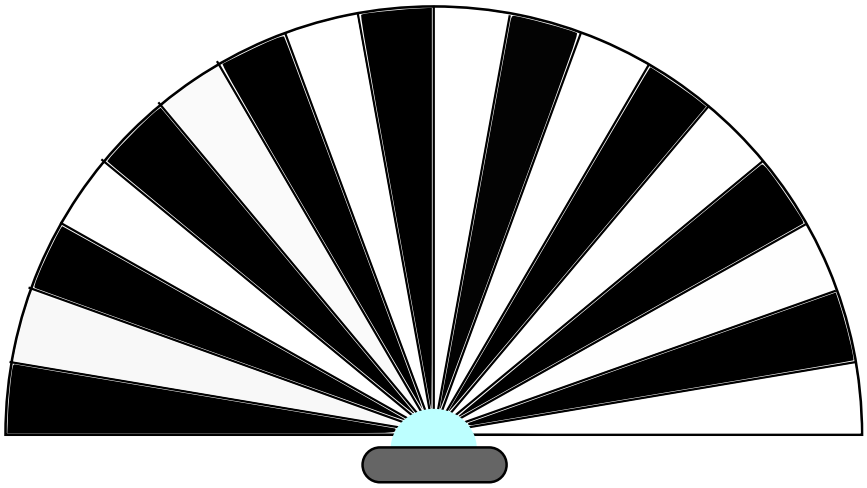


FIGURE 4.5: Calibration board. When there is perfect equidistant distortion, the image shows the same distance between the angle intervals.

After masking the pixels outside the preferred FOV, classification was done using both Otsu's method and iterative selection (K-means clustering) in order to asses the influence of different classification algorithms on the results. Both thresholding methods were applied to the masked RGB images and only the blue band of these. Due to limitations of the OpenCV library, the masked images needed to be converted to 8-bit grayscale images in order to perform the thresholding using Otsu's method.

Due to the high number of mixed pixels (pixels with both plant and sky in them, caused by a relatively low image resolution), iterative selection was done using 3 classes; sky (100), plant (0) and mixed (0 - 100). Mixed class pixels were assigned a value between 0 and 100 based on their place in the histogram. This step was done to improve consistency and accuracy.

Image analysis was done by creating an overlay with a FOV of 160 ° and a angular resolution of 5 ° (giving an overlay with 16 annuli). Inter-annular segments, as proposed by Gardingen et al. (1999), were created, starting with 4 segments at the center and adding 4 segments for each consecutive annulus. Gap fractions were calculated for each segment enabling the calculation of the clumping factor and LAI using Equations 4.1 and 4.2 respectively.

TABLE 4.6: Settings for preprocessing with rawpy.

Setting	Value
Demosaic algorithm	AHD
Noise threshold	Disabled
Auto-bright threshold	0.01
Gamma	(2.222, 4.5)
Chromatic aberration	(1, 1)

TABLE 4.7: Settings for CanEye.

Setting	Value
Image size	(3280, 4948)
Optical center	(1640, 2474)
Projection function	Equidistant
COI	60°
Angular resolution	5°
Clumping parameter	8

4.3.2 DHP

In order to process the RAW images from the Nikon D7000, conversion to .JPG was done using Python and the rawpy (version 0.13.1) library. Parameters for the rawpy.RawPy.preprocess() function were left as default and are shown in Table 4.6.

After preprocessing, the images were processed using CanEye (version 6.4.9). Processing parameters for CanEye are shown in Table 4.7. The images were classified using CanEye's supervised classification using 3 classes (sky, vegetation and mixed pixel).

After data acquisition, TLS data was processed and provided by Wageningen University. LAI was calculated according to Calders, Schenkels, et al. (2015).

4.4 Analysis

For the analysis of the performance of the PiLAI, LAI and gap fractions for each annulus measured with the PiLAI were compared to those measured using the Nikon D7000 and RIEGL VZ-400. This was done for the Speulderbos dataset, using Pearson correlation and the root mean square error (RMSE). An inter-comparison between the Nikon D7000 and RIEGL VZ-400 was done as well. RMSE was calculated as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2}$$

In order to determine the reliability of the PiLAI measurements under different weather and light conditions, the Dassenbos autumn and spring datasets were used to calculate the standard deviation for different measurements taken on the same day. Standard deviation was calculated as:

$$SD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

Analysis was done using Python with the pandas (version 0.24.0), numpy (version 1.16.1), scipy (version 1.2.0) and matplotlib (version 3.0.2) packages.

Chapter 5

Results

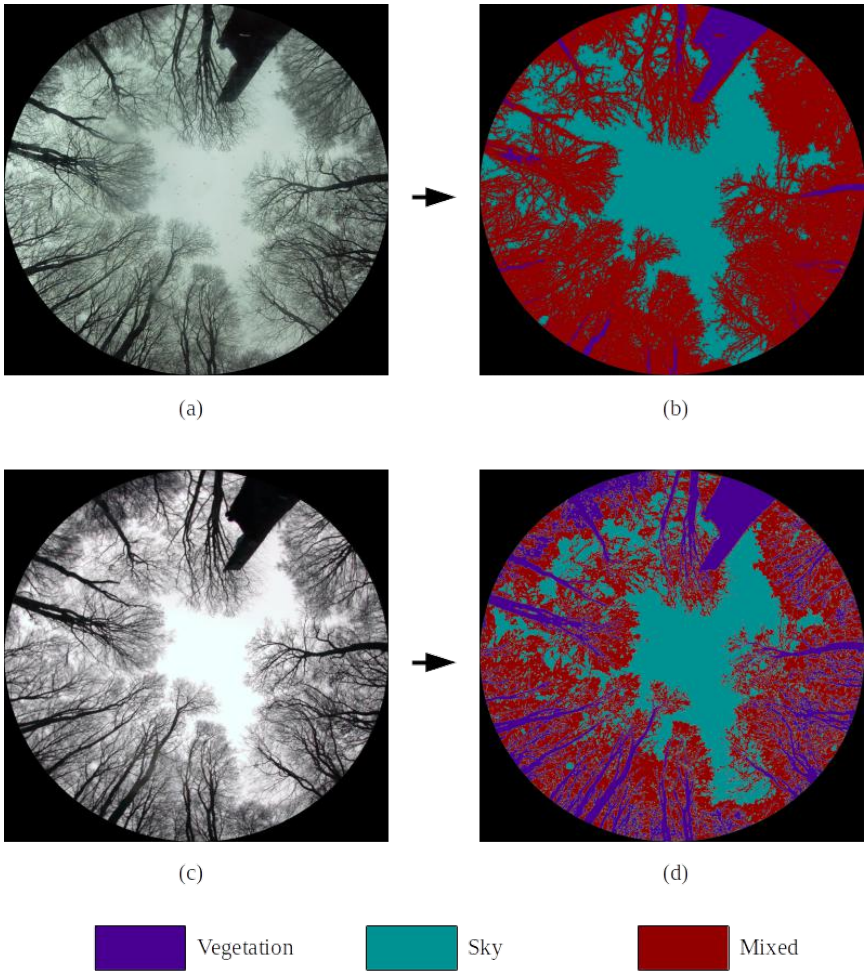


FIGURE 5.1: Images taken at plot Cb02 with PiLAI (a) and Nikon D7000 (c), with their respective classified images using RGB iterative selection (b) and CanEye supervised classification (d).

