SEMANTIC SEGMENTATION OF POLE-LIKE ROAD FURNITURE IN MOBILE LASER SCANNING DATA

Fashuai Li

SEMANTIC SEGMENTATION OF POLE-LIKE ROAD FURNITURE IN MOBILE LASER SCANNING DATA

DISSERTATION

to obtain
the degree of doctor at the University of Twente,
on the authority of the rector magnificus,
prof.dr. T.T.M. Palstra,
on account of the decision of the Doctorate Board,
to be publicly defended
on Thursday July 10, 2019 at 14.45 hrs

by

Fashuai Li

born on April 20, 1988

in Shandong, China

This thesis has been approved by
Prof. dr.ir M.G. Vosselman, supervisor
Dr.ir. S.J. Oude Elberink, co-supervisor

ITC dissertation number 359
ITC, P.O. Box 217, 7500 AE Enschede, The Netherlands

ISBN 978-90-365-4810-6
DOI 10.3990/1.9789036548106

Cover designed by Job Duim
Printed by ITC Printing Department

ITC UNIVERSITY OF TWENTE.
FACULTY OF GEO-INFORMATION SCIENCE AND EARTH OBSERVATION

Graduation committee:

Chairman/Secretary

Prof.dr.ir. A.Veldkamp

Supervisor

Prof.dr.ir. M.G. Vosselman	University of Twente / ITC

Co-supervisor

Dr.ir. S.J. Oude Elberink	University of Twente / ITC

Members

Prof.dr. M.J. Kraak	University of Twente / ITC
Prof.dr. N. Kerle	University of Twente / ITC
Prof.dr. J. Hyyppä	FGI, Finland
Prof.dr.-habil. H.G. Maas	TU Dresden, Germany

Acknowledgements

It is not knowledge,
but the act of learning,
not possession but the act of getting there,
which grants the greatest enjoyment.

Johann Carl Friedrich Gauss, 1808

It rains outside again. Complaints about the Dutch weather never cease in the past four years. Nevertheless, the bad weather never stops me from loving this country and the city, Enschede, where I have been living and studying for four and a half years. Though these years, the Ph.D. study at University of Twente was a marvellous experience. I have received numerous support from many people both in research and life during my Ph.D. journey. This thesis would not be finished without their support. I would like to make use of this opportunity to thank them.

First and foremost, I would like to greatly thank my Ph.D. promoter, Prof. Geroge Vosselman for giving me the opportunity to work on this topic. George has been an extraordinary mentor as well as an academic role model to me. His talents, diligence, deep insights and openness have been impressing and inspiring me. I have learned how to think deeper, to find problems and to cope with them from him. I am also thankful for my Ph.D. co-promoter, Dr. Sander Oude Elberink. He is a nice person always with a smile on his face. I appreciate the freedom he gives me and his comments on my academic papers. He always encourages me to be independent during my research. I hereby thank both of them for their insightful supervision. I am also thankful for Prof. M.J. Kraak, Prof. N. Kerle, Prof. J. Hyyppä and Prof. H.G. Maas for being in my thesis committee.

Speaking of brotherhood, I believe that everyone fellow has stories to tell. I would also like to send my gratitude to Speeder (Shaoqing) and his family (Kiki and Lydia). They ever brought me an unlimited amount of joys. From Enschede to Delft and Den Haag, we have spent the very best of times. Without their support, my Ph.D. life will be hollow. As Lydia’s godfather, I wish her to grow up healthily with happiness.

I would like to show thanks to my Chinese friends in ITC. They are Biao Xiong, Sudan Xu, Fengchao Gu, Mengmeng Li, Zhenchao Zhang, Yifei Xue, Jinfeng Mu, Yaping Lin, Ye Lv, Yiming Zhang, Ting Zeng, Tao Liu, Yunpeng Zhen, Heng Wang, Lina Zheng, Li Li, Xing Da, Dandan Lu, Zhaolan Peng, Yue Wang, Xiaoting Peng, Wanpeng Shao, Yizhen Lao, Baozi, Fangfang, Haichen Zuo, Ling

Chang, Chunjing Yao, Tonggang Zhang, Peng Jia, Michael Yang, Peiqi Yang, Jing Liu and Nelson. We ever shared the very best of ITC times. Sitting in front of the laptop with a cup of coffee held in my hands, I am still lost in the beautiful memories we ever had. Thanks also go to Chinese friends in TU Delft, Jinhu Wang, Kaixuan Zhou and other friends.

I feel very lucky to have kind and helpful officemates. They are Getachew, Rahul, Magnus, Elham and other visitors who ever stayed in my office. We have shared news and knowledge together and made our office as a small family.

Throughout my EOS life, I have met a group of lovely people. Without them, my Ph.D. journey will be gloomy. In the beginning of my Ph.D. journey, they helped me with my spoken English and English writing skills. In the later stage, I learned the cultural diversity from them. The gratitude goes to Marco, Shayan, Caroline, Diogo, Phillipp, Zill, Joep, Anand, Claudia, Andrea, JR, Sophie, Samer and other colleagues. Special thanks go to Teresa. Teresa is always optimistic and helpful during my Ph.D. study. I also would like to thank Markus, Claudio, Michael, Francesco, Valentyn, Ville, Frank, Wan and other EOS staffs for their support in EOS department.

I would like to thank Jasper, Arnold, Melvin, Thomas, Bobo, Muhammad, Marco and other housemates. They have made my student house life brilliant. Those pictures of gathering scene are still in head and will not be wiped out in the future.

I also want to thank my colleagues in Finnish Geospatial Research Institute (FGI). In the half-year exchanging program, I received tremendous happiness from the daily talk and the joint research with these lovely people. They are Hui Zhou, Ruizhi Chen, Yuwei Chen, Lingli Zhu, Xiaowei Yu, Xinlian Liang, Ziyi Feng, Eetu, Matti, Leena, Antero and other colleagues in FGI. I would also like to thank my Finnish landlord Hannu and his big family. Their warm-hearted treat had let me feel back home.

Last but not least, I would like to give sincere thanks to my parents and my sister. They are always there for me whatever situation I am. Without their support, encouragement, and love, I wouldn’t have reached where I am standing now. Special thanks go to my beloved girlfriend Ruru. You complete me. I wish my little nephew to grow up healthily full of joy and explore the world with scientific curiosities.

Table of Contents

List of figures

List of tables

Chapter 1 - Introduction

1.1 Background

Road safety has been one of the primary focuses in public safety concerns for many years. In the 2015 global status report on road safety, the total number of global road traffic deaths remains unacceptably high at 1.25 million per year (**World Health Organization, 2015**). To conduct road safety inspections in Europe, the European Union has adopted a road infrastructure safety management directive (**Adesiyun et al., 2016**). The EU will eventually have a role in the safety management of the roads belonging to European transportation networks—which is set to encompass 90,000 km of motorway and high-quality roads by 2020—through safety audits at the design stage and regular safety inspections of the network (**Bauer et al., 2017**). Similar policies also had been made in the U.S. In 2014, the U.S. Department of Transportation developed a system named Model Inventory of Roadway Elements (MIRE) to inventory road furniture and improve roadway safety (**U.S. Department of Transportation Federal Highway Administration, 2014**). Road safety can be enhanced by the inventory of road furniture which is strongly related to road furniture detection (**Soilán et al., 2018**).

To make the road environment safer and life more convenient, autonomous driving has been advanced in the past few years (the left figure of **Figure 1.1**). It is assumed that the number of road accidents will be reduced and drivers need to put less effort on driving in the near future with the assistance of autonomous driving systems (**Ibañez-Guzman et al., 2012**). There are many sensors amounted on autonomous vehicles to increase their reliability (the right figure in **Figure 1.1**). These sensors, however, are not able to work regularly under adverse weather conditions such as foggy days. It consequently leads to the unreliability of autonomous driving systems. Therefore, other assistants such as navigation in combination with 3D precise maps are needed to support the system under such weather conditions. Road furniture as an essential clue plays a significant role in the acquisition of the surrounding environment of autonomous vehicles. For instance, the autonomous driving system can adjust the velocity of a moving car by searching traffic signs in precise maps.

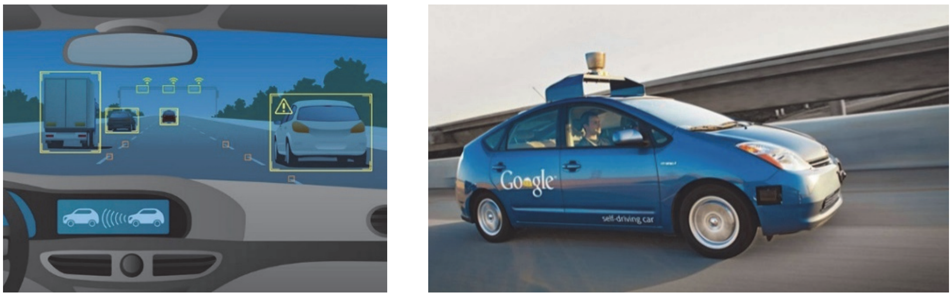


Figure 1.1*: An autonomous driving system (left) (Image source: vesttech blog) and Google autonomous driving car (right) (Image source: EXTREMETECH)*

Traffic congestion has obsessed urban citizens over a long period of time. To tackle this problem, many measures, e.g. to encourage citizens to use public traffic, have been taken. Intelligent traffic as one of these measures adopted can mitigate the traffic congestion problem to some extent. For instance, the switching of traffic lights is adapted based on traffic conditions. In Europe, this type of traffic system has been put into practical use (**Figure 1.2**). The inventory of road furniture is essential in this type of traffic system.

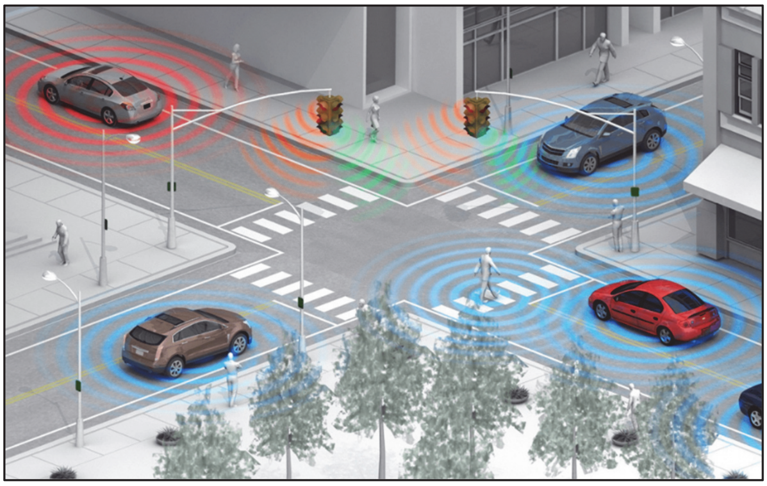


Figure 1.2*: The intelligent traffic system in which the shifting of traffic lights is dynamic (Image source: Tumblr)*

In view of these aforementioned issues, road furniture is of great importance not only because of its pivotal role in road safety and autonomous driving systems but also because of its significant role in intelligent traffic. Therefore, an automatic road furniture recognition is in great demand.

1.2 Laser scanning

To support the inventory of urban objects, systems, in which high-quality data is needed, have been developed. In urban scenes, 3D point clouds are commonly used as the data source for road-side objects inventory due to its availability of 3D location and geometric structure. The emerging of mobile laser scanning provides a solution to this problem. Mobile laser scanning systems have been well developed in recent years (**Puente et al., 2013**), which makes it possible to capture 3D point clouds of urban scenes with higher precision.

Three types of 3D laser scanning systems are often used to collect 3D data in outdoor scenes, airborne laser scanning (ALS) systems, mobile laser scanning systems (MLS) and terrestrial laser scanning (TLS) systems (**Figure 1.3**). ALS systems capture point clouds efficiently but with a low point density. For this reason, ALS systems are widely used for large scene surveying and mapping, e.g., national terrain surveying. TLS systems collect point cloud with higher point density. Therefore TLS systems are commonly used in mine surveying, e.g., open-pit mine surveying. Compared to TLS systems, MLS systems are more flexible and faster when acquiring data. Compared to the data collected by ALS systems, MLS data is denser and more precise. For these reasons, MLS systems are widely used to collect 3D point cloud data in urban scenes. On the other hand, optical images (e.g., KITTI dataset, **Fritsch et al., 2013**) have been widely utilised as a type of source data to interpret urban scenes. Accurate 3D positions of roadside objects can be easily acquired from 3D point clouds, whereas it is more difficult to obtain the 3D precise information from optical images.

(a)

(b) (c)

Figure 1.3: *(a) A mobile lasers canning system (Image source: RIEGL Laser Measurement Systems GmbH). (b) An airborne laser scanning system (Image source: RIEGL Laser Measurement Systems GmbH). (c) A terrestrial laser scanning system (Image source: C.R. KENNEDY & Company Pty. Ltd)*

With the support of high-quality MLS data, road furniture recognition has been well-studied in the past few years. A primary current focus in road furniture inventory is pole-like road furniture identification. Numerous studies have been carried out on pole-like road furniture recognition (**Brenner, 2009; Golovinskiy et al., 2009; Lehtomäki et al., 2010; Pu et al., 2011; Xiong et al., 2011; El-Halawany and Lichti, 2013; Bremer et al., 2013; Li and Oude Elberink, 2013; Cabo et al., 2014; Fukano and Masuda, 2015; Huang and You, 2015; Yang et al., 2015; Yu et al., 2015a; Li et al., 2017; Wang et al., 2017a**).

1.3 Machine learning

In recent years, machine learning algorithms have been explored in many research areas ranging from 2D image classification to 3D point cloud classification. In the framework of machine learning classifiers applied to classification problems, the training data and testing data are prepared first. Then the pre-processing and feature extraction are performed to obtain a number of multidimensional features. After that, a machine learning classifier is trained with manually labelled samples. Validation is used to cope with the overfitting problem. In the end, the trained machine learning classifier is employed to predict the label of every unit of data such as pixel or point, etc. The basic workflow of the application of machine learning classifiers is as shown in **Figure 1.4**.

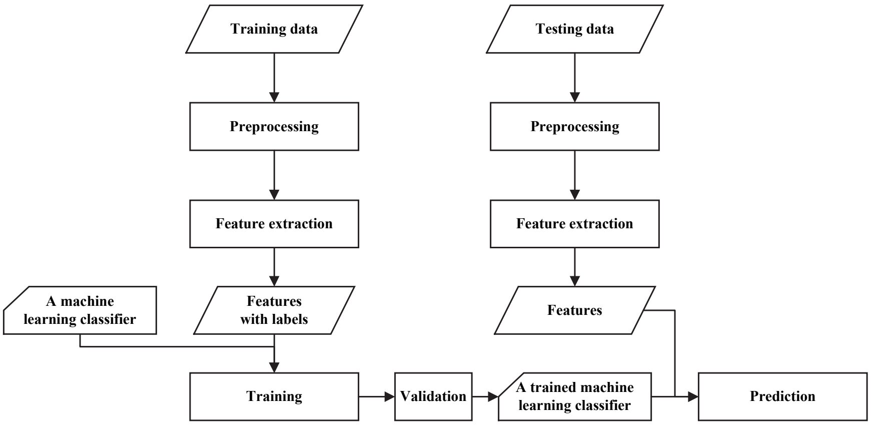


Figure 1.4*: The basic workflow of the application of machine learning techniques*

Popularly used machine learning classifiers have been applied to 3D point cloud processing. Random forests (**Chehata et al., 2009; Yu et al., 2011; Weinmann et al., 2014**), support vector machine (**Brodu and Lague, 2012; Yang and Dong, 2013; Lehtomäki et al., 2016**), Gaussian mixture model (**Vandapel et al., 2004; Belton et al., 2013**), naïve Bayes (**Lai et al., 2010; Tang et al., 2012; Yang and Tian, 2014**), Markov random fields (**Anguelov et al., 2005; Triebel et al., 2006; Munoz et al., 2008; Koppula et al., 2011**) and conditional random fields (**Lim and Suter, 2008; Rusu et al., 2009; Niemeyer et al., 2013; Müller and Behnke, 2014; Wolf et al., 2015**), etc. have been used for 3D point classification. The performance of 3D point cloud classification has been significantly improved. More recently, the rising deep learning techniques have been widely applied to numerous studies. Deep neural networks have been diversely built to facilitate 3D point cloud classification (**Maturana and Scherer, 2015; Huang and You, 2016; Qi et al., 2017; Su et al., 2018**). These emerging machine learning techniques have been greatly arousing 3D point cloud processing.

1.4 Research problems

Much progress has been made in street furniture detection and classification using mobile laser scanning data. There are still research problems making the result not good enough for practical use. It is essential to list them.

1.4.1 Problems with MLS data

Unevenly distributed data

As mobile laser scanning data is captured with a moving vehicle, the speed of laser scanner carrier is not constant, and the distance between the scanner and objects varies dramatically, which leads to the unevenness of point clouds.

Therefore algorithms relying on point density are not suitable for the process. **Figure 1.5** is an example of points with an uneven point density. Points in the upper red outlined area are much denser than points in the lower red outlined area. This is because the upper outlined area is much nearer to the scanner (points in the trajectory line) than the lower red outlined area.

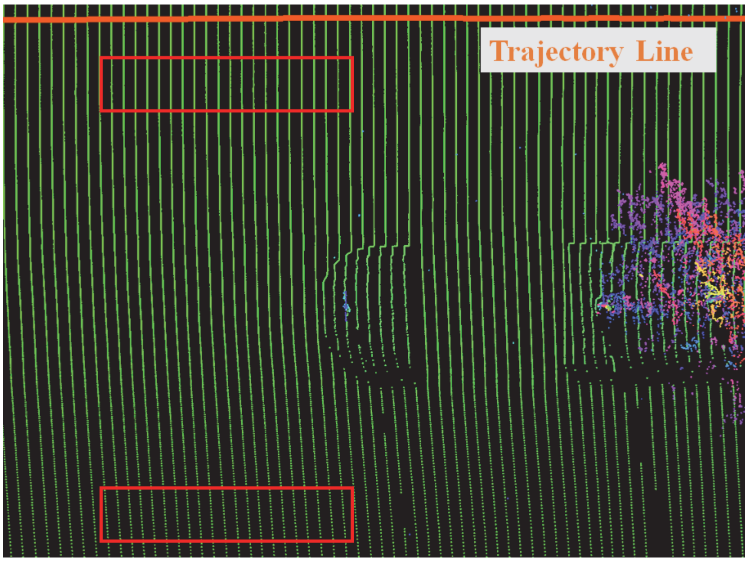


Figure 1.5*: Point clouds with an uneven point density. The point density of point clouds in the upper red outlined area is much higher than the point density of point clouds in the lower red outlined area.*

Discrete data with a low point density

Unlike 3D surface meshes, mobile laser scanning data is discrete and scattered. For small components of street furniture such as traffic lights, their shape information is not as reliable as 3D meshes. As aforementioned, compared to TLS systems and ALS systems, MLS systems better leverage efficiency and accuracy. Nevertheless, collected point clouds are not dense enough to depict street furniture far from laser scanners such as the thin part of poles. Their detailed information (poles of street furniture are scanned as lines rather than tubular objects) is lost. An example is as shown in **Figure 1.6**. It is difficult to distinguish the shape of the cross-section of the pole in the yellow outlined area.

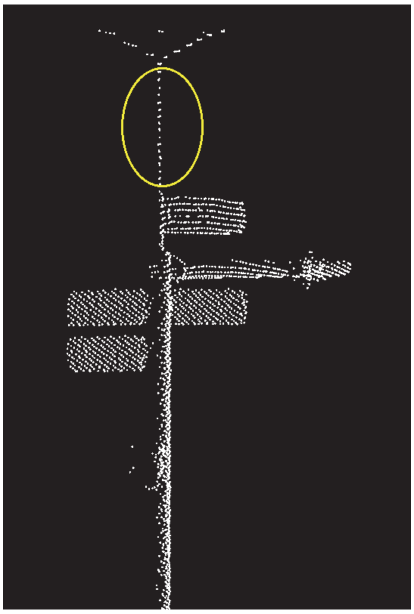


***Figure 1.6**: Point clouds (in the yellow outlined area) with a low point density*

1.4.2 Problems with scenes

Occlusion, connection and incompleteness

Due to the complex street environment, many pieces of street furniture are occluded by other objects such as moving vehicles and pedestrians. This makes it challenging to detect them. In addition, there are pieces of street furniture connected to each other or other objects, which makes it difficult to detect those target objects. Those occlusions mentioned above also makes it difficult to obtain the complete data of street furniture. Apart from that, mobile laser scanners are only able to acquire part of street furniture, because of the geometric scanning pattern. For instance, at most three-quarters of poles of road furniture can be scanned by MLS systems. This makes it difficult to design reliable features to recognise them.

Complex structures, various shapes and a small number of samples

Different shape elements make the classification of road furniture more difficult. Unlike building detection and reconstruction, street furniture contains plane elements, cylinder elements and some other irregular shapes.

Shapes of street furniture vary a lot. They differ from a ball-shape street light to an elongated street light. For example, there are more than 500 types of street lamps on the streets of Rotterdam. There is even more variation if other street furniture objects are taken into account. This makes a big challenge for the detection and classification of street furniture.

Another problem is there are not many samples. For example, less than 150 street lights were scanned in one of our testing sites. Many emerging techniques, which require big data such as deep learning neural networks, are not feasible to solve the classification task.

1.4.3 Research problems to be solved

According to the problems mentioned above, there are four main research problems summarised as follow:

Problem Ⅰ: the challenge with the detection and classification of uneven, sparse and incomplete point clouds of road furniture

Because of the principle of mobile laser scanner capturing data, street furniture is often scanned with uneven density, low resolution and discrete data, which makes street furniture lose part of shape information. Other objects such as vehicles occlude some street furniture, and this leads to the incompleteness of street furniture point cloud. These data problems make it difficult to detect and classify street furniture accurately.

Problem Ⅱ: the problem with the detection and classification of road furniture connected to other objects

There is street furniture connected with other objects such as bicycles. Those connected objects make the structure complex, and some features cannot be directly used on such street furniture. This makes it difficult to detect and classify.

Problem Ⅲ: the difficulty with the detection and classification of road furniture with various shapes

In one class of street furniture, there are various kinds of shapes and structures. This leads to many subclasses of street furniture. For instance, there are hundreds of types of street light heads. This makes it difficult to detect and classify street.

Problem Ⅳ: the problem with the classification of road furniture with multi-functions

As some street furniture has multi-functions, there are many different components with different functions in the street furniture. For example, there are some street signs and traffic signs attached to the same pole. This makes it difficult to classify such street furniture. The current study classifies a piece of road furniture as one object, which is not detailed enough especially for road furniture with multiple functions.

1.5 Research objectives and research questions

1.5.1 Objectives

The main objective is:
To design, implement and evaluate accurate methods for street furniture detection and classification from MLS data in urban environments.

Sub-objectives are:
1. To develop a method to detect pole-like road furniture with incompletely scanned, unevenly and sparsely distributed data.

2. To design and implement an algorithm to separate multi-functionalities pole-like road furniture into poles and their attachments in MLS data. The proposed algorithm should also be able to separate poles and other objects which are attached to poles. The designed algorithm should be able to segment pieces of pole-like road furniture at a more detailed level instead of considering the pole with all attached signs and lights as a single object.

3. To develop a method that interprets pole-like road furniture into detailed subclasses based on their functionalities in MLS data, such as a street light head and a traffic sign connected to a pole.

4. To test the performance of the proposed pole-like street furniture detection, decomposition and classification methods.

1.5.2 Research questions

Based on the objectives of our research, research questions can be divided into three groups: pole-like road furniture detection, pole-like road furniture decomposition, pole-like road furniture interpretation.

Road furniture detection

1. Which features can be used to remove objects that do not belong to street furniture?

2. How to use these features to remove other objects (buildings and trees, etc.)?

3. How to detect complex street furniture with incomplete, unevenly distributed and low-resolution data, or connected with other objects?

4. How to use the result of road furniture decomposition to refine the detection result?

Road furniture decomposition

1. Which multi-scale features and contextual features can be used to decompose street furniture into meaningful components?

2. How to use these features to decompose road furniture?

3. How to decompose street furniture with incomplete, unevenly distributed and sparse data? How to separate street furniture connected with other objects?

4. How to decompose a wide range of complex street furniture with many irregular elements?

5. How to address the over-decomposition and the under-segmentation of road furniture?

Road furniture interpretation

1. Which features of components (local features) can be selected to describe specific street furniture?

2. How to use these local features (features of components) to interpret street furniture?

3. Which contextual features of street furniture can be used to refine street furniture interpretation?

4. How to use primitive fitting and part template fitting to label various kinds of street furniture components?

5. Which machine learning classifiers can be used to improve the performance of road furniture interpretation if the proposed template fitting method does not work decently?

1.6 Thesis outline

This thesis consists of six chapters, of which the third, fourth and fifth chapter contain the main scientific contribution.

Chapter 1 introduce the background and the motivation of our research. We also briefly give an overview of laser scanning and machine learning classifiers. In the following two sections, we illustrate the research gaps and objectives.

Chapter 2 introduces mobile laser scanning systems mounted on different platforms, laser ranging technologies and applications. We also describe the four test sites used in this thesis.

Chapter 3, **chapter 4** and **chapter 5** are the whole framework for the interpretation of pole-like road furniture in mobile laser scanning data. We first detect pole-like road furniture in mobile laser scanning data. Then the point clouds of detected road furniture are decomposed into different segments based on spatial relations. In the end, we design a series of unary features and contextual features. Machine learning classifiers in combination with these designed features are utilised to classify these separated segments to different classes. The overall workflow is as shown in **Figure 1.7**.

In **chapter 3**, we present a framework to detect pole-like road furniture detection in unevenly and sparsely distributed mobile laser scanning data. In the first stage, we remove ground points and obtain above ground components. Then we remove points which belong to dynamic objects and façades. Afterwards, a slice-based method in combination with cylinder fitting is proposed to detect pole-like road furniture. At the last stage, an occlusion analysis is introduced to remove elongated objects behind building façades.

Chapter 4 describes pole like road furniture decomposition in mobile laser scanning data. In this chapter, we first conduct a pre-recognition to identify the type of road furniture and select the corresponding method to extract poles. After poles are extracted and removed from road furniture, a connected component analysis is conducted to separate segments attached to poles. A rule-based splitting and merging step was carried out to refine the decomposition.

Chapter 5 depicts our method for the interpretation of pole-like road furniture in mobile laser scanning data. We compare the performance of different machine learning classifiers and knowledge-driven method on the interpretation of road furniture. The transferability of the machine learning classifier with the best performance was then tested. In addition, we analysed the importance of our designed features.

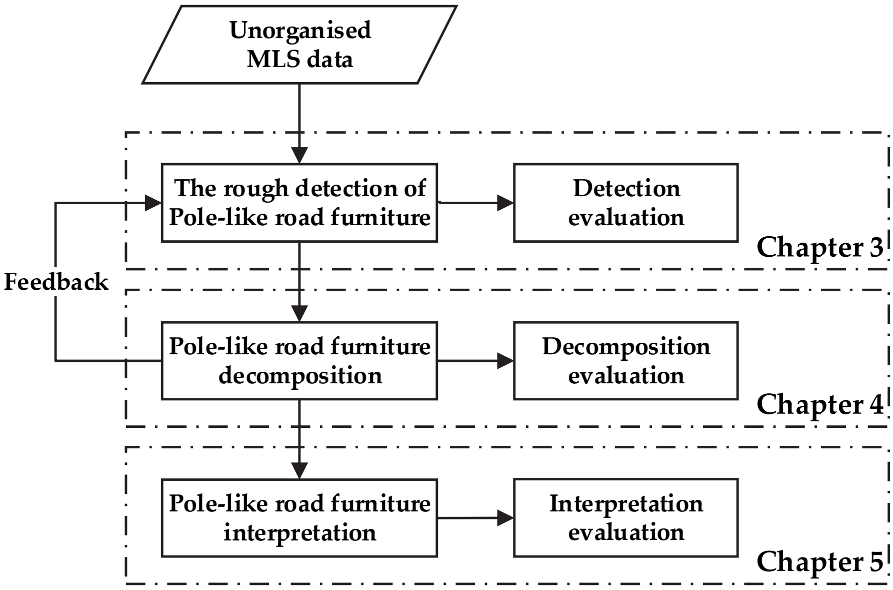

Unorganised MLS data
The rough detection of Pole-like road furniture
Detection evaluation
Chapter 3
Feedback
Pole-like road furniture decomposition
Decomposition evaluation
Chapter 4
Pole-like road furniture interpretation
Interpretation evaluation
Chapter 5

***Figure 1.7**: The workflow of the interpretation of pole-like road furniture in mobile laser scanning data*

Chapter 6 describes the main conclusions of our carried out research and recommendations for future research.

Chapter 2 - Mobile Laser Scanning

This chapter is organised as two sections. In the first section, we give an overview of MLS systems which includes their composition, laser ranging technologies, geometry scanning patterns and applications. The second section introduces four mobile laser scanning datasets used in this thesis.

2.1 Mobile laser scanning systems

Mobile laser scanning systems have been widely used to collect point cloud data in urban scenes because of their favourable balance of efficiency and precision. In this section, we first introduce the assembly of mobile laser scanning systems. Then two types of laser ranging technology are presented, and geometry scanning patterns are explained. Applications of mobile laser scanning systems with respect to corresponding platforms are illustrated in the last subsection.

2.1.1 Assembly of systems

The recognition of road furniture has relied on visual inspection or semi-manual interpretation in the past few years, which is time-consuming and tedious. To facilitate this procedure, scholars have proposed methods to automatically recognise road furniture, and these methods highly rely on high-quality data. Tools for capturing three-dimensional road scene data are Mobile Mapping Systems (MMS) which have been developed rapidly in recent years. They are often mounted on a vehicle and mainly composed of four parts: Light Detection and Ranging (LiDAR) sensors that collect 3D point clouds, cameras that capture 2D imagery data, an accurate Global Navigation Satellite Systems (GNSS) that records the position of the previous two sensors, and an Inertial Measure Unit (IMU) that measures the pose of the sensors. A mobile laser scanning system is as shown in **Figure 2.1**. Compared to the acquirement of optical images, the collection of MLS data is not restricted by the illumination conditions such as good weather and daytime.

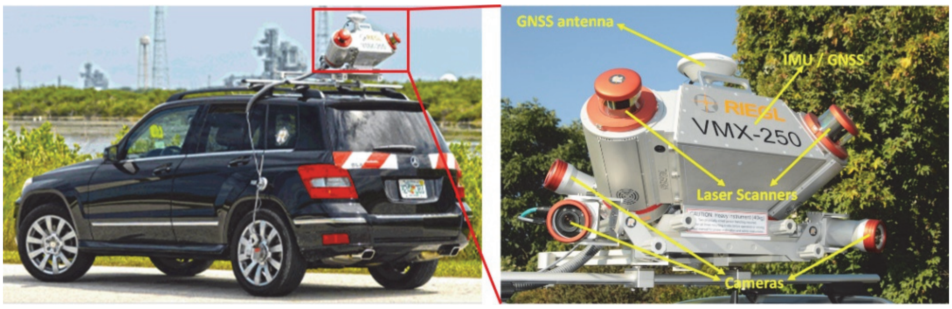


Figure 2.1*: A mobile laser scanning system (Left, image source: SPAR3D blogs; Right, image source: RIEGL Laser Measurement Systems GmbH)*

2.1.2 Laser ranging technologies

As mentioned in the above MLS systems, laser scanners are used to capture 3D point clouds. These 3D point clouds are the accumulation of single point measured by a laser beam. The measurement of the coordinate of a point from a laser beam is based on its transmission distance r and its corresponding angles α and θ. As shown in **Figure 2.2**, the 3D coordinate of a point P is solved as follows

$$x = r \cdot sin(\theta) \cdot sin(\alpha) \tag{2.1}$$

$$y = r \cdot sin(\theta) \cdot cos(\alpha) \tag{2.2}$$

$$z = r \cdot cos(\theta) \tag{2.3}$$

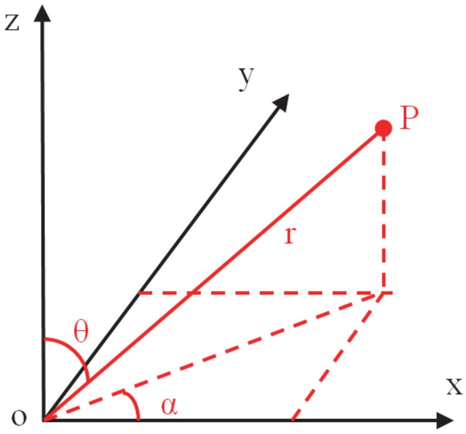


***Figure 2.2**: The solving of the coordinate of a point*

Currently, there are two laser ranging techniques for the measurement of 3D points in mobile laser scanning systems (operating range > 5m). One is time of flight ranging (ToF) technique, and the other one is phase-based ranging (PB) technique (**Vosselman and Maas, 2010**).

In the ToF ranging, the coordinate of a point is calculated by the measured. The transmitter first emits a laser beam. Then the laser beam transmits in the air, hits an object and transmits back. Finally, the signal of the laser beam is captured by the receiver (**Figure 2.3**). The transmission time is recorded as dt. The speed of a laser beam equals the light speed c. The transmission distance r is calculated as follows

$$r = \frac{dt \cdot c}{2} \tag{2.4}$$

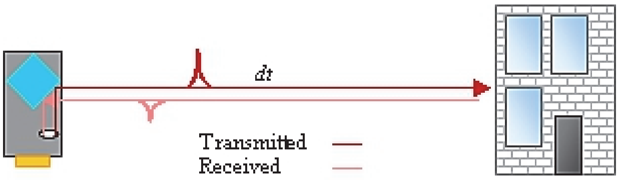


***Figure 2.3**: Principle of ToF laser ranging (Image source: **Kukko, 2013**)*

Compared to the measuring of transmission distance in the ToF ranging, the principle of measuring distance in the phase-based ranging technique is different. A transmitter continuously projects an outbound signal. After the signal hits an object and is collected by the receiver, there is a phase shift $d\varphi$ between the outbound signal and the received signal. Based on the wave length of the outbound signal λ_0 and the number of wavelength N (obtained by the modulated frequency) the signal ever transmits (**Figure 2.4**), the transmission distance r is resolved as

$$r = \left(N + \frac{d\varphi}{2\pi}\right)\lambda_0 \tag{2.5}$$

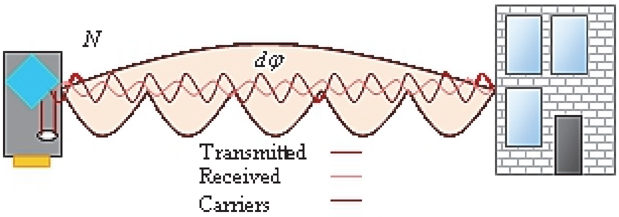


***Figure 2.4**: Principle of PB laser ranging (Image source: **Kukko, 2013**)*

The operating range of the ToF technique is much higher than PB pattern. For instance, the operating range of the ToF technique can be as high as 6000 meters. In contrast, the range of the PB technique is less than 100 meters. The ToF ranging, on the other hand, is characterised by a lower point density. More detail of these two ranging techniques can be found in the work of **Vosselman and Maas (2010).**

2.1.3 Applications

Based on the number of transmitters from a laser scanner, there are two types of laser scanners. The first one is the type of laser scanner scanning with only one laser beam transmitter and one receiver. The representative products are Riegl and Faro (**Figure 2.5**), etc. The second type is with multiple laser beam transmitters and multiple receivers (**Figure 2.6**). The commonly used products are Velodyne and RoboSense, etc.

