# Classification of High Vegetation in an Urban environment: A Performance Comparison of Machine Learning Methods in a LiDAR dataset


Master thesis

MSc. Geoinformatics

Aalborg University Copenhagen

Nijole Makovskaja

June 2018

AALBORG UNIVERSITY
STUDENT REPORT

Aalborg University – Copenhagen

A.C. Meyers Vænge 15

2450 Copenhagen SW

www.plan.aau.dk

**Title**

Classification of High Vegetation in an Urban environment: A Performance Comparison of Machine Learning Methods in a LiDAR dataset

**Semester**

3rd – 4th semesters

**Project period**

September 2017 – June 2018

**Supervisor**

Jamal Jokar Arsanjani

**Page Count**

79

**Preface**

This one-year thesis was written within September 2017 – June 2018 as part of Geoinformatics programme at Aalborg University in Copenhagen.

This thesis analyzes different machine learning methods to classify high vegetation in urban areas given a case study in Copenhagen. The data was collected from various sources of remote sensing as well as national Danish geoportal of https://kortforsyningen.dk/.

The results and discussions in this study can be beneficial to future high vegetation classification or similar analysis

*- Nijole Makovskaja, June 2018*

*Intentionally left blank*

## Abstract

**Introduction:** Mapping high vegetation in the urban areas in different scales (global, regional and local) is important for researchers as well as for authorities. This improves the quality and trustful monitoring of the changing environment. Combining both concepts and using machine learning approach can be helpful in reaching high accuracy classification for high vegetation in urban areas.

**Data and methods:** In this thesis, supervised Machine Learning algorithms were used to compare the effectiveness of three modeling techniques – Support Vector Machines (SVM), Random Forest (RF) and Kernel - for classifying high vegetation in part of Copenhagen urban area. These three classification models were considered/selected in this study since their popularity and high classification performance were proven in the many previous studies. In addition to LiDAR features (intensity, return number, number of returns), orthophoto features such as colors (red, green, blue) and an infrared band were incorporated into the study.

**Results and discussions:** The results indicate that the highest classification accuracy is obtained with Kernel model (85.25%), nevertheless Random Forest was found to be less sensitive when training dataset size was decreased (difference between Sets is 2.4%). In the further direction, classification model can be trained better and might work with larger scale areas, for example all Copenhagen. This model was trained to identify high vegetation in an urban area (part of Copenhagen), but it can be used to classify high vegetation on other cities as well.

**Conclusion:** The highest accuracy was achieved with Kernel classification (85.25%), but with smaller training data size Random Forest can be considered as a very good option. The difference of accuracy changes of Random Forest was the lowest, compared with other models – 5% (the approach I) and 2.6% (the approach II).



**Keywords**: Machine Learning, Classification, Support Vector Machine, Random Forest, Kernel, LiDAR, Orthophoto

# Contents

1. Introduction....5
1.1 Remote Sensing....5
1.1.1 Light Detection and Ranging....6
1.1.2 Airborne Remote Sensing....15
1.2 Machine Learning....19
1.2.1 What is Machine Learning?....19
1.2.2 Learning techniques....21
1.2.3 Tools and toolkits for implementation....24
1.2.4 Classification....28
1.2.5 Validation Techniques....35
1.3 Problem Statement....39
1.4 Research Questions....39
1.5 Report Structure....40
2 Methods and Materials....40
2.1 Data Set....40
2.2 Flowchart of the implementation....41
2.2.1 Approach I....48
2.2.2 Approach II....49
3 Results and discussions....51
4 Conclusion....69
Literature....71

# Table of Figures

**Figure 1.** Feature Spaces (Inspired by Wai Yeung Yan et al., 2015) ..... 8
**Figure 2**. Laser pulse (Wai Yeung Yan et al., 2015)..... 11
**Figure 3.** The three angles (roll, pitch and yaw) of an aircraft that influence the geometry of the acquired images (Khorram S. et al., 2012)..... 15
**Figure 4.** Image bands and DN values in pixel (Khorram S. et al., 2012)..... 16
**Figure 5.** Image characteristics (Inspired by Khorram S. et al., 2012)..... 16
**Figure 6.** Hypothetical perfect fitting spectral reflectance curve of a healthy vegetation (Khorram S. et al., 2012) ..... 18
**Figure 7.** Machine Learning position and relation between other brunches. (Mitchell-Guthrie, P., 2014) ..... 20
**Figure 8.** Machine learning techniques and data requirements (Mitchell M.T., 2006)..... 21
**Figure 9.** Demonstrates how in different scenarios labeling can be done (Mitchell M.T., 2006) ..... 22
**Figure 10.** Hyper plane and support vectors (Kavzoglu, 2009) ..... 29
**Figure 11.** Optimal hyperplane (Kavzoglu, 2009) ..... 30
**Figure 12.** Nonlinear hyper plane (Kavzoglu, 2009) ..... 31
**Figure 13.** Converting to bigger dimension (Kavzoglu, 2009) ..... 31
**Figure 14.** Scheme of decision tree (Stanford, 2017) ..... 33
**Figure 15.** Bad and good splitting (Aggiwal R., 2017) ..... 33
**Figure 16.** Classification example (Aggiwal R., 2017) ..... 34
**Figure 17.** Linear regression and decision tree classification (Inspired by Aggiwal R., 2017) ..... 34
**Figure 18.** Total number of examples split into training and test sets (Bronshtein A., 2017) ..... 36
**Figure 19.** Data set splitting for validation (Bronshtein A., 2017) ..... 37
**Figure 20.** K- folds Cross Validation (Bronshtein A., 2017) ..... 38
**Figure 21.** Leave One Out Cross Validation (LOOCV) (Bravo H.C., 2018) ..... 39
**Figure 22.** Study area in Copenhagen ..... 41
**Figure 23.** The main flowchart of implementation..... 42
**Figure 24.** Converting *.laz to *.txt format ..... 43
**Figure 25.** LiDAR misclassified points..... 44
**Figure 26.** LiDAR point cloud on orthophoto ..... 46
**Figure 27.** The flowchart of the first approach ..... 48
**Figure 28.** Workflow of the approach based on combination of LiDAR and Orthophoto data ..... 50
**Figure 29.** Model training analysis..... 58
**Figure 30.** Testing data accuracy ..... 60

**Figure 31.** Accuracy only with LiDAR data ..... 61
**Figure 32.** Accuracy with LiDAR and orthophoto data ..... 62
**Figure 33.** Accuracy with LiDAR and orthophoto data ..... 63
**Figure 34.** Data Set analysis ..... 64
**Figure 35.** Support Vector Machine classification comparison for different Sets in approach I and approach II ..... 65
**Figure 36.** Random forest classification comparison for different Sets in approach I and approach II ..... 66
**Figure 37.** Kernel classification comparison for different Sets in approach I and approach II ..... 67

## Table of tables

**Table 1.** Training sets and their sizes ..... 47
**Table 2.** Support Vector Machine k-fold train and validation accuracy results for approach I. ..... 52
**Table 3.** Random Forest k-fold train and validation accuracy results for approach I. ..... 53
**Table 4.** KERNEL k-fold train and validation accuracy results for approach I. ..... 54
**Table 5.** Support Vector Machine k-fold train and validation accuracy results for approach II. ..... 55
**Table 6.** Random Forest k-fold train and validation accuracy results for approach II. ..... 56
**Table 7.** Kernel k-fold train and validation accuracy results for approach II. ..... 57
**Table 8.** Test dataset for approach I only with LiDAR data ..... 59
**Table 9.** Test dataset for approach II with LiDAR and Orthophoto data ..... 59

# 1. Introduction

More and more people have started living in a city rather than the countryside. Copenhagen is known as a green and ecological city where urban environment and vegetation goes together (Bothe K., et al., 2018). Moreover, vegetation is very important for ecology and the ecosystem. It takes many years for trees to grow, so high vegetation is very sensitive and time-consuming subject. Today we have remote sensing gadgets to track high vegetation. Most popular of these gadgets are LiDAR scanning and orthophoto from airplane or satellite. Machine learning with different algorithms can classify high vegetation from LiDAR and orthophoto features. This method optimization leads to faster and more precise high vegetation tracking and planning task. This chapter will review previous literature on the subject in order to get general knowledge about the main questions, problems and some solutions, remote sensing features (LiDAR and orthophoto), as well machine learning part with different classification algorithm analysis.

## 1.1 Remote Sensing

The definition of remote sensing can be described as measurement and information achievement without physical contact with the objects by registration gadget.

This definition fits for many things. For example, in the medical description refers to X-rays or magnetic resonance. In an environmental context it stands for obtaining information about electromagnetic energy (EM), which comes from surfaces and objects on the earth. Differences in emitted electromagnetic energy provides possibility of object identification and differentiation between them (Khorram S. et al., 2012).

### 1.1.1 Light Detection and Ranging

“LIDAR stands for Light Detection and Ranging”. It was presented for commercial use in the mid-1990s. Together with direct georeferencing technique the laser scanning equipment installed in planes gathers a cloud of laser range measurements used in calculating the 3D coordinates (XYZ) of the observation area.

The main difference from 2D planimetric remote sensing data is that the explicit LiDAR data point cloud defines the 3D topographic profile of the earth’s surface. Another important advantage of airborne LiDAR is that it is not affected by relief displacement, lightning conditions or penetration of tree canopy. For this reason, LiDAR technique is widely used in topographic mapping, generating digital terrain model (DTM), creating digital 3D city model, natural hazard assessment, etc. (Yan, W.Y, 2012).

LiDAR works by the with echo-return principle, by recording the time differences between the pulse of energy that is transmitted and comes back after touching the object. The wave is emitted from the anchored (to fixed-wing or helicopter type aircraft) sensor.
The Z coordinate is very important here and it reflects pulse time to reach the object and come back. According to Davenport I.J. (2004) vertical accuracy is ±15cm (T.R. Tooke et al., 2014).

The scanning area depends on:

- Pulse speed;
- Aircraft altitude;
- Geographical position (T.R. Tooke et al., 2014).

It is a remote sensing technology to get information from the environment in 3D with the optical sensors.

In present time, many tasks are not as time consuming as they were before LiDAR. Many earthworks today are done with LiDAR, instead of using expensive and slow fieldworks techniques. LiDAR data achieved meaningful technological development and now is taking advantage in natural resource science (J.D. Muss, 2010). The information from LiDAR point cloud is precise and efficient. Digital Elevation Model is the one of the most conventional products from LiDAR (L. Goncalves-Seco et al., 2006, W.L. Lu et al., 2009). The potential LiDAR point in an urban context is capability automatically provide digital models with form and comprehensive structural information (J. Osborne, 2002). Hence, extraction of the willing research object is very important. Additionally, the usage of the object must be taken in to consideration (García-Gutiérrez, 2015). Those points make a sufficient impact in the map based on LiDAR data, regardless of whether it is an environmental or urban map. Artificial intelligence or machine learning is the solution to focus on the object of the interest. Usually, classification and regression techniques are used depending on variables of interest
(Tookea et al., 2014).

#### *1.1.1.1 Airborne LiDAR applications in Urban Environment*

The use of LiDAR technology has been quickly increased in the urban areas, because of its benefits compared to traditional remote sensing methods. In many studies, it is proven that LiDAR can be used not only for classification, but also for object recognition, extraction and different type of analysis.

The urbanization processes can be organized and planned in a more efficient way with LiDAR technology. Precise Classification and recognition results lead to a new level of urban planning. According to Wai Yeung Yan (2015), LiDAR usage can be sorted into two different sections:

1. Urban Morphology
2. Green analysis

Wai Yeung Yan refers to “urban impervious surface extraction (Germaine & Hung, 2011; Hodgson et al., 2003), urban environmental quality assessment (Garcia-Gutierrez, Gonçalves-Seco, & Riquelme-Santos, 2011) and urban change detection (Stal, T., et al., 2013; Teo & Shih, 2013)”.

Environmentally friendly and ecology-based city planning faces issues, which can be solved with data provided by LiDAR green analysis. Hecht et al. (2008) utilized digital surface, where deciduous trees are replicated with fuzzy logic method. This model provides information to evaluate urban green volume. Yao and Wei (2013) suggested the AdaBoost classifier. This classifier focuses on trees in urban areas. The precision is 0.65m in longitude and latitude, and 0.12m in elevation. Huang et al. (2013) provides object-based technique based on LiDAR and images data to calculate green volume. This technique has a few steps. Everything starts with DSM model. When it is created, the object of interest (urban vegetation) can be extracted according to NDVI. Vegetation can be separated into 2 types as tree pattern and grass pattern. Having diverse types of vegetation, the willing one can be analyzed, and different green volume can be calculated.

#### *1.1.1.2 Feature Spaces of LiDAR*

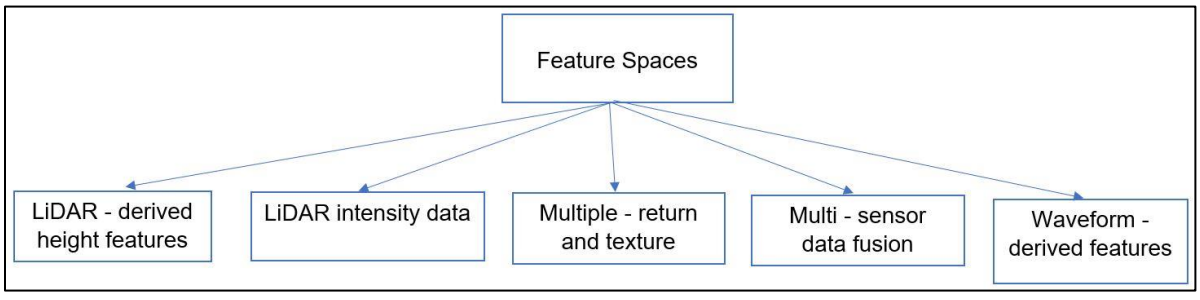



**Figure 1.** Feature Spaces (Inspired by Wai Yeung Yan et al., 2015)

Figure 1 represents feature spaces of the LiDAR. They are LiDAR derived height features, LiDAR intensity data, Multiple - return and texture, multi - sensor data and waveform - derived features.

- LiDAR - derived height features

Growing perception if LiDAR classification and object identification can be assign by the prevailing height feature derived. With the progress or the LiDAR sensor, which is able to provide Z value, arrives advanced land cover classification. Data with Z value shown influence on detailed and truthful delineation of Earth features despite urban or natural surroundings. Generally, the DSM model is created by interpolating 3D LiDAR data. This assists to identify and split particular classes in land cover. According to Hartfield, Landau, & Van Leeuwen (2011) and Priestnall, Jaafar, & Duncan (2000), precision can be increased 5% - 6% by fusing LiDAR - l derived height features on multispectral images. Still, there might be a difference between DEM and DSM from LiDAR. That difference might result in an unreliable establishment of the above - ground component. With sophisticated filtering methods for LiDAR data (Sithole & Vosselman, 2004; Zhang et al., 2003), terrain can be produced combining LiDAR point cloud with a normalized height component. Bartels & Wei (2006), Brennan & Webster (2006), Hartfield et al. (2011), proved that this normalization demonstrates efficiency in increasing classification accuracy. Scientists (Charaniya et al., 2004; Hecht, Meinel, & Buchroithner, 2008; Huang et al., 2013) conducted experiments, which found that LiDAR - derived height features can considerably recognize and divide high and low vegetation. More height transformation types were observed, for example “the height variation (Charaniya et al., 2004), mean, variance and standard deviation of height in the first echo (Bartels & Wei, 2006), homogeneity, contrast, and entropy of height (Im, Jensen, & Hodgson, 2008)”. Nevertheless, none of these performances were as good as using a combination of LiDAR height and intensity data (Wai Yeung Yan et al., 2015).

- LiDAR intensity data

Intensity is a radiometric component. It aids as a supplementary feature for classification. For a distinct return LiDAR sensor, the intensity shows the highest amplitudes registered in the laser from the backscattered of the objects. Here intensity normally consist of 8-12 bits.

As shown in Figure 2 the sensor doesn't only take the number of echoes into account, it also considers the additional storing pulse emission and the backscattered echoes. Intensity data as used for land cover classification, as well as for many different purposes. Mazzarini et al. (2007) was working with lava flow identification and mapping, Lang & McCarty (2009) used intensity to analyze forest wetland, Garroway, Hopkinson, & Jamieson (2011) used LiDAR intensity data for agricultural watershed, Kaasalainen et al. (2010) analyses moisture, Burton, Dunlap, Wood, & Flaig (2011) were observing rock properties.

Song at al. (2002) focused on intensity feature for land classification. The tested data showed that numerous different objects, like roads, grass, roofs, tree, can be classified.

Charaniya et al. (2004) demonstrated that some sensitive objects, which have a similar level, for example, roads and grass, can be separated. Brennan and Webster (2006) provided that intensity data is effective by separating objects with unlike reflectance, for example, bright and dark surfaces. Im et al. (2008) supervised sensitive analysis. It was raised from 10% to 20% when the intensity feature was added in the analysis. Zhou et al. (2009) presented, that satellite images can be combined with LiDAR intensity data. This combination helps to solve issues with shades areas in urban environment.

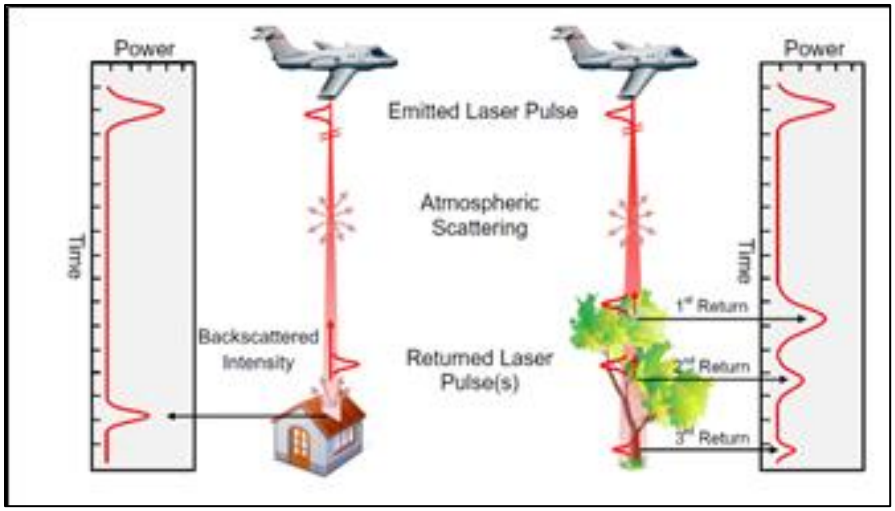



**Figure 2**. Laser pulse (Wai Yeung Yan et al., 2015)

- Multiple - return and texture features

Next to the height and intensity there is a multiple return, which assists in the progress of land cover classification. Analyzing the first and the final returns in connection with some other features such as height or intensity or with differences in that group can provide bigger feature space. In 2004, Charaniya et al. proposed a method on how to increase precision in roads and building by 5%-6%. The solution was to make one more component, which is the difference between the first and last return. This method was repeated by Bartels and Wei (2006). He confirmed that precision was increased. Brennan and Webster (2006) and Buján et al. (2012) identify penetrable objects against non – penetrable ones by including multiple-return data in the object-oriented decision tree classifier. This provides a more precise solution to differentiate objects like trees and buildings, because normally trees have multiple returns due to their different layers (i.e. leaves, stems and branches). Singh et al. (2012) conducted research in a large area in the USA. This research revealed that despite the classification method, the difference in the first and final return improve the classification accuracy. However, the productivity of this method is not always granted. In order to get satisfactory results, the environment should be include a combination of randomly dispersed buildings and trees.