5.1 Accuracy

For the assessment of the accuracy of the PiLAI, the Speulderbos dataset was used. From the 41 locations measured in Speulderbos, 35 were used in the data analysis. This was due to missing or failed measurements from either PiLAI, TLS or DHP. Due to the limited zenith angle ($<= 60°$) used by CanEye, annuli with zenith angles $> 60°$ were not taken into account for Pearson correlation between gap fraction measurements of PiLAI and DHP and those of DHP and TLS. As a result, 420 pairs of

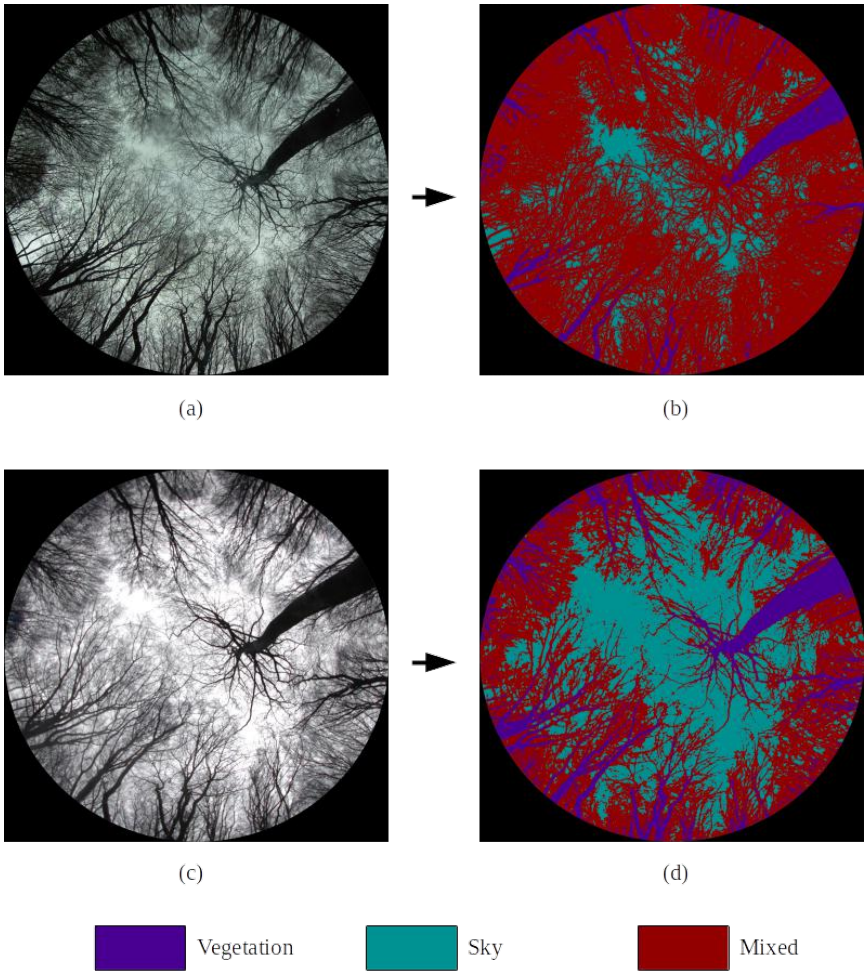


FIGURE 5.2: Images taken at plot Ea05 with PiLAI (a) and Nikon D7000 (c), with their respective classified images using RGB iterative selection (b) and CanEye supervised classification (d).

gap fraction measurements of 5° annuli were used for these correlation analyses. Correlation between PiLAI and TLS gap fraction was done with the full range of zenith angles (560 pairs) as well as with the $<= 60°$ angle subset.

5.1.1 Comparison with TLS

Results from Pearson correlation between LAI estimates from PiLAI and TLS (35 samples) are shown in Table 5.1. Although a significant correlation is found using $\alpha = 0.05$, correlation between LAI estimates of PiLAI and TLS is not high, ranging from 0.611 to 0.647.

TABLE 5.1: Results of Pearson correlation between LAI estimates made with PiLAI and TLS.

Method	**correlation coefficient**	**p**	**RMSE**
K-means (RGB)	0.647	< 0.01	2.58
K-means (blue)	0.647	< 0.01	2.59
Otsu's (RGB)	0.616	< 0.01	2.58
Otsu's (blue)	0.611	< 0.01	2.59

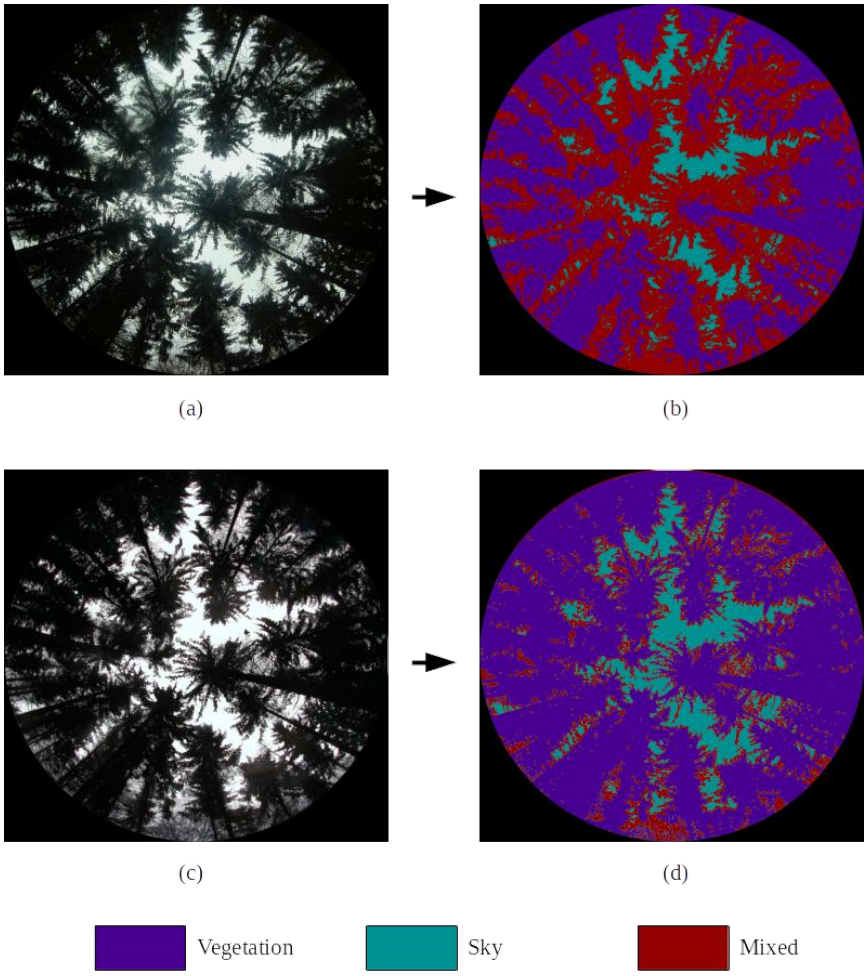


FIGURE 5.3: Images taken at plot Ff02 with PiLAI (a) and Nikon D7000 (c), with their respective classified images using RGB iterative selection (b) and CanEye supervised classification (d).