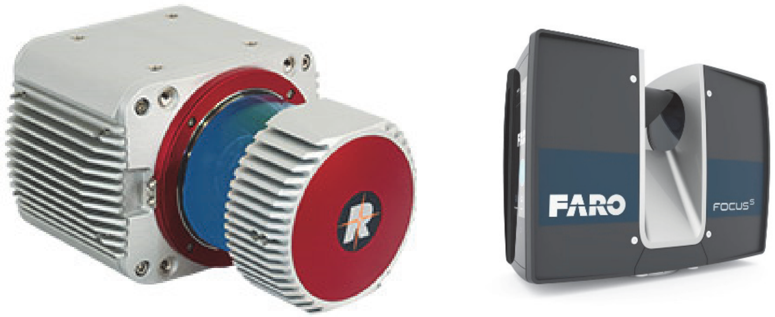


Figure 2.5*: Laser scanners with one transmitter and one receiver (Left, image source: RIEGL Laser Measurement Systems GmbH; Right, image source: FARO Technologies, Inc.)*

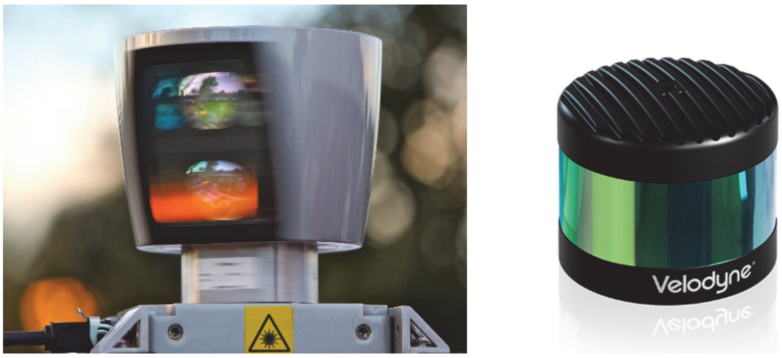


Figure 2.6*: Laser scanners with multiple transmitters and multiple receivers (Image source: Velodyne LiDAR, Inc.)*

The first type of laser scanner is widely used for precise mapping service. Except being amounted on driving vehicles for road mapping, they are also mounted on other moving platforms. These commonly used moving platforms are trains, backpacks, trolleys, boats and all-terrain vehicles etc. The train-based MLS system (**Figure 2.7**) is usually used to detect and model rails (**Oude Elberink et al., 2013**). MLS systems mounted on all-terrain vehicles (**Figure 2.8**) are flexibly used for various applications, of which forestry study is a typical one. Boat-based MLS systems (**Figure 2.9**) have been built up for the river environment study. Apart from the applications mentioned above, these MLS systems also can be applied to other studies or industry usage. In this thesis, we will not review all of them because they are not commonly used.

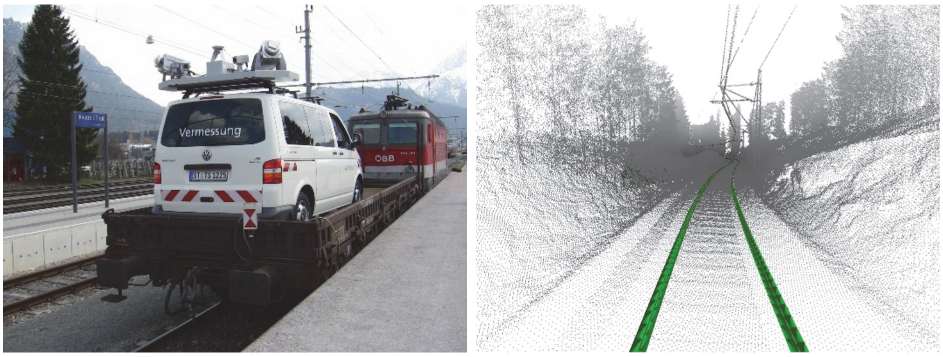


***Figure 2.7**: The train-based MLS system and rail track modelling (Image source: **Oude Elberink et al., 2013**)*

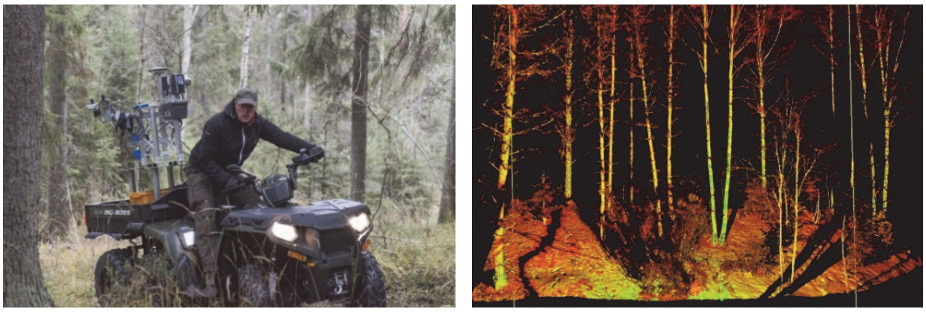


***Figure 2.8**: An all-terrain vehicle MLS system and forestry investigation (Image source: Centre of Excellence in Laser Scanning Research, Finnish Geospatial Research Institute)*

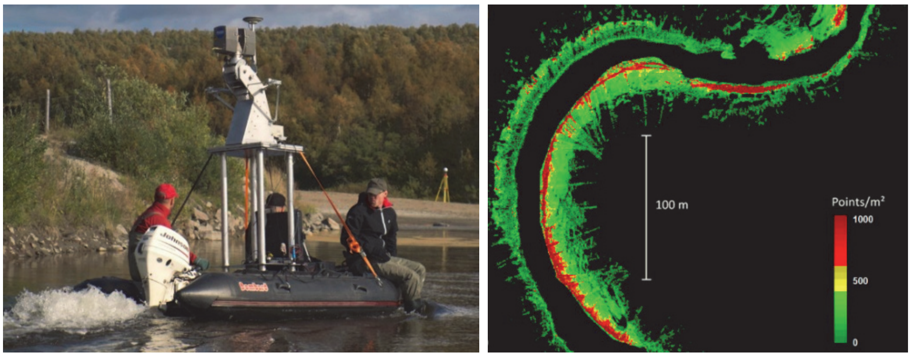


***Figure 2.9**: A boat-based MLS system and river environment study (**Vaaja et al., 2013**)*

Compared to the first type of laser scanner, the second type of laser scanner collects much sparser point clouds. They are widely used for autonomous driving, robotics and indoor mapping because of its featherweight compared to the first type of laser scanner. And there is no need to collect point clouds at a long-ranging distance in most required scenarios. Therefore, the frequency of laser beams emitted by such laser scanners is lower. This type of laser scanner plays a significant role in manoeuver planning for autonomous driving cars

(**Figure 2.10**) by semantically interpreting collected point clouds of road scene in real time. To compare the performance of different proposed methods, **Geiger et al., (2012)** made and released one benchmark dataset, the KITTI dataset. Backpack MLS systems are commonly used to collect point clouds for indoor mapping (**Figure 2.11**). For this type of laser scanner, another widely applied area is robotics (**Figure 2.12**). The scenario information is obtained from the interpretation of collected point clouds to support robots to finish pre-set tasks. With the development of laser scanners, more applications will be explored by MLS systems in future.

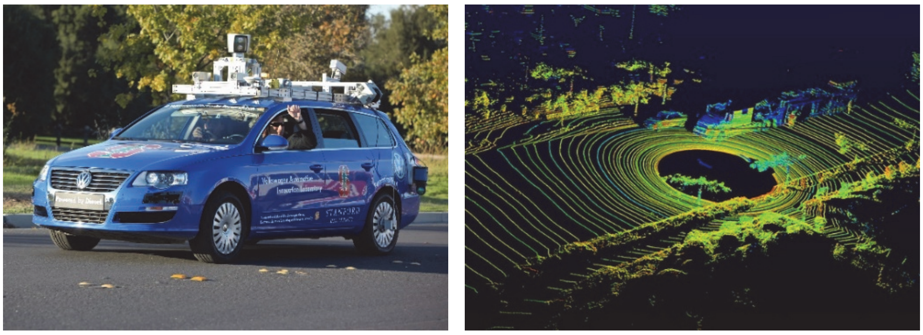


***Figure 2.10**: An autonomous driving car (Left, image source: Flickr photostream) and the collected point clouds (Right, image source: Velodyne LiDAR, Inc.)*

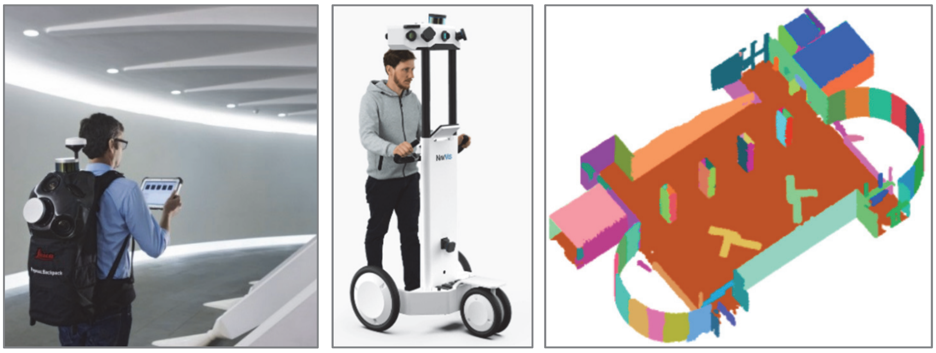


***Figure 2.11**: An backpack MLS system (Left, image source: Leica Geosystems AG), a trolley-based MLS system(Middle, image source: NavVis GmbH) and the indoor mapping (Right, **Nikoohemat et al., 2017**)*

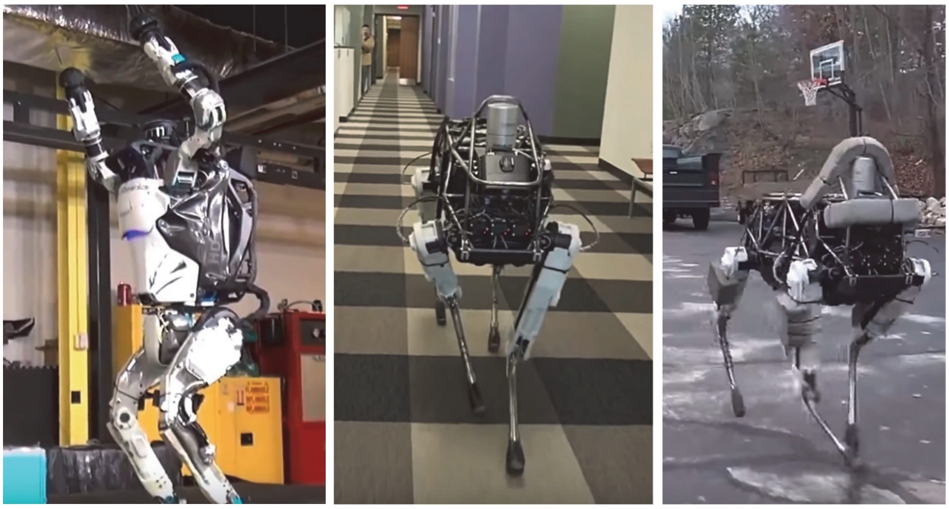


***Figure 2.12**: Bionic robots (Image source: Boston Dynamics)*

2.2 Test datasets

To conduct the experiments, we used datasets collected on four test sites to evaluate the performance of our framework. In these four test sites, there are lots of road furniture and different types of road furniture. These four test sites are located in three European countries, Netherlands, Finland and France. The first three test sites were collected by using laser scanners with the ToF ranging technique. The last test site was collected by using laser scanners with the PB ranging technique.

2.2.1 Enschede

The first MLS dataset was collected in Enschede, a medium-size city located in the eastern side of Netherlands. In December of 2008, it was collected by TopScan GmbH, which utilised the Optech LYNX mobile mapping system. In this system, there are two laser scanners amounted at the backside of a driving vehicle. The directions of rotating planes of these two laser beams are perpendicular to each other and form a 45° angle to the driving direction. An example of our adopted system is as shown in **Figure 2.13**. The frequency of both laser scanners was 100 kHz. The maximum driving speed of this platform was 50 km/h. In this dataset, pulse count and reflectance were also collected. The Enschede MLS trajectory is about 5.5 km long along which more than one hundred million points were captured. The point density ranges from 35 350 points per square metre.

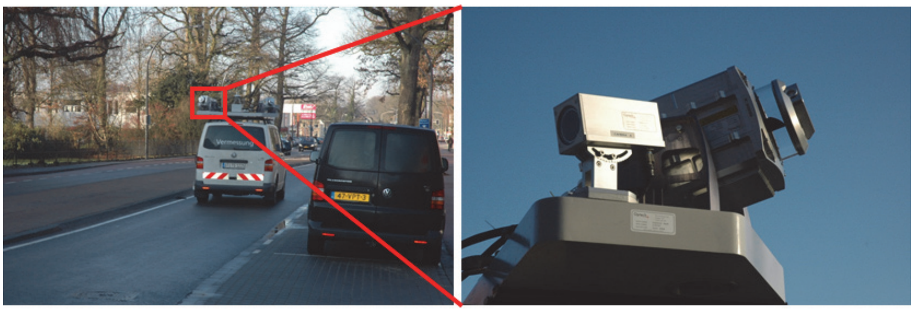


***Figure 2.13**: The Optech LYNX mobile mapping system*

2.2.2 Saunalahti

The second test site is located in the Saunalahti district of Espoo, the second largest city in Finland. It was collected in two different epochs. The first epoch was collected in December 2016, and the dataset is named Saunalahti winter. The second dataset, named Saunalahti spring, was collected in May 2017. Both datasets were collected using an Finnish Geospatial Research Institute (FGI) proprietary MLS system for road and urban environment mapping (**Figure 2.14**). The system was equipped with a Riegl VUX-1HA laser scanner for the collection of 3D point clouds. The scanner was set to scan at 250 Hz frequency for cross-track profiles and transmit 1017000 pulses per second. The scanner is capable of detecting up to seven echoes per pulse on-line, depending on the object characteristics. The setup resulted in 4068 transmitted pulses per 360-degree profile giving thus 1.5 mrad angular resolution. The area was captured mainly from streets at normal urban speed limits 20-40 km/h. At the approximate sensor elevation of about 2.8 m from the ground the point spacing right below the sensor was about 4 mm; at 50 m distance from the sensor the point spacing on a perpendicular plane was 77 mm, and the respective footprint size is 15 mm (0.3 mrad divergence). The Saunalahti winter dataset covers about 5.0 km of road, and the spring dataset covers about 4.0 km of road, consisting of more than one hundred million points.

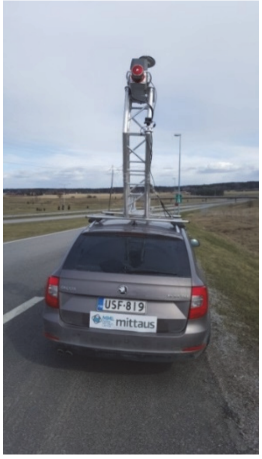


***Figure 2.14**: The FGI proprietary mobile mapping system*

2.2.3 Paris

The test data of Paris was captured in January 2013 by a Stereopolis II system (**Figure 2.15**), an MLS system developed by the French National Mapping Agency IGN. This is the IQmulus & TerraMobilita Contest dataset (**Vallet et al., 2015**). This dataset contains MLS data from a dense urban environment, composed of 300 million points, covering approximately 10 km. In addition, it is an accessible benchmark dataset. In this dataset, there are two available attributes: reflectance and pulse count information. The point density ranged from 72 points per square meter to 500 points per square meter. Compared to the Enschede dataset, the point distribution of the road furniture in the Paris dataset was quite uneven. This is because the geometry scanning pattern was different. The Enschede test data was collected by an MLS system consisting of two laser scanners, while the MLS system used to collect Paris dataset was mounted with only one laser scanner.

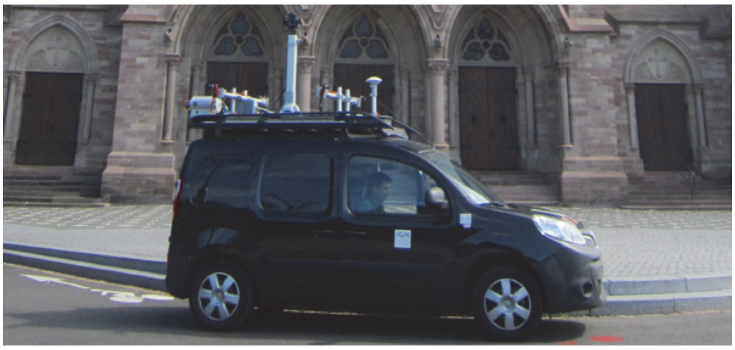


***Figure 2.15**: The Stereopolis II system (Image source: Institut Géographique National)*

2.2.4 Espoolahti

The Espoonlahti dataset was collected in Espoo, about 15 km west of Helsinki, the second largest city in Finland. The study area covers one block around the shopping mall Lippulaiva. It was acquired in 2009 by the ROAMER system (**Figure 2.16**). It consists of a Faro Photon™ 80 terrestrial laser scanner and NovAtel SPAN™ positioning system (NovAtel DL-4 plus GPS-receiver, NovAtel GPS-702-GG antenna and Honeywell HG1700 AG58 inertial measurement unit (IMU)) (**Kukko et al., 2007; Kukko et al., 2012**). The ROAMER system adopts the phase shift ranging technology. The dataset covers about 1.0 km of road scene. The point density of the dataset varies from 50 points per square metre to 250 points per square metre. The ratio of the distance between neighbouring points along scanlines and the distance between adjacent scanlines ranges from 0.1 to 0.42. The distribution of this dataset is strongly uneven and sparse. The Espoonlahti dataset was collected without multi-echo information.

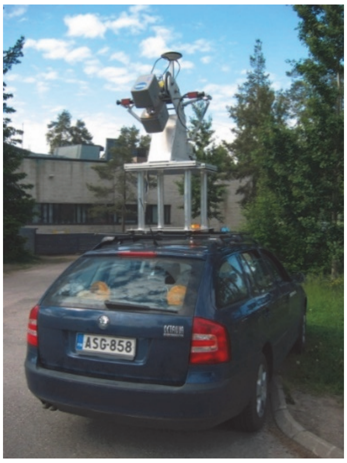


***Figure 2.16**: The ROAMER system (Image Source: **Kaartinen et al., 2012**)*

Chapter 3 - Pole-like Road Furniture Detection[1]

[1] This chapter is based on:

Li, F., Lehtomäki, M., Oude Elberink, S., Vosselman, G., Puttonen, E., Kukko, A. and Hyyppä, J., 2018. Pole-like road furniture detection in sparse and unevenly distributed mobile laser scanning data. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 4(2).
Li, F., Oude Elberink, S. and Vosselman, G., 2018. Pole-Like Road Furniture Detection and Decomposition in Mobile Laser Scanning Data Based on Spatial Relations. *Remote sensing*, 10(4), p.531.

3.1 Introduction

In recent years, road safety has been stressed in many countries. As the modernisation and urbanisation arise in developing countries, an increasing number of vehicles emerges on urban roads. To improve road safety, many measures, which include urban objects inventory, are adopted by governments. During urban objects inventory, urban objects can be counted based on their categories, and their locations can also be recorded. The obtained inventory information can be used to check the suitability of existing road furniture or to assist in the planning of placing road furniture. Road safety can be thereby enhanced by such an analysis. MLS data are denser and more precise. For these reasons, MLS systems are widely used to collect 3D point cloud data in urban scenes. Autonomous driving as a popular topic has been widely studied not only because of its convenience but also its contribution to road safety. 3D urban object detection in sparse and unevenly distributed points (e.g. 3D point clouds collected by Velodyne sensors) becomes crucial.

Currently, urban objects inventory mainly relies on manual labelling. Manual inventory of urban objects can be tedious and time-consuming. Therefore, automatic road-side objects inventory is in urgent demand. Much research has been carried out in the inventory of automatic urban road-side objects (**Golovinskiy et al., 2009; Pu et al., 2011; Yang et al., 2015; Lehtomäki et al., 2016; Wang et al., 2017a; Li et al., 2017**). One of the major focuses is pole-like road furniture detection due to their essential traffic functionalities. Except for their traffic functionalities, detected pole-like road furniture could also be potentially used as features for autonomous driving systems (**Hofmann and Brenner, 2009**) or Simultaneously Localisation and Mapping (SLAM) in urban scenes. Significant progress has been achieved in pole-like road furniture detection using mobile laser scanning data (**Brenner et al., 2009; Lehtomäki et al., 2010; El-Halawany and Lichti, 2013; Cabo et al., 2014**). However, the accuracy of detection is still modest especially in sparse and unevenly distributed point cloud data, which is crucial for 3D object detection in MLS data collected by autonomous driving systems. One reason is that these methods rely on geometric features such as eigenvalue-based features which are not robust in sparse and unevenly distributed MLS data. Another reason is that objects behind building façades are not eliminated. Therefore, in this chapter, we propose a method to detect pole-like road furniture from sparse and unevenly distributed mobile laser scanning data. Our method does not rely on the point density and point distribution. It is capable of detecting poles that are more distant from the road and can potentially be used for pole-like road furniture detection in autonomous driving systems.

The reminder of this chapter is organised as follows. The related work of pole-like road furniture detection in mobile laser scanning data is reviewed in

Section 3.2. Our proposed method is explained as in **Section 3.3**. In **Section 3.4** we test our algorithm and analyse the results. Finally, we draw conclusions from the above observations and analysis (**Section 3.5**).

3.2 Related work

As one of the focus topics in the research field of urban objects identification, pole-like road furniture detection has attracted much attention in recent years. Much progress has been made in this research field. Numerous methods have been proposed to detect pole-like road furniture in mobile laser scanning data. Previous proposed methods are assorted into two types, supervised learning and knowledge-driven methods. In supervised learning methods, distinctive features are utilised as input and trained to generate a model to make the prediction of candidates. In knowledge-driven methods, by contrast, rules or constraints are defined to make predictions based on inductive experience or knowledge.

Numerous knowledge-driven methods have been proposed to detect pole-like road furniture in MLS data. The work of **Brenner et al. (2009)** represents an early attempt to extract pole-like objects from MLS data. A cylindrical stack model is utilised to analyse the structure of measured laser points. **Lehtomäki et al. (2010)** further develop this method by using the scanline information of MLS data. At first, they extract potential sweeps from the data, remove elongated segments and keep short segments. Then these segments are grouped based on their profile information and horizontal plane position before these clusters that belong to one pole are merged by distance and orientation constraint. Finally, these clusters are classified into poles and non-poles according to their properties, such as height, shape, and orientation. The detection rate of the poles is 77.7%, and correctness is 81.0%. It is hard for this method to detect some complex pole-like objects that contain many points in the outer cylinder such as slanted pole-like road furniture or traffic signs that consist of many signboards. With a similar cylinder model mask, **Cabo et al. (2014)** proposed a voxel-based algorithm to detect pole-like road furniture objects from MLS data. To make the point clouds more uniform, they first voxelise the point clouds into grids. For every urban object, they then use two concentric cylinders to fit these voxels. Finally, pole-like objects are classified from point clouds. Although this method has acquired pretty good results, there are still some limitations such as the detection of poles that are close to bushes or guardrails. **Fukano and Masuda (2015)** detect pole-like objects and trees by using scanline information and a slice cutting algorithm. However, this method strongly relies on the triangulation of points, which does not work in sparse and unevenly distributed data. **Pu et al. (2011)** proposed a percentile-based method to recognise pole-like structures from mobile laser scanning data for road inventory studies. Firstly, they use trajectory information to partition the unorganised MLS data into different parts. Then they roughly classify the point clouds into three categories: ground points, points connected to the ground, and off-ground points. Finally, they adopt a knowledge-based method to recognise the above-ground segments. The method is able to detect 61–87% of poles in the point clouds. Among all the subclasses, the detection rate of trees is the lowest (29.5–63.5%) because of the lack use of multiple pulse count information. However, for this method, there are problems with the detection of jointly connected pole-like objects such as trees connected to pole-like objects. **Li and Oude Elberink (2013)** optimised the method of **Pu et al. (2011**) by adding reflectivity information. Because of the use of reflectance information and pulse count information, the rate of street sign detection and the rate of tree detection are significantly improved. However, connected pieces of road furniture cannot be detected and recognised in these two methods. **Huang and You (2015)** detect pole-like road furniture in MLS data by using slicing, seed generation and bucket augmentation. Multi-echo information is not adopted in this method.

Eigen-based features have been widely used for pole-like objects detection from MLS data. **Liberge et al. (2010)** extract above ground objects by local discontinuity. Three types of vertical lamp posts are detected by using eigenvalue features. Another eigenvalue based method to extract pole-like objects is investigated by **El-Halawany and Lichti (2013)**. They extract points with high linearity to detect poles. Then the pole-like road-side objects are detected by using a region growing method. **Bremer et al. (2013)** adopt multi-scale eigenvalue features in combination with defined rules to detect pole-like objects from MLS data. **Aijazi et al. (2013)** segment above ground points by applying voxelisation and a link-chain rule. Descriptors of supervoxels are used to categorise segmented objects into five classes. **Yokoyama et al. (2013)** utilise the Laplacian filter to smooth above-ground components. Eigenvalue features are then used to describe points with linearity, planarity and scattering. A designed model is fitted to decide whether an above-ground component is a pole-like object. **Yang et al. (2015)** voxelise MLS data at multiple scales based on their point attributes, and define a set of rules to recognise objects in urban scenes. A normalised cut algorithm is proposed to separate above-ground objects in **Yu et al. (2015b)**. They subsequently extract pairwise 3D shape context for a set of defined models to perform feature matching and recognise pole-like road furniture in urban scenes.

In contrast to knowledge-driven methods, supervised learning method do not require defined rules or models to make predictions. For supervised learning methods, after feature extraction, it is an end-to-end process. **Golovinskiy et al. (2009)** segment above ground objects by using a min-cut approach and recognise urban objects by shape features allied with machine learning techniques. **Weinmann et al. (2015)** employ an optimal neighbourhood selection and SVM to semantically label 3D urban objects which include pole-like objects. By using lost of features and the random forests classifier, **Hackel et al. (2016)** semantically label urban scene in MLS point cloud. **Lehtomäki et al. (2016)** develop an object wise classification framework on the basis of SVM to identify urban substances.

The performance of pole-like road furniture detection in sparse and unevenly distributed MLS data is still modest, less than 70%. Most of the methods aforementioned strongly depend on the point density during the calculation of features to decide whether road furniture is pole-like or not. Supervised learning methods need many training samples. In this chapter, we utilise the number of slices in combination with cylinder masking, which are not related to point density and are based on generic knowledge, to detect pole-like road furniture from MLS data (**Li et al., 2018b**). Numerous training samples are not needed in our method.

3.3 Methodology

The primary objective of this chapter is to segment road furniture within a certain distance from laser scanners. It is difficult to segment points that are captured distant to scanners because of their low point density. Therefore, we defined a distance to remove points that are far away from the road (trajectory line), which also helps to reduce the computation cost. At this stage, we started with pre-processing, which divided the unstructured data into roadblocks, removed the ground points, and obtained the above-ground components. Then, an initial classification was performed to remove dynamic objects and detect building components. In the last step, we proposed a slice-based method to extract pole-like road furniture, trees, and pole-like road furniture connected with trees and excluded objects inside buildings by occlusion analysis. The workflow of the initial pole-like road furniture detection is as shown in **Figure 3.1**.

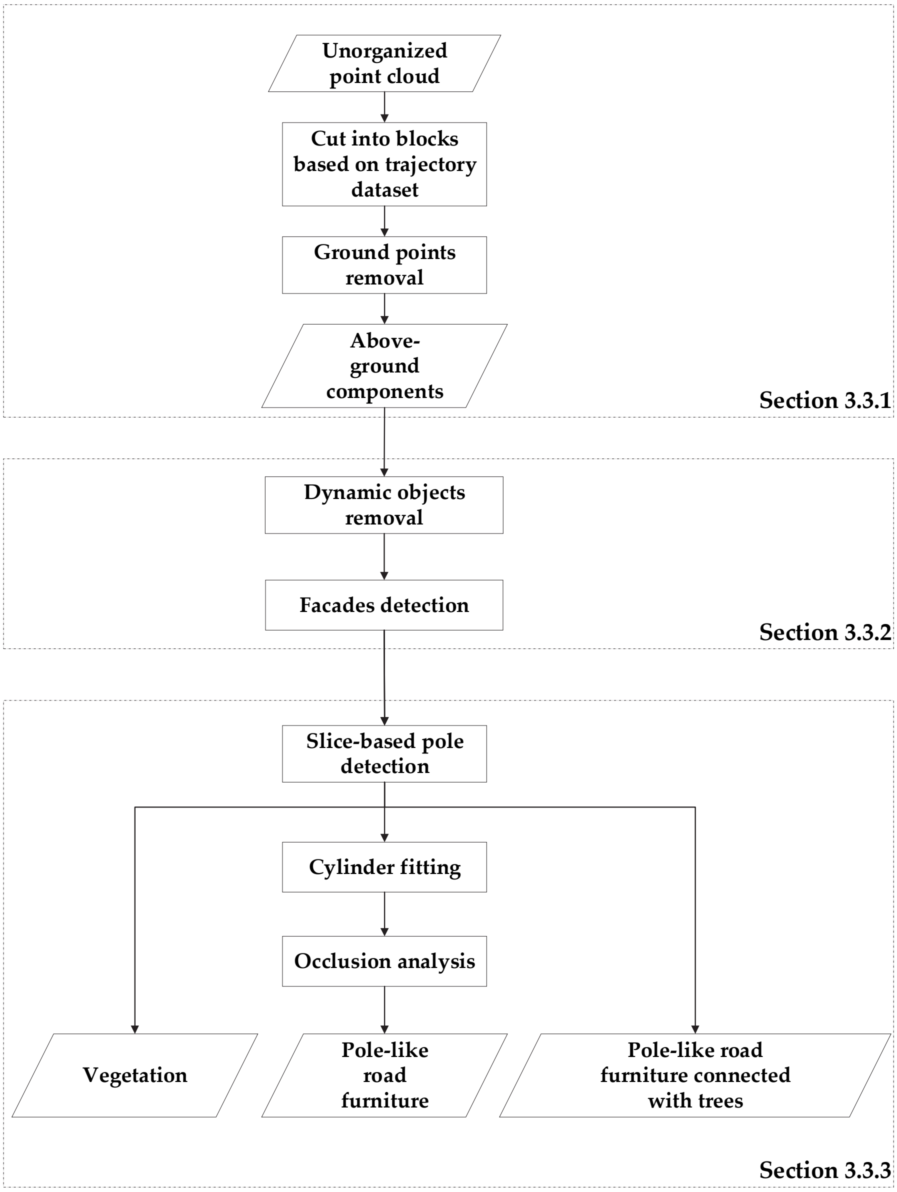


Figure 3.1: *The schematic of the pole-like road furniture detection.*

3.3.1 Pre-Processing

3.3.1.1 Data partitioning

The original mobile laser point cloud files are often very large, and this results in computation and memory problems when processed in one go. To circumvent these difficulties, we split the unorganised point cloud into road parts along the trajectory line as described in (**Pu et al., 2011**). In this research, the length and width of a road part are specified as 50 m and 40 m (**Figure 3.2**), respectively. Since one piece of road furniture could be separated into two parts, the length of the overlapping zone between two neighbouring road parts is set to 5 m.

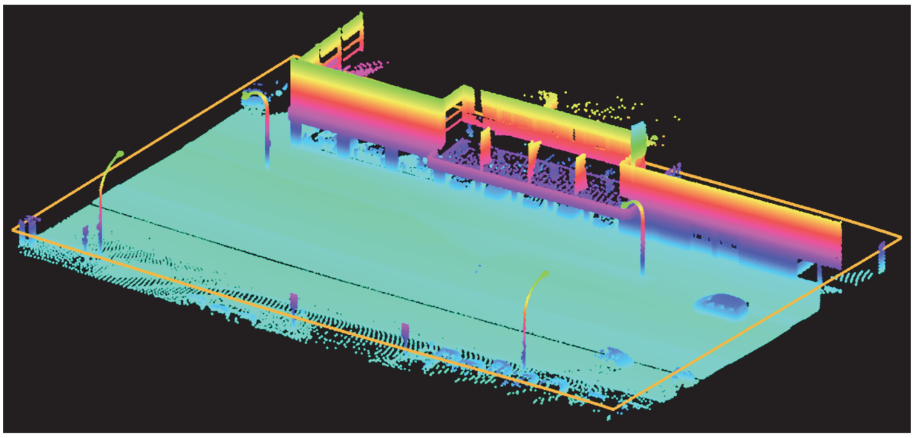


Figure 3.2: *One block of MLS data (points are coloured by their elevation)*

3.3.1.2 Ground removal

Ground points are the main connection between different above-ground objects. To separate different objects, we should remove ground points first. After splitting the point cloud into road parts, the ground points were removed in each road part. As the road surfaces are smooth, the height difference between all nearby points within a certain neighbourhood was small. In contrast, for above-ground objects, there was a large height difference between all nearby points within a certain neighbourhood. We calculated the height difference as the difference between the maximum height and the minimum height within a point's neighbourhood (**Oude Elberink and Kemboi, 2014**). In our research, the neighbourhood size was set to be quite large (e.g., 100 nearest neighbouring points) because of the high point density of the ground points. When the point cloud is very sparse, it should be set lower. The height difference threshold was set to be 0.15 m. Another constraint is that ground points are underneath their corresponding trajectory locations. As shown in **Figure 3.3**, compared to the two feature of above ground points, the height difference $\boldsymbol{p_{hd}}$ (the left orange bin in the figure) with ground points is small, and their height $\boldsymbol{p_h}$ (the left blue bin in the figure) is low. Hereby these two feature constraints in terms of first-order logic (**Russell and Norvig, 2003**) is derived as follows,

$$\forall p \; IsClass\,(p, Ground) \;\Rightarrow IsSmall\,(p_{hd}) \wedge IsLow\,(p_h) \tag{3.1}$$

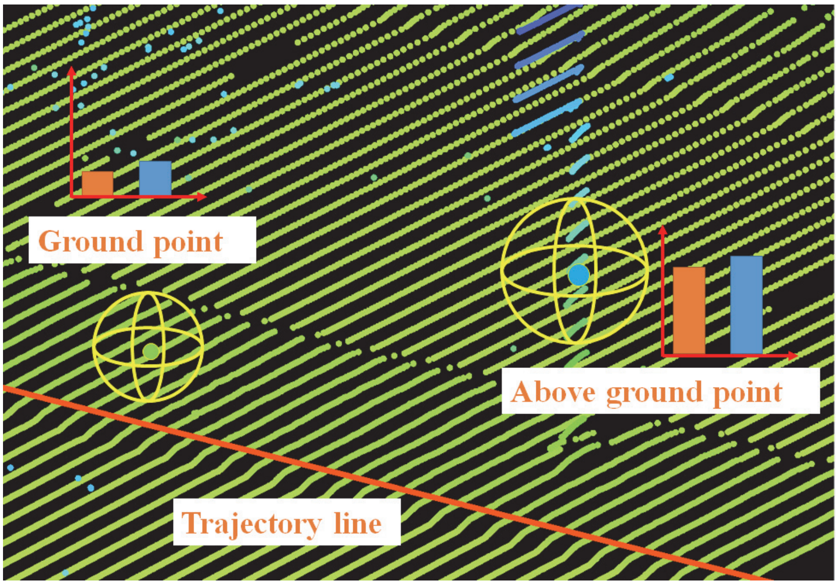


Figure 3.3: *The detection of ground points.*

The ground points were detected and removed by these two feature constraints. Compared to surface growing, this method proved to be more efficient.