Texture analysis considers the allocation and deviation of nearby pixels data. Therefore, the classification criteria should include spatial properties. Im et al. (2008) produced GLCM (Gray-Level Co-Occurrence Matrix) in order to promote classification. The GLCM consisted of homogeneity, contrast, entropy and correlation. However, that solution did not prove better classification when compared to LiDAR derived heights combined with intensity. Samadzadegan, Bigdeli, and Ramzi (2010) used mean, entropy, standard deviation and homogeneity to extract tree, buildings, and ground. They concluded that only entropy texture increasing accuracy. Huang, Zhang, and Gong (2011) generated homogeneity, angular second moment, entropy and dissimilarity. The outcome was that only 19 x 19 window can commit with SVM classifiers. Regardless of these research studies there are not many methods and proofs with texture features and LiDAR data classifications (Yan, Shaker, & El-Ashmawy, 2015).

- Multi - sensor data fusion

Spectral information is not included in LiDAR data, but it can be attached from other sources. LiDAR data combined with spectral information significantly increases the precision of classification. There are two factors which should be completed:

1. The coordinate system of the image should match coordinate system of the LiDAR data.
2. The spatial resolution has to be the same in both data sources.

For the first point, a simple solution is to take aerial pictures from the same flight as the LiDAR.

For large resolution images as QuickBird (Chen et al., 2009) and WorldView (Kim & Kim, 2014; Minh & Hien, 2011) georeferencing can be done before merging with LiDAR data.

The second point can be reached by using regionalization. The procedure's outcome is common resolution for LiDAR and image data. The selecting rules are made by taking into account the content of LiDAR and image data (Huang et al., 2008). For example, Chen et al. (2009) used QuickBird Normalized Difference Water Index (NDWI) and Spectral Shape Index (SSI) to select shadow and water. To differentiate objects, which are higher than the ground, like buildings, nDSM was taken from LiDAR data. Very much alike the idea of classification was used by Sasaki et al., (2012) and Buján et al. (2012). Hartfield et al. (2011) claims, that after combining multi-spectral image and LiDAR data for classification with 8 classes, the precision increased 5.2% (from 84% to 89.2%). This is because LiDAR data clarifies mistakes between herbaceous and tree/shrub classes. Those and other similar research like Zhou et al. (2009), Guan, Ji, Zhong, Li, and Ren (2013), MacFaden et al. (2012) proves that multi-sensor data fusion is practical and beneficial explanation, especially in a big territory land cover mapping (Wai Yeung Yan et al., 2015).

- Waveform - derived features

Besides multi-echo LiDAR-derived features, full-waveform LiDAR provides astonishing performance by producing data for topography and land cover classification. Combined with the onboard equipped waveform digitizer, airborne LiDAR sensor can store full waveform of the backscattered laser pulse signal. The time consumed is just nanosecond (ns) and the outcome is a 1-D signal profile (Mallet & Bretar, 2009). Scanning techniques usually require more than one backscattered. For accurate classification the data should be kept in the same order. Wagner et al. (2006) suggested use Gaussian components for waveform decomposition. However, the symmetric hypothesis of Gaussian decomposition

might not demonstrate the backscattered signal in real life due to the multiplicity of terrain and LiDAR system settings. Consequently, new techniques and methodologies were proposed to enhance estimation of the backscattered waveform geometry. Chauve et al. (2007) reviewed two classic expansions of Gaussian: Lognormal and generalized Gaussian functions for better waveform modeling of the LiDAR. The generalized Gaussian modeling method was presented as an advanced solution for the top point recognition with the same extra parameters. The first time full - waveform data type for classification was used by Mallet et al. (2008). Different waveform characteristics might be obtained from the Gaussian decomposition function to make land cover classification. "Commonly, the waveform amplitude, number of echoes, echo width, and the difference between the first and last echo pulse are tested for urban land cover classification (Alexander et al., 2010; Chehata et al., 2009; Neuenschwander, Magruder, & Tyler, 2009; Niemeyer, Wegner, Mallet, Rottensteiner, & Soergel, 2011)." Even though some tests extract from 10 to 18 features only a few can really provide help in differentiating particular land cover classes. Mallet et al. (2008) conducted research which presented that the echo width can very accurately differentiate vegetation from human made objects. Chehata et al. (2009) *proved* Mallet et al. (2008) research one more time by differentiating trees from land cover.

Lin and Mills (2010) published a paper in which they proved that even though the pulse width is influenced by an areas' roughness, it is still lower in mistaken data than the intensity data. Vaughn, Moskal, and Turnblom (2011) worked with Fast Fourier transformation. They changed waveform into frequency. The case was to classify tree species, they achieved their goal with 75% precision. Alexander et al. (2010) showed that backscattered performed better in differentiating low ground features. Neuenschwander et al. (2009) analyzed classification land cover from LiDAR and QuickBird. The difference in precision was 14.6% (85.8% in LiDAR and 71.2% in QuickBird).

### 1.1.2 Airborne Remote Sensing

The earth surface can be measured remotely by airborne remote sensing with the sensors fixed downward or sideways on an aircraft. The benefit of this remote sensing method compared with satellite imagery is that it produces very high spatial resolution images (< 20 cm). However, it is only useful for a small area mapping. Normally, airborne are planned for collecting the data one time, when satellite observations are used for monitoring (Liew, S.C., 2001).

There are many types of aircrafts which can be used. The options depend of the project type and budget. The speed depends on sensor system which is installed, but normally it is 150 km/h and 750 km/h. The flying altitude depends on the desired resolution. Moreover, the aircrafts position plays an important role in the quality and precision of the data geometry. The positioning is represented by 3 rotation angles: roll, pitch and yaw angles (Figure 3. rotating angles) (Khorram S. et al., 2012).

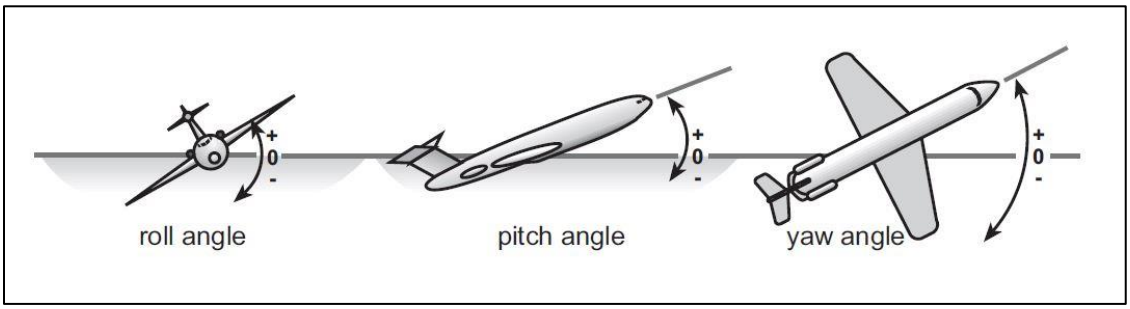



**Figure 3.** The three angles (roll, pitch and yaw) of an aircraft that influence the geometry of the acquired images (Khorram S. et al., 2012).

### 1.1.2.1 *Image data characteristics*

RS image looks like a picture, but it is much more than a picture. Image contain information about EM energy. The data is kept in a grid with rows and columns. One piece of this grid is called a pixel. A pixel holds information of the picture element as Digital Number (DN). Normally, the data is divided by bands, one band has information about one measured wavelength range (Figure 4) (Khorram S. et al., 2012).

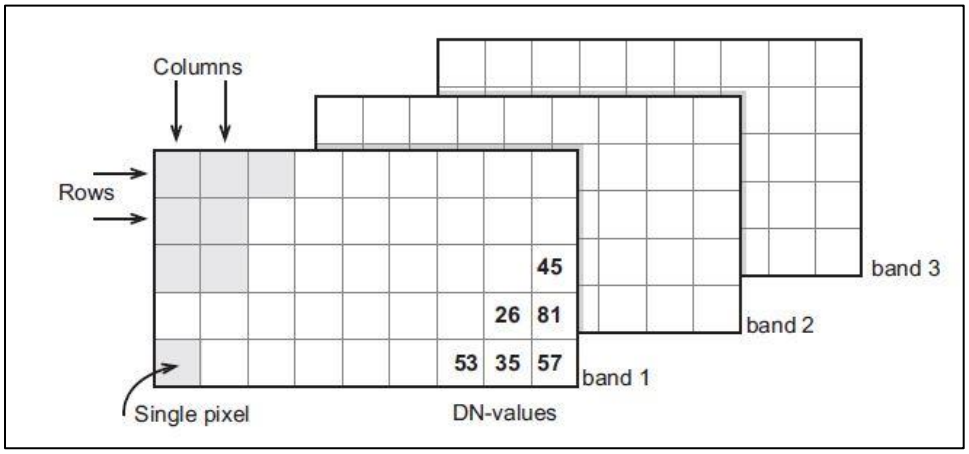



**Figure 4.** Image bands and DN values in pixel (Khorram S. et al., 2012)

Figure 5 represents image characteristics.

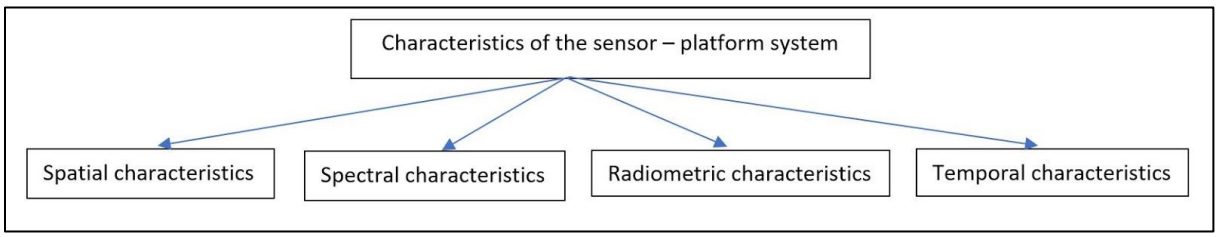



**Figure 5.** Image characteristics (Inspired by Khorram S. et al., 2012)

Image characteristics usually are defined as the characteristics of the sensor-platform structure.

- Spatial characteristics are about the area of interest
- Spectral characteristics specifies spectral wavelengths of the sensor
- Radiometric characteristics indicate energy level measurements
- Temporal characteristics presents time of the information/measurement gaining (Khorram S. et al., 2012).
- Spectral reflectance curves

Irradiance and radiance are two important parameters. The energy which going to the surface is irradiance, the energy which coming from the surface is called radiance.

Every metal can be represented by special reflectance curve. The curves provide information about radiation, which comes from reflected wavelengths. This leads to the new information about reflection angle of wavelength.

Normally, remote sensing sensors are precise and can expand wavelength bands. This is why the curve is valuable for determining overall reflectance. The curves can be collected in specific libraries, because each material has a different reflectance curve (Khorram S. et al., 2012).

- Vegetation

The main parameters of reflectance in vegetation are direction and structure of leaf canopy. Features like pigmentation, thickness and composition (cell structure), amount of the water in the leaf tissue affects distribution of the reflected radiation.

Figure 6 provides an ideal reflectance curve.

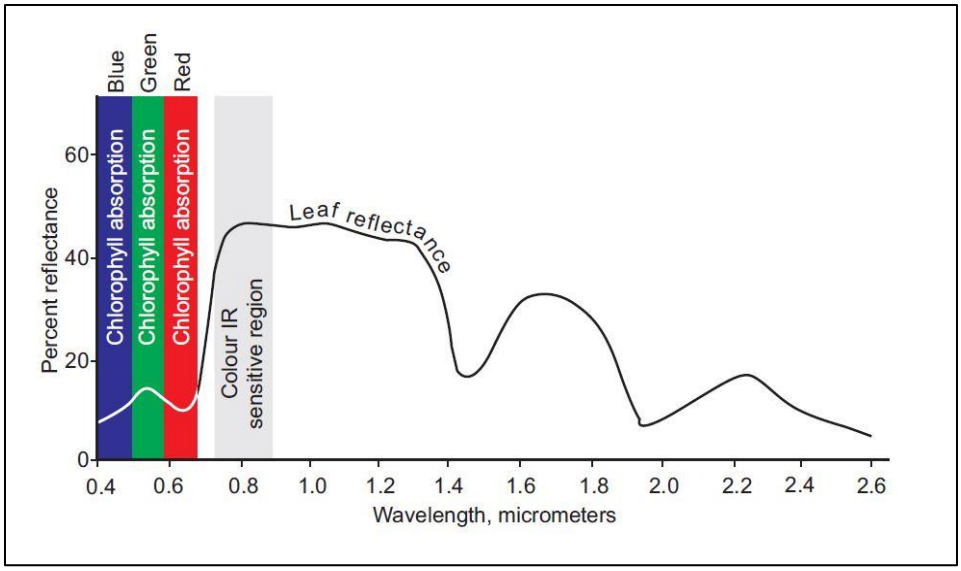



**Figure 6.** Hypothetical perfect fitting spectral reflectance curve of a healthy vegetation (Khorram S. et al., 2012)

From the Figure 6 it seems, that two colors - blue and red are reflected low amount of energy, all energy is absorbed during the photosynthesis process. But green light is reflected in a larger proportion. The most notable reflectance comes from near-infrared. This quantity depends on the leaves growing stage and cell structure. The reflectance has a tendency to go lower, when the amount of the water is higher. For reflectance and water correlation, they are referred to as water absorption bands. When the leaves are losing chlorophyll, they change color. The spectral reflectance curve changing the parameters and now, we can notice, that some bands will have different distribution of reflection. If the leaves are becoming yellow, the reflectance of the red color will be unusually high, and the middle infrared reflectance will increase, while at the same time near-infrared will act the opposite.

These parameters and their changes are showing information about vegetation type and also the health condition of the tree. (Khorram S. et al., 2012).

## 1.2 Machine Learning

This chapter will discuss different machine learning techniques for classification. An overview of different libraries and software where machine learning can be implemented and some main classification algorithms and how they work, will also be discussed. This is an important part for the technical classification aspect because the results depend on the methodology of classification and the chosen algorithm.

### 1.2.1 What is Machine Learning?

The purpose of machines is to be faster and more accurate than human. Mechanical jobs such as traffic control (traffic lights), deep holes digging, and many others were overtaken by machines and our life became easier. Even though machines cannot understand and evaluate emotions, art and many other things, they have a huge advantage over us when it comes to mechanical job, which they can perform much faster and with more precise than us. For example, there are many ways in which human can find the smallest number in an unordered list. As well, there are many different algorithms for machine to calculate it. All algorithms will find the smallest number, but the time cost, the data size can be different. The main difference between machines and humans is intelligence. Humans are able to learn from previous experience by analyzing data and making decisions from our past knowledge. Artificial intelligence (AI) brings machines closer to human. These machines are programmed to remember and take human-like decisions. The data is operating by AI the way, that computer can remember and recognize specific sequence in the data. Machine learning (ML) is one section of AI. Figure 7 illustrates ML position and relationship with related fields (Mohssen M. et al., 2017).

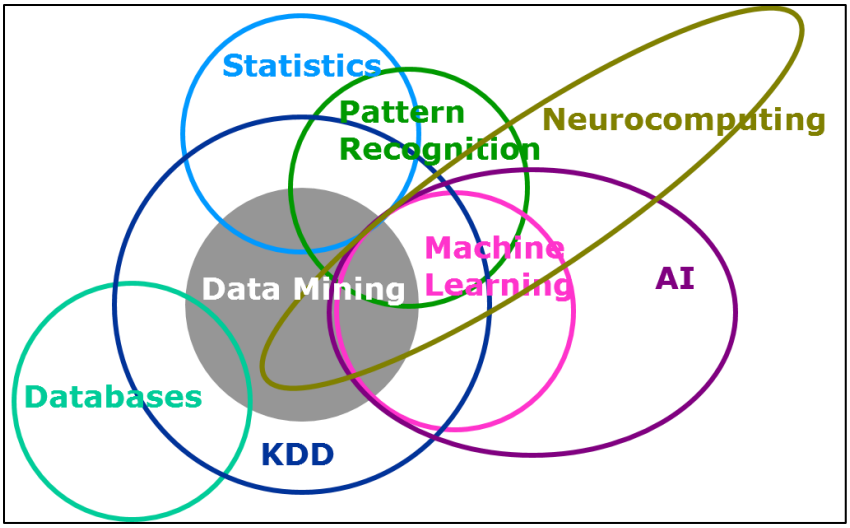



**Figure 7.** Machine Learning position and relation between other brunches. (Mitchell-Guthrie, P., 2014)

According to Tom M. Mitchell ML is a combination of Computer Science and Statistics. The defining question of Computer Science is “How can we build machines that solve problems, and which problems are inherently tractable/intractable?”

The question that largely defines Statistics is “What can be inferred from data plus a set of modeling assumptions, with what reliability?” The defining question for Machine Learning builds on both, but it is a distinct question “How can we build computer systems that automatically improve with experience, and what are the fundamental laws that govern all learning processes?””. ML target is how to reach the step, where computers program themselves (Mitchell M.T., 2006). The idea is to write computer programs, that will make machines learn. The aim is to teach machines how to learn by writing computer programs. With the learning machines doing some tasks, for example predictions. The main target is to have a strong model, which can produce the desired output by data which was imputed. The model can be just an approximation. This means, that in some cases ML output will have some errors but most of the time the model provides the desired output.