Figure 5.4 shows that there is a large discrepancy between LAI estimates from PiLAI and TLS, with all values far below the 1:1 line. PiLAI LAI do not reach 2 while LAI estimates from TLS exceed 5.

The results of Pearson correlation between gap fraction measurements of 5° annuli from PiLAI and TLS (560 pairs) are shown in Table 5.2. A high correlation is found for the 4 combinations of classification method and used bands, with K-means clustering achieving higher accuracies than Otsu's algorithm and RGB marginally outperforming the single blue band.

As shown in Figure 5.5, TLS measurements of gap fraction are consistently lower than those from PiLAI for zenith angles $< 60°$, while slightly higher at zenith angles $> 70°$. Overestimation of the gap fraction (using PiLAI), and thus underestimation of the LAI, at lower zenith angles is described in Calders, Origo, et al. (2018), Hancock et al. (2014) and Seidel, Fleck, and Leuschner (2012), and can in part be attributed to relatively low resolution images, classification issues and blooming effects. Overestimation of the gap fraction at higher zenith angles using TLS has in similar research been attributed to the limited range of the laser and the large distance of objects at these angles (Seidel, Fleck, and Leuschner, 2012). The VZ-400 used in this research has, however, a laser with a high range.

Using the $<= 60°$ zenith angle subset for Pearson correlation showed lower correlation coefficients compared to the full range of zenith angles (Table 5.2). In part this can

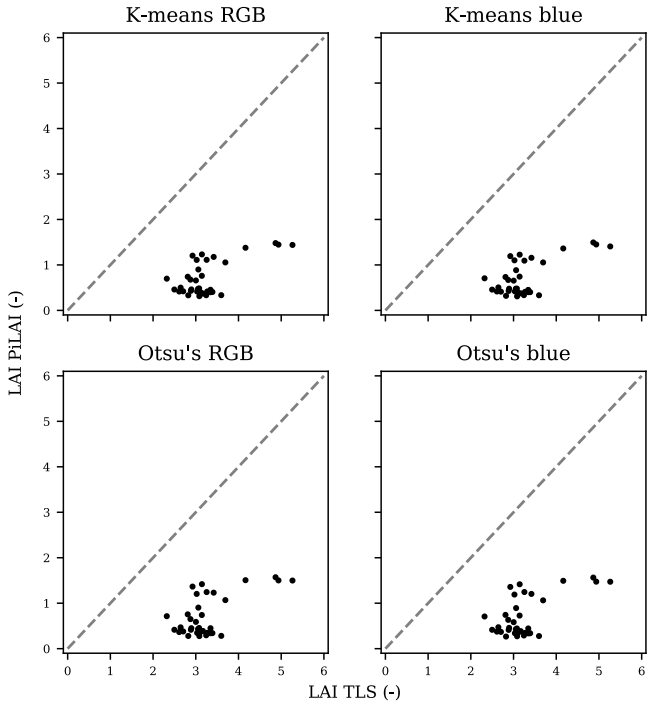


FIGURE 5.4: Scatter plot of LAI estimates from PiLAI and TLS.

be explained by the smaller sample size. The problem of underestimation of the gap fraction at smaller zenith angles is greater as is shown in Figure 5.6.

5.1.2 Comparison with DHP

As shown in Table 5.3, a close correlation was found between LAI estimates from PiLAI and DHP. However, the relatively high RMSE indicates a large absolute difference in LAI values K-means clustering performed slightly better than Otsu's algorithm. Figure 5.7 shows the high correlation found with the relatively straight lines that the points form. The high RMSE is indicated by the large deviation from the 1:1 line.

The results of Pearson correlation between gap fraction measurements from PiLAI and DHP is shown in Table 5.4. Similarly to the correlation of LAI values between PiLAI and DHP, a close correlation is found, as well as the K-means clustering performing slightly better than Otsu's algorithm.

Figure 5.8 shows the majority of points closely following the 1:1 line, with PiLAI gap fraction values tending to be sligthly higher than those from DHP. Outliers are primarily found in the outer zenith angles (50° to 60° annuli). This might be explained by the lower resolution of the PiLAI compared to that of the Nikon D7000, resulting in smaller leaves and branches being classified as sky pixels.

5.1.3 Cross-comparison DHP and TLS

Table 5.5 shows the results of the cross-comparison between DHP and TLS for both LAI and gap fraction. Pearson correlation for LAI is relatively low, which is lower than between PiLAI and TLS (Table 5.1). However, RMSE is lower as well, indicating a better match of the absolute values of the LAI. Figure 5.9 shows the closer proximity of the points to the 1:1 line.

TABLE 5.2: Results of Pearson correlation between gap fraction measurements taken with PiLAI and TLS.

Method	Angles used	Correlation coefficient	p	RMSE
K-means (RGB)	0°-80°	0.883	< 0.01	0.199
K-means (blue)	0°-80°	0.881	< 0.01	0.205
Otsu's (RGB)	0°-80°	0.868	< 0.01	0.235
Otsu's (blue)	0°-80°	0.868	< 0.01	0.242
K-means (RGB)	0°-60°	0.810	< 0.01	0.221
K-means (blue)	0°-60°	0.807	< 0.01	0.228
Otsu's (RGB)	0°-60°	0.783	< 0.01	0.263
Otsu's (blue)	0°-60°	0.783	< 0.01	0.271

TABLE 5.3: Results of Pearson correlation between LAI estimates made with PiLAI and DHP.

Method	correlation coefficient	p	RMSE
K-means (RGB)	0.847	< 0.01	2.30
K-means (blue)	0.851	< 0.01	2.30
Otsu's (RGB)	0.844	< 0.01	2.28
Otsu's (blue)	0.843	< 0.01	2.28

For the cross-comparison between DHP and TLS 420 pairs of gap fraction measurements of 5° annuli were used. Table 5.5 shows that conventional DHP has a Pearson correlation coefficient for TLS than the PiLAI using the same range of zenith angles (Table 5.2).

Figure 5.10 shows a pattern similar to that of Figure 5.6, which is in agreement with the theory that the discrepancy between TLS and PiLAI is due to the passive sensing method of DHP (which the PiLAI uses).

5.2 Reliability

Table 5.6 shows the LAI results from the Dassenbos autumn 2018 and spring 2019 timeseries. Standard deviation of the LAI is relatively high compared to the mean. Standard deviation seems to be slightly higher for days with precipitation. Figure 5.11 shows the timeseries for Dassenbos autumn 2018. Daily LAI estimates have a large range compared to the daily mean. The same holds, to a lesser degree, for the spring 2019 timeseries (Figure 5.12).