After the ground points were removed, the remaining above-ground points were clustered by conducting a connected component analysis (**Vosselman et al., 2004**). After the connected component analysis, we obtained the above-ground objects for the initial classification.

3.3.2 Initial Classification

During the collection of MLS data, many moving objects or pedestrians were scanned. These irrelevant objects should be removed to mitigate false positive detection of road furniture. As mentioned in the introduction, our objects of interest were road-side objects which were assumed to have traffic functionalities. However, large building façades also frequently occur in road environments. Consequently, it is necessary to remove them to reduce the computation consumption for road furniture detection. On the other hand, buildings façades can provide useful information for pole-like road furniture detection. For instance, some pole-like objects inside buildings can be eliminated by using façade information. In this research, dynamic objects, buildings, and fences that are not related to road furniture were detected in this period and removed before decomposition.

3.3.2.1 Dynamic objects detection

For MLS systems with two laser scanners, the scanning of moving objects is as shown in **Figure 3.4**. When the moving direction of MLS platform and the moving direction of objects are the same, the laser scanner (scanner B in the figure) near to the moving object first collects the point cloud (the blue box in the figure). When the further laser scanner (scanner A the figure) scans the moving object, the object is already in another position. As long as the speed of moving object is not extremely slow, the collected point cloud (the orange box in the figure) with the further scanner and the collected point cloud with the closer scanner are detached. Therefore, in **Figure 3.4** the point cloud A is scanned with the scanner A, and the point cloud B is scanned with the scanner B. The scanning case is the same with the scenario when the moving direction of MLS platform and the moving direction of the object are opposite.

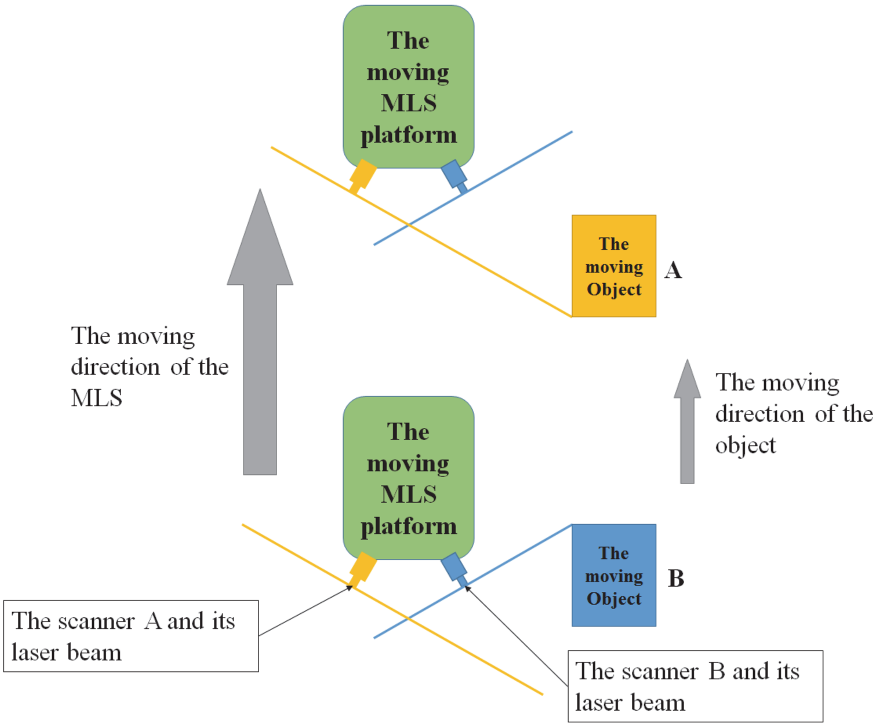

The moving MLS platform
The moving Object
A
The moving direction of the MLS
The moving direction of the object
The moving MLS platform
The moving Object
B
The scanner A and its laser beam
The scanner B and its laser beam

***Figure 3.4**: The scanning of moving objects.*

In the initial classification, we use with the number of scanners collecting a point cloud to remove dynamic objects. A component was labelled as a dynamic object if more than 90% of the points only had k-neighbours from the same scanner. The detection of dynamic objects can only be used for the data collected by multiple laser scanners. The feature constraint is derived as follows,

$$\forall c\ IsClass\ (c, Dynamic)\ \Rightarrow IsHigh\ (c_s) \tag{3.2}$$

Where c_s is the percentage of the points with k-neighbours from the same scanner in one component c. A more detailed explanation is given in the work of **Oude Elberink and Kemboi (2014)**.

3.3.2.2 Building façade detection

A large number of building points in the point cloud would lead to unnecessarily high computation time. Compared to other road objects, a façade plane is perpendicular to the ground, its area is large, and it is above a certain height. Similar to the method in (**Rutzinger et al., 2009; Demantké et al., 2012**), the orientation, height, and area of the façade were selected as distinctive features to detect the façade components. In this step, the surface growing algorithm was performed to extract the planes from the components. Based on an area and angle threshold, the large vertical planes were retained afterwards. The feature constraints are derived as follows

$$\forall c\ IsClass\ (c, Building)\ \Rightarrow IsLarge\ (c_{ap}) \wedge IsVertial\ (c_p) \wedge IsHigh\ (c_p) \wedge IsLarge\ (c_{wp}) \tag{3.3}$$

Where c_{map} is the area of the detected plane c_p in the component c. c_{wp} is the median width of cross-sections of detected plane c_p in the component c.

3.3.3 Pole-like road furniture detection

In this step, pole-like road furniture, vegetation, and pole-like road furniture connected to vegetation were extracted. As known, many types of vegetation have small branches and a large number of leaves. When a laser pulse hits small branches or leaves, this pulse usually splits into multiple pulses before reaching the receiver sensor. In contrast, most points of other objects exclusively have a single pulse count attribute (**Vosselman and Maas, 2010**). Consequently, the ratio of points with a multi-pulse count is useful for tree detection. However, there are points which belong to the edges of road furniture with multiple counts as well. If we use the number of returns, the points belong to the edges of above-ground objects are also labelled with multiple returns. Therefore, the value of the return number was utilised instead of the number of returns as a feature to extract trees. In our research, the ratio $\boldsymbol{R_{pc}}$ of points with the first return in above-ground components was also used as a feature to detect trees. In order to extract pole-like road furniture, **Pu et al. (2011)** proposed a framework which allows for the general interpretation of road furniture. Their work represents an early attempt to utilise a percentile-based algorithm to detect pole-like road furniture. However, it has difficulty in detecting road furniture with many attached components. For instance, there is difficulty with extracting traffic lights that are connected to many traffic signs and street signs. To overcome these difficulties, a slice-based method was presented to detect pole-like road furniture. The occlusion analysis was used to remove pole-like objects behind façades afterward. The workflow of this method is as shown in **Figure 3.5**.

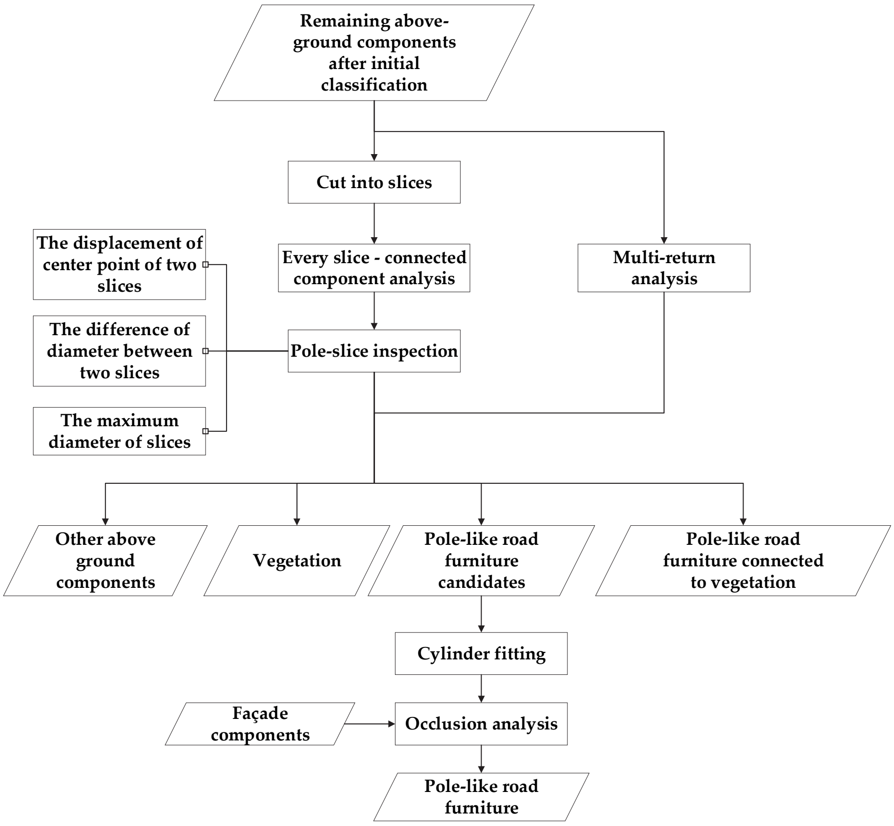

Remaining above-ground components after initial classification
Cut into slices
Every slice - connected component analysis
Multi-return analysis
The displacement of center point of two slices
The difference of diameter between two slices
The maximum diameter of slices
Pole-slice inspection
Other above ground components
Vegetation
Pole-like road furniture candidates
Pole-like road furniture connected to vegetation
Cylinder fitting
Façade components
Occlusion analysis
Pole-like road furniture

Figure 3.5*: The workflow of detecting pole-like road furniture.*

3.3.3.1 Slice-based pole-like road furniture detection

On the dense point cloud

First, every individual above-ground component was cut into horizontal slices (**Figure 3.6**). Then a connected component analysis was performed for every slice to produce separated components. The centre point of every separated slice component was computed, and a 2D connected component analysis was applied to connect the centre points of slices which were very close to each other in the horizontal plane. Here, three constraints were used to compute the number of pole-slices for the connected slice components in every individual above-ground component. The first constraint was the displacement ds of the centre points of two neighbouring slices (**Figure 3.6**). There should be no large displacement between the two neighbouring slices. The second one was the difference of the diameter of two neighbouring slices. The diameter of a slice was the 2D largest distance between two points in this slice. The difference of diameters between the two neighbouring slices should be small (e.g., 0.2 m), assuming that a part of the pole will have no attached objects. The third constraint was the diameter $\boldsymbol{d}$ of a slice (**Figure 3.6**). The diameter of a slice should be smaller than a pre-determined threshold. Finally, the number of pole slices was checked for every individual above-ground component. If the number of pole-slices was larger than a specified threshold (set to 3) and the ratio $\boldsymbol{R_{pc}}$ was smaller than a threshold (0.05), this component was labeled a pole-like road furniture. If the number of pole slices was larger than a specified threshold and the ratio $\boldsymbol{R_{pc}}$ was larger than the threshold, this component was labelled pole-like road furniture connected to trees. If both of them were smaller than their corresponding thresholds, this component was labelled trees. These feature constraints are derived as follows,

$$\forall s\ IsClass\ (s, Pole\_slice)\ \Rightarrow IsSmall\ (s_d) \wedge IsSmall\ (s_{dc}) \wedge IsSmall\ (s_{dd}) \qquad (3.4)$$

$$\forall c\ IsClass\ (c, Pole\_like)\ \Rightarrow IsLarge\ (c_{nps}) \wedge IsSmall\ (c_{rpc}) \qquad (3.5)$$

$$\forall c\ IsClass\ (c, Pole\_like\_connected\_tree)\ \Rightarrow IsLarge\ (c_{nps}) \wedge IsLarge\ (c_{rpc}) \qquad (3.6)$$

$$\forall c\ IsClass\ (c, Tree)\ \Rightarrow IsSmall\ (c_{nps}) \wedge IsLarge\ (c_{rpc}) \qquad (3.7)$$

Where s_d is the maximum diameter of two neighbouring slices$s$, s_{dc} is the displacement of centre points of two neighbouring slices s, s_{dd} is the difference of diameters of two neighbouring slices s, c_{nps} is the number of contiguous pole slices of the component$c$, c_{rpc} is the ratio of points with the first return in the component c.

On the unevenly and sparsely distributed point cloud

In order to detect pole-like road furniture in unevenly and sparsely distributed point clouds, we first cut above ground components into horizontal slices (**Figure 3.6**). Then slices with small diameters were selected, and their centre points were calculated. A 2D connected component analysis was performed on the centre points of these selected slices. The number of centre points in every connected component was checked. If the number is larger than a threshold, this object is labelled as pole-like road furniture. The threshold is leveraged based on the height of slices and the length of the shortest pole.

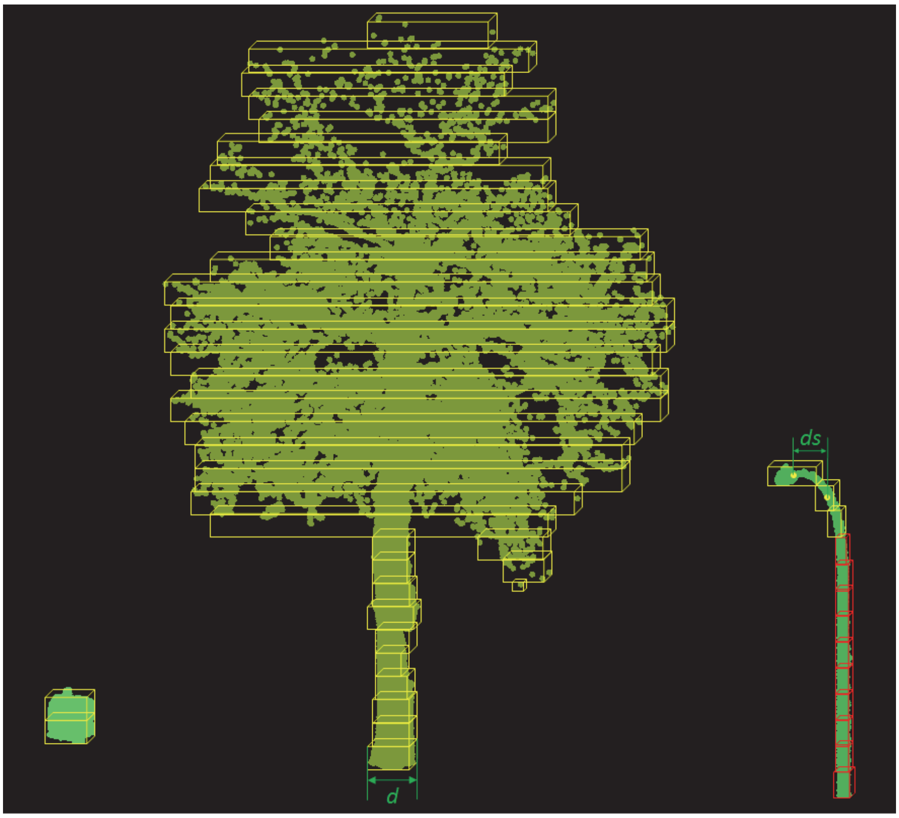


Figure 3.6*: The slice-based detection of pole-like road furniture.*

3.3.3.2 Cylinder fitting

After the slicing analysis, there remain above-ground components other than road furniture such as undetected trees. To eliminate these objects, we constructed two coaxial cylinders as shown in **Figure 3.7**. The ratio of the number of points inside the inner cylinder C_1 and the number of points inside the cylinder C_2 is the discriminative feature to decide whether these remaining above ground components are pole-like road furniture or not. This ratio with pole-like road furniture is high. In contrast, this ratio with trees, which have many branches, is smaller. Therefore, the remaining trees can be eliminated. This step is similar to the cylinder masking described in **Brenner et al. (2009)** and **Lehtomäki et al. (2010)**. The difference is that instead of using pole-like clusters retrieved from the profile information, we use connected slices to fit two coaxial cylinders. In this algorithm, r_1 is set to be the median width of these connected slices. r_2 is to be r_1+0.5m.

This step does not rely on the point density and the evenness of point distribution. It is because we use the number of slices and 2D connected component analysis to detect slices which belong to poles. The calculation of the number of slices and cylinder masking does not strongly rely on the point density and the evenness of point distribution. Therefore, high-quality data is not needed for this method.

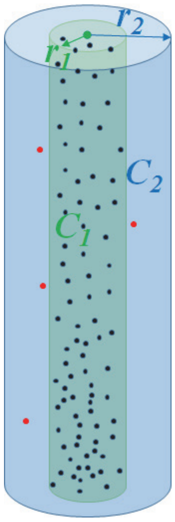


***Figure 3.7**: Cylinder masking*

3.3.3.3 Occlusion analysis

Among the detected pole-like road furniture candidates, there were incorrectly identified objects which were located behind façades. To exclude them, an occlusion analysis is performed. In the step of the initial classification, façade planes were obtained through surface growing. Then these façade planes were computed as constraints to determine if these pole candidates were located outside of the façade planes (**Figure 3.8**). If they were positioned outside of the façade, they were labelled as pole-like road furniture. Otherwise, they were not.

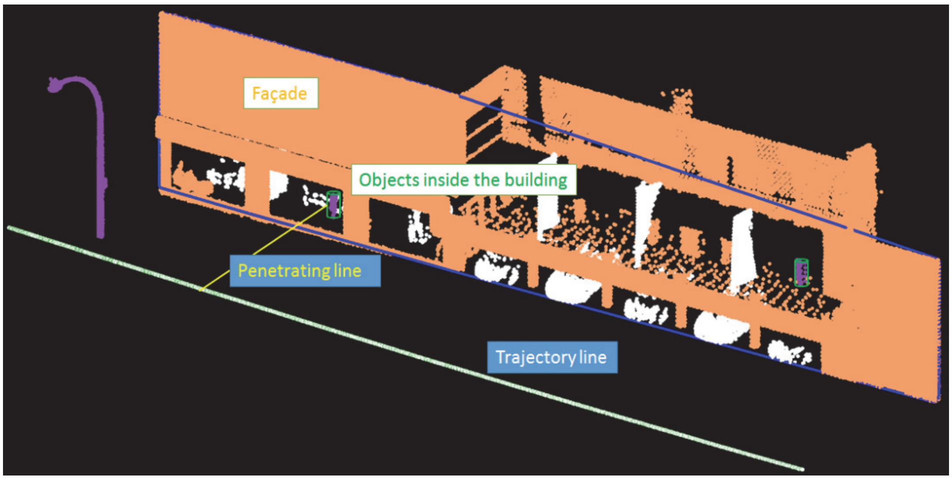


***Figure 3.8**: Occlusion analysis.*

3.4 Study Area and Experimental Result

To test the performance of the proposed framework, three datasets—one in Enschede, one in Paris and another one in Espoo—are chosen. The Enschede dataset was used as a test case because there are many types of road furniture and it is a typical Dutch city. The Paris dataset was a benchmark dataset, and Paris is a French city. The shapes of the road furniture in these three test sites are different because they are from two different countries. The first two datasets were used to test our framework on dense mobile laser scanning data. The Espoonlahti dataset was collected in Finland, and it was used to test the algorithm of pole-like road furniture detection in sparsely and unevenly distributed mobile laser scanning data.

The test of the presented framework on the Enschede dataset is given in **Section 3.4.1**, and **Section 3.4.2** gives the experimental results of the Paris dataset. The analysis of the results is explained in the corresponding sections. In **Section 3.4.3**, the algorithm for pole-like road furniture detection in sparsely and unevenly distributed mobile laser scanning data was tested on the Espoonlahti data and the analysis was given.

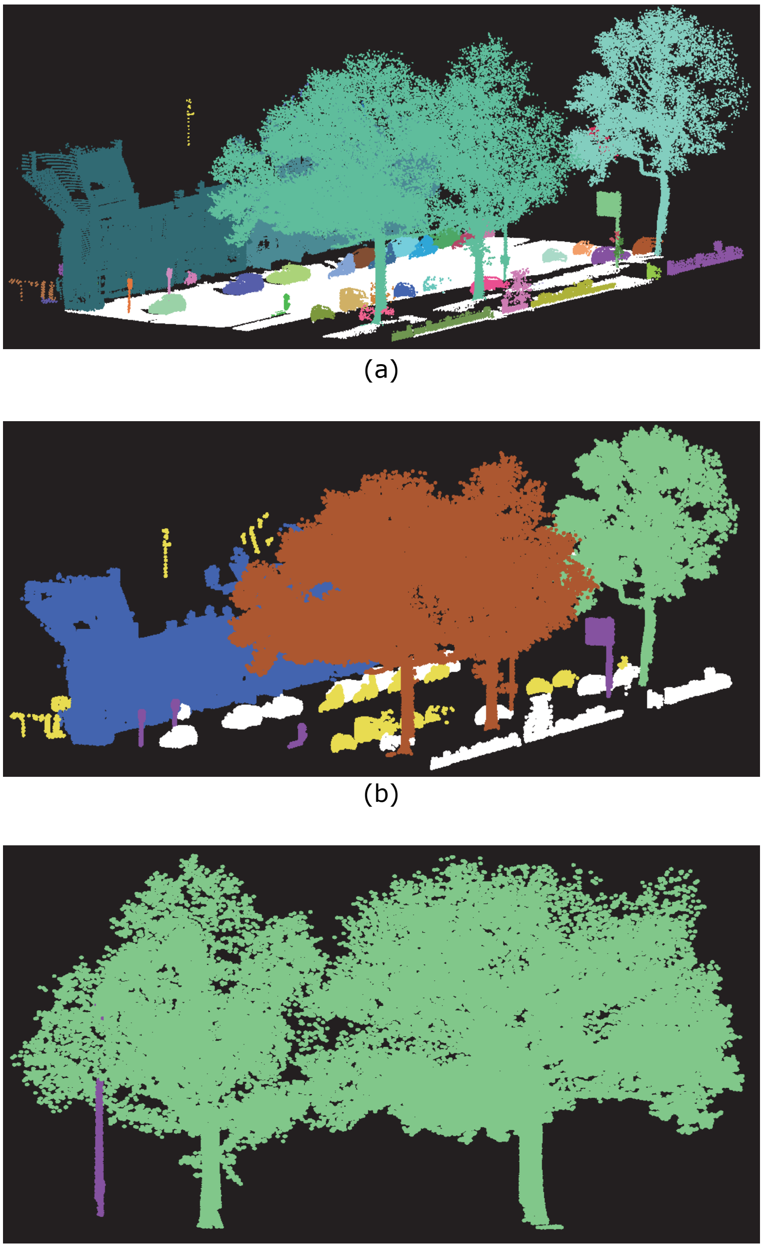
3.4.1 Enschede

3.4.1.1 Test data

The Enschede MLS data are about 1.25 km long, consisting of 19,945,716 points and 151 pieces of road furniture. The point density ranges from 35 points per square meter to 350 points per square meter. The test area was partitioned into 25 blocks along the trajectory line, with each block 50 m long and 40 m wide. In this dataset, the coordinates plus two attributes—the pulse count and reflectance strength—are available. The detail of the collection of Enschede dataset is described in **Section 2.2.1** of the previous chapter.

3.4.1.2 Results

The road furniture detection of a single road part in the Enschede dataset is as shown in **Figure 3.9**. The height difference was firstly estimated for every point in this road part. The height difference threshold between all nearby points within a certain neighbourhood to filter ground points was set to 0.15 m. The neighbouring size for the height difference calculation was set to be high (for example, 100) due to the high point density of the road surfaces. Then points with a small height difference were categorised as ground points (**Figure 3.9a**) and removed. The filtered ground points were coloured white. Subsequently, the above-ground components (**Figure 3.9a**) were produced by removing the ground points and the connected component analysis. The dynamic objects were labelled (**Figure 3.9b**) and eliminated from the above-ground components afterwards. Most of them were cycling bicycles, pedestrians, and moving vehicles. Because of the angles and occlusions, several objects behind building façades were scanned by only one laser scanner. This leads to these objects behind building façades being labelled as dynamic objects. Although this was incorrect, it does not affect our results because we would like to eliminate these objects behind the façades anyway. Then the façades were recognised (**Figure 3.9b**). More detailed information on this process was explained in **Section 3.3.1.2**. At the end of the detection phase, the slice-based poles were analysed. Vegetation, pole-like road furniture, and road furniture connected with vegetation were detected as in **Figure 3.9b**. Due to the proximity, some vegetation connected to road furniture was also detected. Two trees connected with one streetlight were coloured as brown in this figure. Their separation was as shown in **Figure 3.9c**. The detected pole-like road furniture was coloured in purple. The detected pole-like road furniture was as shown in **Figure 3.10a**. False positively detected road furniture inside buildings was excluded after the occlusion analysis, shown in **Figure 3.10b**. In this dataset, 115 pole-like road furniture with attachments were detected, given in **Figure 3.11**.

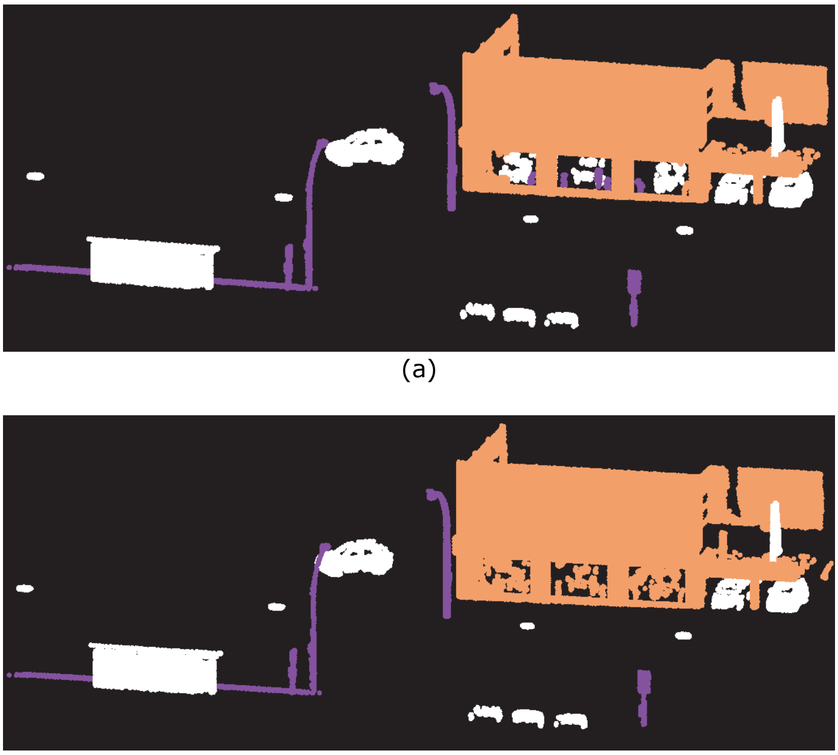


(a)

(b)

(c)

Figure 3.9*: The pole-like road furniture detection from the Enschede dataset. (a) ground points (white points) and above-ground connected components; (b) detected vegetation (light green), façade (light blue), dynamic objects (light yellow), detected pole-like road furniture (purple), and pole-like furniture connected to vegetation (brown); and (c) the separation of pole-like road furniture connected to trees.*



(b)

Figure 3.10*: The occlusion analysis to exclude incorrectly detected pole-like objects inside buildings. (a) The detected pole-like road furniture before occlusion analysis; (b) Pole-like road furniture after the occlusion analysis.*

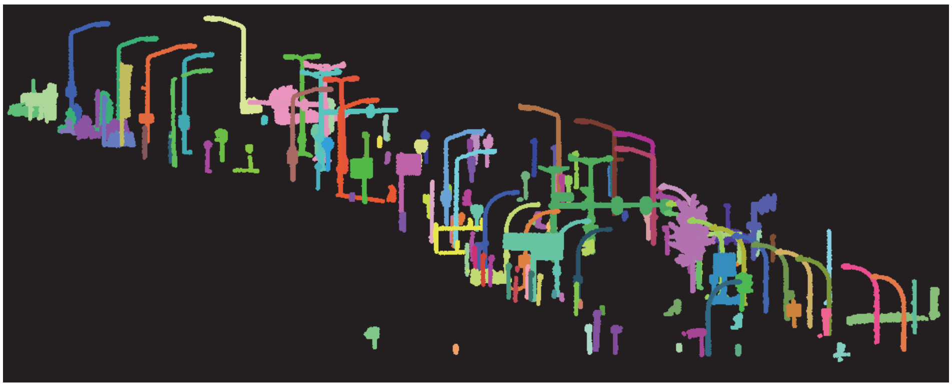


Figure 3.11*: The detected pole-like road furniture in Enschede dataset (coloured by the component number)*

3.4.1.3 Performance analysis and parametric sensitivity analysis

For the evaluation of pole-like road furniture detection, we manually inspected every detected pole-like road furniture. The completeness and correctness of pole-like road furniture detection are as shown in **Table 3.1**. The correctness is 96.0%, and the completeness is 94.2%. In total, 154 pieces of pole-like road furniture were detected, 9 of which were incorrectly recognised. Among these false positive detected entities, there were small garbage boxes and thin trees. Small garbage boxes were partially scanned, and they were similar to short poles such as road piles. Thin trees without many branches and leaves were also easily recognised as pole-like road furniture. This was because it was difficult to distinguish them from pole-like road furniture only from the structure information. Future research could be to investigate the colour information in combination with the point cloud. Two pole-like road furniture items were not identified because of the noisy points and low point density. Four traffic boards were identified as pole-like road furniture. Although their structure is non-pole-like, they can still be categorised as road furniture because of their salient traffic functionalities. Nevertheless, we strictly consider the detected traffic boards as negative results.

To set proper parameters, we chose one block which contained 22 pieces of road furniture to train the parameter settings. In the detection phase, the threshold of the maximum diameter of pole-like slices was obtained by using a grid search. The optimal threshold was 0.50 m. Then we used the trained parameters to detect pole-like road furniture.

***Table 3.1**: The accuracy of the road furniture detection in the two test sites.*

Test Site	**Enschede, NL**
Visual interpretation	151
Correctly detected	145
Total detected	154
Correctness	94.2%
Completeness	96.0%

3.4.2 Paris

3.4.2.1 Test site

The Paris MLS data was acquired in 10 zones. Compared with other zones, most road furniture was in zone 7. For this reason, it was selected as the test data zone. This test area is about 0.43 km long, consisting of 13,776,061 points and containing 132 pieces of road furniture. The test area was partitioned into 13 blocks along the trajectory line. We designed each block to be 40 m long and 30 m wide because the Paris streets are relatively narrow. The point density ranged from 72 points per square meter to 500 points per square meter. Compared with the Enschede dataset, the point distribution of the road furniture in the Paris dataset was quite uneven. This was because the scanning pattern was different. The Paris dataset was collected by only one laser scanner.

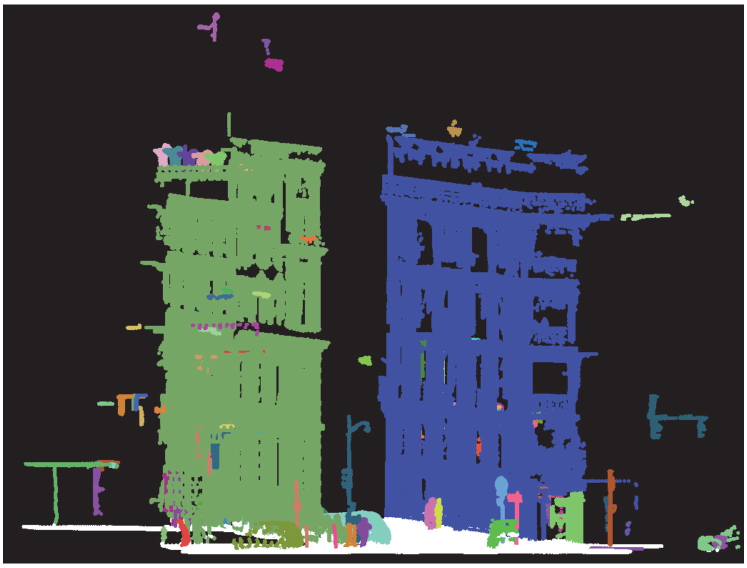
More detail of the data collection of this dataset can be found in **Section 2.2.3**, the second chapter.

3.4.2.2 Results

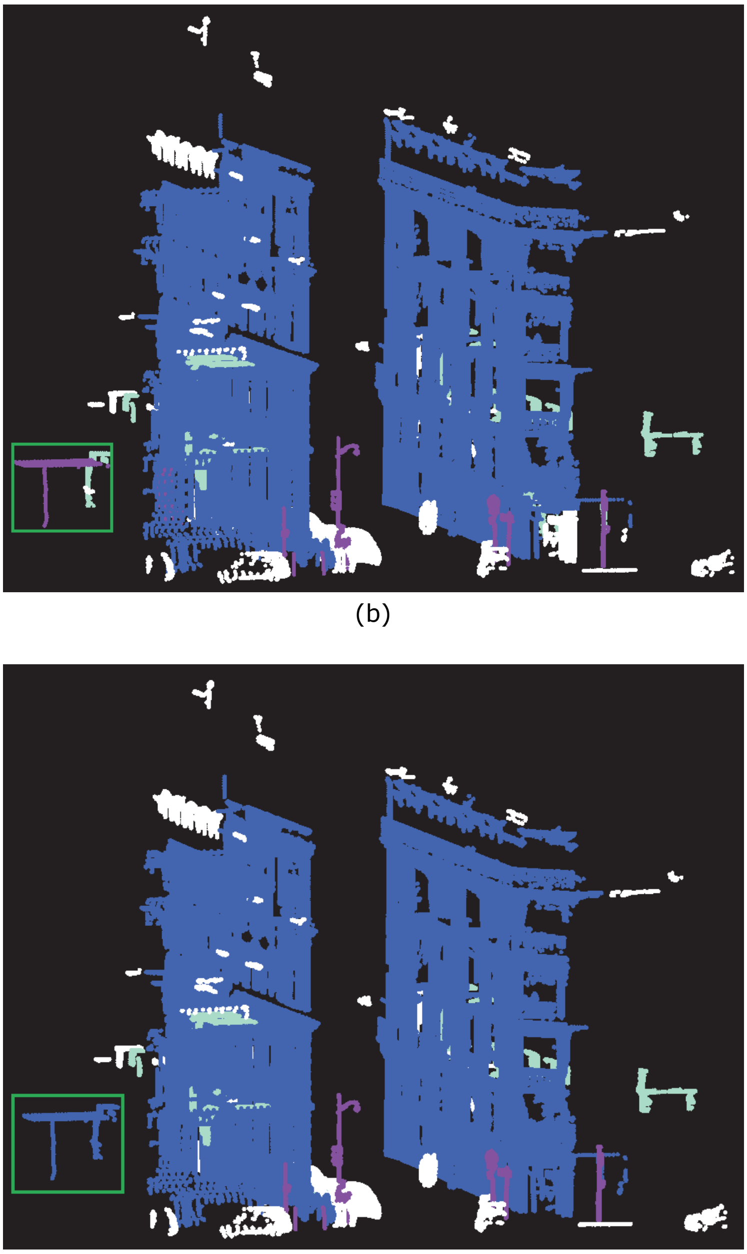
The original point cloud of one road part was as shown in **Figure 3.12**. The detected ground points, above-ground components, and façade points were as shown in **Figure 3.12a**. The detected pole-like road furniture before and after occlusion analysis are illustrated in **Figure 3.12b, c**.