There are four Machine Learning techniques, which will be discussed in the following chapter.

### 1.2.2 Learning techniques

The four Machine Learning techniques are: Supervised learning, Unsupervised learning, Semi-supervised and reinforcement learning. Figure 8 shows ML techniques with the appropriate data.

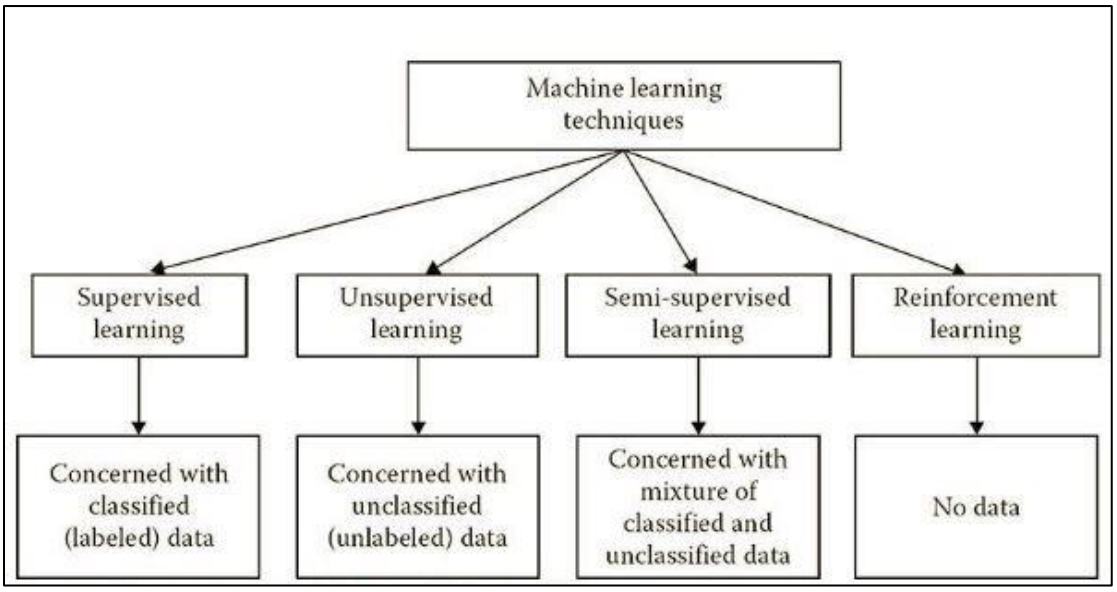



**Figure 8.** Machine learning techniques and data requirements (Mitchell M.T., 2006)

- Supervised learning

The supervised learning technique has a goal to recon a function from a training set, which has labels. The training data has a training example. Training example is input vector X and output vector Y. Vector Y is explanatory vector. That is to say, training data covers training examples.

Every training example has a label corresponding to every output vector Y in the training data. The Y vectors comes after inspection. The inspection can be done by machines, but normally it is done by humans. The labeling which is done by machines results in more errors, whereas, labeling done by humans is more accurate.

Figure 9 presents how labeling can be done according to different criteria. The column named “Example judgment for labeling" describes examples what kind of criterion can be. For example, to know if there is any real estate property in the picture we can look for a house or for a car, which is usually parked next to the house or urban areas.

The next column represents all possible labels for the criterions. The last column shows who can provide this labeling. For example, the criterion is or was the word “football” mentioned in the voice recording? The answer is quite intuitive, either yes or no. This label can be given by machine or by human. Even though speech detection has been progressing for quite a long time it still requires improvement. The last example on the Figure 9 with criterion “Tumor presence in X-ray?”, requires a different level of labeling. The labels must be given not just by any simple human, but rather by an expert of this field.

| *Unlabeled Data Example* | *Example Judgment for Labeling* | *Possible Labels* | *Possible Supervisor* |
|---|---|---|---|
| Tweet | Sentiment of the tweet | *Positive/ negative* | Human/ machine |
| Photo | Contains *house* and *car* | *Yes/No* | Human/ machine |
| Audio recording | The word *football* is uttered | *Yes/No* | Human/ machine |
| Video | Are weapons used in the video? | *Violent/ nonviolent* | Human/ machine |
| X-ray | Tumor presence in X-ray | *Present/ absent* | Experts/ machine |

**Figure 9.** Demonstrates how in different scenarios labeling can be done (Mitchell M.T., 2006)

- Unsupervised learning

Unsupervised learning is the learning without the use of labels. Instead of having direct labels, the aim is to find specific data structures. The algorithms should entirely unattendant find answer. There can be a lot of reasons why data is unlabeled. For example, manual labeling is too expensive or data by itself cannot have labels. “The variety, velocity, and the volume are the dimensions in which *Big Data* is seen and judged”. So, to gain something from the data without inspection and labeling is very important. This is one of the main questions for machine learning development (Mitchell M.T., 2006).

- Semi-supervised learning

Semi-supervised learning is a combination of supervised and unsupervised machine learning. This means, that we have two different type of data. One has labels and another one does not. This type of learning is very similar to human learning. The living environment is very new and strange for a child. This means, he has a lot of unlabeled data. But there are some people, like parents, who introduced him/her to this environment. They are giving “labels” for cat, dog and other subjects. This is how big data is becoming labeled and unlabeled at the same time (Mitchell M.T., 2006). Very common example is a picture. Usually, in a picture there are only a few objects which are known (labeled) - apple, table, all others are unknown (unlabeled).

- Reinforcement learning

Reinforcement learning collects the information which comes from interplay with the environment. Learning has a target to take actions that increase profit or reduce the danger. Reinforcement learning is producing intelligent programs. Intelligent programs, sometimes are called agents.
Reinforcement learning has to perform following the steps:

1. Input is analyzed by the intelligent program.

2. The action is taken according to the decision-making function.
3. When the action is complete, the intelligent program gets reward or reinforcement.

The sequence of the actions, which were taken to achieve reward is collected.

### 1.2.3 Tools and toolkits for implementation

- Scikit

Scikit is presented as easy, understandable and effective tool for machine learning. Moreover, it is an open source so it is freely available for everyone to use. It uses python language and libraries such as NumPy, SciPy and others. With Scikit it is possible to manage classification, regression and clustering. For example, Support Vector Machines, Random Forests, Gradient Boosting, K-Mean (Scikit-learn 2018).

According to data scientists Ben Lorica, Scikit is very well documented. It is obligatory to add script with examples chronologically, with small data set. Moreover, global API is protected and public API have strong documentation, contributors are doing unit test to analyze or all different pieces of the software working together and are ready to use.

SciKit has very a qualified team. Contributors are specialized in ML and software development. In this case, all models are reliable. SciKit has a significant list of the available tools and it contains a lot from ML tasks. Even though ML tasks are always changing and developing because of a large community of expert volunteers, new tendencies are updated very fast. Moreover, users are protected from the variations of the same algorithm written by different people, this problem is common for R users. Python is one of the most popular language between data scientists. Python interpreter gives opportunity to connect datasets and can be

modified according to the needs of the user. Furthermore, PyData has been strongly developed in the last years. Numerous data scientists work with a few pydata components. They start using IPython notebook and creating multi-step analytic projects, which aim to consolidate results from various pydata tools. Python is one of the preferable languages, that shows PySpark, GraphLab (GraphLab notebook), and Adatao, new analytic tools, with Python support. SciKit, as Machine learning library, has a clear target give a package of the most popular algorithms in stable interface (Lorica B., 2015).

- R

R is open source environment, which is used for statistical calculations and visualization. R is similar to S, but R is an open source product. R support numerous statistical and graphical methods. One of the biggest advantage of R is graphical presentation of data. The default parameters were carefully chosen by designers, but users have full control of it.

R is a collection of software for data handling, estimation and visualization. It contains:

- Productive data management
- Package of operations for calculations
- Big, sequential set of tools for data analysis
- Data analysis in graphical view
- Advanced, but simple and productive language.

R is an entirely planned and rational system. C, C++ and Fortran languages can be run on R and C code can be used for direct R control. R has very robust statistical background and its availabilities can be expanded by packages (r-project 2018).

The biggest advantages of R are:

- Open source: everyone can download and use it.
- Packages: packages are libraries. They are mainly created by academics, so it is straightforward way to state-of-the-art methods.
- Maturity: R was born by S, so the methods and algorithms were improved.

The biggest disadvantages of R are:

- Inconsistency: language requires a lot of documentation reading for each package. All algorithms are developed differently, with different parameters.
- Documentation: documentations usually are too general and short. The help is often not strong enough for specific cases.
- Scalability: the data can be used only in one machine. It is not possible to work with R in different machines or flow data. (Brownlee J., 2014).

- Tensorflow

In 2011, a project called Google Brain was started. The aim was to find a solution for deep neural networks in Google products and for further research. Tensorflow was created by google as open source library. At first it was created for machine learning and deep neural network but because of system flexibility, it can be used for other domains as well. Moreover, computations can be done in different machines, without changing, or with a small change in the code, starting with small mobile devices (tablets, phones) and finishing to large scale computation machines. System is available to work with different algorithms in different subjects. For example, “speech recognition, computer vision, robotics, information retrieval, natural language processing, geographic information extraction, and computational drug discovery”. (Dean J. et al., 2015).

The leading examples of TensorFlow are:

- Voice/sound recognition

In security, voice recognition function is widely used, opinion mining is used with customer relationship management, voice search is used for telecoms purposes, and engine noise is mainly used in aviation and automatics.

In everyday usage, the common speech recognition functions are in google search, google translator, Cortana and many others. Voice recognition supports language understanding which is used to convert speech in to text.

- Text based applications

One of the well-known usage of text-based applications is language detection. Google translator has more than 100 languages. It does not only translate ordinary dictionary words but also understands slang, gives sentence translation synonyms. Google uses sequence-to-sequence learning to summarize text. Text summary provides fast article headline prediction. Moreover, Google has a well-developed “smart replay” function, which creates auto answers to emails.

- Image recognition

Image recognition is used for many different purposes, from social media to aviation - whenever people or object need to be identified. In engineering, it is widely used for 2D building transformation to 3D. The buildings are identified from pictures and then the model is created. Ordinary people have interactions with image recognition on Facebook, for example, when they want to tag someone in a picture. Image recognition is becoming more and more popular in healthcare systems. Computers can identify same illness patterns from scanning which is very useful for doctors.

- Time series

Time series are used for predictions and recommendations. They analyze the of flow data in some period and give statistics with forecast. It is widely used in markets like Amazon, Netflix, where according to previous customer choice, algorithm tries to predict what customer might want after.
It is also used for financial predictions, for example, risk detections and so on.

- Video detections

Tensorflow can work with video detection. Neural networks can detect motion and is widely used in games, airports security. Tensorflow is used in a big project, YouTube - M8. The purpose is to speed up large video understanding. Moreover, NASA uses Tensorflow to find out in advance and forecast what object are near the earth.

The list of Tensorflow availabilities and usage is not limited, because it is open source library with a strong contribution.

### 1.2.4 Classification

- Support Vector Machines

“Support vector machines are supervised learning algorithms based on statistical learning theory, which are considered as heuristic algorithms.” (Kavzoglu, 2009). SVM method is based on hyper plane. This plane should be separated in to two different classes. The hyper plane is found by train data sets and then it is checked by test data.

If the data set has k dimensions, then SVM hyperplane will be k-1. Figure 10a presents the diversity of hyper planes, which separate the classes. Nevertheless, there is only one hyperplane, which keeps the biggest separation between two classes (Figure 10b). This plane is called the optimal hyperplane. In Figure 10b there are points called “support vectors”. Support vectors are points which restrain the width of the margins.

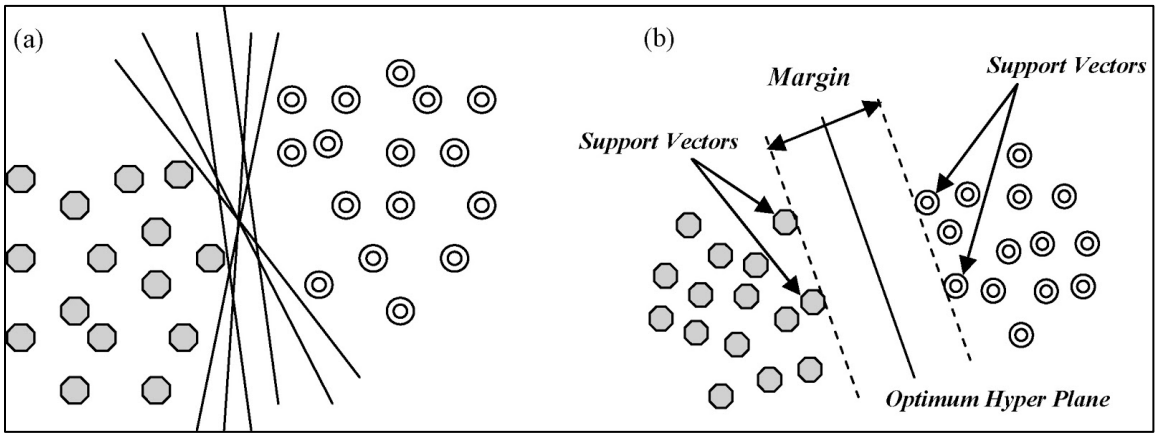



**Figure 10.** Hyper plane and support vectors (Kavzoglu, 2009)

“SVM attempt to detect the best place to hyper plane, where the margins reach its maximum between two classes.” Assume that a training data set containing k number of samples is represented by {xi, yi} (i = 1,…, k) where x ∈ RN is an N-dimensional space, and y ∈ {−1, +1} is class label.” (Kavzoglu, 2009). Figure 11 presents, that the optimal hyperplane is

$$\omega * x_i + b = 0 \quad (1)$$

where x - point on the hyperplane, w - orientation in the space of the hyperplane, b - shift in the distance from the center.

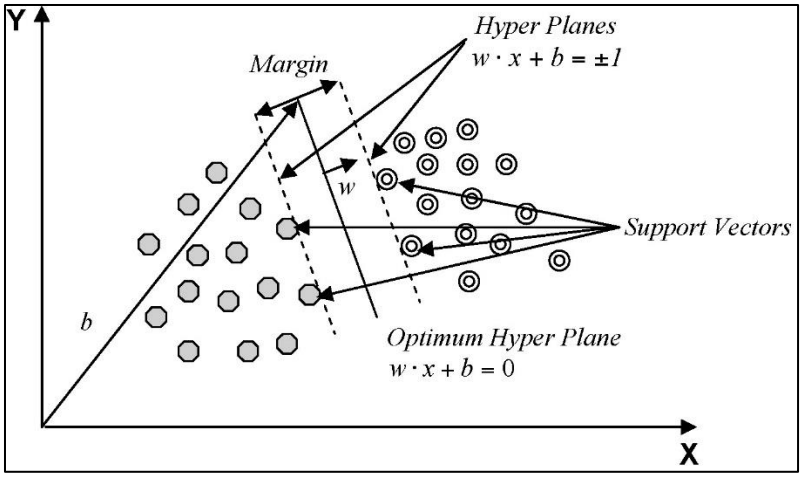



**Figure 11.** Optimal hyperplane (Kavzoglu, 2009)

The hyperplane can be categorized into two classes:

$\omega * x_i + b \geq +1 \text{forally} = +1$ (2)

$\omega * x_i + b \leq -1 \text{forally} = -1$ (3)

Or can be written:

$y_i (\omega * x_i + b) - 1 \geq 0$ (4)

Support vectors are defined as

$\omega * x_i + b = \pm 1$ (5)

They are parallel to the optimum hyperplane (Mathur and Foody. 2008).

As shown in Figure 12a some data is not linear. Usually, remote sensing data is not linear as well. The hyperplane cannot be placed with previous equations (Figure 12b).

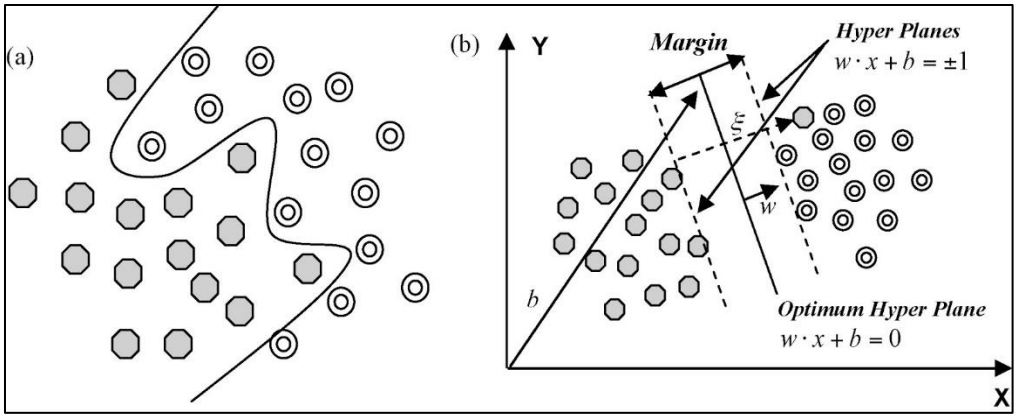



**Figure 12.** Nonlinear hyper plane (Kavzoglu, 2009)

In nonlinear cases there are ξ slack variable

$$\min \left[\frac{\|\omega\|^2}{2} + C \sum_{i=1}^{r} \xi_i\right] \quad (6)$$

$$y_i (\omega * x_i + b) - 1 \geq 1 - \xi_i, \xi_i \geq 0, i = 1,\ldots,N \quad (7)$$

C - constant.

C provides sharp balance between margin maximization and error minimization.

m ξ - is the distance between incorrectly classified points and the hyperplane (Oommen, 2008). The value is directly correlated with misclassified samples.

Figure 13 shows, how to rise to bigger dimensional space if linear equations are not working.