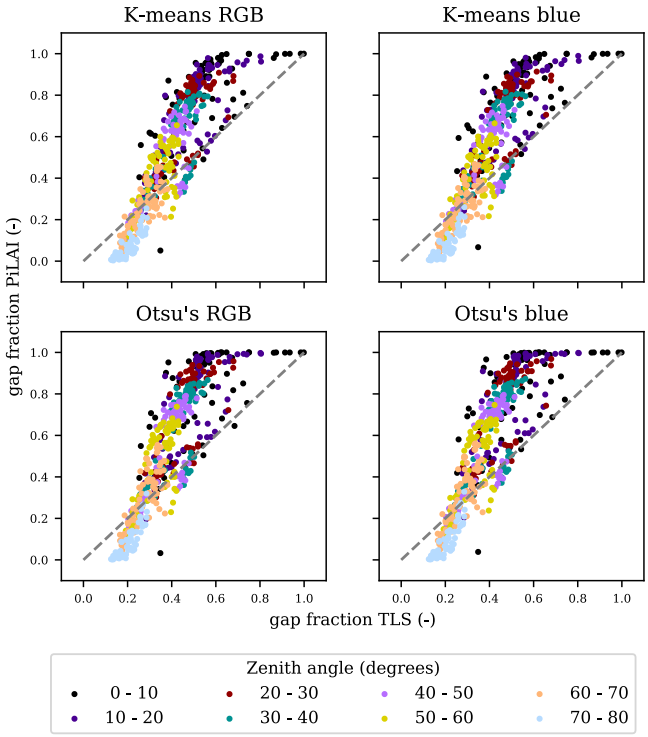


FIGURE 5.5: Scatter plot of gap fraction measurements from PiLAI and TLS using all zenith angles (0° to 80°).

TABLE 5.4: Results of Pearson correlation between gap fraction measurements taken with PiLAI and DHP.

Method	**correlation coefficient**	**p**	**RMSE**
K-means (RGB)	0.897	< 0.01	0.132
K-means (blue)	0.898	< 0.01	0.137
Otsu's (RGB)	0.875	< 0.01	0.169
Otsu's (blue)	0.876	< 0.01	0.176

TABLE 5.5: Results of Pearson correlation between gap fraction and LAI from DHP and TLS.

Variable	**correlation coefficient**	**p**	**RMSE**
LAI	0.585	< 0.01	1.30
Gap fraction	0.733	< 0.01	0.189

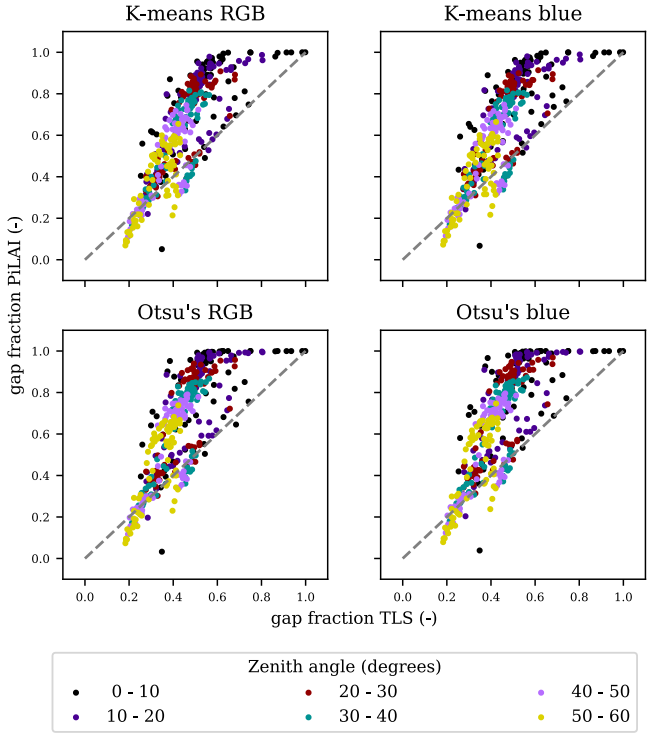


FIGURE 5.6: Scatter plot of gap fraction measurements from PiLAI and TLS using zenith angles 0° to 60°.

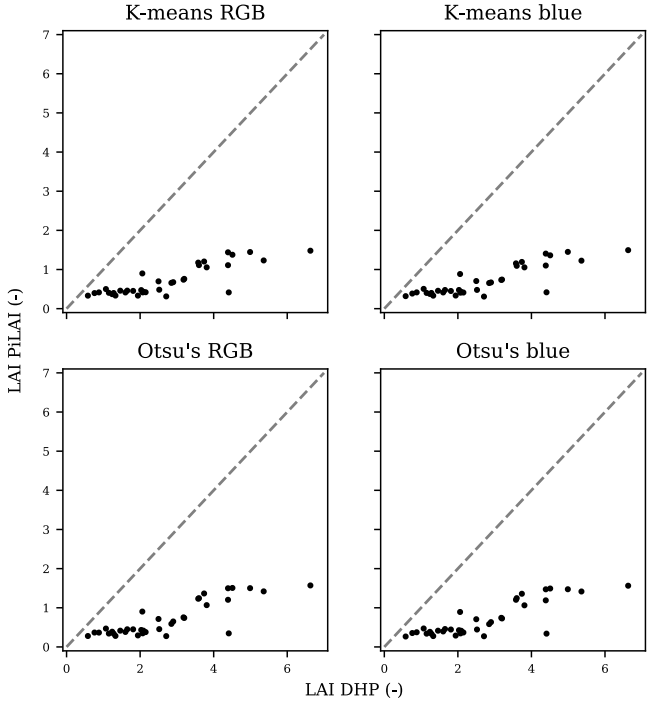


FIGURE 5.7: Scatter plot of LAI estimates from PiLAI and DHP.

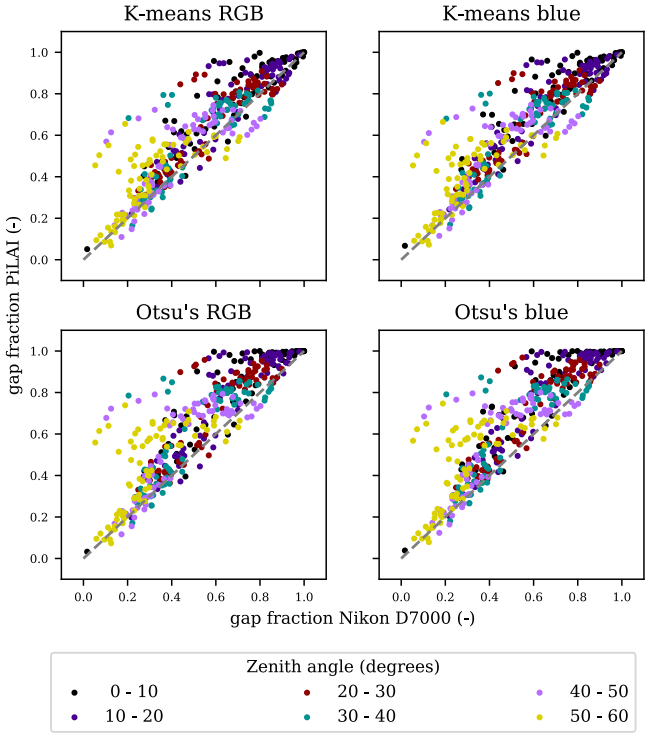


FIGURE 5.8: Scatter plot of gap fraction measurements from PiLAI and DHP.