Some entities behind the façades were detected as vegetation. The reason for this was that these objects were collected with multiple echo count information when the laser pulses passed through the windows and hit objects behind the windows. As this dataset was captured by using only one laser scanner, dynamic objects were not detectable by our framework.

In this dataset, 45 pole-like road furniture items were identified as poles with attachments (see **Figure 3.13**), and 97 were identified as bare poles.



(a)



(b)

(c)

Figure 3.12*: The pole-like road furniture detection from the Paris dataset. (a) The ground points and above-ground components; (b) The detected façade (light blue) and road furniture (purple). The entities behind façades were incorrectly detected as road furniture in the left green frame; (c) The label of incorrectly identified pole-like objects inside the buildings in the green frame was redressed after occlusion analysis.*

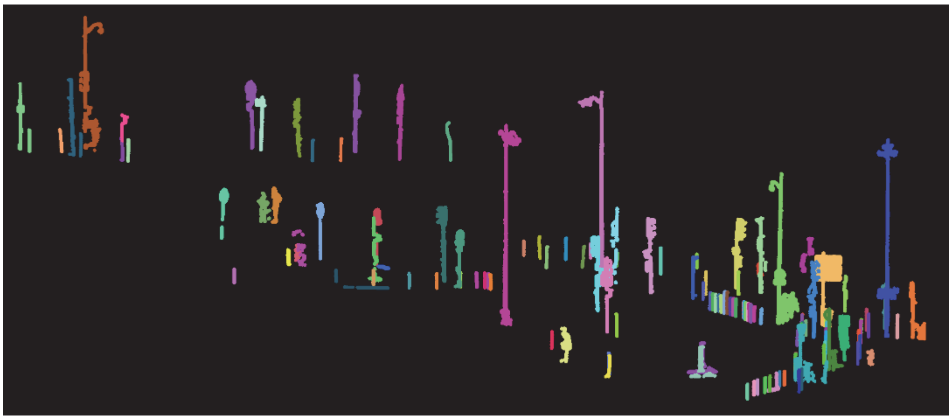


***Figure 3.13**: The detected pole-like road furniture in Pairs dataset(coloured by component number)*

3.4.2.3 Performance analysis and parametric sensitivity analysis

By means of visual inspection, the performance of pole-like road furniture detection in the Paris dataset was evaluated. **Table 3.2** shows that the correctness was 88.7% and the completeness was 95.4% in this dataset. In total, there were 142 pole-like road furniture items detected, 16 of which were incorrectly recognised. Among these false positive detected entities, most of them were pedestrians and entities whose point clouds were partially acquired. Pedestrians are similar to pole-like road furniture, especially in a noisy dataset. Another factor is that some road objects like fences were only partially scanned and are similar to pole-like road furniture in appearance. It is still a challenging task to distinguish them. Six pole-like road furniture items were not identified because of the low point density. Similar to the Enschede dataset, we also selected the optimal parameters by using a grid search in the Paris dataset. The optimal threshold of the maximum diameter of the pole-like slices was determined to be 0.30 m. The optimal ratio of the width of every slice for pole extraction was 0.50.

***Table 3.2**: The accuracy evaluation of road furniture detection in the Paris dataset.*

Test Site	**Paris, FR**
Visual interpretation	132
Correctly detected	126
Total detected	142
Correctness	88.7%
Completeness	95.4%

Besides that, frequent false alarms also originate from the connected objects. For example, in Paris dataset, there is pole-like road furniture extremely close to building façades. In the left figure of **Figure 3.14**, there is a street light (red circle in the left figure of **Figure 3.14**) close to the building façade. A road sign (red circle in the right figure of **Figure 3.14**) is connected to building façade. The smallest distance between pole-like road furniture and building façade in **Figure 3.14** is only 0.01m. It is difficult to extract such pole-like road furniture, as our algorithms groups points within 0.5 m.

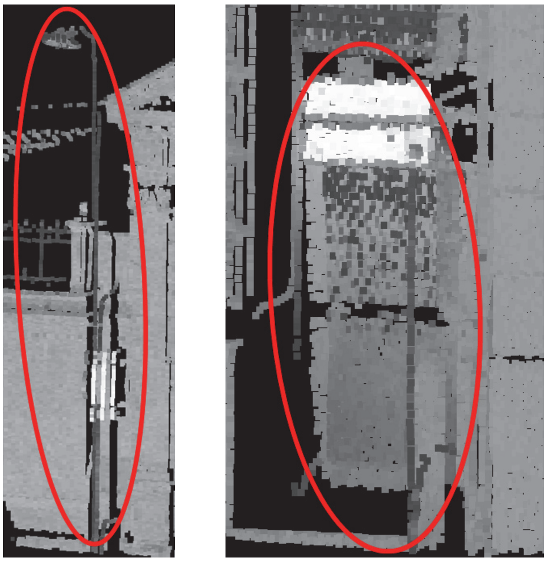


***Figure 3.14**: Undetected pole-like road furniture in Paris dataset.*

3.4.3 Espoonlahti

3.4.3.1 Test sites

The Espoonlahti dataset was collected in Espoo, the second largest city in Finland. It was acquired in 2009 by the ROAMER system, which consists of a Faro laser scanner and other sensors (**Kukko et al., 2007; Kukko et al., 2012**). A phase shift ranging system was adopted to collect the Espoonlahti dataset. It covers about 1.0 km of road scene. The point density of the Espoonlahti dataset ranges from 50 points per square metre to 250 points per square metre. The ratio of the distance between neighbouring points along scanlines and the distance between adjacent scanlines ranges from 0.1 to 0.42. The distribution of the Espoonlahti dataset is strongly uneven and sparse. The Espoonlahti dataset was collected without multi-echo information. The detailed description of this dataset can be found in **Section 2.2.4** of the second chapter.

3.4.3.2 Results and analysis

Experiments are carried out with these two introduced datasets. The aim is to detect pole-like road furniture higher than 0.5 metres. Experimental results are as shown in **Figure 3.15**. We evaluate the detection result by computing the completeness and correctness, which is as provided in **Table 3.3**. The completeness of the detection is 80.0%, and 87.0% of all detected objects are pole-like road furniture in this test site. The large segments in the right side of **Figure 3.15** and the left side of **Figure 3.15** are street lights connected to fences and trees. In our framework, we are still able to detect connected pole-like road furniture such as street lights attached to fences.

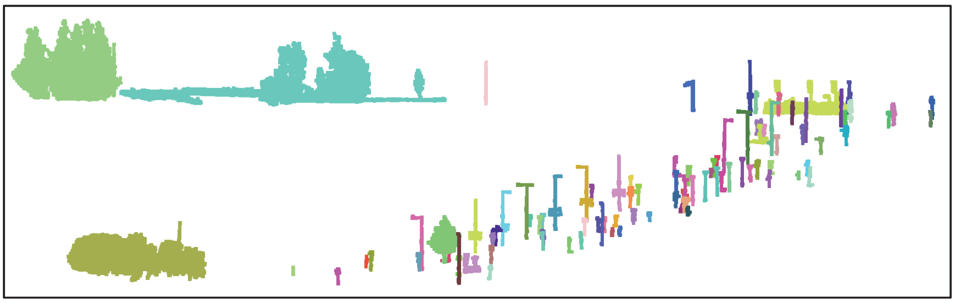


***Figure 3.15**: Detected pole-like road furniture in the Espoonlahti dataset.*

***Table 3.3**: Detection evaluation in two test sites*

Test Site	**Espoonlahti, FIN**
Visual interpretation	113
Correctly detected	85
Total detected	13
Correctness	87.0%
Completeness	80.0%

In this dataset, there is pole-like road furniture connected with other objects (as indicated in the red outlined area of **Figure 3.16**). When we cut them into slices, these slices of poles are still connected with each other unless we set the distance for connected component analysis to be very small. Small distance for connected components analysis nevertheless leads to fragments. Therefore, trading-off the parameters of the connected component analysis is insufficient to detect these poles. Another case is that trees, building pillars and pedestrians are mis-detected as pole-like road furniture. There is a tree with a few branches in this dataset (as shown in the right figure of **Figure 3.16**) mis-detected as pole-like road furniture. A few partially scanned pedestrians in both test sites are also mis-detected as objects of interest. The reason for falsely classifying pillars and pedestrians as pole-like road furniture is that we only use the width and the ratio as features to eliminate them. Trees with few branches are not detected correctly. It is because our method excludes trees based on the ratio of multi-echo points and the high ratio value in the cylinder masking. There are not sufficient points with multiple counts in such trees, and the ratio value is high in the cylinder masking. These features are not enough to make these bare trees, partially scanned pedestrians and building pillars discernible.

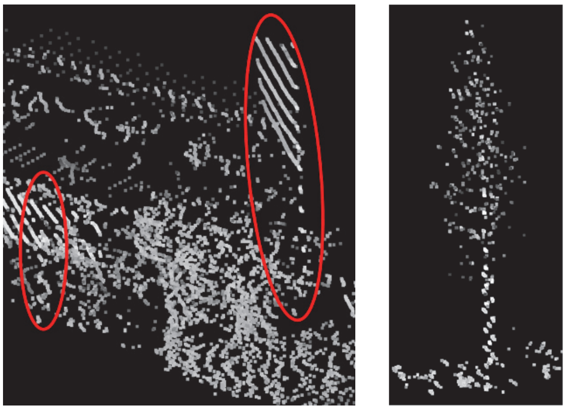


Figure 3.16*: Undetected pole-like road furniture and mis-detected pole-like road furniture in the Espoonlahti dataset.*

3.5 Discussion

In this study, we developed a new framework to detect pole-like road furniture in MLS data automatically. As our results show, pole-like road furniture were detected accurately and automatically. More than 90% of pieces of pole-like road furniture items were detected, and more than 88.5% were correctly detected in both test sites. The correctness of road furniture detection was 92.3%, and completeness was 83.3% in the work of **Cabo et al. (2014)**. In the work of **Yang et al. (2016)**, their achieved correctness and completeness of road furniture detection was higher than 90.0%. The average completeness of our framework is 91.4%, and the average correctness is 95.2%. Our framework is competitive with the current state of the art in the field of pole-like road furniture detection.

A significant advantage of this framework is the adoption of occlusion analysis. All pole-like objects inside buildings were eliminated in the Enschede test case, and only four pole-like objects inside buildings were not filtered out in the Paris test site. These four objects were located close to building façades whose normal direction was almost parallel to the trajectory direction. Our occlusion analysis was not able to cope with such situations.

Even though the accuracy of pole-like road furniture detection is already high, some problems remain. In both of the two test sites, small booths supported by pillars were recognise as pole-like road furniture. These booths have pillars whose appearance is very similar to poles. The detection algorithm was not good enough to discriminate the difference. Therefore, additional structural relations will be added later to address this problem. In the slicing-based pole-like road furniture detection step, trees connected to the pole-like road furniture were detected. We have yet been unable to separate them (**Figure 3.9c**) thoroughly.

In the third test site, compared to the method presented in **Lehtomäki et al. (2010)**, we are able to eliminate trees and retain pole-like road furniture with traffic function. The detection rate of our method was 80%, and correctness was 87%, while in the same dataset their detection rate was 69.7%, and the correctness was 86.7%. The detection rate is significantly improved. It is noted that the reference of earlier work also contained other than road furniture, such as tree trunks. Compared to our result, trees and objects behind facades are not eliminated in the study by **Cabo et al. (2014)**. Compared with the work of **Li and Oude Elberink (2013)**, our method is able to exclude false positively detected pole-like objects inside buildings. **Wang et al. (2017a)** have achieved pole-like road furniture detection with high accuracy, approximate 95%. This, however, required point clouds with high point density and good quality. Two laser scanners were used to collect their experimental datasets. Only one scanner is used to collect data in our test dataset. Point repetition frequency strongly affects the point density. Normally higher point repetition frequency leads to higher point density. Point repetition frequency is 1333 kHz in **Wang et al. (2017a)**. In **Yang et al. (2016)**, the used point repetition frequency is 1100 kHz. Our used point repetition is much lower, about 120kHz. Point density in the Espoonlahti dataset ranges from 50 points per square metre to 250 points per square metre. The point density is considerably higher (1500 points per square metre) in **Wang et al. (2017a)**. The mirror frequency has a significant effect on the evenness of point distribution. The higher mirror frequency, the more even point distribution. The mirror frequency of laser scanning collecting data is above 70 Hz in **Wang et al. (2017a)**. Compared with their datasets, the point clouds used in our experiment are of lesser quality, collected at the mirror frequency of 30 Hz.

3.6 Conclusions

In this chapter, we proposed a new framework to detect pole-like road furniture both from dense point clouds and sparsely and unevenly distributed mobile laser scanning data. Three test sites with two different laser ranging techniques were conducted for experiments. In this framework, we improved road furniture detection by combining dynamic objects removal, pole slicing, cylinder fitting, and occlusion analysis. The average completeness and correctness values of the pole-like road furniture detection were higher than 90% in both the Enschede dataset and the Paris dataset. Features used in this method do not rely on the high point density or evenly distributed points. This method works well in sparse and unevenly distributed mobile laser scanning data. Our algorithm can potentially be of practical use in the pole-like road furniture extraction in extremely sparse point clouds. An example of such sparse data is MLS data collected using Velodyne laser scanners, also used in autonomously driving vehicles. However, it is still difficult to separate pole-like road furniture connected to other objects such as trees and building façades.

The use of imagery or multispectral LiDAR could further help in distinguishing between the objects, i.e. a pole close to a facade should be detectable even with a single channel intensity when there is a large enough intensity gradient between the two. However, the use of colour information may lead to over-segmentation of individual objects. How to separate objects and avoid over-segmentation by adding colour information or other feature constraints remains to be explored in the future.

Chapter 4 - Pole-like Road Furniture Decomposition[2]

[2] This chapter is based on:

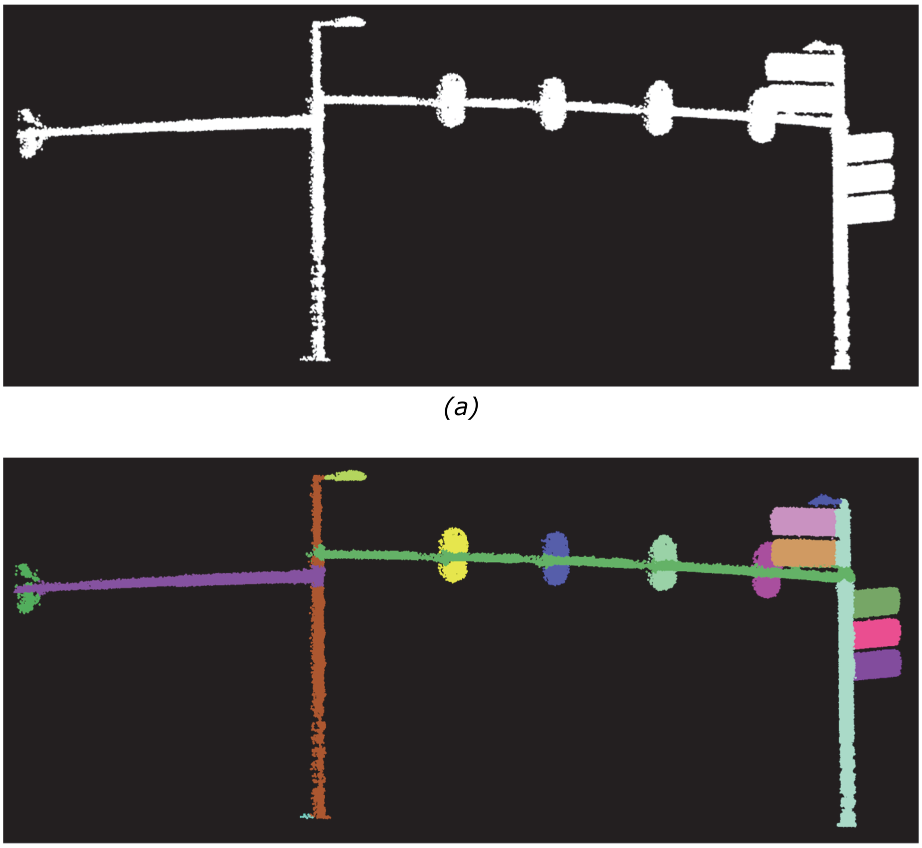
Li, F., Oude Elberink, S. and Vosselman, G., 2016. Pole-like street furniture decomposition in mobile laser scanning data. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 3(3).

Li, F., Oude Elberink, S. and Vosselman, G., 2018. Pole-Like Road Furniture Detection and Decomposition in Mobile Laser Scanning Data Based on Spatial Relations. *Remote sensing*, 10(4), p.531.

4.1 Introduction

Road furniture inventory has received much attention not only because of its importance in road safety but also because of its significant role in intelligent traffic system and autonomous driving. Currently, the road furniture inventory mainly relies on visual inspection or semi-manual interpretation, which is time-consuming and tedious. To facilitate this procedure, scholars have proposed methods for automatic road furniture detection relying on high-quality data. Tools for capturing three-dimensional road scene data are Mobile Mapping Systems (MMS) which have been developed rapidly in recent years. Significant progress has been made on research related to road furniture inventory in the laser scanning data field, which includes road detection and modelling (**Jaakkola et al., 2008; Oude Elberink and Vosselman, 2009; Zhang, 2010**), curbstone mapping (**Zhou and Vosselman, 2012**), railway modelling (**Oude Elberink et al., 2013; Yang and Fang, 2014**), pole-like road furniture interpretation (**Lehtomäki et al., 2010, Pu et al., 2011; El-Halawany and Lichti, 2013; Li and Oude Elberink, 2013; Cabo et al., 2014; Huang and You, 2015; Fukano and Masuda, 2015; Yang et al., 2015; Yu et al., 2015a; Wang et al., 2017a**), tree inventory (**Puttonen et al., 2011**), and building detection (**Rutzinger et al., 2011; Demantké et al., 2012; Yang and Dong, 2013; Niemeyer et al., 2014**). In the urban scene, MLS data analysis has become popular and essential for urban road environment analysis.

Current researches mainly focus on segmentation, street furniture detection and classification. However, earlier segmentation can only segment point cloud into small patches. Furthermore, generic detection and classification recognise classes of street furniture rather than describe the composition of street furniture. They do not take the meaning of components into account. For instance, a streetlight with an attached traffic sign can be recognised as a streetlight at the object level. It can also be interpreted as multiple connected parts: a pole connected to a traffic sign and a streetlight. One example is as shown in **Figure 4.1a**. Traditional segmentation obviously is not detailed enough to describe such types of road furniture. It is more appropriate to segment this piece of road furniture into different parts as shown in **Figure 4.1b**. For this reason, approaches with generic segmentation, detection and classification are restricted. For a better understanding of street furniture, decomposition needs to be investigated. This chapter combines basic segmentation methods with feature constraints to decompose street furniture into meaningful parts. The result of this chapter is that points on individual poles are grouped into one component, and points are grouped for each attached objects such as traffic signs and traffic lights. The assumption is that street furniture is already detected in a previous step.



(a)

(b)

Figure 4.1*: A piece of road furniture with multiple classes. (a) The point cloud of one piece of road furniture. (b) The decomposed road furniture.*

The remainder of this chapter is organised as follows. **Section 4.2** provides a review of related work on 3D point cloud segmentation and 3D mesh segmentation techniques. In **Section 4.3**, we explained our three-stage decomposition method. We introduce the test sites, analyse the experimental results, and give a comparison to other work in **Section 4.4**. The conclusions that are drawn and possible future work can be found in **Section 4.5**.

4.2 Related work

Much research has been carried out on relevant work of 3D data segmentation which can be categorised as 3D point cloud segmentation and 3D mesh segmentation. In this section, we reviewed the state of the art technologies on these two research topics.

4.2.1 3D point cloud segmentation

The related work of 3D point cloud segmentation is summarised as two types of frameworks: rule-based methods and machine learning-based methods. In rule-based methods, feature constraints are used to segment point clouds. For instance, region-growing and primitive fitting methods belong to rule-based methods. Machine learning-based methods segment point clouds based on extracted features in combination with optimization strategies.

Rule-based methods

Rabbani et al. (2006) present a smoothness constraint method to segment point clouds into parts. They firstly compute the normal and residual for every point. Then a region growing procedure is implemented based on local connectivity and surface smoothness. The method of segmentation can be seen in **Figure 4.2**. This method can be used for point clouds decomposition. However, due to the variety in the structure of objects, fixed distance neighbours are not representative enough for learning the features of the neighbourhood of every point.

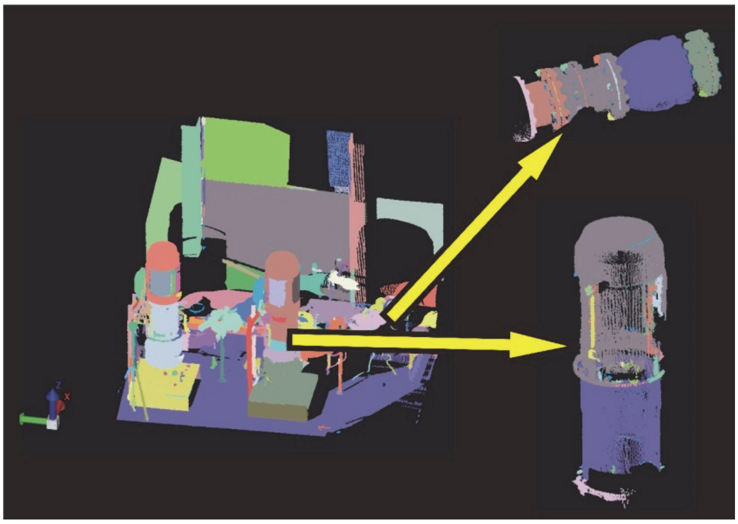


Figure 4.2*: Point cloud segmentation (****Rabbani et al., 2006****)*

Similar to the method proposed by **Rabbani et al. (2006)**, to overcome the difficulties in getting accurate and robust segmentation in the presence of outliers, **Nurunnabi et al. (2012)** propose a fast Minimum Covariance Determinant based approach to segment laser scanning 3D point cloud into several meaningful parts (**Figure 4.3**) based on robust PCA. At the first stage, they try to find the proper size of neighbourhoods. After that, the influence of noisy data is reduced, the robust saliency features are estimated, and the seed point of a region is selected. The next stage is to implement a region growing procedure. Finally, the object is decomposed into different components. This method provides a robust calculation of covariance statistics, normals and curvatures to support the estimation of saliency features. However, in some cases, this method cannot decompose poles and planes (**Figure 4.3**). **Nurunnabi et al. (2016)** utilised a robust diagnostic principal component analysis (RDPCA) in combination with a region growing algorithm to segment road furniture into different parts. However, the calculation of saliency features in these two methods strongly relies on a dense point cloud.

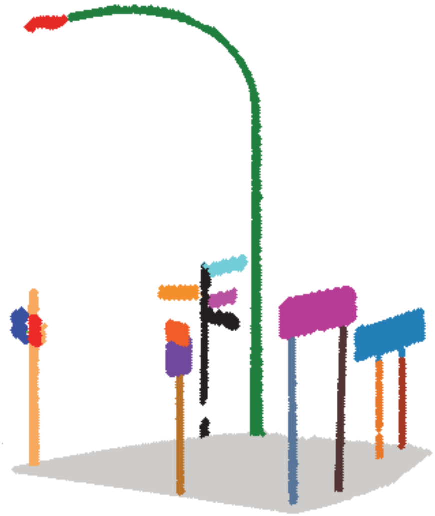


Figure 4.3*: Street furniture segmentation (****Nurunnabi et al., 2012****)*

Vosselman (2013) proposed a method to segment point clouds in urban environments. Various segmentation methods such as surface growing and connected components are combined to support the segmentation of point clouds in urban scenes. An example of classification can be seen in **Figure 4.4**. The author describes a post-processing approach to segment point clouds of large segments by combining segment merging and majority filtering.

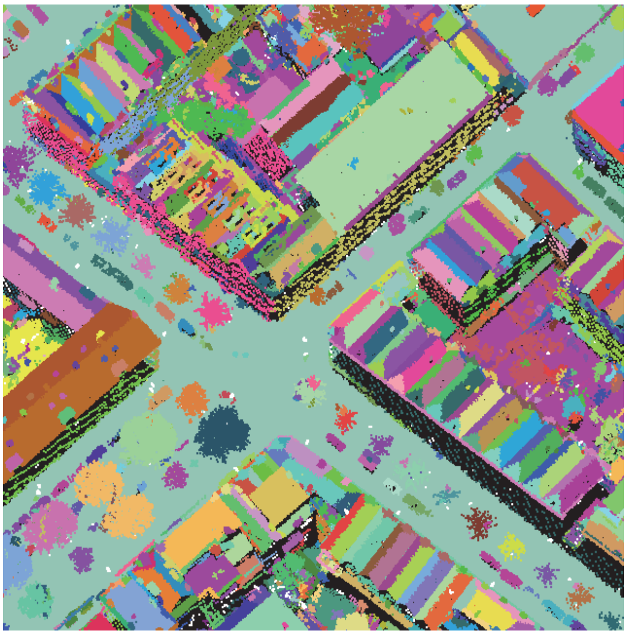


Figure 4.4*: Urban scene segmentation (****Vosselman, 2013****)*

Except for those above feature-based methods, there are many primitive fitting segmentation methods proposed by scholars. **Wani and Arabnia (2003)** describe an edge-based segmentation algorithm to divide range image into patches. **Vosselman et al. (2004)** review different techniques for surface extraction of point clouds. They describe various approaches for the segmentation of smooth planes, the extraction of planar surfaces and the recognition of specific shapes in point clouds such as cylinder and sphere (**Figure 4.5**). **Schnabel et al. (2007)** propose an automatic algorithm that can detect basic shapes in mess point clouds. They utilise RANSAC and normal estimation to recognize plane, sphere, cylinder, cone and torus. The proposed algorithm performs well enough to represent the 3D model. However, due to the restraint that it can only detect basic shapes of point clouds, the present method is not able to segment point clouds of objects with irregular attachments into meaningful components.

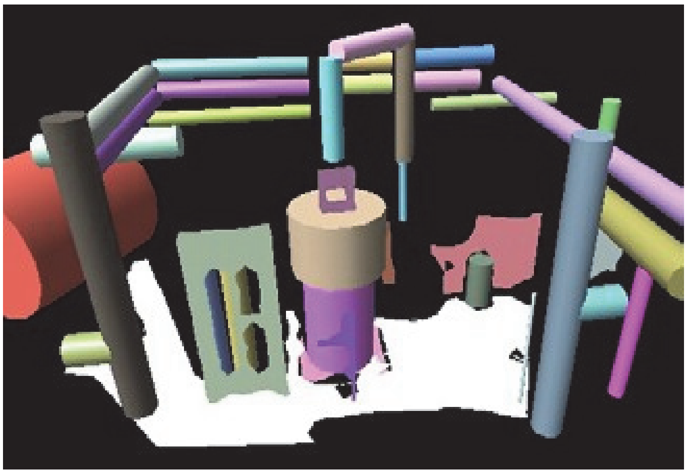


***Figure 4.5**: Planes and cylinders detection in an industrial scene (**Vosselman et al., 2004**)*

By analysing approximate convexity features, **Kaick et al. (2014)** present an innovative method to decompose point clouds into meaningful components. Firstly visibility between each point in the shape is computed. Then the shape is roughly decomposed into parts with weak-convex constraints by using the visibility information. At last, the similarity of adjacent parts is calculated based on geometric properties and adjacent similar components are merged to obtain the final segmentation result (**Figure 4.6**).

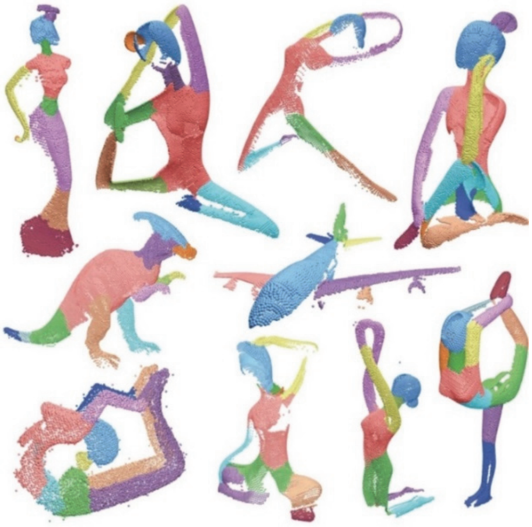


***Figure 4.6**: Point clouds decomposition with convexity features (**Kaick et al., 2014**)*

Machine learning-based methods

Dimensionality features of optimised neighbourhood size are used to segment 3D point clouds in the work of (**Demantke et al., 2011**). They present a min-cut based method to segment point clouds into components which represent different types of objects. A shape-based method is proposed to segment mobile laser scanning (MLS) point clouds into linear segments, planar segments and scattering (**Yang and Dong, 2013**).

Anguelov et al. (2005) use Markov Random Fields (MRF) and graph cuts to detect and segment complex objects from 3D laser scanning data). By combining an MRF-based algorithm and a maximum-margin approach to learn feature sets and their weights, the proposed method utilises graph-cuts and learned features to segment tens of millions of points into multiple object classes efficiently. With the contextually spatial balance of MRF, the present learning model is able to make use of complex surfaces and volumetric features. In addition, the authors compare the segmentation results of the proposed method with the SVM-based method and prove its superiority. However, this method relies on the local surface features of objects heavily and it is difficult to decompose objects with some other features such as linear features.

With coloured point clouds and their surface normal, **Strom et al. (2010)** present a graph-based method to segment 3D laser point clouds into different components (**Figure 4.7**). This method integrates colour information to segment point clouds. However, some MLS data do not contain colour information, and colour information is not strongly reliable because of the variation of light and exposure conditions.

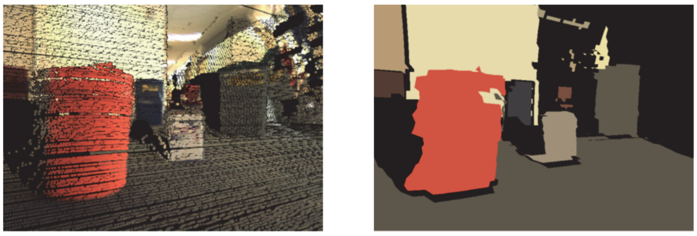


***Figure 4.7**: Coloured point clouds decomposition (**Strom et al., 2010**)*

Golovinskiy and Funkhouser (2009) present a min-cut based method to segment point clouds into different kinds of objects. Given an object location, the proposed method constructs a k-nearest neighbours graph with the point clouds of this object, assumes a background prior and adds hard foreground (and optionally background) constraints. Finally, they use the min-cut algorithm to compute the foreground-background segmentation. The flowchart is shown in **Figure 4.8**. This method can segment individual objects directly. However, it is not able to decompose these objects into meaningful parts. As a consequence, it cannot be used in street furniture detection.

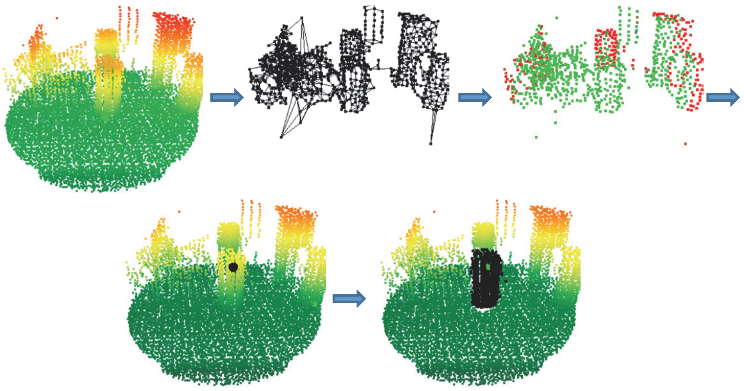


***Figure 4.8**: The flowchart of min-cut segmentation of point clouds (**Golovinskiy and Funkhouser, 2009**)*

In order to reconstruct polygonal structures, **Sedlacek et al. (2009)** propose a graph cut based point cloud segmentation method to separate surface elements from unorganised point clouds (**Figure 4.9**).

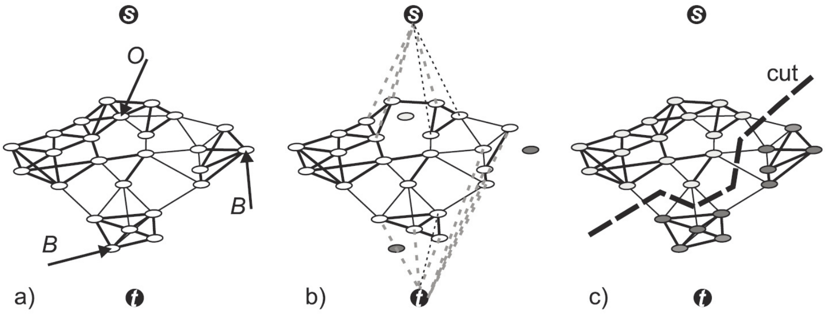


***Figure 4.9**: Graph-based decomposition method (**Sedlacek et al., 2009**)*

Deep learning methods as ascendant techniques have been rapidly developed and widely used for the segmentation of point clouds in recent years. Compared to traditional machine-learning methods, handcrafted features are not required with deep learning methods. **Qi et al. (2017)** proposed PointNet to semantically segment point clouds into different parts by adding a symmetric function. It shows strong performance on part segmentation of objects in point clouds. In addition, PointNet is robust to permutation and corrupted data. However, the length of the input of PointNet is consistent and not flexible. A deep KD-Net was proposed by **Klokov et al. (2017)** to segment objects in point clouds. KD-Net uses kd-trees rather than regular voxels to organise point cloud. For bigger 3D point clouds, the construction of kd-trees severely restricts the efficiency of KD-Net. Another data structure applied to coordinate point cloud is octree. O-CNN was constructed by **Wang et al. (2017b)** to segment point clouds of objects into semantic parts. The hash table was built to speed up the searching of the neighbourhood and the computation of 3D convolutions. **Yi et al. (2017)** built SyncSpecCNN to segment point clouds of objects to different parts. In SyncSpecCNN, the multi-scale kernel was used to aggregate information, and spectral transformer network was proposed to synchronize different spectral domains. SyncSpecCNN represents the state-of-the-art performance on part segmentation. SplatNet was built by **Su et al. (2018)** to segment objects in point clouds to different semantic parts. Lattice space was utilised to leverage spatial relations and calculated features.