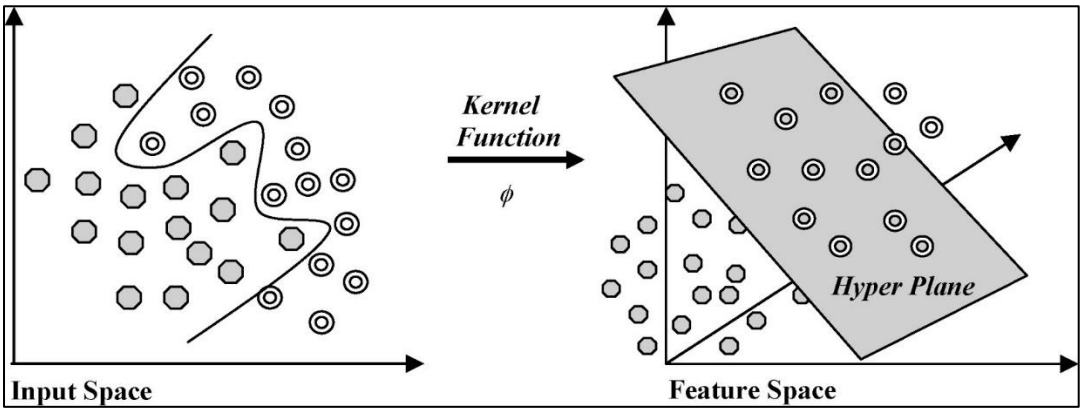



**Figure 13.** Converting to bigger dimension (Kavzoglu, 2009)

Kernel has a function $\phi$, then input points are $\phi$(x) in dimension H. the classification function is

$$f(x) = \text{sign}\left(\sum_i^r \alpha_i y_i K(x, x_i) + b\right) \quad (8)$$

$K(x,x_i)$ - Kernel function

$(y_i)\alpha_i(i = 1,\ldots,r)$ - Lagrange multiplier.

Kernel function allows to transmit data points so that hyperplane can be defined (Dixon and Candade, 2008).

Kernel Radial Basis Function (RBF) is defined as:

$$e^{-\gamma\|(x-x_i)\|^2} \quad (9)$$

RBF is usually used for remote sensing classification (Pal and Mather 2005).

- Random Forest

Random forest is a very universal algorithm. For many years it was used for classification, regression, feature selection and gave desired results, that why it became natural to think about it as a solution for all questions. It can be used for both regression and classification.

Decision trees are very clear and understandable data structures. It works by creating decision rules based on classification question. Each node leads to a decision. The tree is stopped when the final result is reached. Figure 14 shows an example of a hypothetical decision tree.

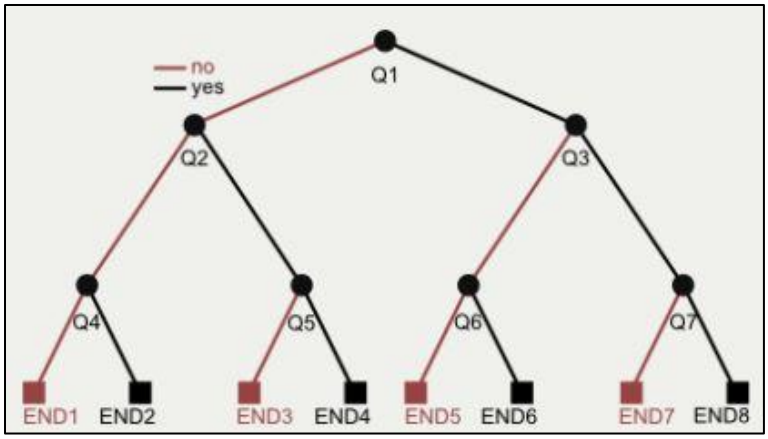



**Figure 14.** Scheme of decision tree (Stanford, 2017)

The critical part of a decision tree is splitting action. Splitting is done on each node, for data to stay clean and get closer to desire classification. Figure 15 presents an example of splitting.


**Figure 15.** Bad and good splitting (Aggiwal R., 2017)

Every node splits the data with a straight line into 2 parts. This is why the final output is outlined with straight lines or boxes. Figure 16 presents classification example.

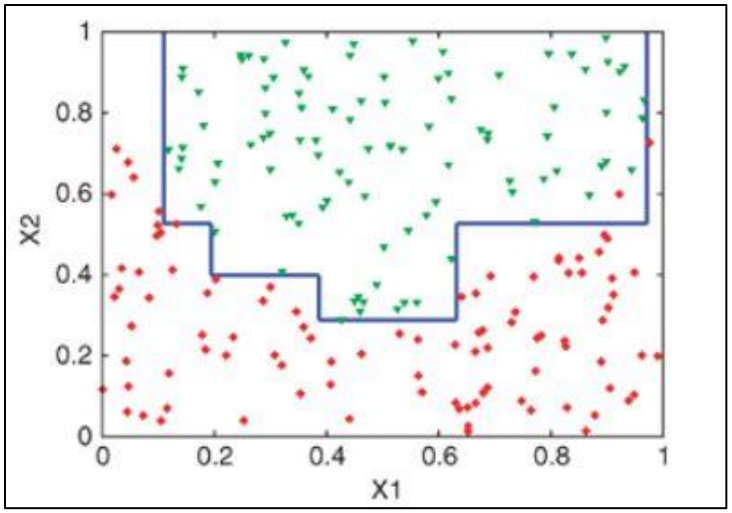



**Figure 16.** Classification example (Aggiwal R., 2017)

For example, if linear regression provides straight line, decision tree can give more advanced staircase boundary (Figure 17).

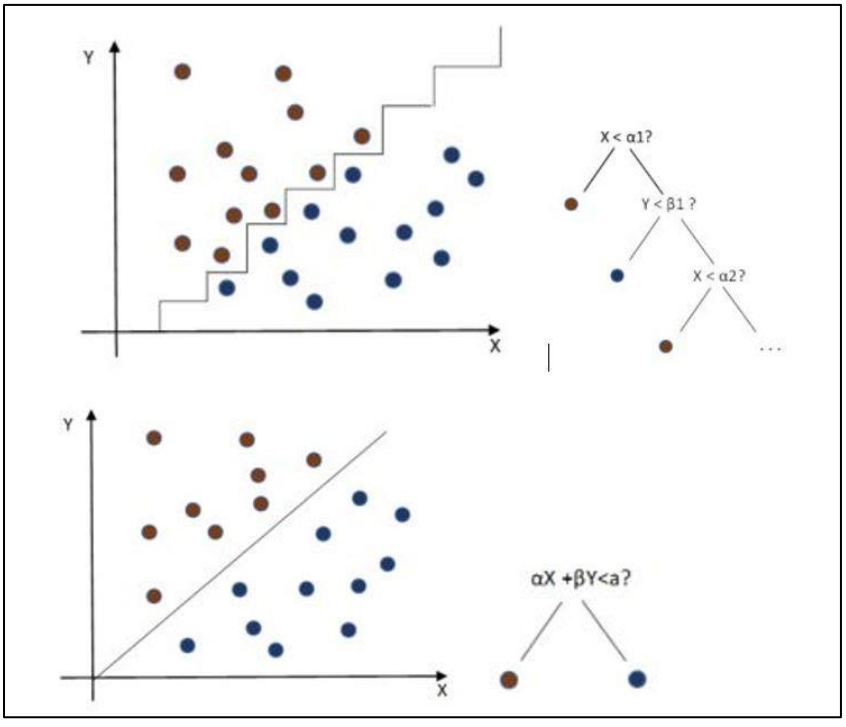



**Figure 17.** Linear regression and decision tree classification (Inspired by Aggiwal R., 2017)

Random forest is a type of advanced decision tree. The main idea is to create small decision trees from random data groups. Every decision tree captures a particular tendency in the data.

During the classification, the biggest number of votes contemplate the class. In real life, this situation would look like asking the same question to many different experts and trusting the majority of answers. (Aggiwal R., 2017)

### 1.2.5 Validation Techniques

In machine learning, data is split into 2 or 3 subsets – train and test or train, validate and test. Then the data model should fit the train model. Unfortunately, what might happen is underfitting or overfitting. So, the model will be inaccurate, or fit only for one dataset.

- Overfitting

Overfitting occurs when a model is over trained. This kind of model remembers the pattern and new data is too confusing for it. This might happen if the model is too complex, with a lot of features in a small dataset. The accuracy is very high in training data, but with unseen, new data accuracy will be very low. Overfitting models are not generalized, such models cannot work with any other data.

- Underfitting

Underfitting is the opposite of overfitting. If in overfitting data was not generalized, then in underfitting data is too general. The model cannot follow the data trend. Moreover, in overfitting underfitted model cannot classify new datasets.

Underfitting happening for too simple models, which don’t have enough independent variables. The accuracy of underfit model is very low.

There are train/test split and cross-validation, which leads to prevention from overfitting and underfitting.

- Train/test split

The dataset should be split into 2 parts: training and testing (Figure 18). Training dataset is for the model to train what we want to achieve. Training part should generalize the model so that after it will recognize pattern of interest in unknown datasets. In training model data goes with answers. Test set is data simulation before real dataset. It gives a general overview of how a model is trained. Usually, train and test datasets are divided 80/20 or 70/30. The split should be random, otherwise, one feature can appear only in one part of splitting. The features distribution should be randomly equal in train and test datasets. If not, overfitting will appear in the data. To avoid this cross-validation method can be used.

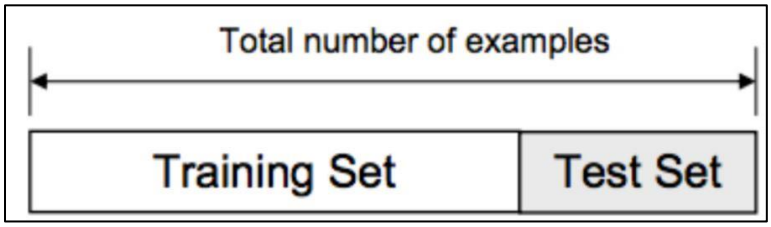



**Figure 18.** Total number of examples split into training and test sets (Bronshtein A., 2017)

- Cross-validation

The cross-validation concept is very similar to train/test. The main difference here is the number of subsets. All data should be split into k subsets and trained with k-1 subsets. In this case, one subset is left for testing (Figure 19).

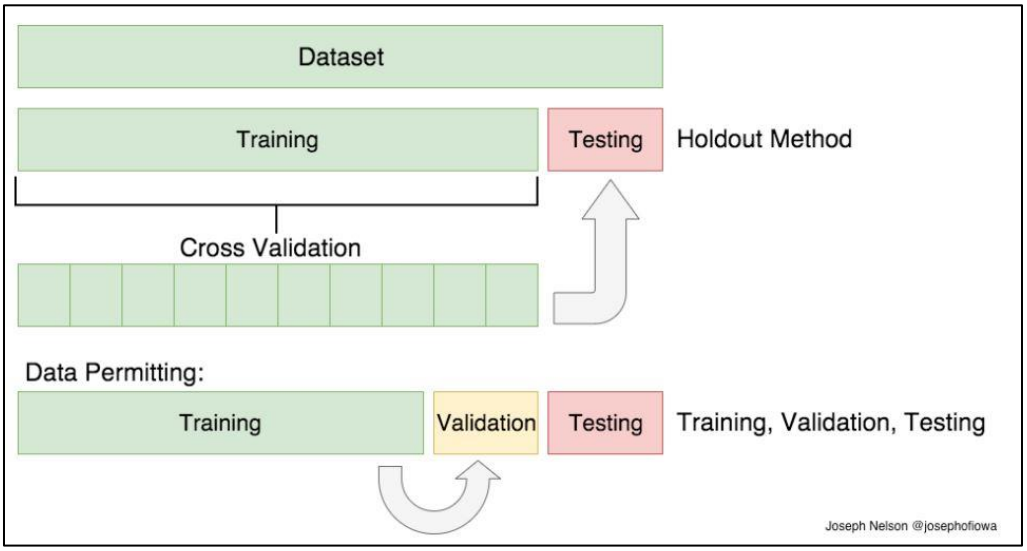



**Figure 19.** Data set splitting for validation (Bronshtein A., 2017)

The 2 main cross validation methods are:

- K-Folds Cross Validation
- Leave One Out Cross Validation (LOOCV)

- K-Folds Cross Validation

K-Folds cross validation method starts with dividing the dataset in k subsets. Then k-1 is used for training and the last one for validation (Figure 20). After that, it is tested with test dataset.

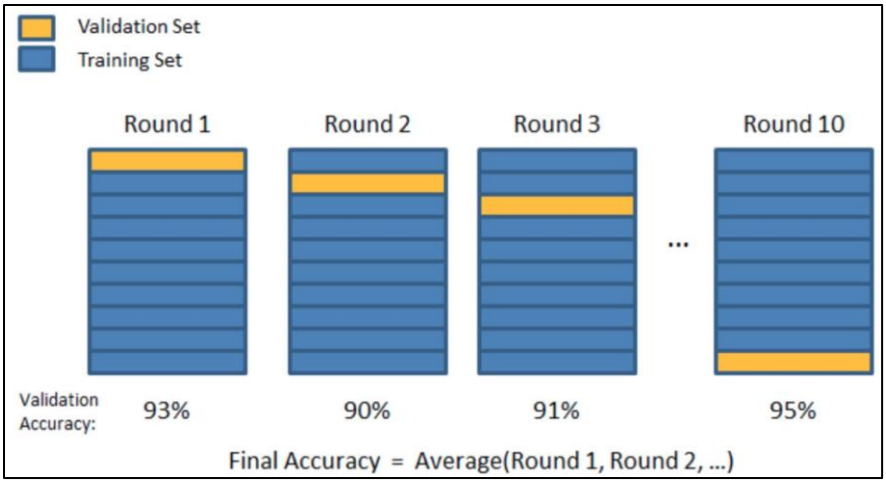



**Figure 20.** K- folds Cross Validation (Bronshtein A., 2017)

- Leave One Out Cross Validation (LOOCV)

In this method, the number of subsets is the same as the number of observations in the dataset. The average is taken from the subsets and then the model is built and finally tested against the last fold. The error estimation formula is:

$$CV_{(n)} = \frac{1}{n}\sum (y_i - y'_i)^2 \qquad (10)$$

In this method, n-1 observations are used, and error is estimated in each sample (Figure 21). The disadvantage here is that this method if effective only for small datasets. The size affects time and computer resources (Bronshtein A., 2017).

In this method, n-1 observations are used, and error is estimated in each sample.

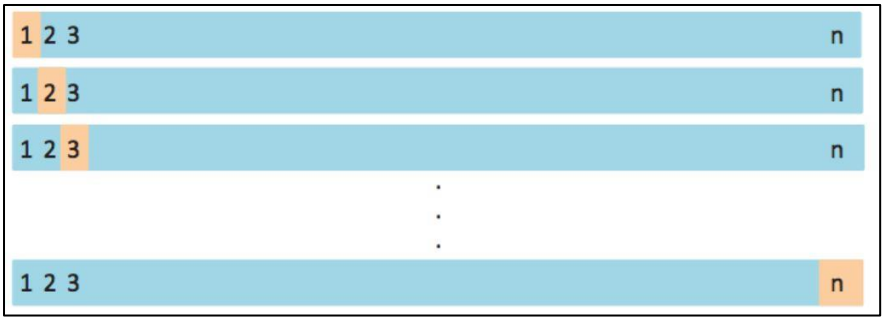



**Figure 21.** Leave One Out Cross Validation (LOOCV) (Bravo H.C., 2018)

## 1.3 Problem Statement

High vegetation in LiDAR dataset of Copenhagen is misclassified which can cause confusions and misleading information while extracting objects/features from it. The aim of this study is to analyze high vegetation classification in a sample urban area – Copenhagen, Denmark- from LiDAR data based on different classification methods in order to find out to what degree the high vegetation can be correctly classified.

## 1.4 Research Questions

- What machine learning based classification can be used for classification of misclassified features of LiDAR data? Which one performs best for the given dataset and geographical region?
- What is the role of training datasets size in the obtained accuracy of each classifier?

## 1.5 Report Structure

This thesis consists of the following chapters:

**Introduction** - Scientific literature review, which gives general impression what has been already done, who and why comes to the similar problem and conclusions. Remote Sensing gives theory about remote sensing types, machine learning is overview about the main viable options to use classification. All those chapters lead to problem statement and research questions.

**Methods and Materials** - road map is a schematic high vegetation classification technique. Data set assist with raw data description and analysis.

**Results** - high vegetation classification output, result comparison and analysis can be found in this chapter.

**Discussion and conclusion** – this chapter examines research questions and concludes with solutions and possible future developments.

# 2 Methods and Materials

## 2.1 Data Set

The city of Copenhagen has always been called, a green city. Where nature and architecture exist in harmony (Bothe, K., 2018). Owing to case studies of high vegetation classification in urban area, Copenhagen is considered an ideal place for experimentation. It is a perfect place to study how classification works in an area where buildings and high vegetation are in close proximity to each other.

Free LiDAR data in available on the internet, the data is provided by Danish government and is reliable. The LiDAR data is provided in *.laz format, it was collected in 2014-2015. The density of the points are 4,5 points in 1sq.m, accuracy, approximately 5 cm in longitude and latitude and 15 cm in elevation. Orthophoto is taken from aircraft and was downloaded in GeoTiff file format. The flight was made from March to May. During those months in Denmark there is no snow or

leaves, so the surface is clean and pictures can represent a precise surface and elevation model. The data is from 2017 with 12.5 cm resolution. The chosen part is shown in Figure 22.

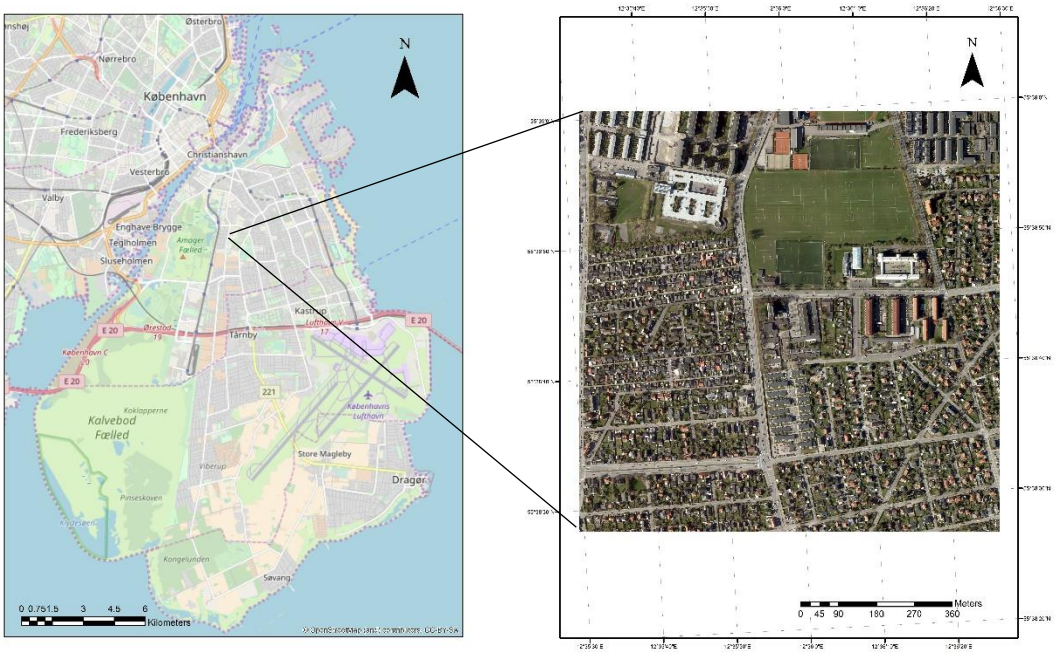



**Figure 22.** Study area in Copenhagen

## 2.2 Flowchart of the implementation

The implementation was done with five main steps: Geographical Location Selection, Machine Learning Tool Selection, Classification Model Analysis, Data Preparation, Accuracy assessment and Comparison Analysis (Figure 23).