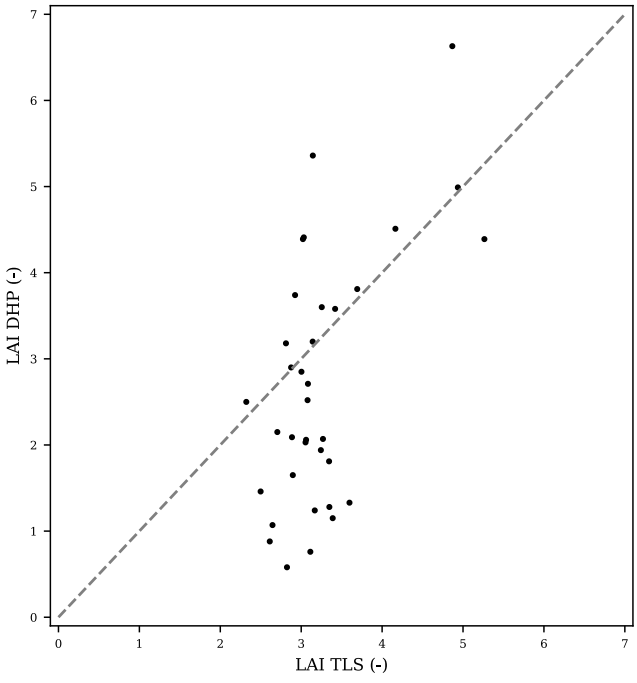


FIGURE 5.9: Scatter plot of LAI estimates from DHP and TLS.

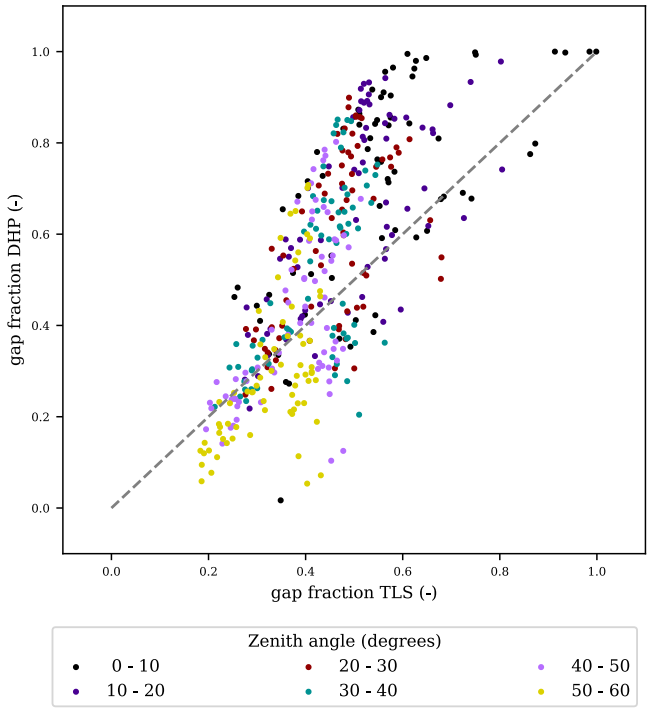


FIGURE 5.10: Scatter plot of gap fraction measurements from DHP and TLS.

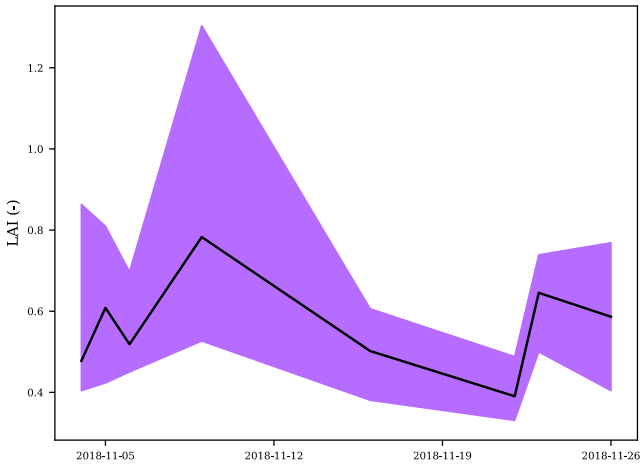


FIGURE 5.11: Dassenbos autumn 2018 mean LAI and range time-series.

TABLE 5.6: Dassenbos standard deviation of LAI estimates from Pi-LAI.

Date	Mean	St.Dev.	Precipitation (mm / day)	Cloud cover (oktas)	Humidity (%)	Measurements #
2018-11-04	0.478	0.124	0	4	90	12
2018-11-05	0.608	0.144	0	8	95	10
2018-11-06	0.519	0.072	0	8	79	12
2018-11-09	0.783	0.191	0	7	88	19
2018-11-16	0.502	0.073	0	2	92	10
2018-11-22	0.390	0.058	0	8	86	9
2018-11-23	0.645	0.096	0	6	89	8
2018-11-26	0.586	0.138	0	8	88	9
2019-04-13	0.204	0.103	0	3	75	20
2019-04-14	0.237	0.071	0	4	68	3
2019-04-15	0.099	0.042	0	3	59	14
2019-04-18	0.320	0.064	0	5	42	13
2019-04-20	0.366	0.044	0	0	43	7
2019-04-22	0.610	0.109	0	4	42	8
2019-04-23	0.636	N.A.	0	6	50	1
2019-04-24	0.757	0.278	9	7	64	18
2019-04-25	0.499	0.099	5	7	79	8
2019-04-26	0.893	0.263	0	7	72	10
2019-04-27	0.845	0.182	5	8	76	15
2019-04-29	0.760	0.284	0	6	75	20
2019-05-02	0.811	0.141	4	7	86	16
2019-05-03	0.712	0.170	2	8	73	18
2019-05-09	0.895	0.217	2	7	78	24
2019-05-10	0.768	0.131	1	8	77	11
2019-05-14	0.725	0.139	0	4	62	19
2019-05-15	0.629	0.131	0	4	51	14

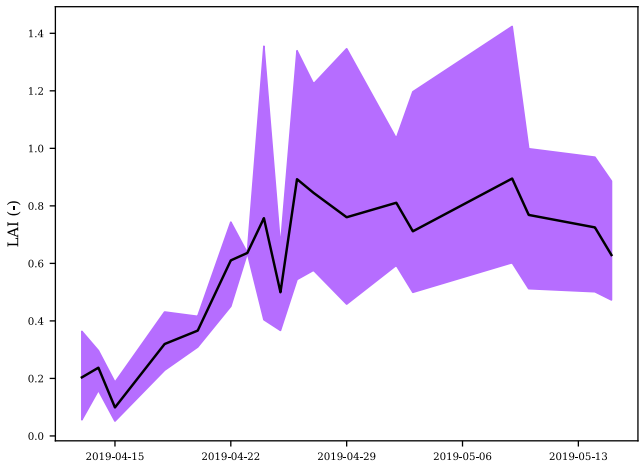


FIGURE 5.12: Dassenbos spring 2019 mean LAI and range timeseries.