4.2.2 3D mesh segmentation

Similar to part segmentation of objects in 3D point clouds, 3D mesh segmentation methods can be categorised as rule-based methods and machine learning-based methods. Rule-based methods are summarised as skeleton-based methods, fitting-primitives methods or hybrid methods. Method-learning methods consist of traditional machine learning methods and deep learning methods.

Rule-based methods

Skeleton-based methods have been widely proposed to segment 3D mesh into parts. Based on homogeneous geometric and topological features, **Li et al. (2001)** present a framework to decompose polygon meshes into components as sub-volumes of objects. They use a sweep plane to extract the skeleton from the 3D mesh (**Figure 4.10**). However, it is difficult for this method to decompose point clouds data into meaningful parts because the proposed method is restricted to polygon data.

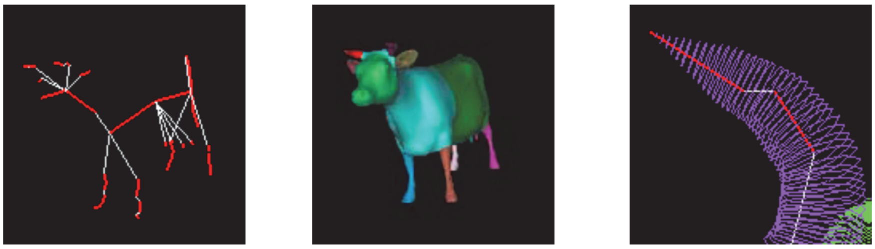


***Figure 4.10**: skeleton extraction and decomposition (**Li et al. 2001**)*

As the skeleton is an important clue for mesh decomposition, **Cornea et al. (2005)** proposed a hierarchical curve skeleton extraction method to segment 3D meshes including scattered points into meaningful components. They start by generating a force field inside the mesh by repulsive force functions. Then the curve-skeleton is extracted based on the core skeleton. According to the skeleton, 3D point clouds are segmented into different parts (**Figure 4.11**). However, the skeleton extraction method performs well on the segmentation of articulated objects. For non-articulated objects such as planes connected to poles, it is difficult to extract the skeleton in this method. Therefore, it is challenging to decompose the planar segments connected to other objects by this skeleton extraction method.

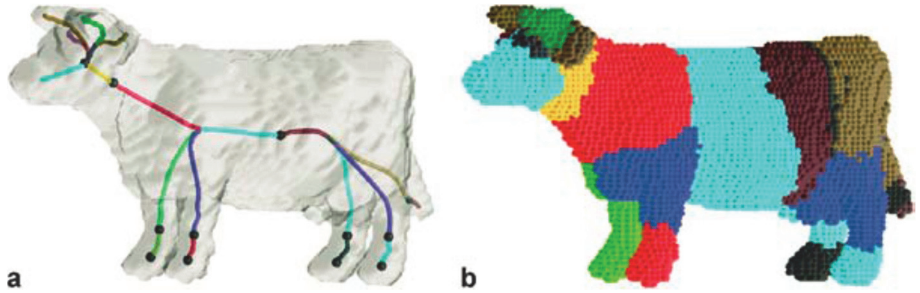


Figure 4.11*: Curve skeleton based decomposition (****Cornea et al., 2005****)*

Lien et al. (2006) and **Lien and Amato (2007)** present early work to decompose 3D meshes by using approximate concave and convex parts. **Lien et al. (2006)** present an approach to do shape decomposition and extract a skeleton simultaneously. They use approximate convex decomposition methods to extract components. At the same time, they use the principal axis of decomposed components to construct the skeleton. After that, the refinement of the skeleton quality is implemented based on the measurement of convexity. Finally, the refined decomposed components and the skeleton are generated. Besides 2D polygons, this framework could also be general enough to work for 3D polyhedral. In contrast to approximate convex decomposition, the proposed approach is able to extract the main skeleton of a model. However, concerning the topological relationship between decomposed components, the result of this method is not accurate. Based on the approximate convex decomposition algorithm, **Lien and Amato (2007)** update the decomposition of polyhedral by adding approximate feature grouping constraints that enable sets of similar and adjacent features to be merged together. Firstly, the concave features are computed by adding some extensions based on region growing rules. Afterwards, the global cuts are extracted by feature grouping. At last the concave features are resolved and the model is split into different components. This method is able to generate tiny and good quality decompositions even for some complex models. But this method has the limitation that some uncommon types of open surfaces on arbitrary genus with a great majority of convex vertices cannot be segmented correctly and some models with non-linearly separate features cannot be decomposed successfully.

Another commonly used method for 3D mesh segmentation is primitive fitting methods. The method of **Attene et al. (2006)** attempts to segment triangular meshes using a hierarchical face clustering algorithm based on fitting primitives belonging to an arbitrary set. In this method, they decompose the 3D triangular meshes into basic shape elements such as planes and cylinders that represent parts of the 3D triangle meshes. Segmentation with different levels is also adopted within this method. However, not all the segmented basic elements are meaningful. For instance, **Figure 4.12** shows that the hand can be decomposed into several parts and the palm is segmented into two different basic shapes. Also, the proposed hierarchical method is not adaptive for decomposition tasks with irregular shapes.

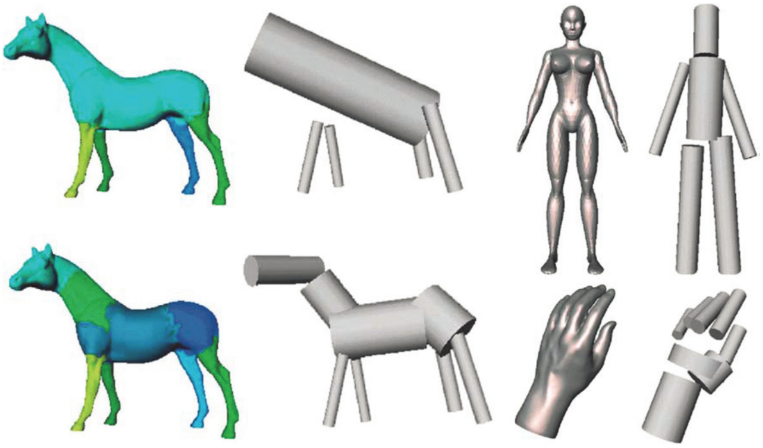


Figure 4.12**: Mesh decomposition by fitting primitives (**Attene et al., 2006)

On the basis of a generic tubular model, **Mortara et al. (2004)** proposed a specific shape segmentation method to detect tubular shapes in 3D objects. In this method, the 3D mesh is decomposed into tubular parts and the main body (**Figure 4.13**). Although this approach builds upon the 3D triangular mesh, it provides a novel way of decomposing some tube objects such as street furniture.

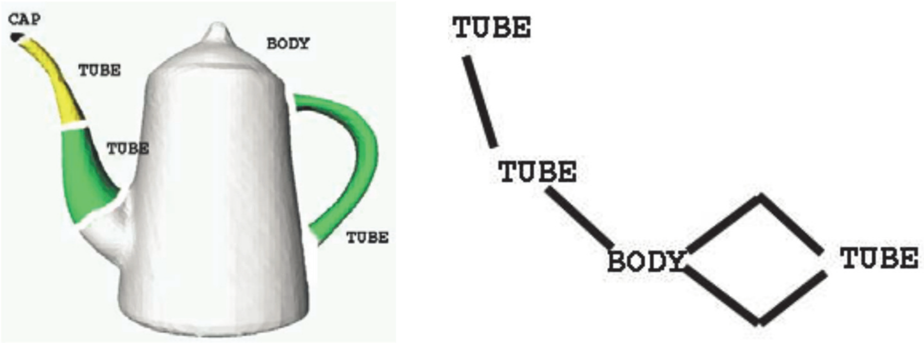


Figure 4.13*: Tubular segments and the main body (****Mortara et al., 2004****)*

The work of **Aleotti and Caselli (2012)** aimed at segmenting 3D shape into parts for robot grasping based on the Reeb graph. At first, they use the Reeb graph to decompose 3D object model. Then the extracted graph is matched with a graph database, and the object model is recognised. However, the graph may vary a lot. It is difficult for the graph database to cover every object's model.

Apart from those two types of methods, there are other proposed rule-based methods. Among the geometric features of shapes, symmetry is fundamental. According to the symmetry property of objects, **Podolak et al. (2006)** propose a plane-reflective symmetry transform method to segment 3D meshes. This method decomposes a mesh to make faces of each segment with the same distinct symmetries. However, this method cannot separate the object into meaningful parts accurately because those parts which have the same distinct symmetries may belong to different components. For instance, the hand in **Figure 4.14** is decomposed into four parts, and two fingers are not distinguished clearly.

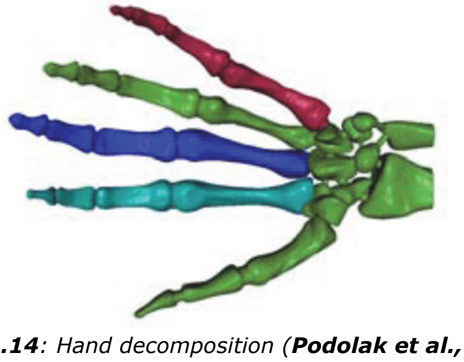


Figure 4.14*: Hand decomposition (****Podolak et al., 2006****)*

Inspired by spectral and contour analysis, **Liu and Zhang (2007)** propose a segmentation algorithm to decompose 3D meshes into useful components.

Firstly, the 3D mesh is projected into the max area surface, and the 2D contour is extracted. The projection relies on the graph Laplacian and the geometric operator which emphasizes concavity. Then the 2D contour is analysed and segmented. After that, 2D segmentation samples are mapped to the faces of the 3D mesh respectively. Finally, according to the salience score, the mesh is cut into different parts.

Katz et al. (2005) present a hierarchical mesh segmentation method to decompose 3D mesh into semantic parts. The method works based on salient feature points and core construction. Firstly they transform the mesh into a representative mesh and compute its convex hull. Then they find points with prominent features in this mesh. Afterwards, they extract the core from the mesh. Finally, they segment the mesh and refine the segmentation to decompose the mesh into meaningful parts. The present algorithm is pose-invariant to the model and can extract tiny segments. And it performs well with articulated objects such as human and animal 3D mesh because joints of articulated objects are very clear. However, when there are not shapes without few prominent features such as planes in a 3D mesh, it will be difficult for the method to segment such a mesh into meaningful components.

Huang et al. (2009) propose a method that segments deformable shapes into meaningful components by using modal analysis. At the first stage, they construct a set of typical deformation models by computing the Hessian. Then they implement a rough initial segmentation. Afterwards, boundaries between patches are optimised by using discriminating deformations. At last, they use a hierarchical refinement strategy to optimise the decomposition result. However, this method was mainly tested on some articulated models such as horse and elephant models. These models have prominent skeletons. Therefore, the proposed method may not be suitable for other models such as street lamps and man-made tools.

The work of **Attene et al. (2009)** aims at segmenting 3D shape into meaningful parts for semantic annotation. They innovatively propose a knowledge-based shape annotation by combining different segmentation algorithms to decompose 3D shapes. Their research provides a promising approach to deal with mesh decomposition by a knowledge-based method.

Machine learning-based methods

The work of **Katz and Tal (2003)** aims at decomposing the mesh into meaningful parts by using fuzzy clustering and cuts. It starts from calculating distances between all the faces in the mesh. Then it implements the initial decomposition, and a probability of belonging to each patch is assigned to each face. Afterwards, a fuzzy decomposition is computed by refining the probability values using an iterative clustering scheme. Finally, the fuzzy decomposition is transformed into the final decomposition by constructing clear boundaries between each component (**Figure 4.15**). This decomposition method is based on computing face distances. However, the point cloud is discrete. Therefore this approach cannot be used for point cloud decomposition. Small concavities result in over-segmentation.

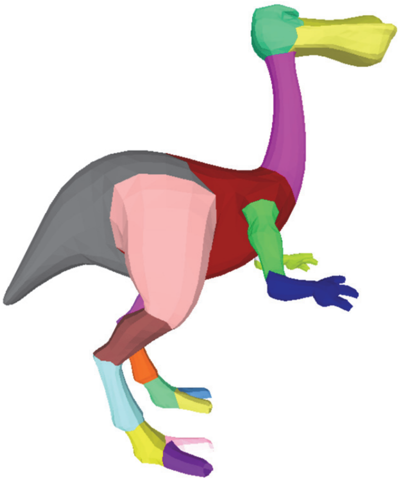


Figure 4.15*: Decomposition result (****Katz and Tal, 2003****)*

Golovinskiy and Funkhouser (2008) investigate a novel shape analysis method based on random cuts of 3D surface meshes. They first construct an edge-weighted graph to represent the 3D mesh. Then a random set of K-way segmentations are generated including K-means, Hierarchical clustering and the Min-cuts algorithm. After that, they estimate the partition function of each edge. At last, they calculate the cut consistency and decompose the mesh into components (**Figure 4.16**). This method combines several algorithms, and it is robust to noisy data. Therefore, this type of method is promising in the decomposition of point clouds with a large amount of noise.

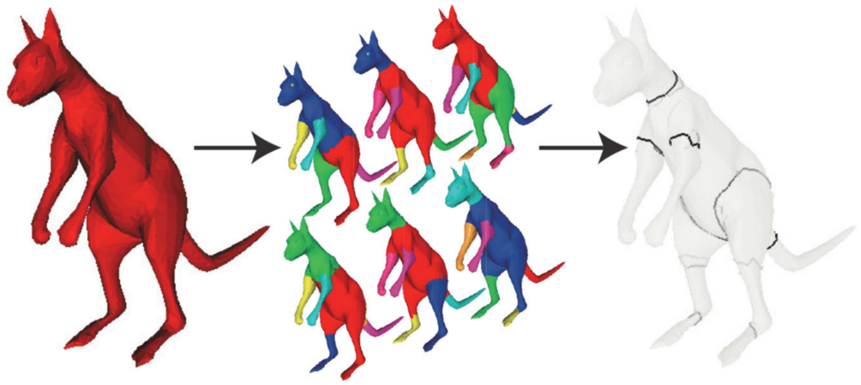


***Figure 4.16**: The decomposition kangaroo model (**Golovinskiy and Funkhouser, 2008**)*

In order to get a better decomposition result of models, **Kalogerakis et al. (2010)** present a data-driven approach to label different parts of 3D meshes. They use the jointboost classifier with unary features to obtain initial predictions. Then they train a CRF model which takes advantages of contextual features to refine the initial results and get the final consistent prediction. From visual inspection, this method has achieved good results of 3D mesh decomposition. An example of aeroplane decomposition can be seen in **Figure 4.17**. The advantage of this method is that it can combine many classification features. However, 3D meshes vary a lot, and it is tedious and inconvenient for people to collect plenty of training datasets.

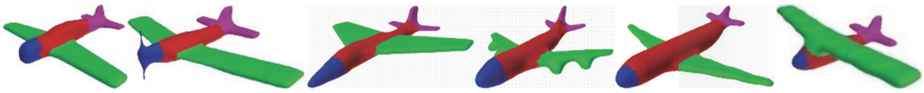


***Figure 4.17**: Aeroplane decomposition (**Kalogerakis et al., 2010**)*

Deep learning methods have been unfolding in the past years. Guo et al. (2015) constructed a deep convolutional neural network to label 3D mesh. Seven types of commonly used features were extracted and fed into the neural network. After the training process and the prediction of the neural network, a graph-cut algorithm is adopted to mitigate the inconsistency of the prediction of adjacent triangles. **Qi et al. (2017)** built two distinct network architectures, volumetric-based network and multi-view network, to segment 3D into semantic parts. Similarly, **Kalogerakis et al. (2017)** proposed a deep neural network to conduct 3D shape segmentation by projecting 3D models to 2D images from multiple views. **Yi et al. (2016)** developed an active learning framework to label 3D mesh. This framework incorporates human verification of predicted results in the optimisation stage to achieve accurate annotated 3D mesh. This piece of work is of great practical use for generating training data.

All in all, significant progress has been achieved on part segmentation of objects in both 3D point clouds and 3D mesh data. In terms of point density and the evenness, MLS data is not as equivalently good as ShapeNet Core dataset. Features designed in the aforementioned methods are not reliable for our research. Another point is that deep learning methods require massive training data. In our research, there is a limited amount of training data. There is difficulty with applying deep learning techniques before we collect and label enough training data.

4.3 Methodology

To address the problem that some road furniture has multiple functions, we presented a method to decompose road furniture into different components based on their spatial relations. We analysed the relationship between attachments and vertical and horizontal poles. Therefore, it is crucial to know which points belong to which pole. The remaining points have spatial relations to either horizontal or vertical poles. Based on these relations, we presented an optimal decomposition procedure. The overall workflow of this method is as shown in **Figure 4.18**. There is a large variety of structures and shapes in pole-like road furniture. It is difficult to extract all the poles using a single method. Therefore, we first identified some critical properties of the road furniture, based on which the most suitable pole extraction method was selected (**Section 4.3.1**). Next, the poles were extracted (**Section 4.3.2**). In **Section 4.3.3**, poles were removed, and the attached components were separated into different parts. The rule-based splitting and merging were carried out to refine the decomposition. The feedback of pole-like road furniture decomposition to detection was performed in **Section 4.3.4.** The evaluation of pole-like road furniture decomposition was introduced in **Section 4.3.5**.

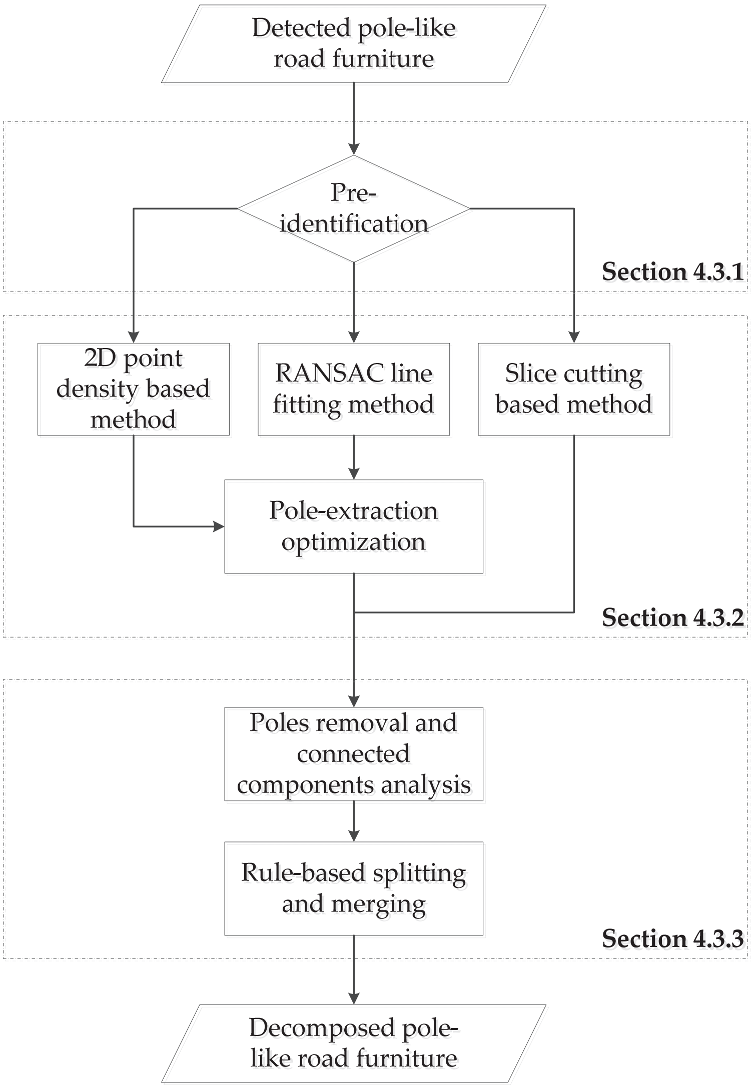

Detected pole-like road furniture
Pre-identification
Section 4.3.1
2D point density based method
RANSAC line fitting method
Slice cutting based method
Pole-extraction optimization
Section 4.3.2
Poles removal and connected components analysis
Rule-based splitting and merging
Section 4.3.3
Decomposed pole-like road furniture

***Figure 4.18**: The flow chart of pole-like road furniture decomposition*

4.3.1 Pre-identification

The assumption is that before the pre-processing, street furniture is already detected in a previous step (the previous chapter). The input data is the connected components above the street level. In order to make the decision for the selection of the most suitable pole extraction method, at the first stage, street furniture is pre-processed to get pre-knowledge such as the height and the width. To understand the structure of the street furniture, we cut the component into slices. The width and the number of points are computed for every slice. The height, the maximum distance in XY direction and the number of poles can now be determined. When the maximum distance in XY direction is large (the threshold is 6.0m), horizontal poles should be included in the street furniture. Based on these features, the corresponding pole extraction method will be decided.

The pole extraction method is selected, as shown in **Figure 4.19**. When there are components attached to more than half of the height of the pole, the 2D point density-based method is utilised. When there are vertical and horizontal poles, the RANSAC line fitting method is used to extract poles. For the rest of the cases, the slice cutting based method is adapted to extract the main poles.

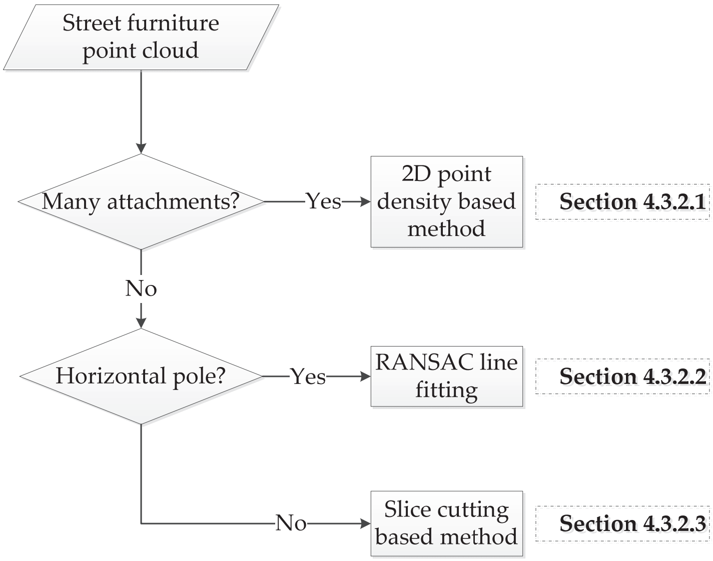


Figure 4.19*: The selection of pole extraction methods*

4.3.2 Pole extraction

In general, pole-like street furniture consists of main poles and other components. These poles are the main link to other components. If the main poles of street furniture are removed, other components will be separated into different parts. At this stage, main poles of street furniture are extracted. Three types of pole extraction methods, the 2D point-density based method, the RANSAC line fitting method and the slice cutting-based method, are introduced in this section. Then an optimised pole extraction is proposed specifically for the first two methods.

4.3.2.1 2D point-density based method

When the pole is vertical, 2D point density of the pole is much higher than other parts when street furniture is projected onto the XY plane, as shown in

Figure 4.20. Based on this property, the 2D point density-based method is proposed. This method can be described in two steps, as shown in **Figure 4.21**.

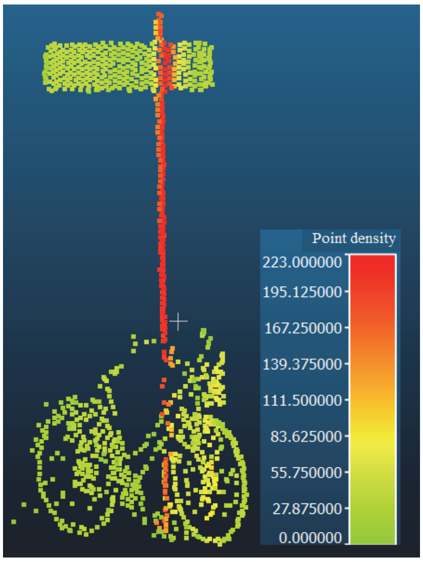


Figure 4.20*: 2D point density of street sign*

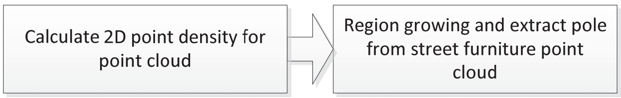

Calculate 2D point density for point cloud
Region growing and extract pole from street furniture point cloud

Figure 4.21*: The workflow of the 2D point density-based method*

To compute 2D point density, we project the point cloud of street furniture point onto the XY plane. Before the calculation of the number of neighbouring points, the 2D distance for searching neighbouring points has to be defined. The 2D distance for searching neighbouring points is obtained by cutting street furniture point cloud into slices and calculating their widths. After that, the number of neighbouring points is calculated by counting the number of points within this circle (as shown in **Figure 4.22a**). The number of 2D neighbouring points around the point is its 2D point density value.

Concretely, in the pre-processing stage, the median width w_{median} of all the cutting slices can be calculated. Then street furniture point cloud is projected onto the XY plane. Based on the distance for searching 2D neighbouring points *dist_neigh_2d* (equal to w_{median}), 2D point density *point_density* is calculated for every point.

After obtaining the point density for the point cloud, region growing is constrained by one attribute of points, 2D point density. By the region growing algorithm, points with high 2D point density are extracted and selected as the points of a pole.

Specifically, the seed point is selected as the point which has the highest 2D point density value. Then we compare the 2D point density between the seed point and its neighbouring points. In this research, the nearest neighbouring points are defined by 2D distance. If the 2D point density of the seed point's neighbouring points is more than 85% of the seed point, this neighbouring point will be categorized as a seed point. This procedure is repeated until all the seed points are found, as shown in **Figure 4.22b**. These high 2D point density points are labelled as the pole. After that *dist_neigh_2d* is used to restrict the region of growing and refine the extracted pole. The extracted pole of a street sign is as shown in **Figure 4.23**.

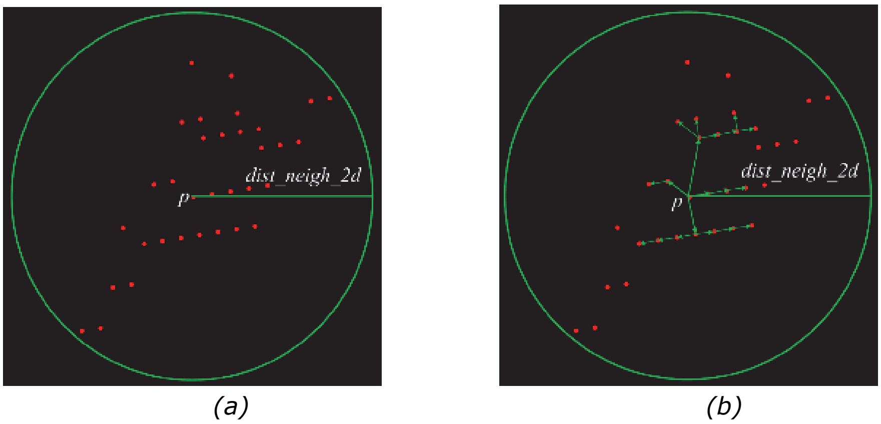


***Figure 4.22**: The region growing process. (a) The neighbouring points of point p. (b) Seed growing*

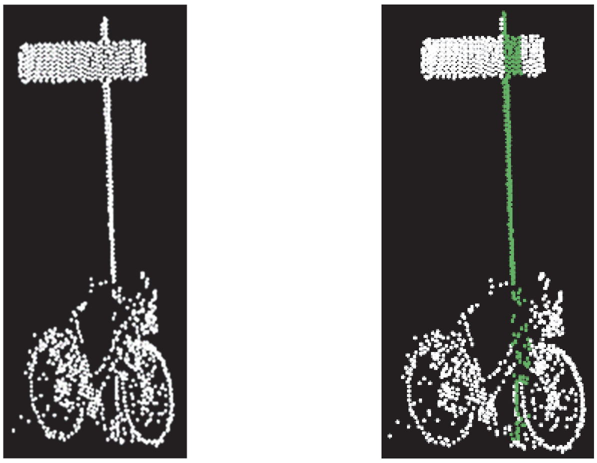


***Figure 4.23**: The left figure is the original point cloud, the right one is the main pole extracted (green points)*

4.3.2.2 RANSAC Line Fitting Method

When there are not many attachments connected with main poles, pole-like street furniture objects have main poles which can be approximated by lines. Based on this, we propose the RANSAC line fitting method to extract both vertical and horizontal poles, which includes three stages.

Geometric features consist of linear, planar and volumetric features. When the neighbourhood size is large, pole points have high linearity feature value.

In order to extract these points with high linearity values, geometric features are calculated for every point by using eigenvalues. Eigenvalues are computed with every point neighbourhood. In this chapter, linear, planar and volumetric features are defined as:

$$f_{1D} = \frac{\sqrt{\lambda_1} - \sqrt{\lambda_2}}{\sqrt{\lambda_1}}, \; f_{2D} = \frac{\sqrt{\lambda_2} - \sqrt{\lambda_3}}{\sqrt{\lambda_1}}, \; f_{3D} = \frac{\sqrt{\lambda_3}}{\sqrt{\lambda_1}} \tag{4.1}$$

$$P_G = \text{argmax}_{d\in[1,3]}(f_{dD}) \tag{4.2}$$

where f_{1D}, f_{2D}, f_{3D} are the linear, planar and volumetric geometric features respectively; λ_1, λ_2, $\lambda_3 \in R$ ($\lambda_1 \geq \lambda_2 \geq \lambda_3$) are the three eigenvalues that correspond to an orthogonal system of eigenvectors, P_G is the geometric feature of the point (**Yang et al., 2015**)

Single neighbourhood size can partly represent geometric features of points. By using a multi-scale neighbourhood to extract features, the information across different scales could be included (**Weinmann et al., 2015**). For sign points, when the neighbourhood size is small, they have high planarity values. When the neighbourhood size is large, they will become points with high linearity values. Short pole points connected with lamps have high linearity values if the neighbourhood size is small. When it is large, they will become points with high planarity or volumetric values. For these reasons, multi-neighborhood size is utilised to calculate geometric features for every point. In this step, a small neighbourhood size *n_small* is used for extracting linear points and excluding sign points by Equations 1 and 2. Simultaneously, a large neighbourhood size *n_large* is used for eliminating short pole points. Here linear points are points with high linearity values. If both points have larger linearity values in both neighbourhood sizes, these points are labelled as linear points. Linear points extraction is shown in **Figure 4.24**.

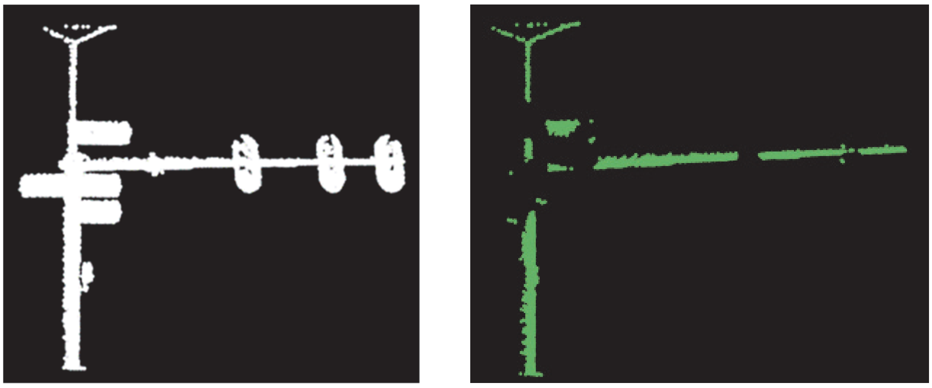


***Figure 4.24**: The extraction of points with high linearity (the left figure is the original point cloud, the right one shows linear points)*

As linear points are the main component of these points extracted from the previous step and there are not many lines with these points, the RANSAC algorithm can be utilised for extracting lines from these points. With only RANSAC algorithm, lines can be incorrectly extracted. Therefore some other constraints are added. In general, street furniture has long poles. Thus, the length of poles must be over a threshold *length_pole*.

After all the linear points are extracted, we use line fitting (RANSAC) and feature constraints to fit the pole and find the main pole points. Once the line is found in the point cloud, points around this line are deleted. This step is iterated until no line can be found in the point cloud. An example of extracted lines is shown in **Figure 4.25**. The left figure shows points with high linearity, and the right one shows extracted lines.

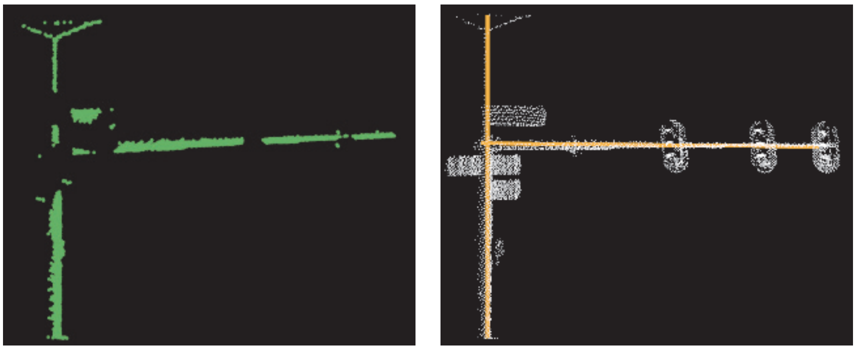


***Figure 4.25**: Extracted lines using RANSAC*

In order to extract points belonging to poles from the point cloud of street furniture, a width is proposed to be with the extracted line to capture those points around the line. In this step, distances between the extracted line and its surrounding points are calculated. If the distance is smaller than the width threshold of this line *diameter_line*, this point will be labelled as a point of the pole. An example of extracted poles is shown in **Figure 4.26**. The left figure shows the extracted line, and the right one shows the extracted poles of street furniture.

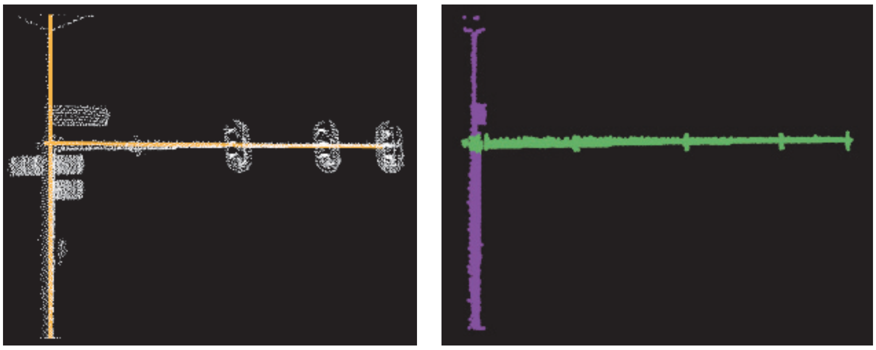


Figure 4.26*: Extracted poles of street furniture*

4.3.2.3 Slice Cutting Based Method

For both vertical and slanted pole-like street furniture especially thick poles that the two previous methods cannot extract, centre points of these poles can be fitted by straight lines. According to this rule, a four-step method is proposed to decompose pole-like street furniture.