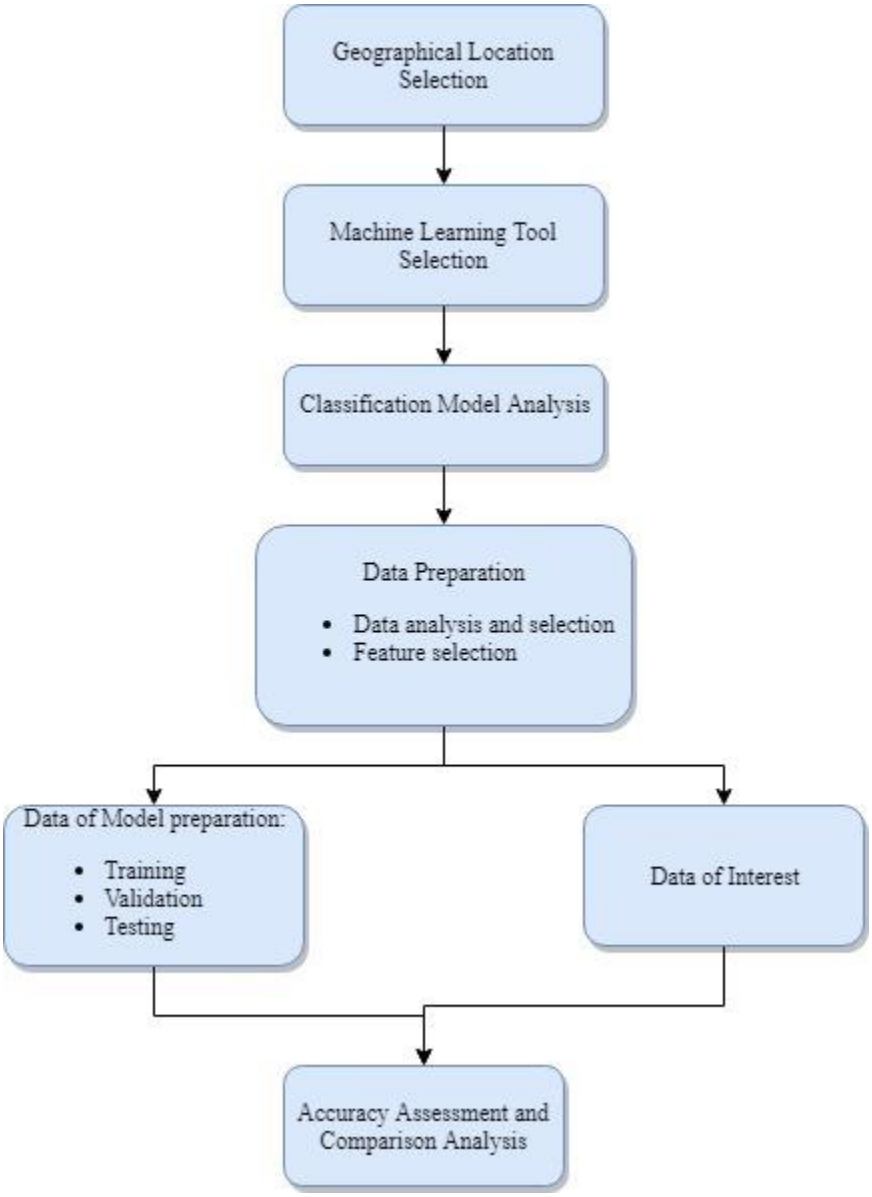



**Figure 23.** The main flowchart of implementation

**1) Geographical Location Selection and primary data review:** This study began with the selection of geographical location and data collection. This step was discussed in previous chapter (2.1 Data Set).

LiDAR data is available in *.laz format. Because the further classification will be done with SciKit – learn, which is Python library, it is better to convert *.laz file into txt. Figure 24 shows how it looks like in *txt format.

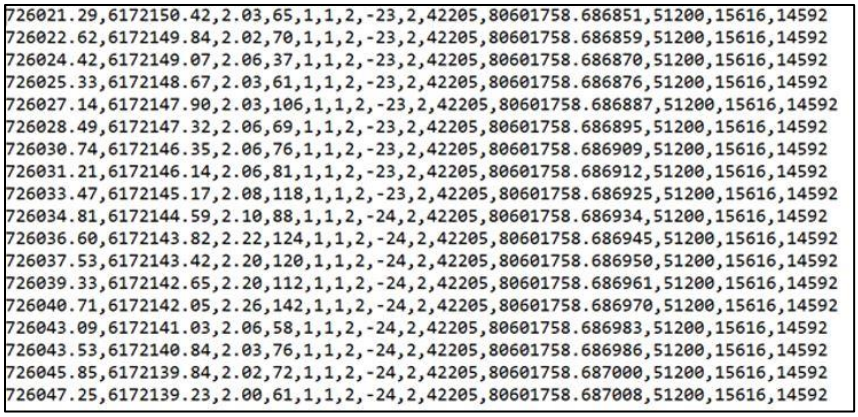



**Figure 24.** Converting *.laz to *.txt format

The information which contains *.LAZ file is:

- X
- Y
- Z
- **Intensity**
- **Return number**
- **Number of returns**
- ***Classification***
- Scan angle
- User data
- Point source ID
- GPS time,
- RGB from LiDAR signal

Intensity, return numbers and the number of returns can assist with high vegetation classification. These three features will be different for high vegetation than for other objects, for example buildings and roads.

One feature is “Classification”. Even though points are classified it is misleading information and cannot be trusted for future analysis. Intensity, return number and

number of returns are features, which comes from sensors, so they can be trusted and lead to more precise classification.

**Figure 25.** LiDAR misclassified points.

When the area of interest is selected and the relevant data is collected (LiDAR point cloud and orthophoto) the next step is to choose the most suitable machine learning tool.

**2) Machine Learning Tool Selection**: As discussed in chapters 1.2.2 Machine learning techniques and 1.2.3 Tools and toolkits for implementation, supervised machine learning will be done for point classification. There are many tools to do that, but because of python power and effectiveness it was decided to use SciKit - learn library. It is an effective machine learning tool. Moreover, it is an open source and it has well documented examples and support SVM, RF and Kernel classifications.

Because of implemented libraries as NumPy, SciPy it is easy to work with matrix format and it manages classification, regression and clustering. Because of big number contributors, which are specialized in ML, I was sure, that classification will be done, without software errors. Models are well developed and checked.

**3) Classification Model Analysis:** with different methods classification will be done differently. To get the best results and at the same time to analyze different strategies for future directions it was decided to do classification with three different algorithms. Support vector machines, Random Forest and Kernel. They are very different in terms of the underlaying algorithm and have gain popularity in GIS science.

SVM and Kernel were chosen because of their popularity. They both are based on statistics. SVM is very powerful on linear objects and Kernel can separate non-linear planes. Both of the functions work with hyperplane and support vectors. Support vectors are points, which restrain the width of the margins. Comparing the methods with the same working style, but different math it was expected to see different, but satisfying results.

Random forests method was chosen as different classification technique compared with previous two. It is a very universal algorithm and used many times for different reasons (see chapter 1.2.4 Classification).

**4) Data Preparation:** Data was prepared for classification model training, validation and testing and data for the area where high vegetation should be classified. Data for training, validation and testing was taken from the block, which is 1km further than classification area. This was done with an intention to avoid data leaking. Machine see the area of interest only one time – during classification when the model is developed well. Data for the area of interest was just cleaned by the previous classification, which has misleading high vegetation classification.

- Training, validation and testing data

Data for model training, validation and testing was taken 1km away from classification area. This was done with intention to have similar point cloud as desired classification area but keep model training points separate from them. Any

kind of data leaking is not safe for model training and affect final classification and accuracy. Training points were labeled with supervision (see Figure 9). This means all training points was selected manually, carefully analyzed and confirmed as trustful for training model. This ensures, that each training point has correct class. The main focus was on high vegetation points. Figure 26 shows the small example of classified points.

**Figure 26.** LiDAR point cloud on orthophoto

Training was done 3 times with different amount of points (Table 1). The area, which needs to be classified has 8 279 149 points in total. The training Set 1 has 133 000 points, the training Set 2 has 66 500 points and training Set 3 has 33 250 points. Different training size will help to analyze accuracy changes. To provide smaller training data size leads to the less time-consuming point selection process. Moreover, smaller training size helps to save space in the machine, which means faster calculations.

**Table 1.** Training Sets and their sizes

| Train Set number | Amount of point |
|---|---|
| Set 1 | 133,000 |
| Set 2 | 66,500 |
| Set 3 | 33,250 |

For validation, the 10-folds validation model was used. 9 folds for training and 1 for validating. With 9 folds for training we will train 90% of data. As shown in Figure 21, 10 rounds were done, and average accuracy was calculated as the final one. In each round the accuracy of training and validation and difference between them to identify overfitting or underfitting was observed. When training and validation results show neither overfitting nor underfitting, testing can be done.

In some cases, it is possible to skip validation and do only testing but, in this thesis, all three steps are performed since there is enough data to implement the entire training Set. The testing data contains 1 300 245 random points. This dataset does not change model work. It gives us an estimate of what accuracy we can expect with a new, never seen dataset and usually it is smaller. Small data set does not require long calculation time and shows the main errors.

Training, validation and test results are restated in the following chapter “Results”.

**5) Accuracy assessment and Comparison Analysis:** Accuracy assessment and comparison analysis are widely discussed in chapter “Results”. This is final result about the model and how it works with new “unknow” data. The main flowchart of implementation is shown in Figure 23.

To get further analysis, the workflow was divided into two mains approaches - approach I and approach II. The first approach is presented in Figure 27. The first classification was done only with LiDAR data. Features as intensity, number of returns and return number was chosen to identify misclassified label “classification”. Those features come from scanner, so they are not misleading,

and we assume, that can give more precise high vegetation classification. The second approach (Figure 28) adds some new features to the first approach. In addition to LiDAR data features, information from orthophoto is added. The features are colors (reg, green, blue) and near-infrared. The main purpose to have two different approaches is to analyze feature influence to classification accuracy and to see or colors and NIR can improve accuracy. The first and second approaches are discussed more detailed in the subchapter below.

### 2.2.1 Approach I

If we look at Figure 23, which shows the main flowchart of implementation, approach I and Approach II, is smaller part for data preparation. The main differ from approach I and approach II is feature selection. Approach one works only with LiDAR data set. Figure 27 presents flowchart of the first approach.

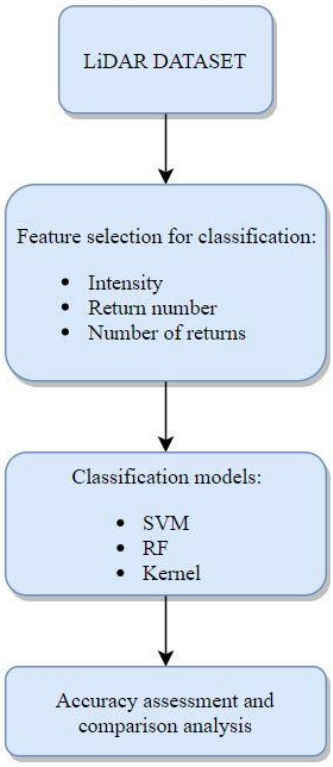



**Figure 27.** The flowchart of the first approach

Raw LiDAR data is taken and converted to txt file as shown in Figure 25. Intensity, return number and number of returns was chosen as independent variables to reclassify points. Classification column during model creation and classification is ignored. This is done with intension not to confuse and not to show incorrect classification. Training, validation and testing data preparation is discussed in chapter 2.2. The data was collected manually, only points with right class was chosen. This method is time consuming, but ensures the best data training. For training and validation class is known, but test dataset is new and unknown for model. Classification models are chosen – SVM, RF and Kernel classification models. All of them are well known, widely used for solving many different types of problems and at the same time, they are very different from each other. The model choice is discussed in chapter 2.2. Accuracy assessment and comparison analysis is the most interesting part. Here the created model is tested with real, new dataset. The target of the approach I is to see how model can classify points only with LiDAR data and different training set sizes.

### 2.2.2 Approach II

We assumed, that combination visual spectrum R, G, B, NIR with LiDAR features can increase classification accuracy. Therefore, approach II was undertaken. Approach II workflow is shown on Figure 28.

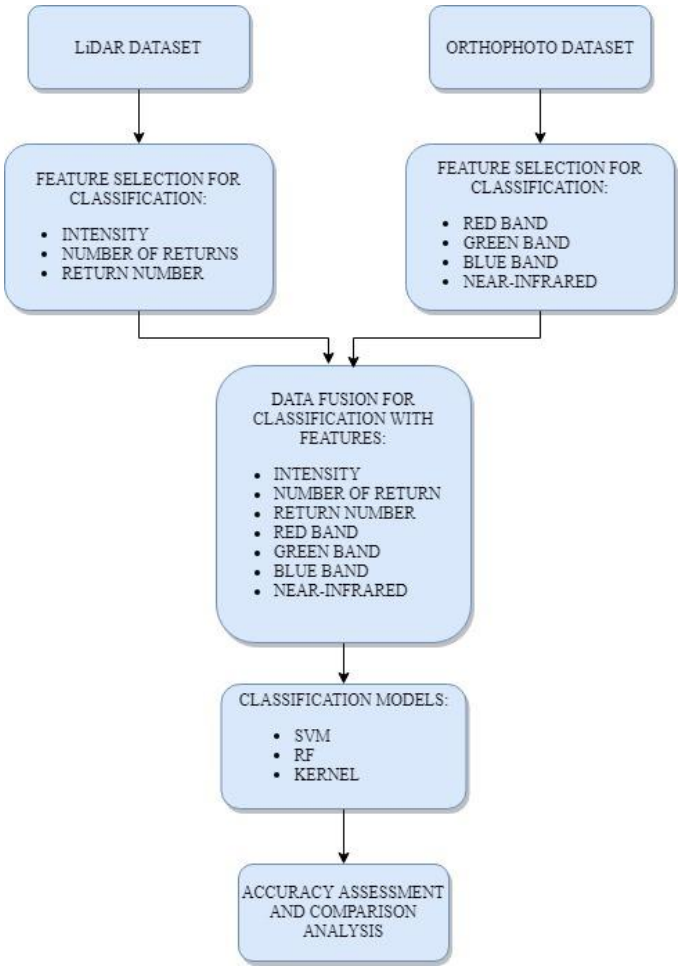



**Figure 28.** Workflow of the approach based on combination of LiDAR and Orthophoto data

The second approach has more features than the first one. Two different datasets were combined to achieve better results. LiDAR dataset has the same parameters: laser intensity, return number and number of returns. Orthophoto dataset comes with four new bands: red, green, blue and NIR. Colors and infrared band is good vegetation identifier. We assume, that colors and NIR should increase accuracy, because vegetation colors are different from other, especially artificial, objects.

Data merging was done according to geolocation. LiDAR points coordinates was compared with orthophoto points coordinates and all points features were

connected together. In this case all training, validation and testing as well as wanted to classify dataset are enlarged with orthophoto features.
The other steps are the same as in approach I. The same classification models: SVM, RF and Kernel were chosen with intension to compare how additional features changing accuracy. As well the same three dataset sizes were used for training, validation and testing. Accuracy analysis is presented in “Results” chapter.

# 3 Results and discussions

Classification model was created by selecting points very carefully. All points must have the correct classification label. The most important of which is the high vegetation label. If the data for model is collected without errors, the model is good and machine learning is ready to classify new, unknown data. To reach the goal there are three important steps:

- Training
- Validation
- Testing

The first step is training. The model should be trained with known data and only then unknown values can be shown for classification. The biggest problems of training are overfitting and underfitting. To ensure, that the model is not overfitted or underfitted, k-folds cross validation test was done. 10 folds were used, and result was averaged. 9 folds were used for training and 1 for validation. This means, that 90% of data will be trained.

- Training and validation

The 10-fold cross validation was done for each classification techniques SVM, RF, KERNEL and for all data sets - Set 1, Set 2 and Set 3. Results for approach I is

shown in Table 2 for SVM classification, in Table 3 for RF classification, and Table 4 for Kernel classification.

**Table 2.** Support Vector Machine k-fold train and validation accuracy results for approach I.

| SVM | SET 1 accuracy | | SET 2 accuracy | | SET 3 accuracy | |
|---|---|---|---|---|---|---|
| | Training, % | Validation, % | Training, % | Validation, % | Training, % | Validation, % |
| Round 1 | 69.12 | 72.31 | 72.13 | 68.27 | 71.58 | 78.51 |
| Round 2 | 65.25 | 76.11 | 62.17 | 69.40 | 65.44 | 65.48 |
| Round 3 | 69.30 | 71.54 | 67.70 | 74.21 | 61.11 | 60.49 |
| Round 4 | 70.11 | 70.54 | 67.20 | 73.62 | 64.57 | 69.52 |
| Round 5 | 78.34 | 74.56 | 62.26 | 67.01 | 65.65 | 68.04 |
| Round 6 | 79.55 | 70.23 | 67.18 | 71.53 | 63.13 | 63.84 |
| Round 7 | 69.37 | 71.74 | 73.18 | 71.81 | 67.62 | 67.56 |
| Round 8 | 71.15 | 75.28 | 72.55 | 78.25 | 64.04 | 70.89 |
| Round 9 | 79.32 | 70.14 | 69.20 | 78.21 | 63.89 | 65.65 |
| Round 10 | 81.05 | 75.03 | 76.56 | 77.44 | 62.66 | 63.97 |
| **Average** | **73.25** | **72.75** | **69.01** | **72.98** | **64.97** | **67.40** |

SVM Set 1 training and testing accuracy difference is 0.5%. Estimating this small difference, which is close to 0, it can be assumed, that the model has no tendency to overfitting or overfitting. The model is trained very well. Set 2, has a 3.97% difference and Set 3 has a 2.43% difference. Both Sets show very good accuracy, but it is not as good as in Set 1 and hence it is more underfitting pattern.