Chapter 6

Discussion

6.1 Development of the sensor system

The base sensor system is easy to build and can be done so at low cost (< 100 €. By keeping the cost low, an extensive set of sensors can be built even with a limited research budget. Costs can be reduced even more by using a Raspberry Pi Zero or choosing off-brand alternatives of the used materials. The system can be built with minimal tools, making it a viable option for people with limited access to power tools. Although not all hardware used in the sensor was open source, all hardware had open source software interfaces, making software development easier. True open source alternatives for the camera module are at the time of writing not readily available.

Software development of the sensor was initially done with Python for the Robotic Operating System[1] (ROS). ROS is a low-level framework that enables easy exchange of data between different components of a system. ROS was chosen to make the sensor system more modular and extensible, however, due to the relatively high learning curve of ROS and the limited availability of compatible Python libraries, ROS was considered unsuitable for the PiLAI and instead a pure Python solution was used. The Python pycamera package provides sufficient options for precise control of the camera. Due to the low learning curve and dominance of Python in Raspberry Pi applications, Python is considered a suitable scripting language for the further development of the PiLAI.

While X. Li et al. (2015) and Lecerf et al. (2010) have shown the potential of an automatic LAI sensor, this research shows that an automatic sensor system can be build with a bare minimum of materials. By keeping the system modular, extra functionality can be added if needed.

6.2 Results

6.2.1 Image quality

Comparing images acquired by the PiLAI with those of the Nikon D7000 (Speulderbos), no significant difference was found (Figures 5.1, 5.2 and 5.3). Although the PiLAI images lack the detail of those from the Nikon, primarily due to the lower resolution of the PiLAI, overall image quality seems to be similar. The lower resolution does lead to a higher number of mixed pixels. Images acquired in Dassenbos show that blooming due to open sky and a high position of the sun do result in overexposed images. After in-depth inspection of the images, it was established that while the camera did capture the full range of pixel values (DN of 0-255) for the lower part of the image, the higher upper part had a maximum pixel value 241. This was most

[1] https://www.ros.org/

likely due to a defect in the camera sensor module. This problem caused the function that tested for overexposure, `check_histogram()` to be virtually useless, causing overexposed images not to be classified as such.

Figures 5.1, 5.2 and 5.3 show a dominance of mixed pixels for the PiLAI image classifications, whereas the Nikon D7000 images classified using CanEye's supervised classification show most solid parts (tree trunks and larger branches) correctly as vegetation. In part this can be attributed to the lower pixel-density of the PiLAI images, leading to more mixed pixels. The automatic classification algorithm of the PiLAI seems to have a larger range for which to label a pixel as mixed pixel than the CanEye supervised classification algorithm. As can be seen in Figures 5.1, 5.2 and 5.3, this can provide nuances that CanEye's supervised classification misses in some cases, including small branches without leaves.

One of the major differences between the range of PiLAI classification methods and CanEye's supervised classifiction is the time required for the processing. Whereas supervised classification took at least 6 minutes of manual work per subplot (set of 3 images), PiLAI's most time-consuming classification method (RGB iterative selection) took no more than 10 seconds per image (including reading and writing). Otsu's algorithm (4 images per second) and the use of only the blue band reduce processing times even more. This reduction in labour is cost-effective and can provide opportunities for large-scale tree canopy monitoring.

The use of only the blue channel instead of full RGB did not result in closer correlation of the LAI estimates, contrary to what is suggested in most literature (Jonckheere, Nackaerts, et al., 2005; Woodgate et al., 2015; Yan et al., 2019). Instead of using only the blue channel, a camera sensitive in the near-infrared (NIR) part of the electromagnetic spectrum (750 nm to 1500 nm) could be used. Since green vegetation has a high reflectance in this part of the spectrum, it is suitable for segmentation. The use of NIR with low-cost cameras has been demonstrated by Petach et al. (2014) and Kirby, L. Chapman, and V. Chapman (2018), and shows potential for further development, especially when separation of leaves from other above-ground vegetation is required.

Overall, K-means clustering performed better as classification algorithm than Otsu's algorithm, although the differences were small. Since processing time differed significantly between the two methods, Otsu's algorithm might be preferred for low-powered hardware and larger datasets.

6.2.2 Accuracy and reliability

Overall, LAI estimates from the PiLAI were low, never exceeding 2. During leaf-on conditions (start of autumn and end of spring), such LAI values are unlikely. Pearson correlation between the LAI estimates of the PiLAI and TLS was not very high, with 0.647 being the highest. Woodgate et al. (2015) showed higher correlation with conventional high resolution DHP using CanEye supervised classification (0.79), however, high resolution DHP classified using an automatic global binary threshold performed similarly (0.69).

Using an automated transmittance sensor, the PASTIS-57, Brede, Gastellu-Etchegorry, et al. (2018) reported a Pearson correlation of 0.98. Compared to this, PiLAI LAI estimates are quite inaccurate. Correlation between LAI estimates of PiLAI and DHP was better, showing the potential of the PiLAI as an automated alternative to conventional DHP. Correlation of gap fraction measurements of the PiLAI with those of DHP and TLS showed promise, with Pearson correlation coefficients exceeding 0.8 and relatively low RMSE. The discrepancy between the Pearson correlation results of gap fraction and LAI indicate a possible error in the calculation of the LAI.

A cross-comparison between DHP and TLS (Table 5.5) showed that the correlation between DHP and TLS in this study was lower than that of the PiLAI and TLS, indicating that the PiLAI performed better than the conventional DHP. The lower RMSE of DHP compared to the PiLAI does show that the absolute LAI estimates of DHP are closer to those of TLS. Since no direct method for determining LAI was used in this research, the correlation of PiLAI LAI estimates to the true LAI could not be assessed.

Reliability of the LAI estimates from the PiLAI seems to be an issue, with daily standard deviations exceeding 25% of the daily mean. Certain weather conditions, especially precipitation and humidity (covering the acrylic dome), can be problematic. Changing weather conditions can lead to a wide range of LAI values over a single day. Relatively large variations in LAI estimates over the day can in part be explained by the low canopy cover (actual LAI) during the measurement periods, leading to a large influence of noise on the measurements. Extended timeseries and measurements taken under dense-canopy conditions are likely to yield better results.

6.3 Limitations and possibilities

6.3.1 Limitations

Over the course of this research a number of limitations of the PiLAI have been identified. One of the main limitations of the PiLAI is the limited time it can operate on a single charge. With the Raspberry Pi 2B draining 220 mA at 5 V when idle[2] (not including the camera module), the 12 V, 7 Ah battery should theoretically last 76 hours. After disabling software components with a large power consumption, such as the graphical user interface, the battery lasted for no more than 36 hours in the field, which is insufficient to be a viable option for automatic LAI monitoring. Using a larger battery might not be a viable solution as it would increase the size and weight of the system, making it less portable. Due to the voltage drop required (12 V to 5 V), a part of the energy is dissipated as heat. As such, the use of high quality 5 V power banks might be a good solution to remove this loss. With the large consumer demand for power banks, developments in capacity are fast and costs are relatively low. Other solutions are the use of low-powered alternatives for the Raspberry Pi 2B. The Raspberry Pi Zero has a significantly lower power consumption, while it is still able to do image processing at > 5 MP resolution. Arduino boards such as the Arduino Nano or Mega consume even less power, however, they lack the processing power for high-resolution image processing. The use of solar panels as secondary power source could lead to extended operational time, although their use is limited when the canopy is dense due to low incoming solar radiation. A second prototype of the PiLAI is in development to extend the maximum operating time by shutting off the Raspberry Pi and cutting the power to it externally using an Arduino Nano (Figure A.1). Due to setbacks and time constraints, the prototype could not be finished.