Firstly, in the pre-processing step, the width of every slice of street furniture has been computed. For every slice, its centre point is calculated, as shown in **Figure 4.27**. After that, a set of width values $W=\{width_1, width_1, \dots, width_n\}$ and centre points $P\_CENTRE=\{P_1, P_2, \dots, P_n\}$ can be generated.

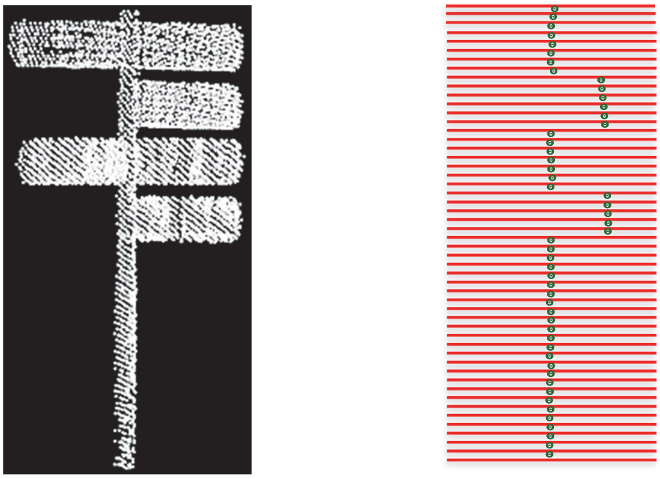


Figure 4.27*: The original point cloud of (left figure) one piece of street furniture is cut into slices, and their centre points (green points in the right figure) are obtained from these cut slices (interstices between red lines in the right figure)*

For width values *W={width₁, width₁, … , widthₙ}*, there are some large changes between widths because of attached components such as street signs. In this step, these widths with large changes will be detected according to their neighbouring widths and median width value. From the bottom slice of street furniture, the width of every slice should not be much higher than the previous one or the median value. If the change is higher than 0.05m, the width of this slice will be changed to the smaller width value of the previous slice. Based on this, the width of every slice is smoothed, as shown in **Figure 4.28**. The blue bins are the original width of every slice, and the red bins are the width of every slice after refining. The red bins are the buffer widths used to extract pole points with the centre line.

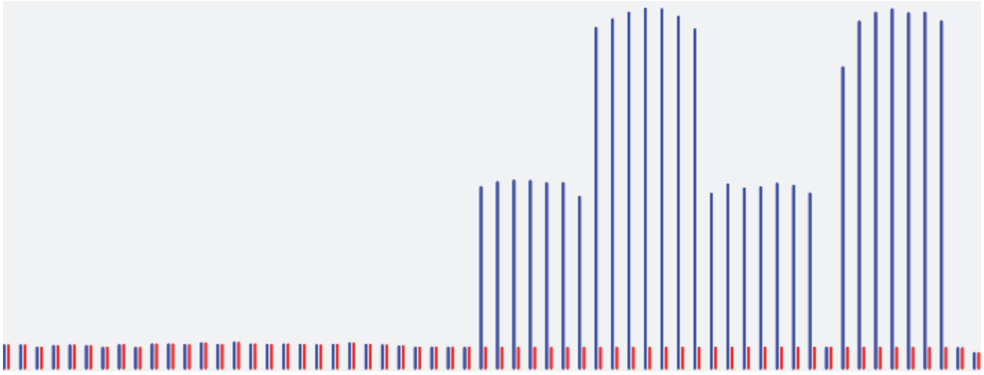


Figure 4.28*: The width of every slice before (blue) and after refining (red)*

At the first stage, centre points of street furniture are extracted. In the second step, some large width changes of slices are detected. In this stage, these centre points, with which there are no big width changes, are grouped to fit lines by using the RANSAC algorithm. The extracted centre points and the centre line is shown in **Figure 4.29**.

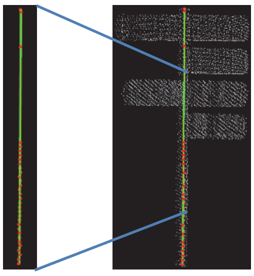


Figure 4.29*: The extracted centre points and the centre line of street furniture*

In order to extract poles from street furniture, all the extracted lines are given widths. Specifically, use every slice and its respective refined width in the extracted line to capture those points around the line. In this step, calculate distances between the extracted line and its surrounding points. If the distance is smaller than the width threshold of the respective slice *diameter_slice*, this point will be labelled as a point of the pole. The extracted pole is shown in **Figure 4.30**.

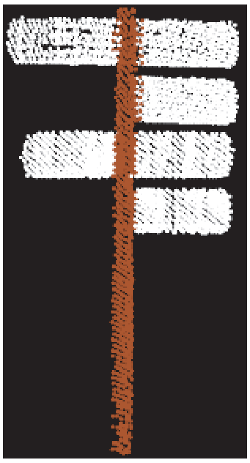
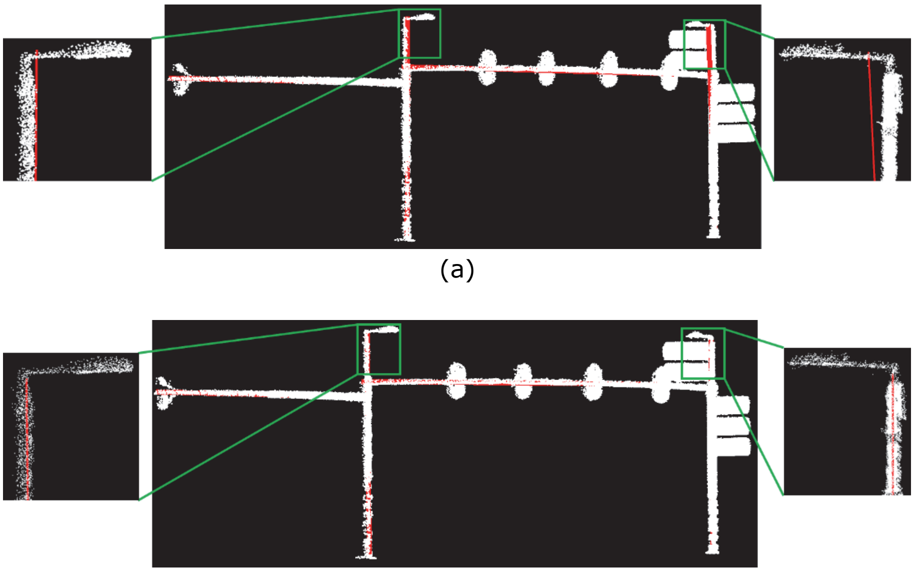


Figure 4.30*: The extracted pole of street furniture (brown points)*

4.3.2.4 Pole extraction optimisation

Poles extracted by the first two methods (2D point density method and RANSAC line fitting method) were often not accurate. For the 2D point density-based method, points of attachments that are near the poles have a high 2D point density. They can thereby be detected as poles. When points with high linearity were not extracted accurately enough, the centre lines of the poles cannot be estimated correctly using the RANSAC algorithm in the second method. If a pole is inaccurately generalized to a line, this causes the imprecise extraction of the pole. For example, if the extracted centre lines inclined towards the street side (**Figure 4.31a**), the part of the points that would be far away from the centre lines and belong to the poles would not be extracted in this stage. To tackle these problems, we optimised the pole extraction by re-estimating the centre lines of the poles. Similar to the slice cutting, we cut the extracted points of poles into slices, computed their centre points, and used the RANSAC algorithm to extract the lines from these centre points. By doing this, poles can be extracted more accurately by using the precisely fitting centre lines. The re-estimated centre lines were shown in Figure **4.31b**.



(a)

(b)

Figure 4.31*: The extracted centre lines of poles. The red lines are the extracted centre lines of the pole. (a) The extracted centre lines were inclined towards the street side and (b) the optimized centre lines.*

4.3.3 Decomposition into poles and attachments

After all the pole points are extracted, street furniture is decomposed into poles and components attached to poles. At this stage, extracted poles are removed from the street furniture point cloud. Then the connected component analysis is applied to the rest of the points. By doing so, street furniture point cloud can be decomposed into the main poles and different components. Since components may be close to each other, the neighbourhood size and the maximum distance for connection analysis should be considered carefully. Considering the distance between scanlines and points distribution on a single scanline, we assume that the neighbourhood size cannot be too small. Otherwise, points on different scanlines cannot be connected. The maximum distance between two scanlines here is 0.05m. For a point, the number of its neighbouring points is 10 within the distance of 0.05m in a single scanline. Here we set the maximum for connected components to 0.15m, and the neighbourhood size to 15. Examples of street furniture decomposition of each pole extraction method are shown in **Figure 4.32**. **Figure 4.32a** shows the partitioning of the remainder of the points which were obtained from the 2D point density-based pole extraction method. Pole extraction in **Figure 4.32b** uses the RANSAC line fitting method and pole extraction in **Figure 4.32c,d** utilised the slice cutting based method.

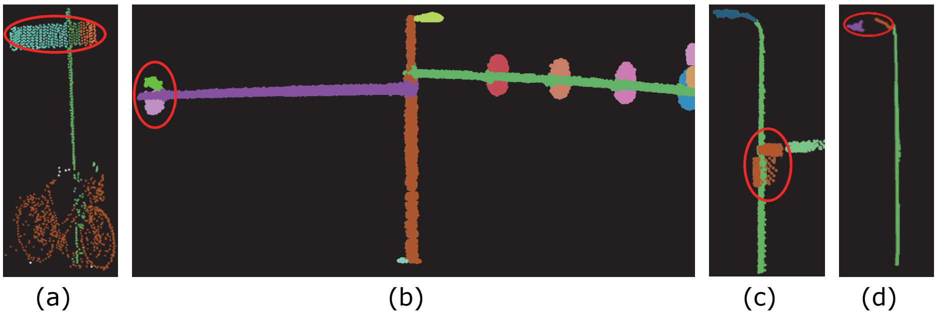


***Figure 4.32**: Examples of the separated components. The separated components after (a) the 2D point density-based pole extraction, (b) the RANSAC line fitting based pole extraction and (c, d) the slice cutting based pole-extraction.*

Because of occlusion, low point density, and noisy points, it is sometimes difficult to connect all the points that belong to the same attachment by leveraging the parameters of the aforementioned connected component analysis. Examples are marked in **Figure 4.32**. To address this problem, we defined a set of generic rules to split and merge the components. We first performed merging rules for the components for horizontal poles. Then the components connected to vertical poles were analysed to see whether they could be merged or split. Then, the detached components were checked on whether they could be merged with their nearby components.

As shown in the red circle of **Figure 4.32b**, one component attached to a horizontal pole can be separated into two parts during the connected component analysis. If two components were attached to the same horizontal pole and their positions overlap in XY-plane, these two components should be merged. The merging analysis of the horizontal pole attached components was repeated until no such components could be found. **Figure 4.33** shows the merged components after applying the merging rule.

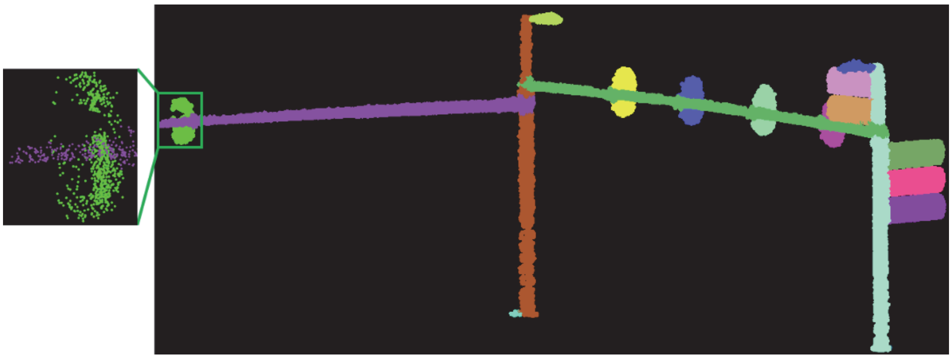


***Figure 4.33**: The left frame is the zoomed in image of the merged component after applying the merging rule.*

Another situation was that the attachments might be connected to each other because of noisy points or imprecise pole extraction. An example is shown in **Figure 4.32c**. In this situation, we increase the width of the pole extraction. If one component can be split into two parts and they are not at the same height, this component was split into two parts. This method is similar to erosion followed by dilation.

Suppose $\boldsymbol{C}_{verti}(\boldsymbol{P}_w)$ is a component which is attached to vertical poles when the width is set to $\boldsymbol{w}$ for pole extraction. We increased $\boldsymbol{w} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_n\}$—where $\boldsymbol{w}_i$ is the width of the cut slice $\boldsymbol{i}$—and performed a connected component analysis. Once $\boldsymbol{C}_{verti}(\boldsymbol{P}_w)$ becomes the two components— $\boldsymbol{C}_{verti}(\boldsymbol{P}_{w'})_i$ and $\boldsymbol{C}_{verti}(\boldsymbol{P}_{w'})_j$—and they are not at the same height, they should be separated into two parts. This operation will continue until the increment of $\boldsymbol{w}$ reaches a predefined value. In this chapter, it is empirically set to 0.15 m for the splitting analysis.

There are single attachments which might be separated into two parts (**Figure 4.32a**). This case occurs when the distance for connection is not large enough during the connected component analysis. In this case, we conducted a merging analysis. We continued increasing the width of the pole extraction to $\boldsymbol{w}''$. Here, we set $\boldsymbol{w}''$ to be the normal size of traffic signs, for example, 0.5 m. Once the two components $\boldsymbol{C}_{verti}(\boldsymbol{P}_{w''})_i$ and $\boldsymbol{C}_{verti}(\boldsymbol{P}_{w''})_j$ are wrapped by their connected vertical pole and once the two components are in the same height, the connection line of their center points comes very near to the pole line. Then they should be merged together. The merging analysis will be iterated until no components can be merged. The components were split and merged as shown in the red circle of **Figure 4.34**.

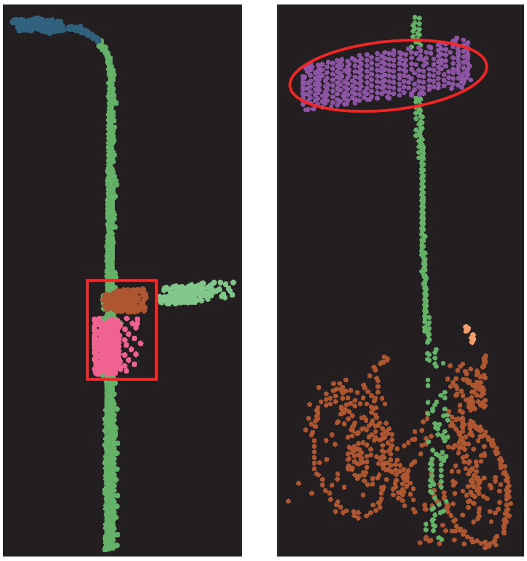


Figure 4.34*: The components connected with vertical poles are split (left) and merged (right) after applying the fitting rules.*

Components that were not connected to poles (**Figure 4.32d**) were described as detached components because of the low point density or occlusion. It is assumed that if one detached component and an attachment were at the same height, on the same side, and their connection line was very close to this pole, they should be one component and merged together. Then, this merged component was added to continue with the merging analysis of the detached components. The detached components were merged as illustrated in **Figure 4.35**.

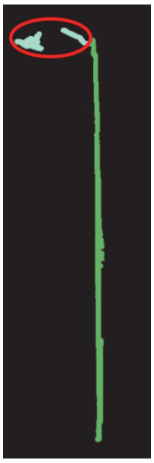


Figure 4.35*: The detached components were merged into one component (red outlined area).*

4.3.4 Final detection of pole-like road furniture

The results of the decomposition were imported as it is feedback for the road furniture detection stage. If there was no pole extracted from a road furniture item such as the pillars of buildings, this road furniture item would not be labelled as pole-like. For example, the pillars of buildings can be detected as a pole-like road furniture item at the detection stage. When such a pillar is decomposed by using RANSAC line fitting, this pillar will probably not be extracted because the percentage of points which belong to the pole is low when there are many points with high linearity from the edges of façades or fragments.

4.3.5 Road furniture decomposition evaluation

In this section, an approach was presented to evaluate the accuracy of the decomposition. Based on their spatial relations, the components of pole-like road furniture were manually labelled. For example, street signs can be manually labelled as "pole and attachments", as shown in **Figure 4.36**.

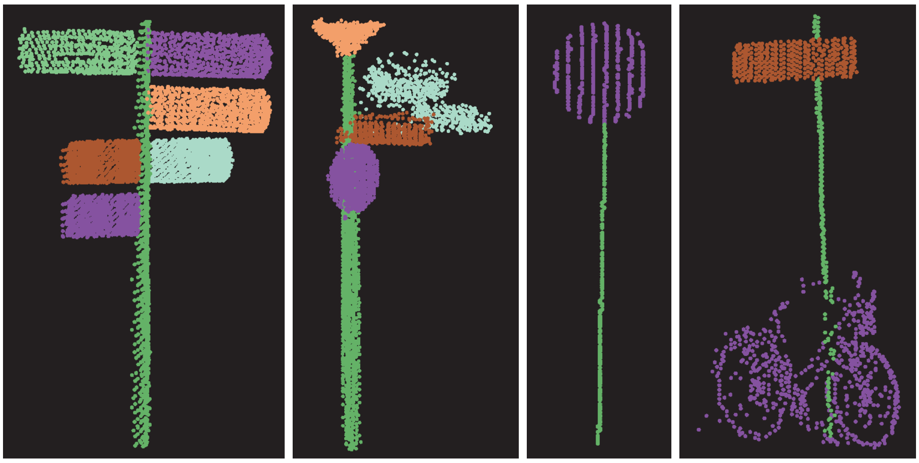


Figure 4.36*: Examples of reference data.*

In order to assess the results of the decomposition, we used a two-level evaluation method. One is the point-based evaluation, and the other one is a component-based evaluation. We used completeness and correctness to quantify the assessment.

In the point-based evaluation, we first selected the corresponding true positive points of the manually labelled ground truth component. We matched the corresponding decomposition result and every ground truth component by selecting the largest decomposed component in this ground truth component. The true positive points in an attachment are the points belonging to both this attachment and its corresponding manually labelled attachment. Then the completeness of every component was computed as the ratio of the number of points of the largest segment in this component to the number of points of this component, $\boldsymbol{Completeness} = \boldsymbol{TP}/(\boldsymbol{TP} + \boldsymbol{FN})$. $\boldsymbol{TP}$ is the number of correctly decomposed points in the manually labeled components, $\boldsymbol{FN}$ is the number of incorrectly decomposed points in the manually labeled components. For example, in **Figure 4.37**, the true positive component of the streetlight head is the decomposed component which is marked by the red circle 1 in the left figure. Thus, the completeness of this component can be computed as the ratio of the number of points which are in the overlap of the component labeled with the red circle 1 in the left figure and the component labeled with green circle 1 in the right figure, to the number of points of the component labeled with the green circle 1 in the right figure.

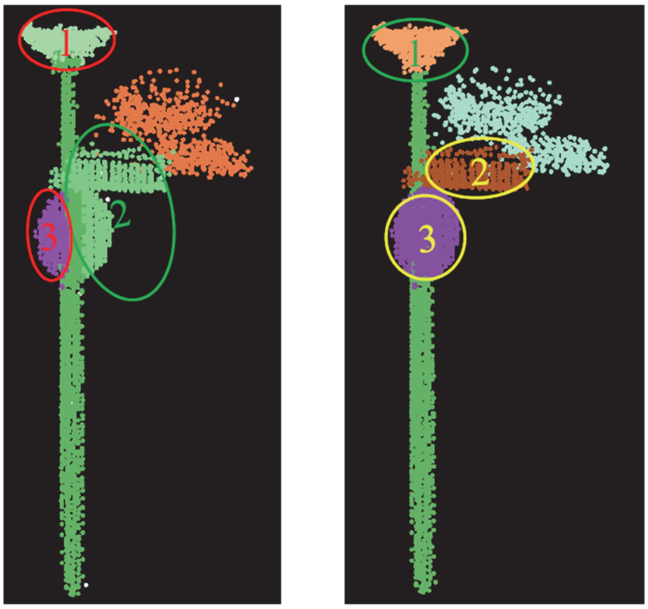


***Figure 4.37**: The decomposed components (left) and manually labelled components (right).*

The correctness of a separated attachment is the number of true positive points divided by the number of points of this separated attachment. For example, in **Figure 4.37**, the correctness of this component is computed as the ratio of the number of points which are the overlap of the component marked with the red circle 1 in the left figure and the component marked with the green circle 1 in the right figure, to the number of points of the component marked with a green circle 1 in the right figure. $\boldsymbol{Correctness} = \boldsymbol{TP}/(\boldsymbol{TP} + \boldsymbol{FP})$. $\boldsymbol{FP}$ is the number of incorrectly decomposed points in the components produced by our algorithm. Those points which are labeled with the green circle 1 in the right figure are included as well in the segment which is labeled with the red circle 1 in the left figure.

In the component-based evaluation, the largest segment in every manually labelled ground truth component was selected as the corresponding segment of this component. Then we calculated the completeness and correctness of every component by using point-based evaluation. If the completeness was lower than the threshold $\boldsymbol{over\_deco_{th}}$, this component was labeled as over-decomposed. If the correctness was lower than the threshold $\boldsymbol{under\_deco_{th}}$, this component was labeled as under- decomposed. Otherwise, this component was an applicable decomposed component. Both $\boldsymbol{over\_deco_{th}}$ and $\boldsymbol{under\_deco_{th}}$ were defined to be 0.6. For example, in the left image of **Figure 4.37**, component 2 labeled with a green circle is under-decomposed with low correctness. In contrast, component 3 is over-decomposed with low completeness.

4.4 Study area and experimental result

In order to evaluate the performance of our proposed decomposition method, we chose two test sites, Enschede and Paris. The Enschede dataset was used as a test case because there are many types of road furniture and it is a medium-sized city in the Netherlands. The Paris dataset was a benchmark dataset, and Paris is a French city. The shapes of the road furniture in these two test sites are different because they are from two different countries.

The test of the presented framework on the Enschede dataset is given in **Section 4.4.1,** and **Section 4.4.2** gives the experimental results of the Paris dataset. The analysis of the results is explained in the corresponding sections. In **Section 4.4.3**, the limitations and applicability are analysed.

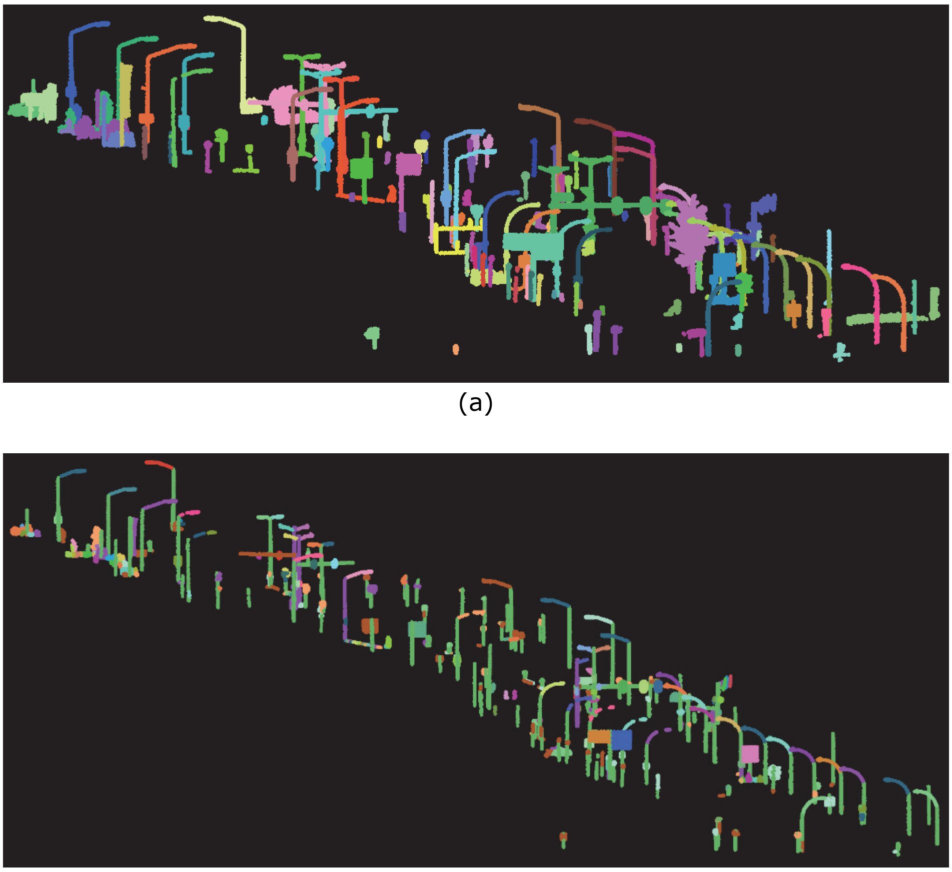
4.4.1 Enschede

4.4.1.1 Test Site

The Enschede MLS dataset was collected in a medium-sized city by the Optech LYNX system which consists of two rotating laser scanners at the back of the moving vehicle. These two laser scanners are in two almost vertical planes, are perpendicular to each other and at 45 degrees with the driving direction. The main area of this dataset is the urban area of Enschede which covers about 5 kilometres. As there are few pieces of road furniture, we chose the part of the dataset containing the most pieces of road furniture as our testing data. The selected Enschede MLS data are about 1.25 km long, consisting of 19,945,716 points and 151 pieces of road furniture. In the selected dataset, there a wide range of pole-like street furniture which can be tested with the proposed algorithm. The point density ranges from 35 points per square meter to 350 points per square meter.

4.4.1.2 Results

At the stage of decomposition, particular attention was paid to complex structures rather than bare poles. Therefore, we removed the bare poles from the detection stage, and only complex structural pole-like road furniture was identified as the input for the decomposition test. In this dataset, 115 pole-like road furniture were detected, given in **Figure 4.38a**. The final results of the pole-like road furniture decomposition can be found in **Figure 4.38b**.



(a)

(b)

***Figure 4.38**: The pole-like road furniture detection and decomposition from the Enschede dataset. (a) The identified pole-like road furniture (coloured by the component number); (b) the decomposition result after applying optimization and rules (coloured by poles and attachments).*

The feedback on the detection of pole-like road furniture from the pole-like road furniture decomposition works. The completeness and correctness of pole-like road furniture detection are as shown in **Table 4.1**. The correctness is improved to be 96.0%, and the completeness is 94.7%.

The accuracy evaluation of the decomposition of the pole-like road furniture from the Enschede dataset is given in **Table 4.2**. The completeness is above 79.5%, and the correctness is 92.3%. Compared to our previous work in which pole extraction optimization and rules were not applied, the performance improved significantly in the point-wise evaluation. Due to the high point density of this dataset, the components were already well-separated at the point level. The ratio of the correctly decomposed components increases from 72.2% to 79.5%, which proves the improvement of the decomposition after the addition of pole extraction optimization and rules for the decomposition part.

In order to set proper parameters, we chose one block which contained 22 pieces of road furniture to train the parameter settings. In the decomposition phase, we use a grid search to obtain the optimal ratio of the width of every slice for pole extraction. A set of parameters {0.25, 0.5, 0.75, 1.0, 1.25, 1.50} was tested on the sampling data. A value of 0.75 proved to be the optimal parameter.

***Table 4.1**: The accuracy of the road furniture detection in the two test sites.*

Test Site	**Enschede, NL**
Visual interpretation	151
Correctly detected (before/after the feedback)	145/143
Total detected (before/after the feedback)	154/149
Correctness (before/after the feedback)	94.2%/96.0%
Completeness (before/after the feedback)	96.0%/94.7%

***Table 4.2**: The accuracy evaluation of the decomposition in the Enschede dataset*

Enschede, NL		**Without optimisation and rules**	**With optimisation and rules**
Point wise	Completeness	79.5%	83.2%
	Correctness	92.3%	92.3%
Component wise	Visual inspection	327	327
	Correctly decomposed	236	260
	Over-decomposed	55	34
	Under-decomposed	58	54
	Correctness	72.2%	79.5%

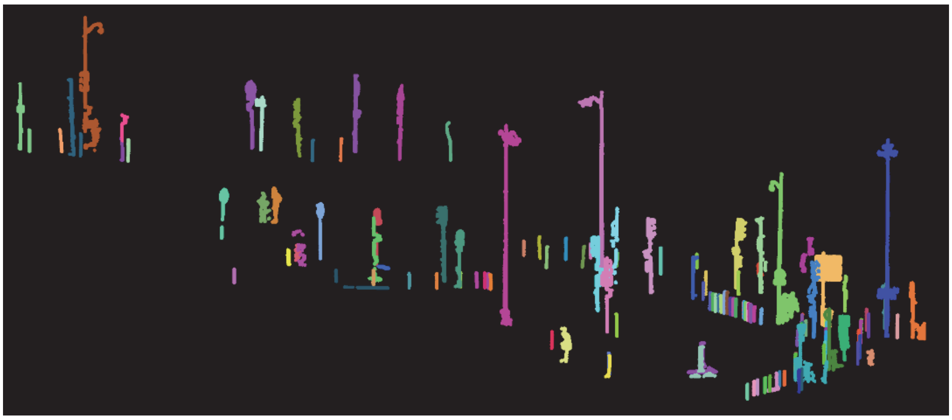
4.4.2 Paris

4.4.2.1 Test Site

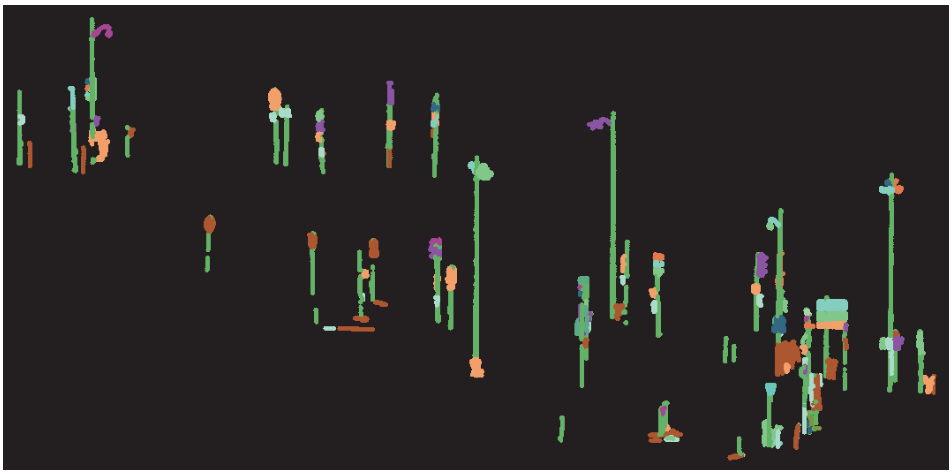
The Paris MLS data was acquired in 10 zones. Compared with other zones, most road furniture was in zone 7. For this reason, it was selected as the test data zone. This test area is about 0.43 km long, consisting of 13,776,061 points and containing 132 pieces of road furniture. The test area was partitioned into 13 blocks along the trajectory line. We designed each block to be 40 m long and 30 m wide because the Paris streets are relatively narrow. The point density ranged from 72 points per square meter to 500 points per square meter. Compared with the Enschede dataset, the point distribution of the road furniture in the Paris dataset was quite uneven. This was because the scanning pattern was different. The Paris dataset was collected by only one laser scanner.

4.4.2.2 Results

In this dataset, 41 pole-like road furniture items were identified as poles with attachments (see **Figure 4.39a**), and 97 were identified as bare poles. The detected poles with attachments were used to quantify the decomposition results. The pole-like road furniture decomposition result was as depicted in **Figure 4.39b**.



(a)



(b)

Figure 4.39*: The pole-like road furniture decomposition from the Paris dataset. (a) The detected pole-like road furniture (coloured by component number); (b) the decomposition result after applying optimization and rules (coloured by poles and attachments).*

By means of visual inspection, the performance of pole-like road furniture detection in the Paris dataset was evaluated. **Table 4.3** shows that the correctness was improved from 88.7% to 91.3% and the completeness was 95.4% in this dataset. Similar to the Enschede dataset, we also selected the optimal parameters by using a grid search in the Paris dataset. The optimal ratio of the width of every slice for pole extraction was 0.50.

The performance of our previous work and the current framework is illustrated in **Table 4.4**. Compared with the Enschede dataset, the quality of the Paris dataset is lower. For example, in the Paris dataset, the point distribution was uneven because of the usage of a single scanner. Because the decomposition method required a high-quality point cloud, the performance of our previous work tested on the Paris dataset was not satisfying. As shown in **Table 4.4**, the results of the initial decomposition were without pole extraction optimization and rules. With the addition of pole extraction optimization and rules in the decomposition part, the completeness was improved to be above 80%, the correctness was improved to be 92.6%, and the rate of correctly decomposed components was enhanced to 72.0%. The results prove the effectiveness of the proposed framework.

***Table 4.3**: The accuracy evaluation of the detection in the Paris dataset.*

Test Site	**Paris, FR**
Visual interpretation	132
Correctly detected (before/after the feedback)	126/126
Total detected (before/after the feedback)	142/138
Correctness (before/after the feedback)	88.7%/91.3%
Completeness (before/after the feedback)	95.4%/95.4%

***Table 4.4**: The accuracy evaluation of the decomposition in the Paris dataset.*

Paris, FR		**Without optimisation and rules**	**With optimisation and rules**
Point wise	Completeness	69.0%	80.0%
	Correctness	90.0%	92.6%
Component wise	Visual inspection	125	125
	Correctly decomposed	67	90
	Over-decomposed	24	12
	Under-decomposed	45	29
	Correctness	53.6%	72.0%

4.4.3 Parameter analysis

For the experiments, three groups of parameters are explained for the respective pole extraction method. The first one is for the 2D point-density based method. There are two critical parameters for this method. These two parameters directly affect the number of extracted pole points and its accuracy.

- 2D neighbouring distance and the constraint radius of a pole is defined by the median width of the cut slice. It affects point density calculation and pole extraction.

- 2D point density similarity is defined as the 2D point density difference between the seed points and its neighbouring points. Here the similarity threshold of their ratio is 0.85.

In the RANSAC line fitting method, three parameters are significant for decomposition.

- The smaller neighbourhood size is defined by the maximum neighbourhood size of main pole points beginning to have high linearity values. The larger neighbourhood size should be larger than the maximum neighbourhood size of short pole points beginning to have high planarity or volumetric values, and it should also be smaller than the maximum neighbourhood size of main pole points beginning to have high planarity or volumetric values. Here the smaller neighbourhood size is 100, and the large neighbourhood size is 300.

- The percentage of points fitting the line is defined as an adaptive variable. This parameter affects the accuracy of the extracted line. The threshold is not a fixed value and is decreasingly set from 70% to 30% until the main poles are found.

- The distance between the extracted line and pole point candidates for pole fitting is defined based on the median value of widths of all cut slices. This parameter directly affects the accuracy of the extracted pole. That's why we calculate this per object.