**Table 3.** Random Forest k-fold train and validation accuracy results for approach I.

| RF | SET 1 accuracy | | SET 2 accuracy | | SET 3 accuracy | |
|---|---|---|---|---|---|---|
| | Training, % | Validation, % | Training, % | Validation, % | Training, % | Validation, % |
| Round 1 | 68.05 | 72.13 | 64.64 | 71.60 | 69.37 | 71.93 |
| Round 2 | 66.87 | 76.96 | 68.16 | 64.49 | 69.29 | 69.73 |
| Round 3 | 71.05 | 71.12 | 69.86 | 68.42 | 65.65 | 68.8 |
| Round 4 | 71.65 | 70.13 | 64.44 | 75.85 | 69.34 | 72.26 |
| Round 5 | 75.86 | 74.81 | 73.53 | 70.90 | 70.01 | 71.24 |
| Round 6 | 74.41 | 70.48 | 63.14 | 65.22 | 73.31 | 62.37 |
| Round 7 | 71.66 | 71.25 | 66.93 | 66.00 | 75.54 | 74.27 |
| Round 8 | 70.31 | 75.35 | 72.25 | 75.50 | 71.95 | 74.96 |
| Round 9 | 71.53 | 70.09 | 65.54 | 65.46 | 66.53 | 74.66 |
| Round 10 | 79.71 | 75.22 | 66.99 | 70.45 | 68.19 | 66.31 |
| **Average** | **72.10** | **72.75** | **67.54** | **69.39** | **69.92** | **70.65** |

Random forest with Set 1 has difference in accuracy 0.65%, the same as in SVM Set1, difference is very small and can be concluded as a very good train model, which is neither overfitted, nor underfitted. Set 3 difference is 0.73% and can be considered as good training as well as Set1. Set 2 leads to underfitting, because validation accuracy is higher than training accuracy. In Set 2 the difference is 1.85%.

**Table 4.** KERNEL k-fold train and validation accuracy results for approach I.

| KERNEL | SET 1 accuracy | | SET 2 accuracy | | SET 3 accuracy | |
|---|---|---|---|---|---|---|
| | Training, % | Validation, % | Training, % | Validation, % | Training, % | Validation, % |
| Round 1 | 77.74 | 64.67 | 65.65 | 75.69 | 69.74 | 71.74 |
| Round 2 | 79.67 | 66.62 | 78.75 | 68.71 | 63.65 | 66.77 |
| Round 3 | 64.62 | 78.67 | 64.8 | 72.74 | 67.66 | 74.76 |
| Round 4 | 75.70 | 67.75 | 80.79 | 66.74 | 74.8 | 66.68 |
| Round 5 | 70.74 | 77.75 | 63.67 | 73.76 | 72.73 | 73.77 |
| Round 6 | 67.79 | 75.77 | 77.68 | 80.68 | 71.67 | 66.8 |
| Round 7 | 63.65 | 66.74 | 67.73 | 78.74 | 67.64 | 68.77 |
| Round 8 | 78.66 | 79.66 | 69.8 | 78.77 | 76.75 | 69.65 |
| Round 9 | 68.79 | 75.73 | 70.69 | 73.79 | 76.7 | 64.76 |
| Round 10 | 73.68 | 66.68 | 68.76 | 64.63 | 66.64 | 66.69 |
| **Average** | **72.11** | **72.00** | **70.83** | **73.42** | **70.79** | **69.04** |

Kernel Set 1 shows difference in accuracy is only 0.11%, which is the best compared with all datasets. Kernel, with Set 1 is the best trained model. Set 2 shows more underfitting, because validation accuracy is bigger 2.59% than training accuracy. Set 3 shows more overfitting, because training accuracy is bigger 1.75% than validation accuracy.

The best result with approach I was from Kernel Set 1, with overfitting 0.11%, and the biggest difference is with RF Set 2 – 3.97% underfitting.

The same 10 – fold validation check was done with approach II data. SVM, RF, Kernel models were checked, and results are presented in Table 5, Table 6 and Table 7. 90% of data was tested and 10% left for validation.

**Table 5.** Support Vector Machine k-fold train and validation accuracy results for approach II.

| SVM | SET 1 accuracy | | SET 2 accuracy | | SET 3 accuracy | |
|---|---|---|---|---|---|---|
| | Training, % | Validation, % | Training, % | Validation, % | Training, % | Validation, % |
| Round 1 | 70.44 | 77.23 | 71.17 | 69.57 | 72.26 | 79.16 |
| Round 2 | 71.34 | 77.12 | 63.18 | 75.60 | 66.29 | 69.28 |
| Round 3 | 69.23 | 74.76 | 69.71 | 74.21 | 60.21 | 75.27 |
| Round 4 | 69.26 | 71.45 | 69.22 | 73.62 | 66.37 | 78.13 |
| Round 5 | 79.32 | 75.66 | 64.28 | 69.12 | 66.48 | 78.27 |
| Round 6 | 80.33 | 77.23 | 68.29 | 75.63 | 65.23 | 77.49 |
| Round 7 | 65.41 | 75.87 | 74.24 | 70.64 | 68.55 | 79.11 |
| Round 8 | 70.25 | 78.19 | 74.65 | 79.55 | 65.59 | 78.58 |
| Round 9 | 80.42 | 71.28 | 70.30 | 79.31 | 65.57 | 70.56 |
| Round 10 | 85.10 | 78.43 | 77.66 | 79.15 | 64.56 | 75.55 |
| **Average** | **74.11** | **75.22** | **70.27** | **74.64** | **66.11** | **76.14** |

In all sets train accuracy is smaller then validation accuracy. The biggest difference is in Set3 – 10.13% (66.11% and 76.11%). This shows that in all sets the model has a pattern to underfitting. However, 10.13% is not a big difference and model can be tested.

**Table 6.** Random Forest k-fold train and validation accuracy results for approach II.

| RF | SET 1 accuracy | | SET 2 accuracy | | SET 3 accuracy | |
|---|---|---|---|---|---|---|
| | Training, % | Validation, % | Training, % | Validation, % | Training, % | Validation, % |
| Round 1 | 88.43 | 87.10 | 77.44 | 84.45 | 83.51 | 85.71 |
| Round 2 | 85.34 | 78.33 | 80.62 | 85.71 | 79.18 | 84.63 |
| Round 3 | 88.72 | 75.53 | 75.75 | 80.12 | 76.52 | 82.81 |
| Round 4 | 85.87 | 75.14 | 78.91 | 80.92 | 87.11 | 85.54 |
| Round 5 | 91.70 | 86.32 | 78.34 | 85.76 | 75.17 | 82.01 |
| Round 6 | 88.61 | 82.54 | 84.55 | 90.69 | 72.13 | 85.66 |
| Round 7 | 95.31 | 88.56 | 75.09 | 80.12 | 74.23 | 80.75 |
| Round 8 | 90.33 | 86.15 | 81.12 | 85.87 | 79.64 | 85.32 |
| Round 9 | 82.34 | 70.16 | 74.19 | 82.77 | 78.22 | 84.53 |
| Round 10 | 80.00 | 71.99 | 84.02 | 90.22 | 85.04 | 92.33 |
| **Average** | **87.67** | **80.18** | **79.00** | **84.66** | **79.08** | **84.93** |

Random Forest model with Set 1 has a higher training accuracy than validation (7.49% bigger). The difference is not sufficient, but the model is more overfitted. Set 2 and Set 3 have very similar accuracy results. Difference between training and validation are 5.66% and 5.85%. Set 2 and set 3 are more underfitted models.

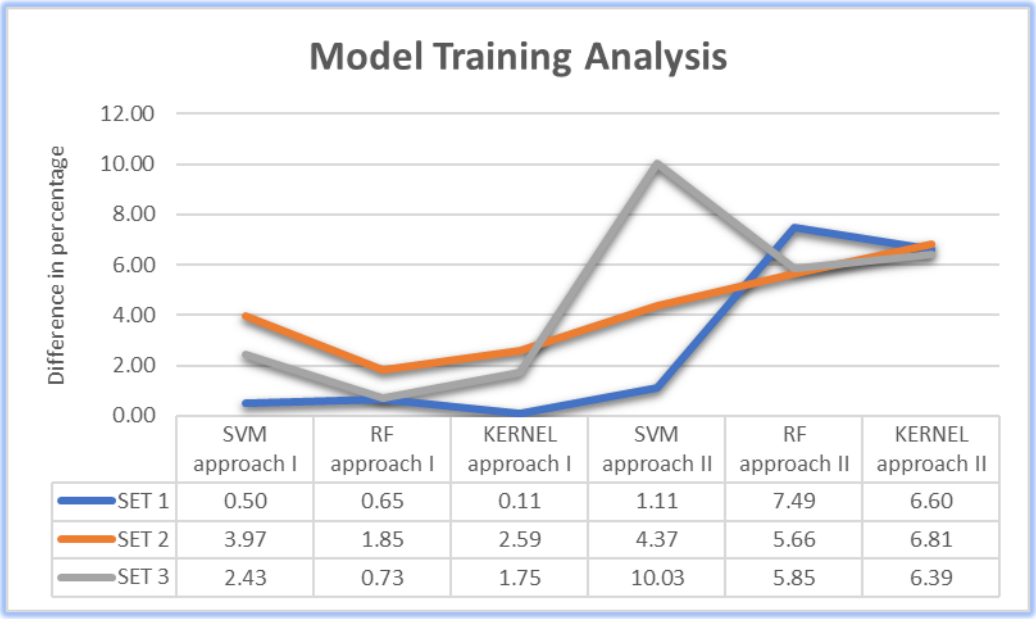
**Table 7.** Kernel k-fold train and validation accuracy results for approach II.

| KERNEL | SET 1 accuracy | | SET 2 accuracy | | SET 3 accuracy | |
|---|---|---|---|---|---|---|
| | Training, % | Validation, % | Training, % | Validation, % | Training, % | Validation, % |
| Round 1 | 88.32 | 83.64 | 82.62 | 75.13 | 92.44 | 84.51 |
| Round 2 | 88.11 | 83.28 | 92.78 | 88.43 | 87.07 | 83.28 |
| Round 3 | 89.67 | 85.18 | 92.83 | 84.74 | 85.13 | 79.23 |
| Round 4 | 88.13 | 81.08 | 82.60 | 75.16 | 92.87 | 85.94 |
| Round 5 | 89.20 | 81.15 | 92.34 | 83.29 | 84.22 | 74.39 |
| Round 6 | 89.12 | 82.04 | 92.79 | 89.19 | 92.12 | 86.93 |
| Round 7 | 90.43 | 81.46 | 95.12 | 88.02 | 85.04 | 71.18 |
| Round 8 | 85.13 | 73.47 | 85.43 | 73.88 | 79.12 | 72.09 |
| Round 9 | 85.61 | 82.34 | 92.06 | 87.32 | 90.07 | 82.44 |
| Round 10 | 90.31 | 84.34 | 93.22 | 88.57 | 79.33 | 83.53 |
| **Average** | **88.40** | **81.80** | **90.18** | **83.37** | **86.74** | **80.35** |

Kernel has opposite results than SVM. All three sets have intention to overfitting. This is indicated by higher accuracy in training then in validation. The differences are 6.60% for Set 1, 6.81% for Set 2 and 6.39% for Set 3. The results are satisfying, and the model can be run for testing, with knowing, that Set 2 has highest intention to overfitting.

The best result with approach II was from SVM Set 1, with underfitting 1.11%, and the biggest difference is with SVM Set 3 – 10.03% underfitting.

Model training analysis is represented in Figure 29. Results from approach I and approach II data are compared. It can be seen, that SVM model with data from LiDAR and orthophoto with training Set 3 run the biggest risk (10.03%) has. Kernel model with only LiDAR data and Set 1, show only 0.11% difference between train and validation data, which can be considered as a very good result. Classification with this model should be without overfitting and without underfitting.




| | SVM approach I | RF approach I | KERNEL approach I | SVM approach II | RF approach II | KERNEL approach II |
|---|---|---|---|---|---|---|
| SET 1 | 0.50 | 0.65 | 0.11 | 1.11 | 7.49 | 6.60 |
| SET 2 | 3.97 | 1.85 | 2.59 | 4.37 | 5.66 | 6.81 |
| SET 3 | 2.43 | 0.73 | 1.75 | 10.03 | 5.85 | 6.39 |

**Figure 29.** Model training analysis

All k-folds validation results show some difference between training and validation accuracy, but all values are small enough to be able to continue classification with trained models.

- Testing

After training and validation gave satisfying results, test data was run to see, how the model can handle never seen data and what to expect. Table 4 and Table 5

presents accuracy and kappa index in test dataset for data only with LiDAR features (Table 5) and data with LiDAR and orthophoto features together (Table 6).

**Table 8.** Test dataset for approach I only with LiDAR data

| TEST LiDAR DATA | SET 1 accuracy % | SET 2 accuracy % | SET 3 accuracy % |
|---|---|---|---|
| SVM | 93.14 | 89.95 | 84.14 |
| RF | 90.41 | 89.54 | 88.87 |
| KERNEL | 94.58 | 91.25 | 90.55 |

The best accuracy score is achieved with Kernel classification model, Set 1 (94.58%). SVM classification was less accurate with Set 3 (84.14%). The difference between the best and the worst classification is 10.44%. Classification test result proves, that the model is well trained, and all classification algorithms can be used for new area classification.

**Table 9.** Test dataset for approach II with LiDAR and Orthophoto data

| TEST LiDAR DATA + ORTHOPHOTO DATA | SET 1 accuracy % | SET 2 accuracy % | SET 3 accuracy % |
|---|---|---|---|
| SVM | 91.14 | 90.95 | 90.14 |
| RF | 94.51 | 93.54 | 93.01 |
| KERNEL | 94.88 | 91.25 | 89.75 |

The best accuracy score is with Kernel classification model, Set 1 (94.88%). SVM classification was less accurate with Set 3 (90.14%). The difference between the best and the worst classification is 4.74%. Approach II dataset was more complex and with four additional features. This is the reason for better accuracy. We can assume, that combination with LiDAR data and orthophoto features should lead to

more accurate classification. Classification test results prove, that the model is well trained, and that all classification algorithms can be used for new area classification.

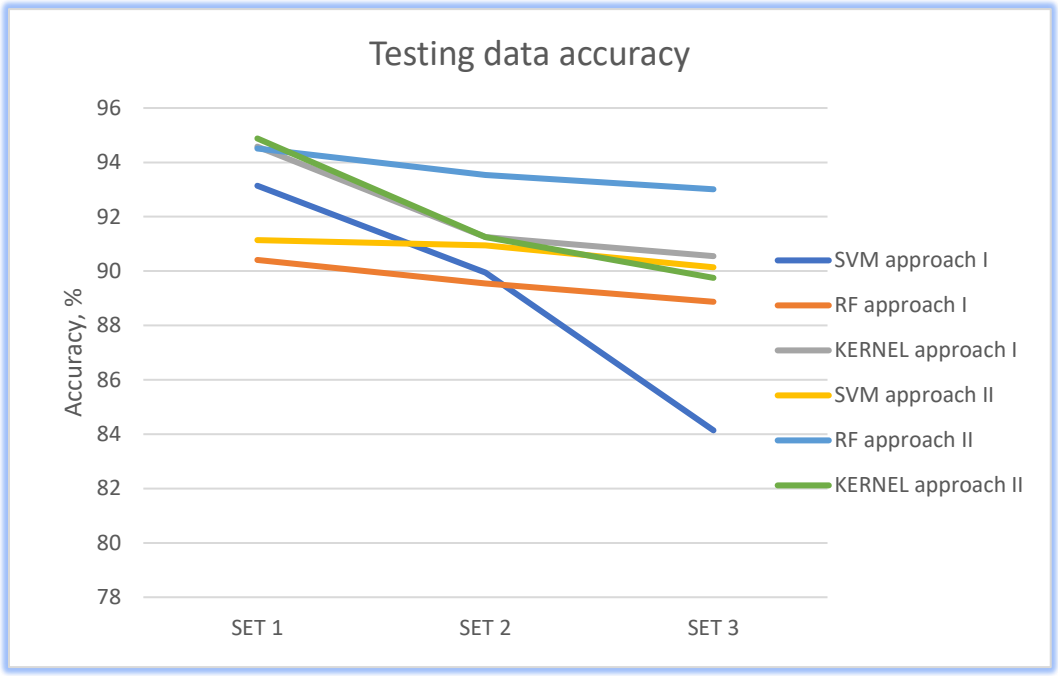



**Figure 30.** Testing data accuracy

In Figure 30 is presenting test data accuracy changes during different sets and approaches. SVM approach I drops down with every new set. The difference from the first and the last set is 9%. In addition, Kernel approach II shows a big changes from Set 1 to Set 2. It decreases 3.63%. At the same time, Kernel Set 1 has the highest accuracy 94.88%. The lowest accuracy is SVM approach I in Set 3. The general pattern is that accuracy decreases from Set 1 to Set 3.

- Classification in the chosen area of Copenhagen

When the model is formed and confirmed as trustful classification for desired area can be started, which is shown in Figure 22, can be started. The real area is 1sq.km

point cloud with density 4,5 point in 1sq.m. Figure 31 and Figure 32 present classification results.