A second limitation is the influence of certain weather conditions on the measurements. During the field tests condensation on the inside of the lens cover was a frequent issue, resulting in large parts of the image being incorrectly classified. Near the end of the field tests this issue was resolved by taping a sachet of silica gel next to the camera module. This low-cost material absorbs moisture and is often used by shipping companies to prevent moisture from damaging shipped goods. Precipitation caused issues as well by covering the lens cover in water drops, although this often resolved itself within a few hours after it stopped raining.

[2] http://www.pidramble.com

A third limitation that was identified was the lack of user-friendliness of the PiLAI. Due to the lack of external buttons and screen, the PiLAI had to be set up with its box open and the Raspberry Pi connected to a laptop. The second prototype of the PiLAI tries to solve this issue by adding buttons and a small LCD screen to the sensor.

6.3.2 Possibilities

Although the performance of the PiLAI on estimating LAI was a bit disappointing, gap fraction measurements showed promising results, especially for a first prototype. The PiLAI has shown that automatic acquisition of hemispherical images is a viable option, as well as unsupervised classification. Due to the open source and extensible nature of the PiLAI, continuous development of the sensor is possible. Contrary to a closed source system, users can add other components to suit their specific needs and develop the system further based on new research. This removes the need to buy new (and expensive) proprietary sensors when the previous version becomes outdated.

To give greater autonomy to the sensor, a module for wireless data (*e.g.* 3G) could be added to the sensor. This would enable the sensor to notify the user when the battery is low or even do the processing *in situ* and sending gap fraction, clumping factor and LAI values to the user. An added benefit of a wireless data module such as the low-cost SIM800L is the availability of location data. Furthermore, developments in Raspbian and the Python picamera library are likely to give even greater control over the image acquisition process.

Chapter 7

Conclusions

The PiLAI is a low-cost, open source, automatic sensing system for continuous canopy monitoring. The system uses widely available components and can be build with limited experience in electronics. The system can provide canopy images at desired intervals, automatically setting exposure parameters and testing for over- or underexposure. It offers the choice to do *in situ* processing or to collect the images from the PiLAI and do the processing elsewhere. Intermediate data are stored, guaranteeing reproducibility of the results. The PiLAI has the following advantages:

- Required hardware is inexpensive and widely available.
- Uses virtually exclusively open source software, reducing the risk of hidden errors/anomalies and providing the opportunity for further development of the system.
- System is easy to build and maintain.
- System is weatherproof in most situations.
- The automation of the workflow reduces manual labour to a minimum, greatly reducing operating costs.

The prototype still has some problems, requiring further development in order to reach its full potential as alternative for manual DHP:

- Battery life is short, voltage conversion results in waste of power due to heat dissipation.
- Certain weather conditions (rain and high humidity) can negatively impact measurements.
- The system still lacks user-friendliness.

Validation of the system showed promising results, especially considering the hard to measure conditions (virtually no leaves in the canopy). Further validation of the PiLAI under leaf-on conditions would be useful for more in-depth quantification of the accuracy and reliability of the system. Overall, the PiLAI shows promise as an automated alternative for conventional DHP.

References

Alonzo, M. et al. (2015). "Mapping urban forest leaf area index with airborne lidar using penetration metrics and allometry". en. In: *Remote Sensing of Environment* 162, pp. 141–153.

Breda, N. J. J. (2003). "Ground-based measurements of leaf area index: a review of methods, instruments and current controversies". en. In: *Journal of Experimental Botany* 54.392, pp. 2403–2417.

Brede, B., Bartholomeus, H., et al. (2016). "The Speulderbos fiducial reference site for continuous monitoring of forest biophysical variables". eng. In: vol. SP-740.

Brede, B., Gastellu-Etchegorry, J.-P., et al. (2018). "Monitoring Forest Phenology and Leaf Area Index with the Autonomous, Low-Cost Transmittance Sensor PASTiS-57". en. In: *Remote Sensing* 10.7, p. 1032.

Calders, K., Armston, J., et al. (2014). "Implications of sensor configuration and topography on vertical plant profiles derived from terrestrial LiDAR". en. In: *Agricultural and Forest Meteorology* 194, pp. 104–117.

Calders, K., Origo, N., et al. (2018). "Variability and bias in active and passive ground-based measurements of effective plant, wood and leaf area index". en. In: *Agricultural and Forest Meteorology* 252, pp. 231–240.

Calders, K., Schenkels, T., et al. (2015). "Monitoring spring phenology with high temporal resolution terrestrial LiDAR measurements". en. In: *Agricultural and Forest Meteorology* 203, pp. 158–168.

Chen, J. M., Pavlic, G., et al. (2002). "Derivation and validation of Canada-wide coarse-resolution leaf area index maps using high-resolution satellite imagery and ground measurements". In: *Remote Sensing of Environment* 80.1, pp. 165–184.

Chen, J. M. and Cihlar, J. (1995). "Plant canopy gap-size analysis theory for improving optical measurements of leaf-area index". en. In: *Applied Optics* 34.27, p. 6211.

Chen, J. M., Rich, P. M., et al. (1997). "Leaf area index of boreal forests: Theory, techniques, and measurements". en. In: *Journal of Geophysical Research: Atmospheres* 102.D24, pp. 29429–29443.

Colombo, R. et al. (2003). "Retrieval of leaf area index in different vegetation types using high resolution satellite data". In: *Remote Sensing of Environment* 86.1, pp. 120–131.

Delta-T Devices Ltd (2017). *WinDIAS 3 Image Analysis System*. URL: https://www.delta-t.co.uk/wp-content/uploads/2017/03/WinDIAS3-version-2.pdf (visited on 02/21/2019).

Fang, H., Li, W., et al. (2014). "Seasonal variation of leaf area index (LAI) over paddy rice fields in NE China: Intercomparison of destructive sampling, LAI-2200, digital hemispherical photography (DHP), and AccuPAR methods". en. In: *Agricultural and Forest Meteorology* 198-199, pp. 126–141.

Fang, H., Ye, Y., et al. (2018). "Continuous estimation of canopy leaf area index (LAI) and clumping index over broadleaf crop fields: An investigation of the PASTIS-57 instrument and smartphone applications". en. In: *Agricultural and Forest Meteorology* 253-254, pp. 48–61.

Ferdoush, S. and Li, X. (2014). "Wireless Sensor Network System Design Using Raspberry Pi and Arduino for Environmental Monitoring Applications". en. In: *Procedia Computer Science* 34, pp. 103–110.

Fernandes, R. et al. (2001). "Examination of error propagation in relationships between leaf area index and spectral vegetation indices from Landsat TM and ETM".

In: *Proceedings of the 23rd Canadian Remote Sensing Symposium, Quebec City, Quebec*, pp. 41–51.

Gardingen, P. van et al. (1999). "Leaf area index estimates obtained for clumped canopies using hemispherical photography". en. In: *Agricultural and Forest Meteorology* 94.3-4, pp. 243–257.