For the slice cutting based method, there are two critical parameters affecting the result of decomposition

- The interval height of every slice is defined as 0.1m. This parameter affects the performance of refining of widths of cut slices.

- The width of extracted lines for pole fitting is defined as the width of every slice. This parameter directly decides whether the pole points of every slice can be extracted accurately.

For the parameters mentioned above, most of them are adaptive or defined by the prior-knowledge of street furniture such as median width.

4.5 Discussion

The pole-extraction optimization, defined merging, and splitting rules also improved the performance of the decomposition. It enhances the ratio of correctly decomposed components by 7.3% in the Enschede test site and 18.4% in the Paris test area. Some well-decomposed road furniture is shown in **Figure 4.40**.

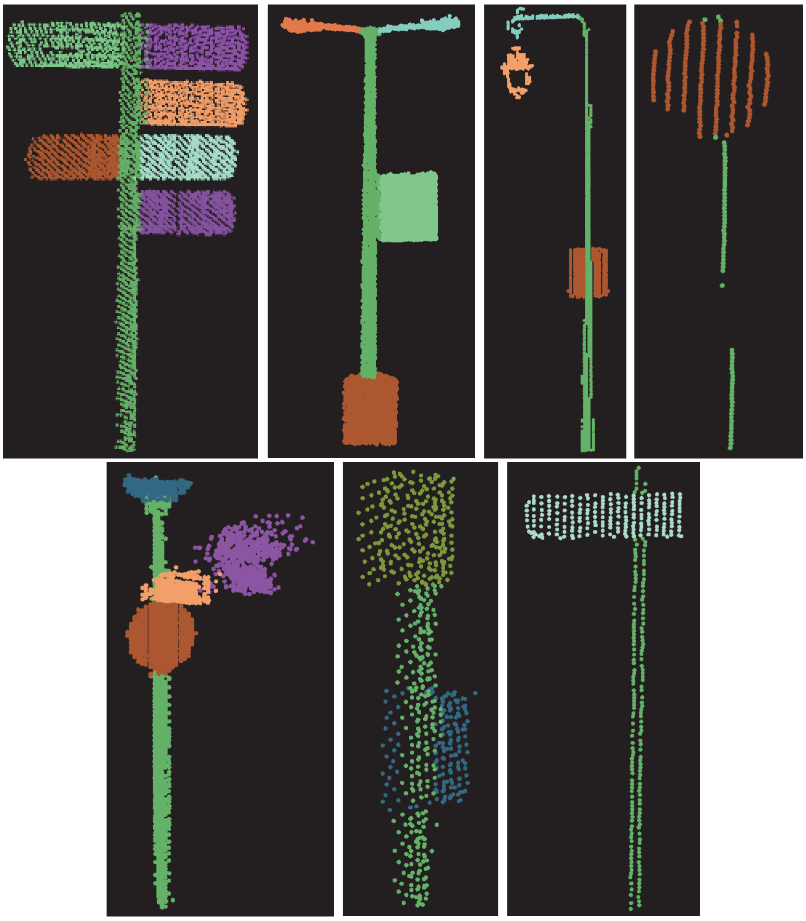


***Figure 4.40**: The correct decomposition results.*

The problems with decomposition are divided into two types. The first one is the problem with the data itself. Specifically, it involves point density, point distribution, and noisy points. In **Figure 4.41a**, the two connected traffic signs should have been separated. If there are many noisy points around one road furniture item, it is even harder to separate these attached components by visual interpretation. Under such circumstances, our decomposition algorithm does not work. If the point density is extremely low, points cannot form an individual component. For example, in the Paris dataset, a few traffic lights were documented by several points, which led to the fragmented and unstructured components during the decomposition (**Figure 4.41b**). In contrast, most of the traffic signs were decomposed as individual components owing to their adequate point density. Therefore, the point density cannot be very low. The point spacing between points in the trajectory direction usually is higher than 0.05 m. The point spacing was about 0.02 m within the scanlines. The point spacing in these two directions was different. The second problem originates from the scene. Currently, curved poles cannot be extracted as the assumption is that pole-like road furniture contains straight poles. Beside this, pedestrian-oriented traffic lights (**Figure 4.41c**) which consist of traffic light heads connected with each other cannot be decomposed correctly (**Figure 4.41d**).

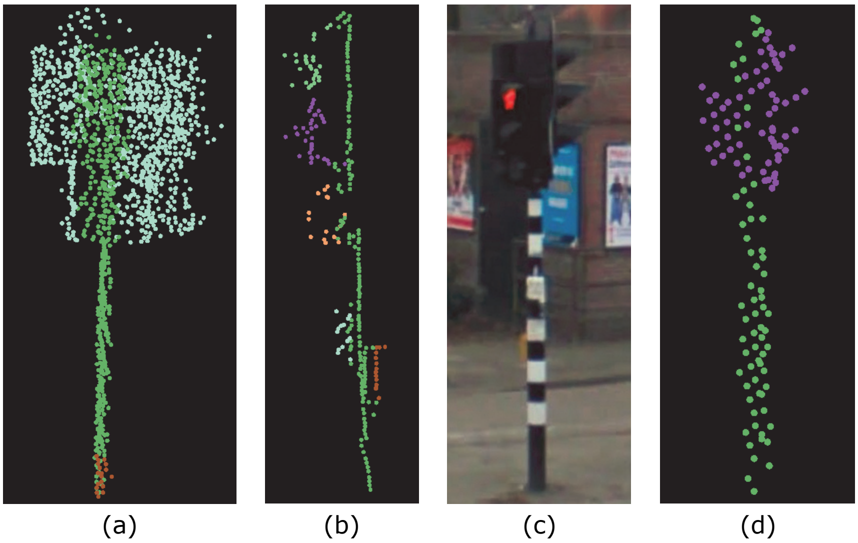


Figure 4.41*: Incorrectly decomposed road furniture. (a) two traffic signs were not separated; (b) one traffic light was decomposed to two fragmented components; (c) the image and (d) the point cloud of two traffic lights which were not separated.*

4.6 Conclusion

In this chapter, we proposed a framework to decompose pole-like road furniture into different components with corresponding to their functionalities. This innovative framework is tested in two test sites. After being processed by our new framework, road furniture was decomposed by spatial relations, which can be used for precise semantics labelling. This proposed framework can be potentially used for high defined 3D mapping. In this framework, we enhanced the initial decomposition of road furniture by applying the optimised pole extraction, splitting and merging rules. The feedback from pole-like road furniture decomposition helps improved pole-like road furniture detection. The main contribution was the decomposition of the road furniture and its evaluation.

Except that curve poles cannot be extracted accurately because the pole extraction method is proposed based on the features of straight poles, most of the incorrect decomposition is caused by connected component analysis. When sign points are close to a pole, points on these signs are categorised as poles. The reason is, in this case, the 2D point density of these sign points is very high, and they are very close to the centre line of a pole. With only Euclidean distance, it is difficult to separate these points from pole points.

In our research, road furniture items have been decomposed into components by using mere geometric features. Colour information has also been beneficial to the detection and decomposition. Many techniques on image semantics labelling such as the convolutional neural network can be applied to our research. 2D image data were also captured by two cameras mounted on a moving vehicle. Better detection and decomposition of road furniture could benefit from the colour information additionally collected with 3D point clouds.

Chapter 5 - Pole-like Road Furniture Interpretation[3]

[3] This chapter is based on:

Li, F., Lehtomäki, M., Oude Elberink, S., Vosselman, G., Puttonen, E., Kukko, A. and Hyyppä, J., 2018. Pole-like road furniture detection in sparse and unevenly distributed mobile laser scanning data. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 4(2).

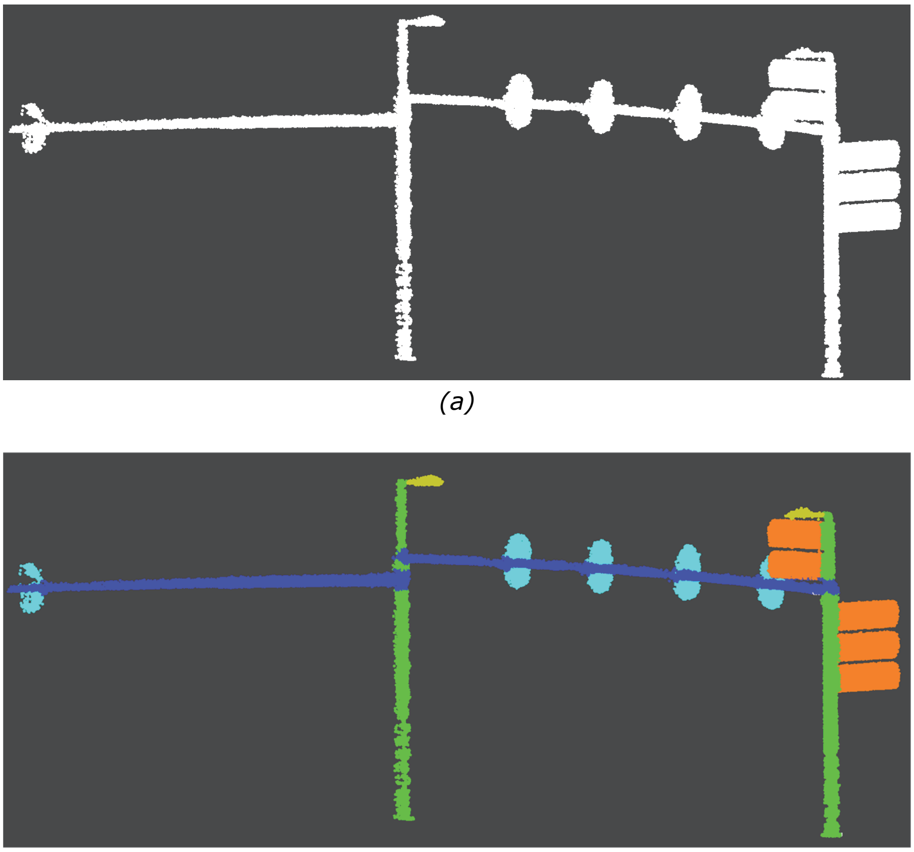
Li, F., Lehtomäki, M., Oude Elberink, S., Vosselman, G., Kukko, A., Puttonen, E., Chen, Y. and Hyyppä, J., 2018. Semantic segmentation of road furniture in mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing. (Under review)*

5.1 Introduction

In recent years, the automatic recognition of road furniture is in great demand because of the pivotal role of road furniture in the road safety analysis. However, road furniture recognition has relied on manual labelling, which is tedious and time-consuming. The emerging of mobile laser scanning provides a solution to this problem. Mobile laser scanning systems have been well developed in recent years, which makes it possible to capture 3D point clouds of urban scenes with higher precision (**Ma et al., 2018**). Compared to laser scanning systems mounted on other platforms (e.g. terrestrial and airborne laser scanners), mobile laser scanning systems are able to collect 3D data in an urban environment more precisely than airborne laser scanning systems and more rapidly than terrestrial laser scanning systems.

With the support of high-quality mobile laser scanning (MLS) data, road furniture recognition has been well-studied in the past few years. A primary current focus in road furniture inventory is pole-like road furniture identification. Numerous attention has been paid on pole-like road furniture recognition (**Brenner, 2009; Golovinskiy et al., 2009; Lehtomäki et al., 2010; Pu et al., 2011; Xiong et al., 2011; El-Halawany and Lichti, 2013; Bremer et al., 2013; Li and Oude Elberink, 2013; Cabo et al., 2014; Fukano and Masuda, 2015; Huang and You, 2015; Yang et al., 2015; Yu et al., 2015a; Li et al., 2017; Wang et al., 2017a**). The performance of pole-like road furniture classification has improved significantly in recent years. Most studies focus on the classification of road furniture as single objects by performing a connected component analysis with above-ground points, whereas little attention has been paid to the interpretation of single pieces of road furniture with multiple classes, such as signs and lamps attached to poles. **Figure 5.1a** shows an example, where road furniture can only be interpreted as a single object by current frameworks. **Figure 5.1b** shows a detailed interpretation of such road furniture, which is more appropriate and of more practical use. To achieve this goal, we propose a framework to interpret pole-like road furniture. In the first stage, road furniture is detected from MLS data. Then a decomposition stage is carried out to separate detected road furniture into poles and their attachments (e.g. traffic signs and lamps). In the final stage, these attachments are classified into multiple classes by using extracted features and machine learning classifiers. Literally, the point cloud of an attachment is a segment. Our contribution is to fully enable the potential of a decomposed point cloud to understand which classes can be assigned to which attachment. Because of the limited number of pieces of road furniture, the number of our training samples is quite small. Therefore, we chose not to use deep neural networks. The performance of different machine learning classifiers is tested on two test sites. We also compare machine learning classifiers to our previously proposed knowledge-driven method (**Li et al.,**

2017). The performance of road furniture interpretation has been significantly improved compared to the knowledge-driven method.



(a)

(b)

Figure 5.1*: A piece of road furniture with multiple classes. (a) The original point cloud of one piece of road furniture. (b) The interpreted road furniture (Orange: Street signs, Yellow: Street lights, Cyan: Traffic lights, Green: vertical poles, Blue: Horizontal poles)*

The remainder of this chapter is organised as follows. In **Section 5.2**, we give an overview of current research carried out on pole-like road furniture classification in MLS data. The proposed framework for detailed road furniture recognition is depicted in **Section 5.3**. Then we test our algorithm on two different test sites and analyse the result in **Section 5.4**. In the end, the conclusion is drawn. In this chapter, we focus on the classification of the attachments, which are generated by the detection and decomposition in our previous work (**Li et al., 2018a**).

5.2 Related work

Below, we divide studies on road furniture recognition into knowledge-driven and machine learning-based methods. We also treat a branch of machine learning, deep learning as a separate group.

5.2.1 Knowledge-driven

Knowledge-driven methods represent early attempts to recognise objects alongside roads from MLS data. This type of methods utilises generic rules to classify target objects. Techniques used for the recognition of structures in point clouds have been reviewed by **Vosselman et al. (2004)**, which includes the identification of smooth surfaces, planar surfaces and parameterised shapes. **Brenner et al. (2009)** use a local scatter matrix and a 3D cylinder stack to detect pole-like road furniture from MLS data. In this method, the generic rule of pole-like road furniture detection is to analyse the distribution of points in both the inner and outer region of a fitted cylinder. Utilising similar rules, **Lehtomäki et al. (2010)** develop a framework to detect vertical-pole objects by using a scan line segmentation strategy. A voxel-based algorithm is proposed by **Cabo et al. (2014)** to detect pole-like street furniture objects from MLS data by using a similar cylinder masking. **Bremer et al. (2013)** employ two radius neighbourhoods to extract linear, planar and volumetric features. Rules are defined to classify objects into three categories in this piece of work. However, it is difficult to extract slanted poles and thick poles like tree trunks for this method. A percentile-based method is proposed by **Pu et al. (2011)** to identify pole-like road furniture from MLS data for road inventory studies. Traffic signs are differentiated by using shape information in this method. **Li and Oude Elberink. (2013)** optimise the work of **Pu et al. (2011)** by the additional use of reflectance information. Because of the addition of reflectance information, the detection rate of traffic signs is significantly improved. Similar to **Pu et al. (2011)**, **Riveiro et al. (2016)** employ shape information to recognise traffic signs. However, these three methods have difficulties with the detection of road furniture with complex structures, such as road furniture connected with other objects. The 2D point density feature is exploited by **El-Halawany and Lichti (2013)** to identify road poles from mobile terrestrial laser scanning data. **Oude Elberink and Kemboi. (2014)** recognise objects by using the slicing features and template matching in MLS data. **Yang et al. (2015)** introduce a supervoxel-based method to classify urban objects from MLS data. However, there is pole-like road furniture connected with trees or other road furniture, which makes it difficult for these methods mentioned above to conduct the detection. Knowledge-driven methods rely on generic rules which are obtained from the inference of empirical observation. It is not appropriate for this type of method to perform classification tasks with many target classes. This is because the number of generic rules in combination with feature constraints is somewhat limited. This type of methods does not require much training data.

5.2.2 Machine learning

Machine learning based methods have been widely applied for point cloud processing in recent years. Four commonly used machine learning classifiers are introduced in this section. They are support vector machine (SVM), random forests (RF), Gaussian mixture model (GMM) and naïve Bayes (NB). After that, we reviewed current work on point cloud classification and interpretation using machine learning classifiers.

Support Vector Machine

One commonly adopted machine learning classifier in classification problems is the support vector machine (**Bishop, 2006**). SVM classifies different objects based on maximising the margin, that is, the distance between the decision hyperplane and closest training vectors. The hyperplane is constructed by defining a mapping function, which is defined as follows,

$$y(x) = w^T \emptyset(x) + b \tag{5.1}$$

where $\emptyset(x)$ denotes a fixed feature-space transformation and w is a vector of weights which is also known as the normal of the hyperplane. b is a bias parameter.

The basic version of the classifier is linear, but by using the so-called kernel trick and, for example, a radial basis function (RBF) kernel, nonlinear decision surfaces can be learned (**Bishop, 2006**). C-SVM is a type of SVM widely utilised in classification problems. The LIBSVM package (**Chang and Lin, 2011**) is popularly used to train the classifiers and to perform predictions.

SVMs are binary classifiers, but several modifications have been developed to tackle the multiclass problem (**Hsu and Lin, 2002**). One approach is to train several binary classifiers and combine their outputs into one classifier. In LIBSVM, the one-versus-one approach is used, also suggested by **Hsu and Lin (2002)**, in which a binary classifier is trained for each pair of classes. The classifiers are combined using the Max Wins strategy in which a new test vector is classified with all binary classifiers, and the test vector is labelled according to the class that gains most votes.

C-SVM that uses the RBF kernel contains two hyperparameters that control the complexity of the classifier, that is, how complex or nonlinear the decision surface is or how accurately the classifier fits the training data. If the classifier overfits to the training data, it predicts poorly a new unseen data set and has a poor generalisation performance. The first hyperparameter is the inverse kernel width γ and the second one is the complexity trade-off parameter C. To find optimal values for γ and C, cross-validation (CV) is used. Cross-validation is a method that can be used to the select optimal model from a set of alternative models without using an external validation data set. Cross-validation estimates the generalisation performance of a model on a new unseen data set by maximising the amount of data used for model training. In CV, the data is split into n disjoint sets or folds. Each fold in turn acts as an independent test set and, the rest n-1 folds are used to train the model. The test set is predicted using the trained model and test set accuracy is calculated. Finally, n test set accuracy estimates are gained. Based on (**Kohavi, 1995**), an n-fold cross-validation with stratification is normally used. The CV accuracy $K(\theta)$, where θ are the model selection parameters to be optimised was calculated based on the number of correctly classified test set samples $q_{\text{correct}}^{(i)}$ in fold i, $i = 1, \dots, n$, and the number of all samples in the test set $q^{(i)}$ using the equation (**Lehtomäki et al., 2016**)

$$K(\theta) = \frac{1}{10} \sum_{i=1}^{10} \frac{q_{\text{correct}}^{(i)}}{q^{(i)}} \tag{5.3}$$

As a result, each model has a cross-validation accuracy calculated with **Eq. 5.2**, and the one with the highest CV accuracy is chosen as the optimal model. When selecting γ and C, cross-validation is combined with a grid search. A 2D grid is constructed in the γ - C plane where each γ - C value combination corresponds to one model. Cross-validation is performed in each vertex of the grid using the corresponding γ-C value combination. The γ-C pair with the highest CV accuracy is selected as the optimal model.

To improve the classification performance and to analyse the goodness of different classification features, also feature selection is performed (**Guyon and Elisseeff, 2003**). We use backward elimination (**Guyon and Elisseeff, 2003**), as it is able to take into account the joint effects of various features on the classification accuracy. Backward elimination starts with taking all features and removing each feature in turn. With each feature removed in turn, the CV accuracy is evaluated using the **Eq. 5.2**. The model with the lowest CV accuracy is discarded, and the corresponding feature is eliminated from the set of all features. This process is repeated until only one feature is left. As a result, an elimination curve is drawn in which the CV accuracy after each elimination is plotted as a function of the elimination round. The feature set with the highest CV accuracy is chosen as optimal. After having found the optimal features, γ and C are optimised using the grid search.

The performance of the final classification model, that is, the trained classifier with optimal γ, C and feature set has to be evaluated on an independent test set in order to perform an unbiased evaluation. First, the data was split into n folds. In each fold, the classification model was constructed (including training, feature selection and optimisation of γ and C) using n-1 fold and the performance of the model was evaluated using the remaining fold (**Eq. 5.2**). In the end, the optimal model obtained from cross validation is used to evaluate the performance of the classification model on an independent test set.

Random Forest

The Random Forests classifier takes advantage of combining multiple weakly learners. It has been widely employed in a number of classification applications (**Criminisi et al., 2012**). Given the labelled training components with an m-dimensional feature vector $v = (x_1, \dots, x_m) \in R^m$, the bootstrap aggregating is used to get a bootstrap sample S_l. A random forest tree T_l is constructed for the bootstrap sample. The handcrafted k dimensional features are picked up and one is selected as the best to split the node j into two children nodes by maximizing the gain of information with **Eq. 5.3** (**Breiman, 2001**).

$$\theta_j^* = \underset{\theta_j \in \tau_j}{\operatorname{argmax}} I_j \tag{5.3}$$

$$I_j = H(S_j) - \sum_{i \epsilon \{L,R\}} \frac{|S_j^i|}{|S_j|} H(S_j^i) \tag{5.4}$$

$$H(S) = -\sum_{c \in C} p(c) \log p(c) \tag{5.5}$$

Where I_j is the gain of information, which indicates the utility of a selected feature. $H(S)$ is the Shannon's entropy of generic set S of training components S_l. $p(c)$ is the probability of class c in the generic set S. This random forest tree T_l grows by repeating the process of constructing binary nodes until the depth of this tree reaches the predefined value. Likewise, the other trees in the forest are constructed and grow by following the same process. The depth of a tree d and the number of trees n are set based on the optimal trained model with the best performance in the training dataset. After the trained model is obtained, we predict the probability of a given component with feature vector v belongs to class c as follows,

$$p(c|v) = \frac{1}{L} \sum_l^L p_l(c|v) \tag{5.6}$$

The predicted label is the class with the highest probability, which is solved as follows,

$$\hat{y} = arg\max_{c} p(c|v) \tag{5.7}$$

In order to analyse the utility of our designed features, the importance of every feature is also computed. The feature importance is calculated by **Eq. 5.8** as described in **Louppe et al. (2013)**.

$$Imp(x_m) = \frac{1}{L}\sum_T \sum_{l\in L:v(s_l)=x_m} p(l)\, I(s_l) \tag{5.8}$$

Gaussian Mixture Model

The Gaussian mixture model (GMM) is often used in combination with k-means clustering which gives initial clustering information. Instead of using the k-means clustering, our training data is manually labelled. In GMM, the distribution of a component belonging to a class follows a multivariate Gaussian with the mean μ_k and covariance matrix $\sum_k$. With the mixing weights π_k added, the maximum-likelihood is estimated as follows,

$$L(x,\theta) = \log p(x,\theta) = \sum_{i=1}^{n} log\left(\sum_{k=1}^{K} \pi_k \mathrm{N}_m(x_i|\mu_k, \textstyle\sum_k)\right) \tag{5.9}$$

To find the maximum-likelihood estimates (MLE) of all the mixture parameters, the expectation maximization (EM) is used to estimate the μ_k and $\sum_k$ of each class and the mixing weights π_k. Similar to the random forests classifier, the label of a component is given to the class with the highest probability.

Naïve Bayes

As a method based on Bayes' theorem, the Naïve Bayes (NB) classifier assumes every pair of features is conditionally independent given the class (**Murphy, 2012**). It is assumed that every single dimension feature value of components given their labels follows a Gaussian distribution.

Based on Bayes' theorem, given the feature vector $\{x_1, \dots, x_n\}$, the probability of the label of a component is:

$$p(y|x) = p(y)\frac{\prod_{i=1}^{n} p(x_i|y)}{p(x)} \tag{5.10}$$

where $p(y)$ is the prior probability of the labels. A Posteriori (MAP) is used to estimate $p(y)$ and $p(x_i|y)$. In the end, every component is given the label with the highest probability.

Machine learning-based methods on point cloud interpretation

Golovinskiy et al. (2009) propose a shape-based approach to recognise 3D point clouds in urban environments by using the min-cut algorithm. They utilised a normalised cut method to localise objects of interest. Different machine learning techniques in combination with shape features were used to classify these above-ground objects into sixteen categories. In the work of **Munoz et al. (2009)**, a functional gradient approach was proposed to label mobile mapping data by using Max-Margin Markov Networks (M3Ns). This method was tested for both 3D point cloud classification and geometric surface estimation in 2D images. However, it is still difficult for this method to separate and classify conjunctions which contain multiple objects. **Ross et al. (2011)** compare five message-passing approaches for predicting labels of point clouds. The proposed message-passing algorithm shows the scalability and efficacy with 3D point cloud classification. Based on their work, **Xiong et al. (2011)** propose a sequenced predictor to do 3-D scene analysis. However, the precision of recognition of poles and tree trunks using M3N is low compared with the identification of other categories. **Yu et al. (2015b)** present a method to separate urban road objects and recognise street light from MLS point clouds using the normalised cut algorithm. **Velizhev et al. (2012)** present an implicit shape model (ISM) based method to localise and recognise cars and light poles automatically. Spin image (SI) descriptor is employed as the keypoint representation for recognition. **Yokoyama et al. (2013)** propose a method to detect and classify pole-like road furniture from MLS data. Both shapes features of pole-like objects and the distribution of surrounding pole-like objects are used in this method.

Different from these methods mentioned above, **Huang and You (2015)** proposed a method in combination with a Supported Vector Machine (SVM) to classify road furniture into four categories. In order to localise pole-like objects, they implemented point cloud slicing, clustering, pole seed generation, and bucket augmentation. In the following stage, ground points are removed, and the pole-like objects were extracted. In the end, six features are trained by the SVM to classify road furniture into four types. The detection rate of pole-like road furniture was 75%. **Soilán et al. (2016)** used mobile laser scanning data in combination with images to recognise traffic signs. At the first stage, the ground points were removed by using height and intensity constraints. Traffic signs were extracted with a reflectance threshold estimated by a Gaussian Mixture Model (GMM). Then, geometric parameters of the extracted traffic signs were generated. Based on these geometric inventories, these extracted components were projected onto the corresponding camera systems, after which traffic signs can be found on the corresponding 2D images. In the end, Histogram of Oriented Gradients (HOG) features trained by the SVM are adapted to recognise traffic signs. More than 85% of pole-like traffic signs were detected.

However, the detection of traffic signs in this method strongly relies on the reflectance information. To classify point clouds of urban scenes, the SVM classifier in combination with a set of handcrafted features is also employed in the work of **Yang and Dong (2013)** and **Lehtomäki et al. (2016)**.

Hackel et al. (2016) described an efficient and effective method for point-wise semantic classification, which can deal with point clouds captured by LiDAR or derived from photogrammetric reconstruction with high-density variations. Instead of computing optimal neighbourhoods for each point, they down-sample the entire point cloud to generate a multi-scale pyramid with decreasing point density and calculate features for every voxel at every scale level. Then these features are trained and discriminant ones are selected. Finally, the random forests classifier is used for semantically labelling. The precision of pole-like road furniture recognition is less than 35%. The random forests classifier was also applied to classify urban objects in the work of **Fukano and Masuda (2015)**. **Weinmann et al. (2015)** compare different machine learning classifiers for the classification of urban environment objects by using extracted features with an optimal neighbourhood size. **Tombari et al. (2014)** combine local descriptors and global descriptors to recognise pole-like road infrastructures automatically. Support vector machine and Markov random fields (MRF) are adopted at a local level and global level respectively. The classification of pole-like road furniture both in point-level and in object-level has been improved significantly by machine learning classifiers in combination with handcrafted features because of the acquisition of higher quality data.

5.2.3 Deep neural network

Deep learning has been studied and applied to object recognition in 3D point clouds. Based on the representation of point clouds, current deep learning techniques on point cloud classification are categorised as three types: convolutional neural networks (CNNs) operating on voxels, point clouds and multi-view images. Voxel-based methods regularise point clouds by using voxels. **Maturana and Scherer (2015)** develop VoxNet, convolutional neural network (CNN) to train and recognise objects in real-time 3D point clouds. A 3D convolutional neural network was proposed by **Huang and You (2016)** to interpret urban scenes into seven categories. They first used small grids to densely voxelise the original point cloud. Then feature maps in combination with a convolutional neural network are trained to label the point clouds. 87% of pole-like road furniture was identified. **Riegler et al. (2017)** conduct the classification of sparse 3D point clouds by building a 3D CNN on the basis of the octree representation. An example of the architecture of VoxelNet is as shown in **Figure 5.2.**

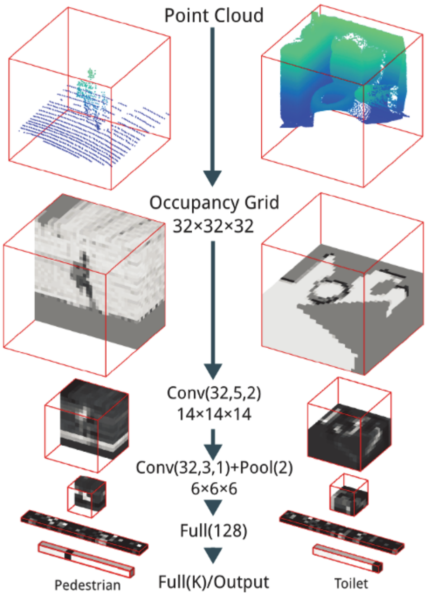


***Figure 5.2**: The architecture of VoxelNet (**Maturana and Scherer, 2015**)*

Compared to methods which utilise CNNs operating on voxels, methods that employ CNNs operating on point clouds directly use point coordinates as inputs to feed neural networks. PointNet is proposed by **Qi et al. (2017)** to semantically segment scene and classify objects in 3D point cloud by respecting the permutation invariance of 3D points. PointNet directly utilises point clouds as an input. Then a symmetry function for the unordered input is proposed to aggregate the formation for each point and make the model invariant to input permutation. These point features are aggregated by max pooling. In the end, the output is the classification score for each class. This network can be flexibly employed for both segmentation and classification of point clouds. The architecture of PointNet is as shown in **Figure 5.3**. **Landrieu and Simonovsky (2017)** embed PointNet with the gated recurrent unit to construct graphs, which segment large-scale point cloud semantically. Similar to the PointNet, **Li et al. (2018c)** exploit a 3D CNN to segment a point cloud semantically by learning a transformation. **Su et al. (2018)** present a network structure to interpret objects in 3D point clouds by representing a collection of points as a sparse set of samples in a high-dimensional lattice. The Bag of Words (BoW) and Deep Boltzmann Machine (DBM) methods are applied by **Yu et al. (2016)** to detect and recognise traffic signs in mobile laser scanning (MLS) data. Here, the authors first constructed a visual word vocabulary by using features encoded by a DBM model to detect traffic signs. Similar to **Li et al. (2016)**, they separated the poles and traffic signs which are projected to 2D images afterwards. Lastly, these cropped traffic sign images were recognised by a pre-trained DBM model. More than 90% of pole-like road furniture was detected in the high-quality MLS data.

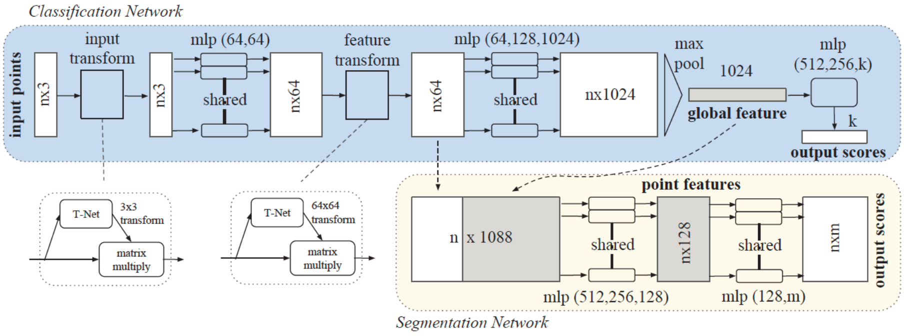


***Figure 5.3**: The architecture of PointNet (**Qi et al., 2017**)*

Different from the other two types of method, multi-view image-based methods project point clouds to multiple planes to generate image features for every point. In this way, deep learning techniques for images processing can be applied to process point clouds. **Su et al. (2015)** project point clouds onto different views to generate 2D rendered images. Then they concatenate different rendered images to train a CNN to recognise objects. **Boulch et al. (2017)** further develop this type of network to process 3D point clouds in the large-scale urban scene. **Figure 5.4** shows the architecture of the multi-view CNNs. However, it requires numerous training data with good quality.

Compared to knowledge-driven methods and traditional machine learning classifiers, deep learning methods are more flexible. There is no need for this type of methods to manually design generic rules and high-level features. Deep learning has been applied to 3D point clouds processing and advanced this field significantly. However, it requires numerous training data with good quality.

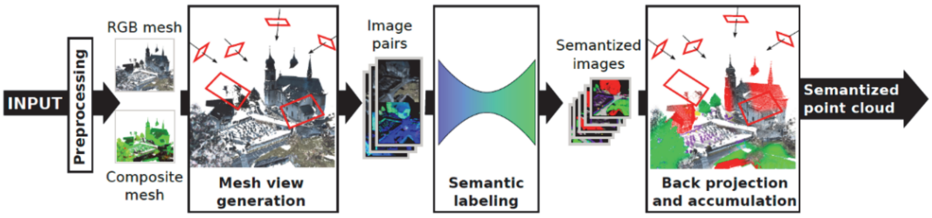


***Figure 5.4**: The architecture of Multi-view CNN (**Boulch et al., 2017**)*

Up to now, these methods mainly focus on the classification of entire pieces of road furniture. We have proposed a knowledge-driven method to interpret decomposed pole-like road furniture in MLS data (**Li et al., 2017**). Otherwise, the detailed interpretation of road furniture has not yet been addressed.

5.3 Methodology

Poles and attachments are separated in the previous chapter. In the current stage, the attachments are classified into different classes such as street light by using features extracted from attachments in combination with defined rules or machine learning classifiers. In this section, extracted features are defined first (**Section 5.3.1**). In **Section 5.3.2**, we introduce the knowledge-driven method. The machine learning classifiers utilised are introduced in **Section 5.3.3**. In **Section 5.3.4**, we explain how to extract non-changed attachments of road furniture between two epochs.