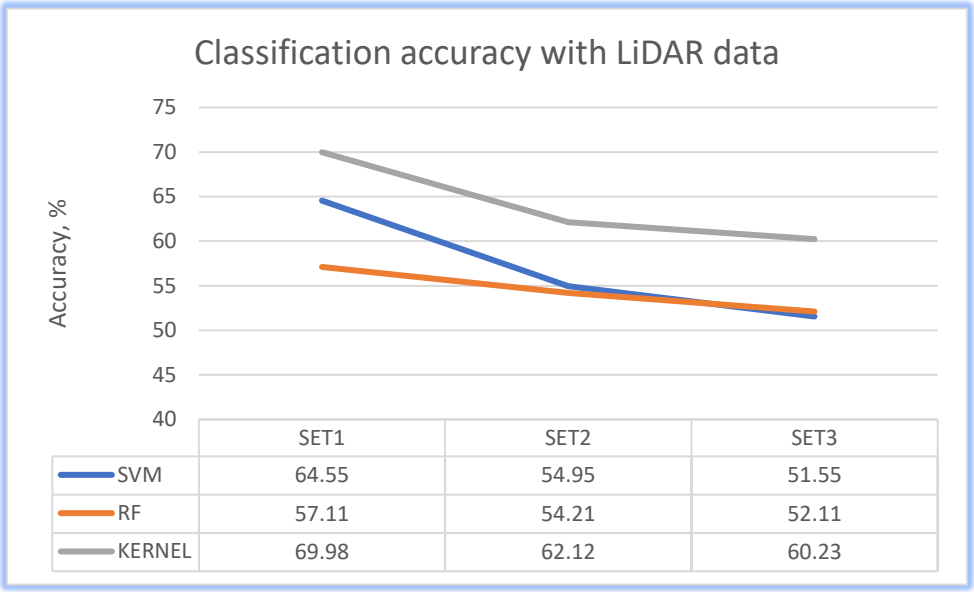



**Figure 31.** Accuracy only with LiDAR data

Figure 31 shows accuracy of the classification with LiDAR data features with SVM, RF and Kernel models. The best results in all sets were with Kernel 69.98%, 62.12% and 60.23%. Random forests were always less accurate than SVM or Kernel, but during the Set changes it was the most stable. It decreases only 5%. RF is recommended for classification, when training dataset cannot be prepared with sufficient number of points or when accuracy comparison is not possible.

Figure 32 shows accuracy of the classification with LiDAR data features combined with orthophoto features with SVM, RF and Kernel models. Classification is done with 7 features: intensity, number of returns, return number, red, green, blue colors and NIR.

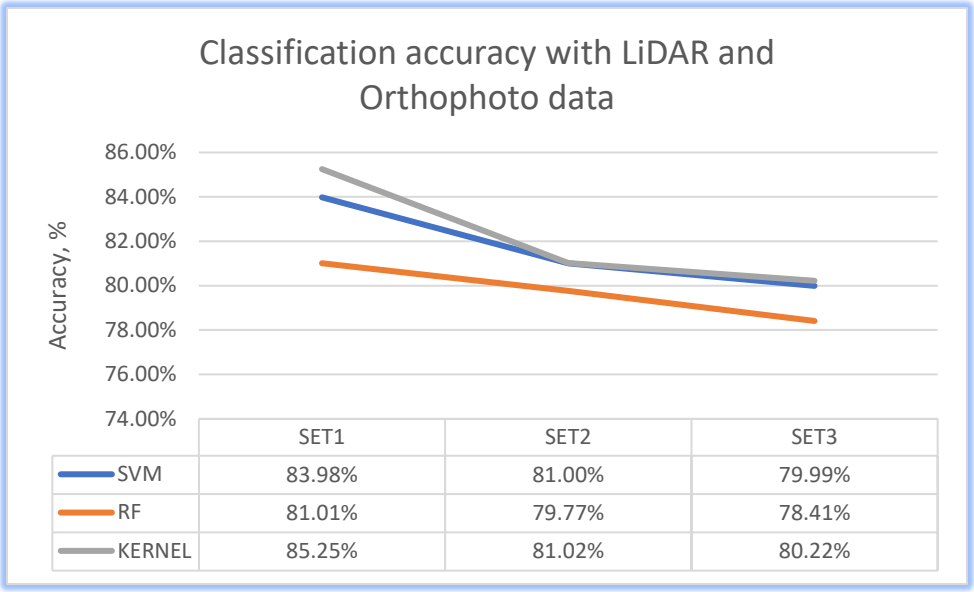



**Figure 32.** Accuracy with LiDAR and orthophoto data

The highest accuracy was achieved by Kernel classification model with Set 1 (85.25%). In Set 2 and Set 3 Kernel and SVM performed almost the same. Kernel was 0.02% and 0.23% better than SVM. Random forest shows, that it is less affected by trained model dataset size. The change during the sets was 2.6%, when at the same time Kernel, which performed the best accuracy, decreased 5.03% and SVM decreased 3.99%. The general trend is the same as with classification only with LiDAR data, decreasing data size for model training, accuracy of final classification decreases.

Figure 33 shows combined accuracy results. All accuracies, which were achieved by classifying with approach I features show significantly smaller accuracy, than those, which were achieved with approach II features. This concludes, that orthophotos features such as colors: red, green, blue and NIR values are very important for high vegetation classification.

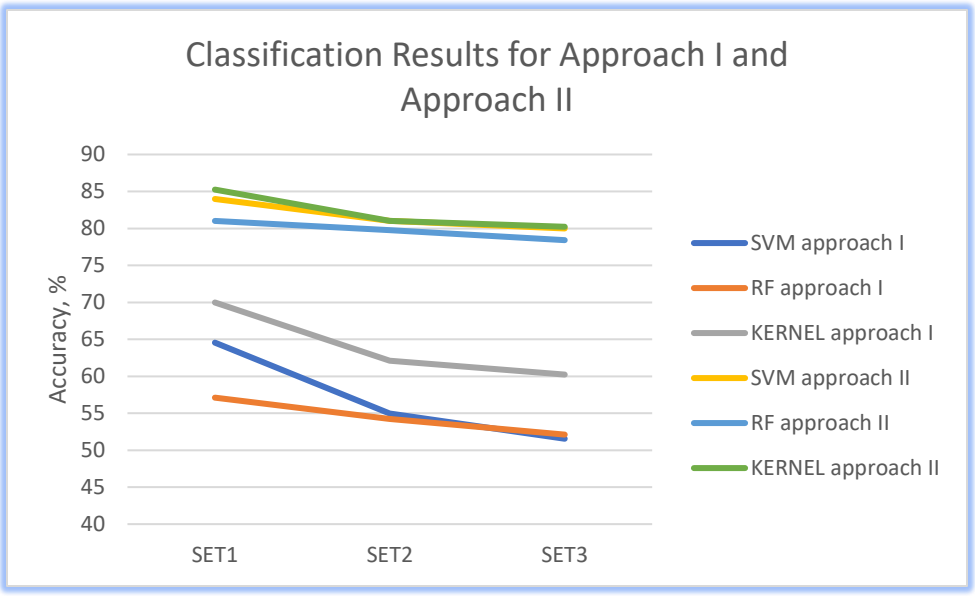



**Figure 33.** Accuracy with LiDAR and orthophoto data

The highest accuracy was achieved with Kernel approach II, Set 1 and the lowest accuracy was achieved with RF approach II, Set 3. With all methods and models, the highest accuracy was achieved by Set 1 and the lowest with Set 3. This results proves, that training Set size has a significant influence in further classification results. The bigger the training set, the better the accuracy of classification. Despite this fact, random forest was the least affected model for training data size (differences is 5% and 2.6%).

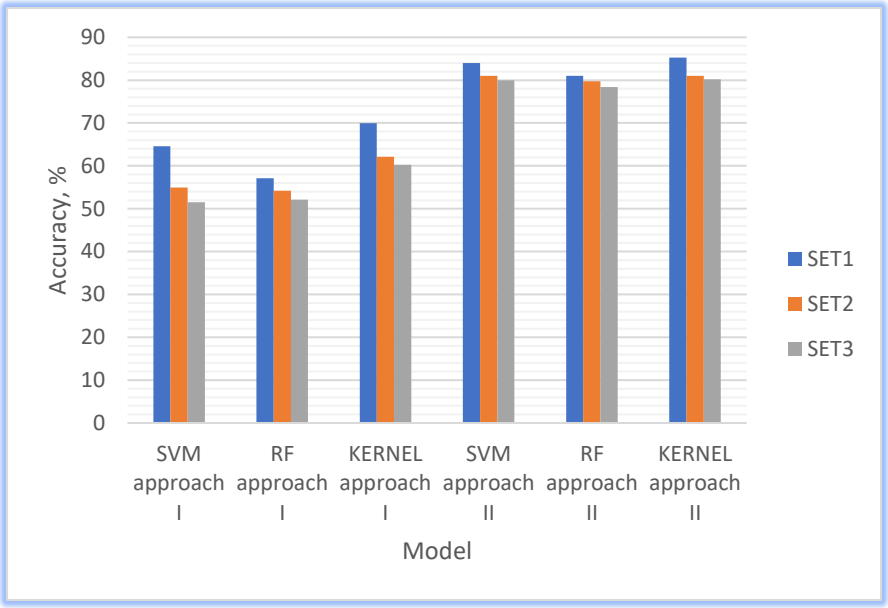



**Figure 34.** Data Set analysis

Analyzing datasets Set 1, Set 2 and Set 3, we can see that RF classification is less sensitive to data quantity than the other types of classifications, especially data with LiDAR and orthophoto (Figure 34). The difference between Set 1 and Set 3 in accuracy is only 2.6%. Data set size has the biggest influence on SVM with features only from LiDAR data. The difference from Set1 and Set3 is 13%.

Figure 35, Figure 36 and Figure 37 represents how classification accuracy changes when additional features from orthophoto, besides LiDAR, are added.

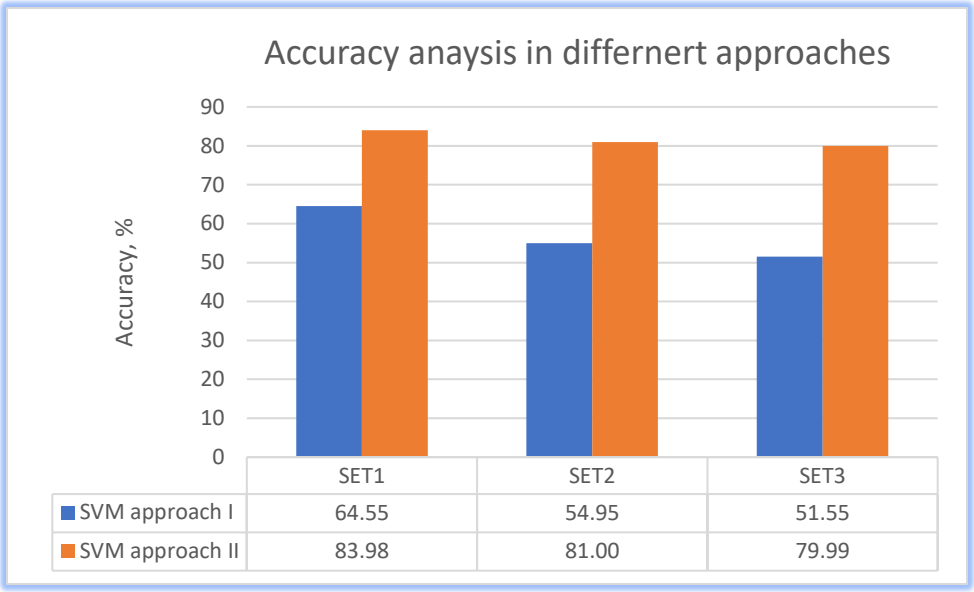



**Figure 35.** Support Vector Machine classification comparison for different Sets in approach I and approach II

With SVM classification model accuracy was increased in all 3 sets, with approach II, where LiDAR features are combined with orthophoto features. The biggest change is seen in Set 3, where training dataset is the smallest. The accuracy was increased by 28.44% (from 51.55% to 79.99%) with additional features from orthophoto. The smallest change appears in Set 1, where training dataset is the biggest. It increases by 19.43% (from 64.55% to 83.98%).

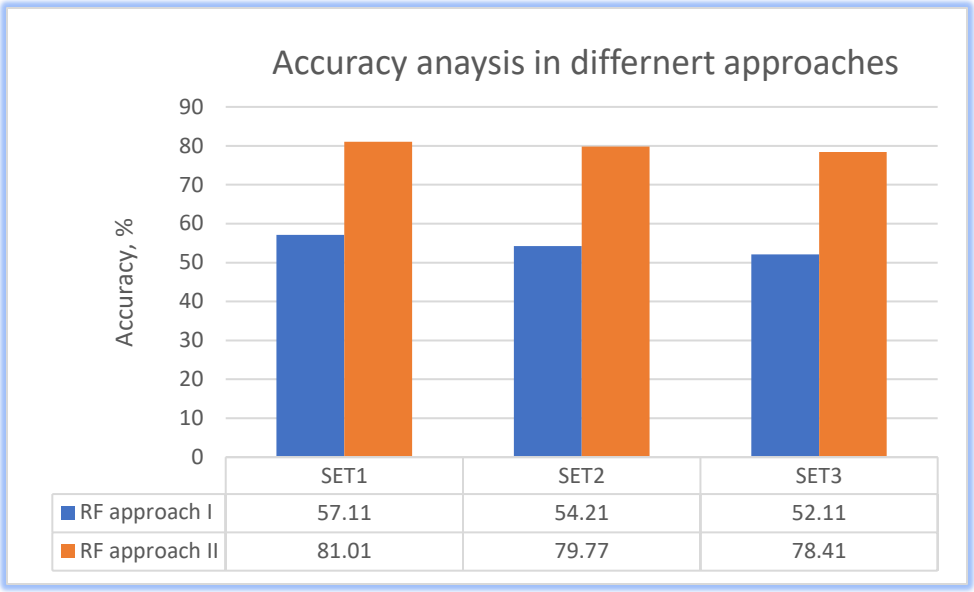



**Figure 36.** Random forest classification comparison for different Sets in approach I and approach II

With RF classification model accuracy was increased in all 3 sets, with approach II, where LiDAR features are combined with orthophoto features. The biggest change is seen in Set 3, where training dataset is the smallest. The accuracy was increased by 26.30% (from 52.11% to 78.41%) with additional features from orthophoto. The smallest change appears in Set 1, where training dataset is the biggest. It increases by 23.90% (from 57.11% to 81.01%).

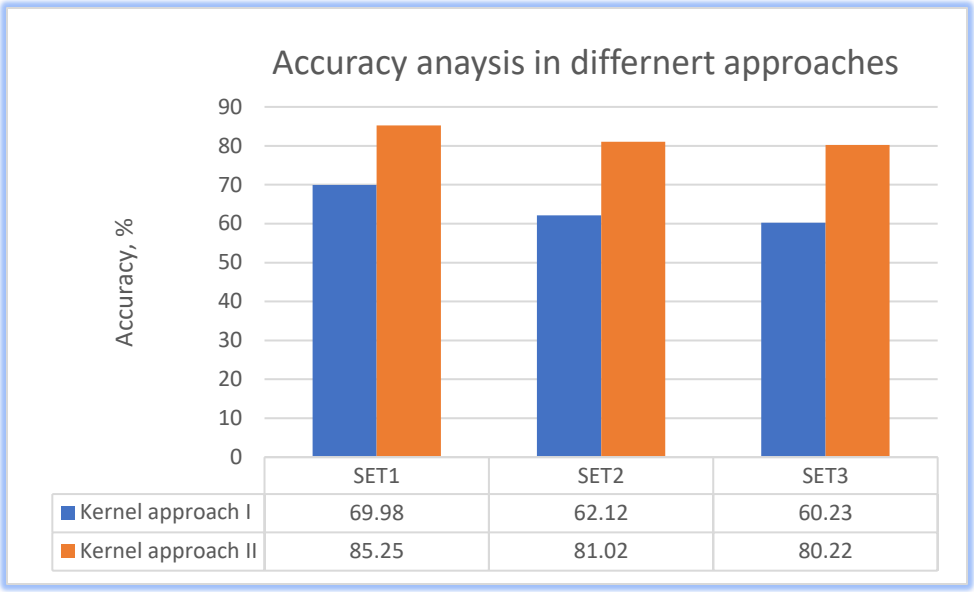



**Figure 37.** Kernel classification comparison for different Sets in approach I and approach II

With Kernel classification model accuracy was increased in all 3 Sets, with approach II, where LiDAR features are combined with orthophoto features. The biggest change is seen in Set 3, where training dataset is the smallest. The accuracy was increased by 19.99% (from 60.23% to 80.22%) with additional features from orthophoto. The smallest change appears in Set 1, where training dataset is the biggest. It increases by 15.27% (from 69.98% to 85.25%).

All model analysis shows the same pattern, that orthophoto, with additional features increases accuracy and the more data was used for model training, the more precise classification we can obtain. The best classification accuracy is with Kernel classification, approach II, Set 1. Set 1 training model was done with 133 000 points. To ensure, that there is no overfitting or underfitting, 10-folds validation method was used and 90% of the data was trained. Approach II includes LiDAR classification features, which comes from scanner: intensity, number of returns and return number and additional to that features from orthophoto was added: red, green and blue bands and NIR. Kernel model is one of the most popular for remote sensing classification (Pal and Mather 2005). RF classification was not with the

highest accuracy, but it keeps stability, when training data set size is changing. RF model is a good option, when training size cannot be large.

# 4 Conclusion

This research discusses the classification of high vegetation in an urban environment. The data used in this study was LiDAR and orthophoto, which was obtained from Danish geodata distribution portal. Three ML-based classification techniques (Support Vector Machine, Random Forests, KERNEL) were used to reclassify high vegetation in the point cloud dataset. The three classifiers were trained with different training sets' sizes: Set 1 – 133,000 points, Set 2 – 66,500 points and Set 3 – 33,250 points (corresponding to 30%, 15% and 7% of the entire training dataset, respectively). Two different approaches were undertaken: For approach I, classification was done only based on LiDAR point cloud dataset and only by looking at LiDAR driven parameters e.g., intensity, number of returns and return number.

Training was successfully done with the 3 training sets. As per validation, 10-fold validation technique was used and the best performance was observed from Kernel classifier and training Set 1 with an accuracy of 69.98%.

For approach II, classification was done with LiDAR point cloud dataset and orthophoto. The parameters for classification was taken from LiDAR: intensity, number of returns and return number, from orthophoto: red, green, blue and infrared bands.

All three different training sets' sizes were trained and 10-folds validation technique was used in order to increase accuracy and to avoid overfitting and underfitting. The best classification result was achieved with Kernel classifier and training Set 1 with an accuracy of 85.25%.

It is important to minimize the risk of model underfitting or overfitting. This was done by 10-folds validation method, when 90% of the data was trained and validated with the remaining 10%. All 10 times results show good trained model.

All classification models have their own pros and cons. SVM advantage is support vectors, which assist to precise separation, but since it is linear model, for a

complex data it may caused underfitting. RF is known as very universal model. It works well with classification, but it might be hard to achieve higher accuracy. Kernel works with support vectors and non-linear data. The disadvantage of this model is complex math, which might be time consuming for big datasets.