Hancock, S. et al. (2014). "Characterising forest gap fraction with terrestrial lidar and photography: An examination of relative limitations". en. In: *Agricultural and Forest Meteorology* 189-190, pp. 105–114.

Jonckheere, I., Fleck, S., et al. (2004). "Review of methods for in situ leaf area index determination". en. In: *Agricultural and Forest Meteorology* 121.1-2, pp. 19–35.

Jonckheere, I., Nackaerts, K., et al. (2005). "Assessment of automatic gap fraction estimation of forests from digital hemispherical photography". en. In: *Agricultural and Forest Meteorology* 132.1-2, pp. 96–114.

Kirby, J., Chapman, L., and Chapman, V. (2018). "Assessing the Raspberry Pi as a low-cost alternative for acquisition of near infrared hemispherical digital imagery". en. In: *Agricultural and Forest Meteorology* 259, pp. 232–239.

Lang, A. (1986). "Estimation of leaf area index from transmission of direct sunlight in discontinuous canopies". en. In: *Agricultural and Forest Meteorology* 37.3, pp. 229–243.

Leblanc, S. G. et al. (2005). "Methodology comparison for canopy structure parameters extraction from digital hemispherical photography in boreal forests". en. In: *Agricultural and Forest Meteorology* 129.3-4, pp. 187–207.

Lecerf, R. et al. (2010). "PASTIS 57: Autonomous light sensors for PAI continuous monitoring. Principles, calibration and application to vegetation phenology". In: *AGU Fall Meeting Abstracts* 52, B52C–08.

Li, X. et al. (2015). "The Design and Implementation of the Leaf Area Index Sensor". en. In: *Sensors* 15.3, pp. 6250–6269.

LICOR (2016). *Leaf Area Meters Brochure*. URL: https://www.licor.com/documents/rnzuch09nob4j3ov2x7m (visited on 02/21/2019).

OpenCV (2018). *OpenCV: Flags for video I/O*. URL: https://docs.opencv.org/4.0.1/d4/d15/group__videoio__flags__base.html (visited on 02/18/2019).

Pearse, G. D., Watt, M. S., and Morgenroth, J. (2016). "Comparison of optical LAI measurements under diffuse and clear skies after correcting for scattered radiation". In: *Agricultural and Forest Meteorology* 221, pp. 61–70.

Petach, A. R. et al. (2014). "Monitoring vegetation phenology using an infrared-enabled security camera". en. In: *Agricultural and Forest Meteorology* 195-196, pp. 143–151.

Prescott, C. E. (2002). "The influence of the forest canopy on nutrient cycling". en. In: *Tree Physiology* 22.15-16, pp. 1193–1200.

Ridler, T. and Calvard, S. (1978). "Picture Thresholding Using an Iterative Selection Method". In: *IEEE Transactions on Systems, Man, and Cybernetics* 8.8, pp. 630–632.

RIEGL (2017). *Data Sheet, RIEGL VZ-400*. URL: http://www.riegl.com/uploads/tx_pxpriegldownloads/10_DataSheet_VZ-400_2017-06-14.pdf (visited on 03/04/2019).

Seidel, D., Fleck, S., and Leuschner, C. (2012). "Analyzing forest canopies with ground-based laser scanning: A comparison with hemispherical photography". en. In: *Agricultural and Forest Meteorology* 154-155, pp. 1–8.

Tillack, A. et al. (2014). "Estimation of the seasonal leaf area index in an alluvial forest using high-resolution satellite-based vegetation indices". In: *Remote Sensing of Environment* 141, pp. 52–63.

Watson, D. J. (1947). "Comparative Physiological Studies on the Growth of Field Crops: I. Variation in Net Assimilation Rate and Leaf Area between Species and Varieties, and within and between Years". In: *Annals of Botany* 11.41, pp. 41–76.

Wilson, J. W. (1960). "Inclined Point Quadrats". en. In: *New Phytologist* 59.1, pp. 1–7.

Woodgate, W. et al. (2015). "Understanding the variability in ground-based methods for retrieving canopy openness, gap fraction, and leaf area index in diverse forest systems". In: *Agricultural and Forest Meteorology* 205, pp. 83–95.

Yan, G. et al. (2019). "Review of indirect optical measurements of leaf area index: Recent advances, challenges, and perspectives". en. In: *Agricultural and Forest Meteorology* 265, pp. 390–411.

Zhang, G. et al. (2014). "Estimation of forest aboveground biomass in California using canopy height and leaf area index estimated from satellite data". en. In: *Remote Sensing of Environment* 151, pp. 44–56.

Appendix A

Schematics

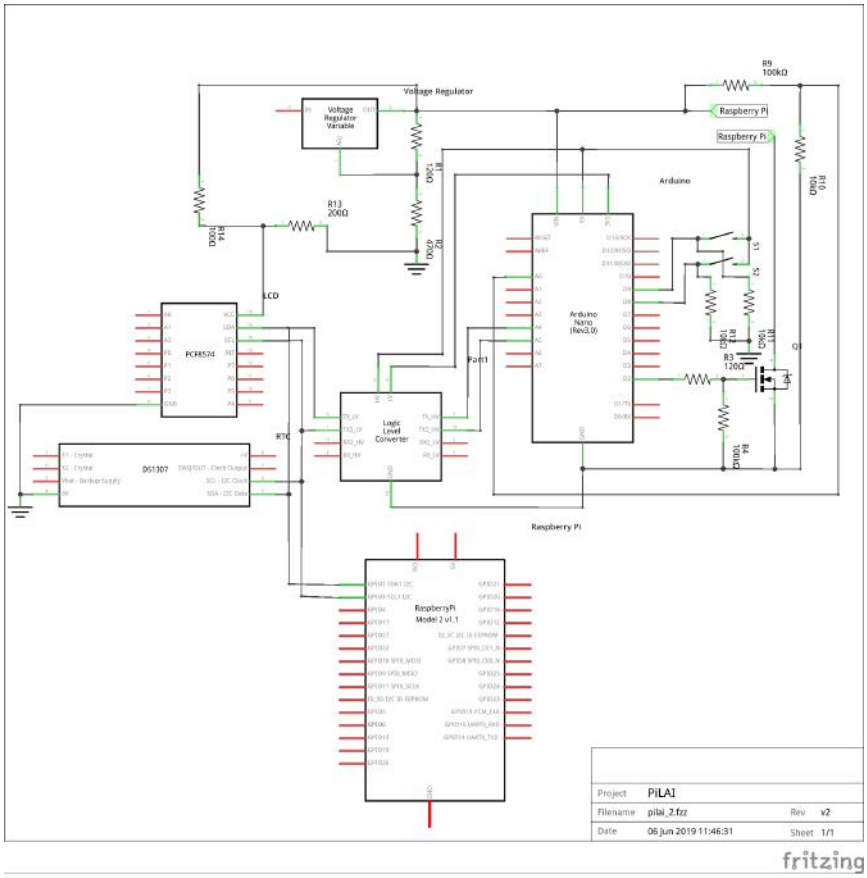


FIGURE A.1: Schematic of the PiLAI v2 prototype (unfinished).