As per further direction, classification model can be trained better. Potential solution is to enlarge point set size because it leads to higher accuracy. Moreover, if LiDAR scanning and orthophoto will be taken at the same date, accuracy should increase as well, because the possibility of changes in the surface will be very small.

This model was trained to identify high vegetation in urban areas (part of Copenhagen). It can be used to classify high vegetation in other cities as well as to test how widely the proposed approach can be applied. This was not done in this thesis, because of time limitation and LiDAR data size. Model training and classification is time consuming. More data requires more computer resources to be processed. So, for relevant dataset size, a relevant machine is required.

# Literature


- Alexander, C., Tansey, K., Kaduk, J., Holland, D., & Tate, N.J. (2010). Backscatter coefficient as an attribute for the classification of full-waveform airborne laser scanning data in urban areas. ISPRS Journal of Photogrammetry and Remote Sensing, 65, 423–432.

- Aggiwal, R., (2018 February 28). Introduction to random forest. Retrieved from www.dimensionless.in

- Anderson, J.R., Hardy, E.E., Roach, J.T., & Witmer, R.E. (1976). A land use and land cover classification system for use with remote sensor data. US Geological Survey Professional Paper 964. Washington, D.C.: USGS.

- Bartels, M., & Wei, H. (2006). Rule-based improvement of maximum likelihood classified LiDAR data fused with coregistered bands. Proceedings of Annual Conference of the Remote Sensing and Photogrammetry Society (pp. 1–9) (Cambridge, UK).

- Bartholomé, E., & Belward, A. (2005). GLC2000: A new approach to global land cover mapping from Earth observation data. International Journal of Remote Sensing, 26, 1959–1977.

- Bothe, K., Hansen, H.,K., Winther, L. (2018). Spatial restructuring and uneven intra-urban employment growth in metro- and non-metro-served areas in Copenhagen. Journal of transport geography, 70, 21–30

- Benediktsson, J., Chanussot, J., & Fauvel, M. (2007). Multiple classifier systems in remote sensing: From basics to recent developments. In M. Haindl, J. Kittler, & F. Roli (Eds.), Multiple classifier systems, volume 4472 of lecture notes in computer science (pp. 501–512). Berlin/Heidelberg: Springer

- Bravo, H.C., (2018 May 1). Model selection. Retrieved from www.hcbravo.org

- Brennan, R., & Webster, T. (2006). Object-oriented land cover classification of LiDAR derived surfaces. Canadian Journal of Remote Sensing, 32, 162–172.

- Bronstein, A., (2018 May 17). Train/Test Split and Cross Validation in Python. Retrieved from www.towardsdatascience.com

- Buján, S., González-Ferreiro, E., Reyes-Bueno, F., Barreiro-Fernández, L., Crecente, R., & Miranda, D. (2012). Land use classification from LiDAR data and ortho-images in a rural area. The Photogrammetric Record, 27, 401–422

- Burton, D., Dunlap, D.B., Wood, L.J., & Flaig, P.P. (2011). LiDAR intensity as a remote sensor of rock properties. Journal of Sedimentary Research, 81, 339–347.

- Charaniya, A., Manduchi, R., & Lodha, S. (2004). Supervised parametric classification of aerial LiDAR data. Proceedings of the IEEE 2004 Conference on Computer Vision and Pattern Recognition Workshop, vol. 3. (pp. 1–8) (Baltimore).

- Chasmer, L., Hopkinson, C., Smith, B., & Treitz, P. (2006). Examining the influence of changing laser pulse repetition frequencies on conifer forest canopy returns. Photogrammetric Engineering & Remote Sensing, 72, 1359–1367.

- Chauve, A., Mallet, C., Bretar, F., Durrieu, S., Deseilligny, M.P., & Puech, W. (2007). Processing full-waveform LiDAR data: Modelling raw signals. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 36, 102–107.

- Chehata, N., Guo, L., & Mallet, C. (2009). Airborne LiDAR feature selection for urban classification using random forests. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 39, 207–212.

- Chen, Y., Su, W., Li, J., & Sun, Z. (2009). Hierarchical object oriented classification using very high resolution imagery and LiDAR data over urban areas. Advances in Space Research, 43, 1101–1110

- Deems, J.S., Painter, T.H., & Finnegan, D.C. (2013). LiDAR measurement of snow depth: A review. Journal of Glaciology, 59, 467–479

- Friedl, M., McIver, D., Hodges, J., Zhang, X., Muchoney, D., Strahler, A., et al. (2002). Global land cover mapping from MODIS: Algorithms and early results. Remote Sensing of Environment, 83, 287–302

- García-Gutiérrez, J., Martínez-Álvarez, F., Troncoso, A., & Riquelme, J. C. (2015). A comparison of machine learning regression techniques for LiDAR-derived estimation of forest variables. Neurocomputing, 167, 24-31.

- Garcia-Gutierrez, J., Gonçalves-Seco, L., & Riquelme-Santos, J.C. (2011). Automatic environmental quality assessment for mixed-land zones using LiDAR and intelligent techniques. Expert Systems with Applications, 38, 6805–6813.

- Garroway, K., Hopkinson, C., & Jamieson, R. (2011). Surface moisture and vegetation influences on LiDAR intensity data in an agricultural watershed. Canadian Journal of Remote Sensing, 37, 275–284

- Germaine, K.A., & Hung, M.C. (2011). Delineation of impervious surface from multispectral imagery and LiDAR incorporating knowledge based expert system rules. Photogrammetric Engineering & Remote Sensing, 77, 75–85.

- Gonzalez-Aguilera, D., Crespo-Matellan, E., Hernandez-Lopez, D., & Rodrguez-Gonzálvez, P. (2013). Automated urban analysis based on LiDAR-derived building models. IEEE Transactions on Geoscience and Remote Sensing, 51, 1844–1851.

- Goncalves-Seco, D., Miranda, R., Crecente, J., Digital terrain model generation using airborne LIDAR in forested area of Galicia, Spain, Lisbon, Portugal, in: Proceedings of 7th International Symposium on Spatial Accuracy Assessment in Natural Resources and Environmental Sciences, 2006, pp. 169– 180.

- Gregorio, A., & Jansen, L. (2000). Land cover classification system, classification concepts and user manual.

- Guan, H., Ji, Z., Zhong, L., Li, J., & Ren, Q. (2013). Partially supervised hierarchical classifi- cation for urban features from LiDAR data with aerial imagery. International Journal of Remote Sensing, 34, 190–210

- Habib, A., Kersting, A., Shaker, A., & Yan, W.Y. (2011). Geometric calibration and radiometric correction of LiDAR data and their impact on the quality of derived products. Sensors, 11, 9069–9097

- Hartfield, K.A., Landau, K.I., & Van Leeuwen, W.J. (2011). Fusion of high resolution aerial multispectral and LiDAR data: Land cover in the context of urban mosquito habitat. Remote Sensing, 3, 2364–2383



- Hartfield, K.A., Landau, K.I., & Van Leeuwen, W.J. (2011). Fusion of high resolution aerial multispectral and LiDAR data: Land cover in the context of urban mosquito habitat. Remote Sensing, 3, 2364–2383

- Hecht, R., Meinel, G., & Buchroithner, M.F. (2008). Estimation of urban green volume based on single-pulse LiDAR data. IEEE Transactions on Geoscience and Remote Sensing, 46, 3832–3840.

- Helbich, M., Jochem, A., Mücke, W., & Höfle, B. (2013). Boosting the predictive accuracy of urban hedonic house price models through airborne laser scanning. Computers, Environment and Urban Systems, 39, 81–92.

- Hodgson, M.E., Jensen, J.R., Tullis, J.A., Riordan, K.D., & Archer, C.M. (2003). Synergistic use of LiDAR and color aerial photography for mapping urban parcel imperviousness. Photogrammetric Engineering & Remote Sensing, 69, 973–980.

- Hopkinson, C. (2007). The influence of flying altitude, beam divergence, and pulse repetition frequency on laser pulse return intensity and canopy frequency distribution. Canadian Journal of Remote Sensing, 33, 312–324

- Huang, M., Shyue, S., Lee, L., & Kao, C. (2008). A knowledge-based approach to urban feature classification using aerial imagery with LiDAR data. Photogrammetric Engineering & Remote Sensing, 74, 1473–1485.

- Huang, X., Zhang, L., & Gong, W. (2011). Information fusion of aerial images and LiDAR data in urban areas: Vector-stacking, re-classification and post-processing approaches. International Journal of Remote Sensing, 32, 69–84

- Huang, Y., Yu, B., Zhou, J., Hu, C., Tan, W., Hu, Z., et al. (2013). Toward automatic estimation of urban green volume using airborne LiDAR data and high resolution remote sensing images. Frontiers of Earth Science, 7, 43–54

- Im, J., Jensen, J.R., & Hodgson, M.E. (2008). Object-based land cover classification using high-posting-density LiDAR data. GIScience & Remote Sensing, 45, 209–228

- J. Osborne, E. Waters, Four assumptions of multiple regression that researchers should always test, Pract. Assess. Res. Eval. 8 (2) (2002).

- J.D. Muss, D.J. Mladenoff, P.A. Townsend, A pseudo-waveform technique to assess forest structure using discrete LiDAR data, Remote Sens. Environ. 115 (3) (2010) 824–835

- Jaboyedoff, M., Oppikofer, T., Abellán, A., Derron, M.H., Loye, A., Metzger, R., et al. (2012). Use of LiDAR in landslide investigations: A review. Natural Hazards, 61, 5–28.

- Johnson, B.D., & Singh, J. (2003). Building the national geobase for Canada. Photogrammetric Engineering & Remote Sensing, 69, 1169–1173.

- Kaasalainen, S., Niittymaki, H., Krooks, A., Koch, K., Kaartinen, H., Vain, A., et al. (2010a). Effect of target moisture on laser scanner intensity. IEEE Transactions on Geoscience and Remote Sensing, 48, 2128–2136

- Khorram S., Koch F.H., van der Wiele C.F., Nelson S.A.C. (2012) Introduction in: Remote Sensing. Springer Briefs in Space Development. Springer, Boston, MA

- Kim, Y., & Kim, Y. (2014). Improved classification accuracy based on the output-level fusion of high-resolution satellite images and airborne LiDAR data in urban area. IEEE Geoscience and Remote Sensing Letters, 11, 636–640.

- Lang, M.W., & McCarty, G.W. (2009). LiDAR intensity for improved detection of inundation below the forest canopy. Wetlands, 29, 1166–1178

- Lever, J., Krzywinski, M., Altman, N., (2017) Points of significance: Principal component analysis. Nature Methods 14:641–642. Retrieved from www.nature.com

- Lim, K., Treitz, P., Wulder, M., St-Onge, B., & Flood, M. (2003). LiDAR remote sensing of forest structure. Progress in Physical Geography, 27, 88–106

- Lohani, B., & Kumar, R. (2008). A model for predicting GPS-GDOP and its probability using LiDAR data and ultra-rapid product. Journal of Applied Geodesy, 2, 213–222.

- Loveland, T., Reed, B., Brown, J., Ohlen, D., Zhu, Z., Yang, L., et al. (2000). Development of a global land cover characteristics database and IGBP DISCover from 1 km AVHRR data. International Journal of Remote Sensing, 21, 1303–1330.

- Lu, Z., Im, J., & Quackenbush, L. (2011). A volumetric approach to population estimation using LiDAR remote sensing. Photogrammetric Engineering & Remote Sensing, 77, 1145–1156.

- MacFaden, S.W., ONeil-Dunne, J.P., Royar, A.R., Lu, J.W., & Rundle, A.G. (2012). Highresolution tree canopy mapping for New York City using LiDAR and object-based image analysis. Journal of Applied Remote Sensing, 6 (6 063567-1-063567-23).

- Mallet, C., & Bretar, F. (2009). Full-waveform topographic LiDAR: State-of-the-art. ISPRS Journal of Photogrammetry and Remote Sensing, 64, 1–16

- Mallet, C., Soergel, U., & Bretar, F. (2008). Analysis of full-waveform LiDAR data for classification of urban areas. International Archives of the Photogrammetry, Remote Sensing and Spatial, Information Sciences, 37, 85–92.

- Mazzarini, F., Pareschi, M.T., Favalli, M., Isola, I., Tarquini, S., & Boschi, E. (2007). Lava flow identification and aging by means of LiDAR intensity: Mount Etna case. Journal of Geophysical Research, Solid Earth, 112

- Morsdorf, F., Frey, O., Meier, E., Itten, K.I., & Allgöwer, B. (2008). Assessment of the influence of flying altitude and scan angle on biophysical vegetation products derived from airborne laser scanning. International Journal of Remote Sensing, 29, 1387–1406.

- National Research Council (2005). Radiative forcing of climate change: Expanding the concept and addressing uncertainties. Washington, DC, USA: The National Academies Press.

- Neuenschwander, A.L., Magruder, L.A., & Tyler, M. (2009). Landcover classification of small-footprint, full-waveform LiDAR data. Journal of Applied Remote Sensing, 3 (033544-033544)

- Niemeyer, J., Rottensteiner, F., & Soergel, U. (2012). Conditional random fields for LiDAR point cloud classification in complex urban areas. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 1, 263–268.

- Priestnall, G., Jaafar, J., & Duncan, A. (2000). Extracting urban features from LiDAR digital surface models. Computers, Environment and Urban Systems, 24, 65–78

- Quackenbush, L.J., Im, I., & Zuo, Y. (2013). Road extraction: A review of LiDAR-focused studies. Remote Sensing of Natural Resources, 155–169.

- Samadzadegan, F., Bigdeli, B., & Ramzi, P. (2010). A multiple classifier system for classifi- cation of LiDAR remote sensing data using multi-class SVM. Multiple classifier systems (pp. 254–263). Springer.

- Sasaki, T., Imanishi, J., Ioki, K., Morimoto, Y., & Kitada, K. (2012). Object-based classification of land cover and tree species by integrating airborne LiDAR and high spatial resolution imagery data. Landscape and Ecological Engineering, 8, 157–171.

- Singh, K.K., Vogler, J.B., Shoemaker, D.A., & Meentemeyer, R.K. (2012). LiDAR–Landsat data fusion for large-area assessment of urban land cover: Balancing spatial resolution, data volume and mapping accuracy. ISPRS Journal of Photogrammetry and Remote Sensing, 74, 110–121.

- Sithole, G., & Vosselman, G. (2004). Experimental comparison of filter algorithms for bare-Earth extraction from airborne laser scanning point clouds. ISPRS Journal of Photogrammetry and Remote Sensing, 59, 85–101

- Song, J.H., Han, S.H., Yu, K.Y., & Kim, Y.I. (2002). Assessing the possibility of land-cover classification using LiDAR intensity data. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 34, 259–262

- Stal, C., Tack, F., De Maeyer, P., De Wulf, A., & Goossens, R. (2013). Airborne photogrammetry and LiDAR for DSM extraction and 3D change detection over an urban area — A comparative study. International Journal of Remote Sensing, 34, 1087–1110.

- T.R. Tooke, N.C. Coops, J. Webster, Predicting building ages from LiDAR data with random forests for building energy modeling, Energy Build. 68 (Part A) (2014) 603–610

- Teo, T.A., & Shih, T.Y. (2013). LiDAR-based change detection and change-type determination in urban areas. International Journal of Remote Sensing, 34, 968–981.

- Tooke, T. R., Coops, N. C., & Webster, J. (2014). Predicting building ages from LiDAR data with random forests for building energy modeling. Energy and Buildings, 68, 603-610.

- Tucker, C., Grant, D., & Dykstra, J. (2004). NASA's global orthorectified Landsat data set. Photogrammetric Engineering & Remote Sensing, 70, 313–322.

- Vaughn, N.R., Moskal, L.M., & Turnblom, E.C. (2011). Fourier transformation of waveform LiDAR for species recognition. Remote Sensing Letters, 2, 347–356.

- W.L. Lu, K.P. Murphy, J.J. Little, A. Sheffer, H. Fu, A hybrid conditional random field for estimating the underlying ground surface from airborne LiDAR data, IEEE Trans. Geosci. Remote Sens. 47 (8/2) (2009) 2913–2922.

- Wagner, W., Ullrich, A., Ducic, V., Melzer, T., & Studnicka, N. (2006). Gaussian decomposition and calibration of a novel small-footprint full-waveform digitising airborne laser scanner. ISPRS Journal of Photogrammetry and Remote Sensing, 60, 100–112

- Wang, R. (2013). 3D building modeling using images and LiDAR: A review. International Journal of Image and Data Fusion, 4, 273–292

- Wilkinson, G. (2005). Results and implications of a study of fifteen years of satellite image classification experiments. IEEE Transactions on Geoscience and Remote Sensing, 43, 433–440.

- Wulder, M.A., White, J.C., Nelson, R.F., Næsset, E., Ørka, H.O., Coops, N.C., et al. (2012). LiDAR sampling for large-area forest characterization: A review. Remote Sensing of Environment, 121, 196–209

- Yan, W. Y., Shaker, A., & El-Ashmawy, N. (2015). Urban land cover classification using airborne LiDAR data: A review. Remote Sensing of Environment, 158, 295-310.

- Yan, W.Y., Shaker, A., Habib, A., & Kersting, A.P. (2012). Improving classification accuracy of airborne LiDAR intensity data by geometric calibration and radiometric correction. ISPRS Journal of Photogrammetry and Remote Sensing, 67, 35–44

- Yao, W., & Wei, Y. (2013). Detection of 3-D individual trees in urban areas by combining airborne LiDAR data and imagery. IEEE Geoscience and Remote Sensing Letters, 10, 1355–1359

- Yu, B., Liu, H., Wu, J., Hu, Y., & Zhang, L. (2010). Automated derivation of urban building density information using airborne LiDAR data and object-based method. Landscape and Urban Planning, 98, 210–219.

- Zhang, J. (2010). Multi-source remote sensing data fusion: Status and trends. International Journal of Image and Data Fusion, 1, 5–24

- Zhang, K., Chen, S.C., Whitman, D., Shyu, M.L., Yan, J., & Zhang, C. (2003a). A progressive morphological filter for removing nonground

measurements from airborne LiDAR data. IEEE Transactions on Geoscience and Remote Sensing, 41, 872–882.

- Zhou, W., Huang, G., Troy, A., & Cadenasso, M. (2009). Object-based land cover classification of shaded areas in high spatial resolution imagery of urban areas: A comparison study. Remote Sensing of Environment, 113, 1769–1777