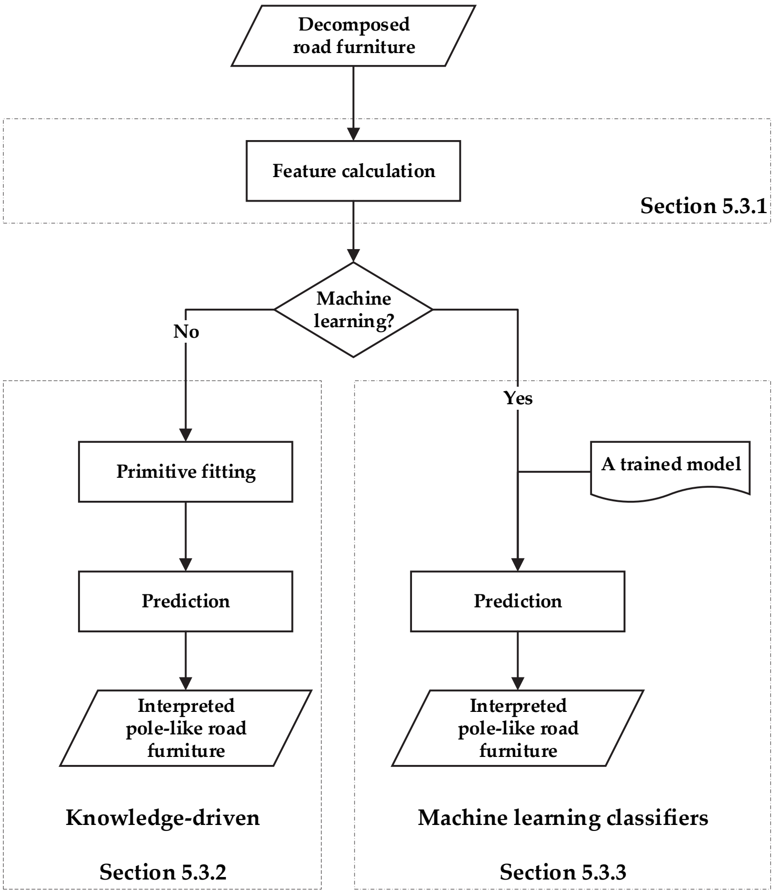


Figure 5.5*: The schematic of the pole-like road furniture interpretation*

5.3.1 Feature notation

Different from pointwise features, features are extracted from attachments in relation to poles, which are regarded as units in this research. Therefore, these utilised features are segment-wise. In addition to unary features, contextual features are extracted as well to feed machine learning classifiers for training and predicting. Unary features are features which are directly obtained from attachments. Three types of unary features are extracted. They are size $\{S\}$, eigenvalue-based features $\{E\}$ and radiometric features $\{R\}$.

To extract size features $\{S\}$, we construct the bounding box for every attachment as shown in **Figure 5.6**. A six dimensional feature $\{S\}$ is calculated for the bounding box. It includes the horizontal length S_L, horizontal width S_W, height S_H, maximum range in the horizontal plane S_D, volume $S_V = S_L \cdot S_W \cdot S_H$ and ratio feature S_{D-H}. S_D is the largest distance between two points of an attachment in the horizontal plane. Ratio feature S_{D-H} is the ratio of the 2D distance S_D to the height of the bounding box S_H. When we compute the largest 2D distance S_D of two points in the point cloud of an attachment, the complexity of the calculation of this feature is $O(n^2)$.

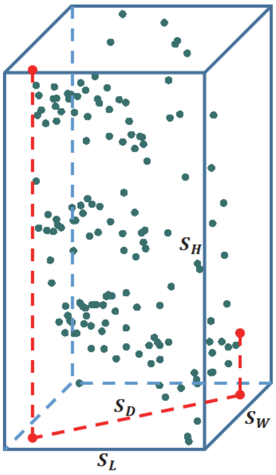


Figure 5.6*: The size features of a traffic light (the blue box is the bounding box of the point cloud of the traffic light)*

Eigenvalue-based features $\{E\}$ are calculated by eigenvalues $\{\lambda_1 > \lambda_2 > \lambda_3 \geq 0\}$ of the covariance matrix of point coordinates belonging to every attachment. They are linearity E_L, planarity E_P, linear planarity E_{LP}, scattering E_S, omnivariance E_O, anisotropy E_A, eigenentropy E_E, and change of curvature E_C. E_L describes the linearity of attachment. E_P denotes how planar an attachment is. E_{LP} is designed for specifying elongated street signs. E_S depicts how scattered the distribution of points of an attachment is. E_O suggests how spherical an attachment is. E_A implies the distribution of points of an attachment in different directions. The entropy of the distribution of points in three directions is indicated by E_E. E_C denotes the change of curvature of an attachment. The calculation of these features is as **Eq. 5.11 to Eq. 5.19**. Compared to the features used by **Weinmann et al. (2015)**, we add E_{LP}.

$$e_i = \frac{\lambda_i}{\sum_i \lambda_i},\ \ i \in \{1, 2, 3\} \tag{5.11}$$

$$E_L = \frac{e_1 - e_2}{e_1} \tag{5.12}$$

$$E_P = \frac{e_2 - e_3}{e_1} \tag{5.13}$$

$$E_{LP} = \frac{e_2 - e_3}{e_2} \tag{5.14}$$

$$E_S = \frac{e_3}{e_1} \tag{5.15}$$

$$E_O = \sqrt[3]{e_1 e_2 e_3} \tag{5.16}$$

$$E_A = \frac{e_1 - e_3}{e_1} \tag{5.17}$$

$$E_E = \sum_{i=1}^{3} e_i ln(e_i) \tag{5.18}$$

$$E_C = e_3 \tag{5.19}$$

Radiometric features $\{R\} = \{R_{MD}, R_{MA}\}$ are obtained directly from the collected radiometric attributes of the point cloud. R_{MD} and R_{MA} are the median and mean value of reflectance of points in every attachment.

In addition, there are two types of contextual features, which describe the spatial relations between attachment and their attached poles.

Unlike the extraction of unary features, contextual features are calculated from attachments and poles. They are angle features $\{A\}$ and relative height features $\{H\}$. The detailed explanation of these features is as follows.

Angle features $\{A\}$ consist of the angle A_{PA} between the principal direction of poles $\overrightarrow{V_P}$ and the normal direction of their attachments $\overrightarrow{V_A}$, and the angle A_{VA} between vertical direction $\overrightarrow{V_V}$and the normal direction of attachments. They are defined in the following two equations:

$$A_{PA} = \overrightarrow{V_P} \cdot \overrightarrow{V_A} = \cos \alpha_{PA} \tag{5.20}$$

$$A_{VA} = \overrightarrow{V_V} \cdot \overrightarrow{V_A} = \cos \alpha_{VA} \tag{5.21}$$

One instance of angle features of a traffic sign is as illustrated in **Figure 5.7**.

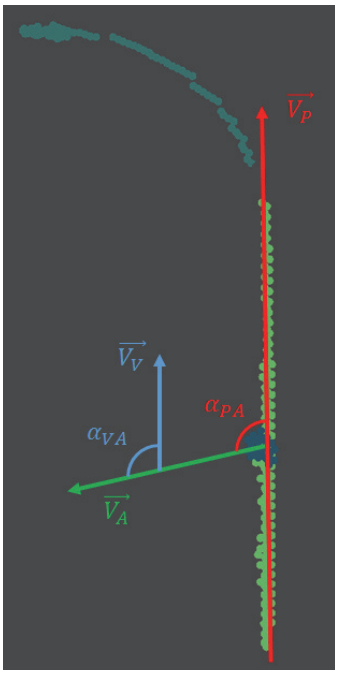


Figure 5.7: *Angle features (see text for the explanation)*

Height features $\{H\}$ are the differences of height between the position of attachments and the position of poles. They include H_{CB}, H_{CT}, H_{BB}, H_{BT}, H_{TB}, H_{TT}. The first sub-index is the index of the position of an attachment, and the second sub-index is the index of the position of its connected pole. The sub-index C denotes the centre position, B the lowest position, and T the highest position.For instance, H_{CB} is the height difference between the centre point of the attachment and the lowest point of its connected pole. The illustration of height features is as shown in **Figure 5.8**. The height features are not only applicable for the demonstrated type of pole. Except for curved poles, this type of feature works with any type of straight poles.

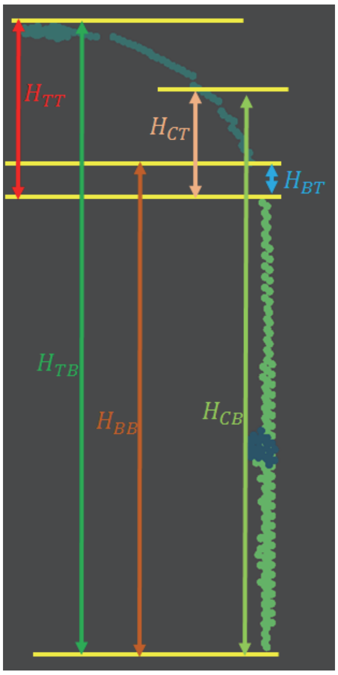


Figure 5.8*: Height features between a pole and an attachment. Green points denote pole, and the street light and the traffic sign are represented by dark cyan points and steel blue points, respectively.*

5.3.2 Knowledge-driven method

In this section, a method is proposed to assign meaningful labels to decomposed road furniture. We describe the formulation of generic rules for the recognition of attachments of road furniture.

We categorise attached segments of road furniture into 6 classes, street light, traffic sign, street sign, traffic light, connected traffic signs and other signs. Instances of these attachments are as indicated in **Table 5.1**.

Table 5.1*: Examples of attached components*

Street light	Traffic sign	Street sign

Traffic light	Connected traffic signs	Other signs

In order to recognise attachment of road furniture, we character them by generic rules based on traffic regulations. Then, based on generic rules of assembling road furniture, we distinguish the topological relations between poles and their attached segments. In this research, the connectivity between a pole and an attachment is decided by their minimum distance. If it is smaller than a threshold, it is considered to be connected.

The following rules are defined to assign a semantic label to each component of road furniture.

Street lights: If there is an attachment connected to a vertical pole, its relative height is larger than a threshold $H_{sl,}$ and it is on the top of this pole, this component will be street light. H_{sl} is the threshold of discriminant feature to recognize street light head.

$$\forall a \; IsClass\,(a, StreetLight) \;\Rightarrow IsHigh\,(H_{BB}) \wedge IsSmall\,(H_{BT}) \tag{5.22}$$

Traffic signs: If there is an attachment connected to a pole, it is not at the bottom of this pole, its relative angle is perpendicular, it is linear or planar, its area is smaller than A_{ts}, its ratio (height to length) is close to 1, and its ratio of high reflectance points is larger than R_{ts}, this component will be a traffic sign. A_{ts} is important for differentiating traffic signs from other traffic functional signs.

$$\forall a \; IsClass\,(a, TrafficSign) \;\Rightarrow IsSmall\,(a_a) \wedge IsEqual\,(S_{D,}S_H) \wedge IsHigh\,(a_{rh}) \tag{5.23}$$

$$a_a = S_D \cdot S_H \tag{5.24}$$

where a_{rh} is the ratio of high reflectance points.

Street signs: Conditions are the same as traffic signs except that the area is smaller than A_{ss} and size ratio is smaller than R_{ss}. Then this component should be a street sign or direction sign. The ratio of size is significant for distinguishing street signs from other traffic functional signs.

$$\forall a\ IsClass\ (a, StreetSign)\ \Rightarrow IsSmall\ (a_a) \wedge IsHigh\ (S_D/S_H) \wedge IsHigh\ (a_{rh}) \quad (5.25)$$

Traffic lights: If there is an attachment connected to a vertical pole, it is not planar, and its relative height is smaller than H_{sl}, this component should be a traffic light. Another case is that if there is a component connected to a horizontal pole and it is not planar, this attachment should be a traffic light.

Connected traffic signs: Traffic information signs are usually elongated in the vertical direction. Conditions are the same as street signs except that the area is larger than max (A_{ts}, A_{ss}) and size ratio is larger than R_{ct}. Then this attachment should be a traffic information sign.

$$\forall a\ IsClass\ (a, StreetSign)\ \Rightarrow Islarge\ (a_a) \wedge IsHigh\ (S_H/S_D) \wedge IsHigh\ (a_{rh}) \quad (5.26)$$

Other signs: If there is an attachment connected to a pole, its normal direction is perpendicular to the principal direction of this pole, and its ratio of high reflectance points is smaller than R_{os} , this attachment will be other sign such as a billboard.

Based on these rules, labels are assigned to the attachments. Poles are first detected by our previous work. Specifically, based on their principal direction, poles are recognised to be vertical and horizontal. Then the connectivity between the attached components and poles is analysed. Based the connectivity relations, attachments are found for every pole. Features mentioned above are produced for every attachment afterwards. Then these attachments are given labels by fitting predefined rules with these generated features. Before giving labels to the attachments, the parameters are optimised by selecting the best combination of parameters in the training area.

5.3.3 Machine learning classifiers

To classify the separated attachments, we employ four different types of machine learning classifier: support vector machine (SVM), random forest (RF), Gaussian mixture model (GMM) and naïve Bayes (NB). First, the classifiers are trained using training data with features extracted in **Section 5.3.1**. Then class labels are predicted for new unseen data using the trained classifiers.

In this research, we apply the C-SVM classifier **(Chang and Lin, 2011; Bishop, 2006)** and use the radial basis function (RBF) kernel, suggested in (**Hsu et al., 2003)**. The LIBSVM package (**Chang and Lin, 2011**) is used to train the classifiers and to perform predictions. As suggested in (**Hsu et al., 2003)** and(**Hsu and Lin, 2002)**, all features are scaled between 0 and 1, and the one-versus-one approach is used for multi-class classification..

C-SVM that uses RBF kernel contains two hyperparameters γ and C that control the complexity of the classifier. To find optimal values for γ and C, cross-validation (CV) (**Kohavi, 1995**) in combination with grid search is used. The γ-C pair with the highest CV accuracy was selected as an optimal model. In the end, the optimal model obtained from cross validation is used to evaluate the performance of the classification model on an independent test set.

To improve the classification performance and to analyse the goodness of different extracted features, also feature selection is performed. We use backward elimination (**Guyon and Elisseeff, 2003**), as it is able to take into account the joint effects of various features on the classification accuracy.

The random forest classifier as an ensemble learning method takes advantage of the combination of multiple weak learners. It has been widely employed in classification applications (**Criminisi et al., 2012**). Bootstrap aggregating in conjunction with information gain is utilised to construct a group of trees called random forest (**Breiman, 2001**). Two hyperparameters, the number of trees N and the depth of trees D, are in the random forest classifier. The optimal combination of these two hyperparameters is selected using a 2D grid search. In order to analyse the utility of our designed features, the importance of every feature is also computed by the Gini index, which is described in **(Louppe et al., 2013)**.

The Gaussian mixture model is often used in combination with k-means clustering which gives initial clustering information. Instead of using the k-means clustering, our training data is manually labelled. In GMM, the distribution of an attachment belonging to a class follows a multivariate Gaussian distribution. We used the Expectation Maximization (EM) to estimate all the mixture parameters. In the end, the label of an attachment is given to the class with the highest probability.

As a method based on Bayes' theorem, the naïve Bayes classifier assumes every pair of features is conditionally independent given the class (**Murphy, 2012**). It is also assumed that every feature given their labels follows a Gaussian distribution. Based on Bayes' theorem and Maximum A Posteriori (MAP), the probability of the label of an attachment is estimated. In the end, every attachment is given the label with the highest probability.

5.3.4 The extraction of non-changed attachments in two epochs

In order to test the robustness of our framework, we test the difference of predictions of non-changed attachments in two epochs. In this research, non-changed attachments are attachments appearing in both epochs of the Saunalahti dataset in the same position. To extract the non-changed attachments, we first detect non-changed poles and obtain their attachments. The non-changed poles are extracted by comparing the 2D distance of centre points of poles from two epochs. If the distance is less than a predefined threshold, they are identified as non-changed poles. The connected attachments of the non-changed poles are obtained afterwards. Then we compare the centre position of bounding boxes of the obtained attachments from two different epochs. If the distance of centre positions is less than a threshold, they are detected as the non-changed attachments. Otherwise, they are identified as changed attachments.

5.4 Study area and experimental results

5.4.1 Test sites

To conduct the experiment, we chose two test sites to evaluate the performance of our framework. In both datasets, there are many types of scanned road furniture.

The Enschede MLS trajectory is about 5.5 km long along which more than one hundred million points were recorded. The point density ranges from 35 points 350 points per square metre. There are six scanning strips used as training data. In this research, one scanning strip is the MLS data corresponding to a continuously scanned street trajectory. There are 210 pieces of road furniture in the training dataset. Another four scanning strips, which contains 260 pieces of road furniture, are taken as a testing area. The size of every block is set to be 50 m long and 40 m wide. The configuration of system collecting this dataset is introduced in **Section 2.2.1**, **Chapter 2**. The strip overview of this dataset is as shown in **Figure 5.9.**

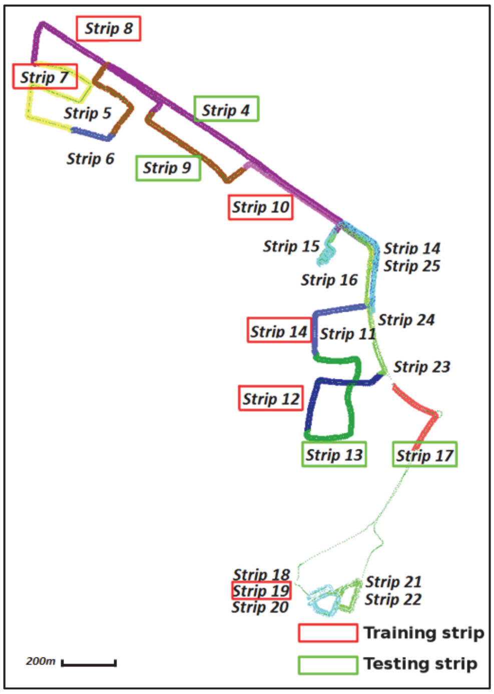


Figure 5.9*: Enschede dataset. The strips without red or green outlines are not utilised in the experiment.*

The Saunalahti winter dataset covers about 5.0 km of road, and the spring dataset covers about 4.0 km of road, consisting of more than one hundred million points. The distance of the common area, of which the data was collected in both the winter of 2016 and the spring of 2017, is about 2.5 km. In the winter dataset, six scanning strips are chosen as training data, and seven strips as testing data. The training datasets and testing datasets contain 111 and 187 pieces of road furniture, respectively. In the Saunalahti spring dataset, four strips are used as training data, which comprises 100 pieces of road furniture. Another six strips are chosen as testing data, in which there are 241 pieces of road furniture. In the data partitioning stage, the size of every block is set to be 50 m long and 40 m wide. The strip overview of this dataset is shown in **Figure 5.10**.

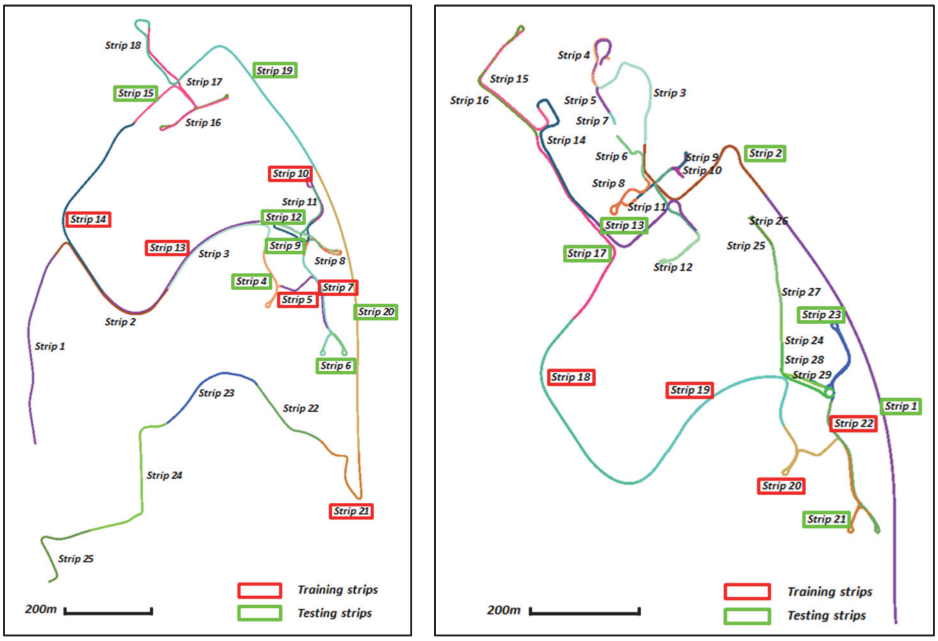


***Figure 5.10**: The strip view of Saunalahti winter dataset (left) and Saunalahti spring dataset (right). The strips without red or green outlines are not utilised in the experiment.*

5.4.2 Results

We evaluate the results of pole-like road furniture interpretation with four aspects. We first compare the performance of the knowledge-driven method and four machine learning classifiers. Then the confusion matrix is given and analysed with the method, which achieves the best performance. The reliability of our framework is evaluated by the difference in the interpretation of corresponding road furniture in two epochs afterwards. In the end, we quantitatively assess the transferability of our framework by swapping trained models.

To produce the ground truth data for the evaluation of interpretation, we only use detected pole-like road furniture. The tag of every attachment is manually labelled. We treat our proposed framework as an automatic procedure. Therefore, undetected pole-like road furniture is not included in the ground truth data. We compare the performance of five different methods for the classification of attachments of pole-like road furniture in the three datasets mentioned above. The first one is the knowledge-driven method (**Li et al., 2017)**. In this method, a set of rules is defined based on the generic mounting patterns of road furniture. For instance, street lights are often mounted on the top of poles and above a certain minimum height. The details of this method can be found in **Li et al. (2017)**. The other four methods are the machine learning classifiers, SVM, random forest, GMM and naïve Bayes introduced in **Section 5.3.3**. The setting of hyperparameters of machine learning classifiers is as illustrated in **table 5.2**. The quantitative evaluation of the performance is as shown in **Table. 5.3.** The table indicates that random forest the other four methods. The precision of correctly interpreted attachments is higher than 80% in Enschede dataset and higher than 90% in both Saunalahti spring and winter datasets. The precision utilised in this research is the number of correctly recognised attachments divided by the number of all attachments. The ratio of complex pole-like road furniture is almost 50.0% in both Enschede and Saunalahti test sites. More than 68.0% attachments in two test sites cannot be classified correctly by previous frameworks (**Yang et al., 2015; Yu et al., 2016; Wang et al., 2017a**).

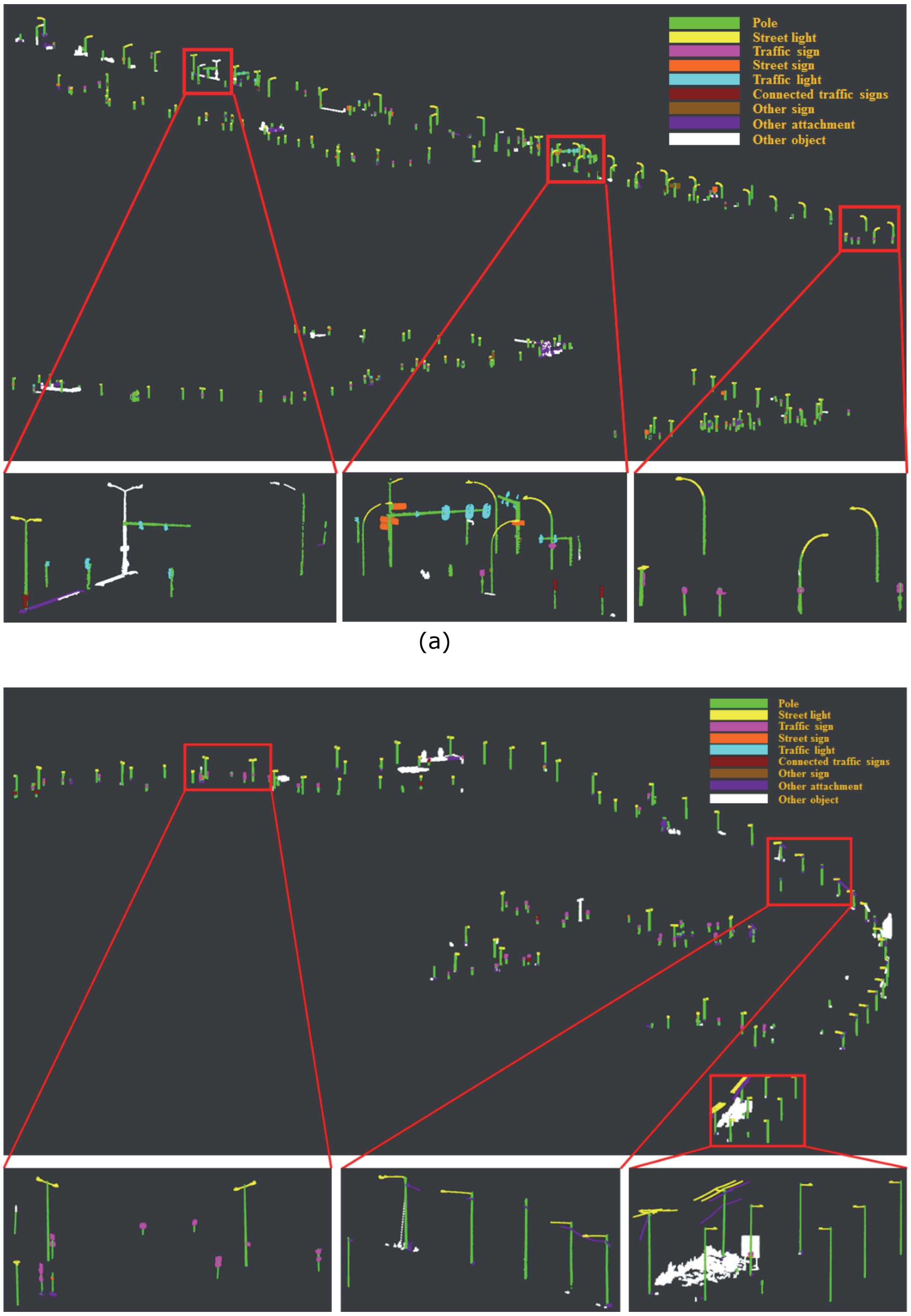
Table 5.2*: The setting of hyperparameters of machine learning classifiers*

Test sites	SVM		Random forest	
	γ	C	D	N
Enschede	0.05	8	10	20
Saunalahti winter	0.05	5	6	33
Saunalahti spring	0.02	5	6	40

Table 5.3*: The performance of road furniture interpretation of five different methods*

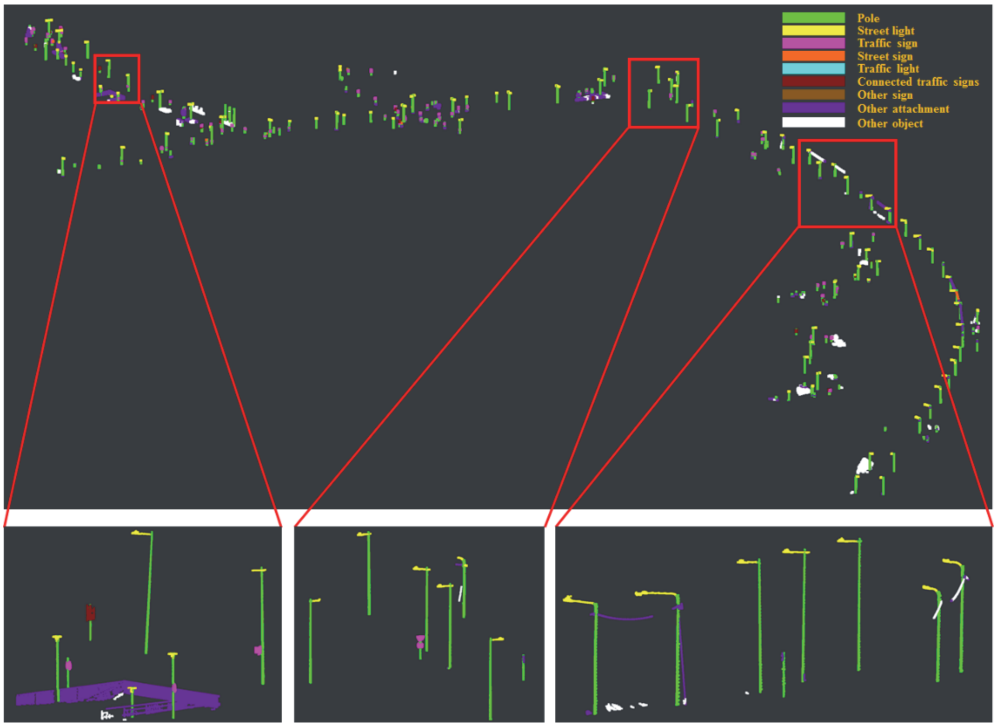
Test sites	Knowledge-driven	SVM	Random forest	GMM	Naïve Bayes
Enschede	64.4%	72.2%	81.0%	29.0%	59.1%
Saunalahti winter	51.5%	79.1%	92.3%	35.4%	86.8 %
Saunalahti spring	59.5%	78.2%	94.1%	40.7%	86.8%

A screenshot of interpreted road furniture (trained and predicted by the random forests model) in these three test sites is as shown in **Figure 5.11.** In this figure, vertical poles and horizontal poles are in green colour, and street lights are coloured to be yellow. Magenta attachments are traffic signs, and orange attachments are street signs. Cyan attachments represent traffic lights and maroon attachments represent connected traffic signs (**Figure 5.12**). Other signs and other attachments are colourised to be brown and purple respectively. White attachments are other objects. **Figure 5.11** indicates that road furniture is well interpreted in both Saunalahti winter and spring datasets.


🟩 Pole
🟨 Street light
Traffic sign
🟧 Street sign
Traffic light
🟥 Connected traffic signs
🟫 Other sign
🟪 Other attachment
⬜ Other object

(a)

(b)



(c)

Figure 5.11*: The visualisation of interpreted road furniture in (a) Enschede dataset, (b) Saunalahti winter dataset and (c) Saunalahti spring dataset*

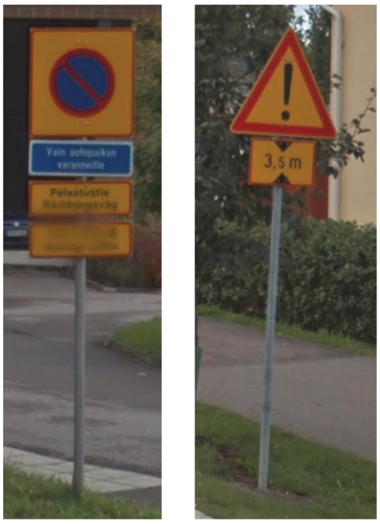


Figure 5.12*: Connected traffic signs*

In addition, we also give the confusion matrices of the results on the three test sites. The confusion matrices for the random forest classifier are as shown in

Table 5.4, Table 5.5 and Table 5.6. From the confusion matrix, it can be seen that the accuracy of recognition of street light is the highest. The recognition rate of the street sign and connected traffic sign is much lower.

Table 5.4: *The confusion matrix of results in Enschede dataset*

	Street light	Traffic sign	Street sign	Traffic light	Connected traffic sign	Other signs	Other attachments	Correctness
Street light	148	0	0	0	0	1	1	98.7%
Traffic sign	0	43	10	0	4	9	1	64.2%
Street sign	0	9	41	0	0	1	0	80.4%
Traffic light	0	1	0	17	0	6	1	68.0%
Connected traffic sign	0	3	3	0	13	0	0	68.4%
Other signs	0	7	0	1	2	13	3	50.0%
Other attachments	1	3	2	5	1	9	84	80.0%
Completeness	99.3%	65.2%	73.2%	73.9%	65.0%	33.3%	93.3%	

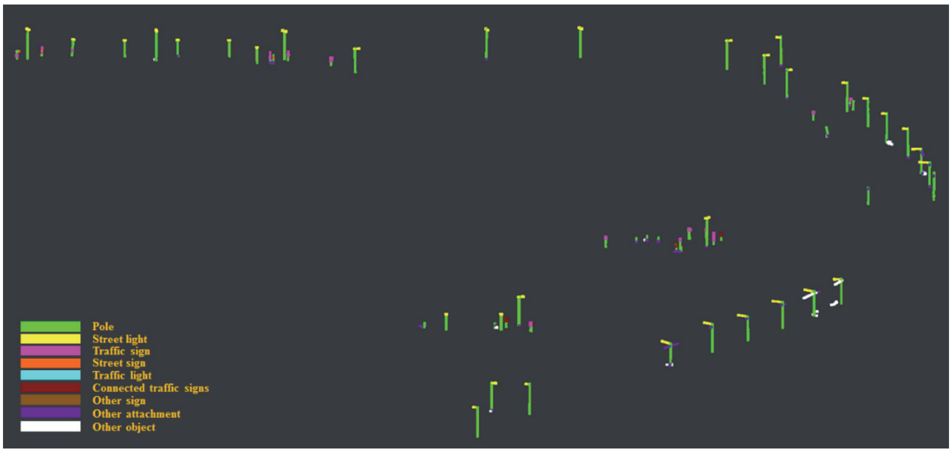
***Table 5.5**: The confusion matrix of results in Saunalahti winter dataset*

	Street light	Traffic sign	Street sign	Connected traffic signs	Other attachment	Correctness
Street light	90	0	0	0	5	94.7%
Traffic sign	0	83	1	4	3	91.2%
Street sign	0	2	4	1	1	50.0%
Connected traffic signs	0	3	0	15	0	83.3%
Other attachment	2	4	0	1	144	95.4%
Completeness	97.8%	90.2%	80%	71.4%	94.1%	

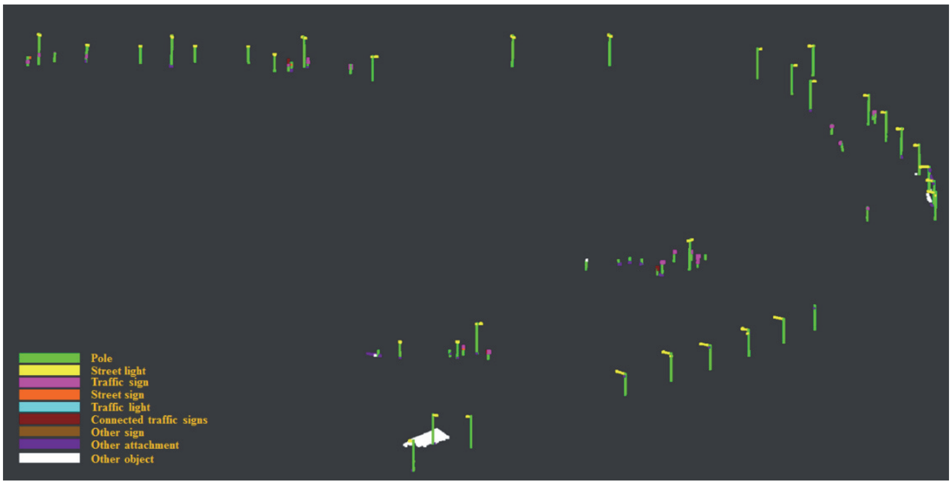
***Table 5.6**: The confusion matrix of results in Saunalahti spring dataset*

	Street light	Traffic sign	Street sign	Connected traffic signs	Other attachment	Correctness
Street light	119	0	0	1	8	93.0%
Traffic sign	0	106	4	6	3	89.1%
Street sign	0	1	13	0	1	86.7%
Connected traffic signs	0	0	0	20	0	100.0%
Other attachments	2	1	0	2	205	97.6%
Completeness	98.3%	98.1%	76.5%	69.0%	94.5%	

In order to analyse the robustness of our framework, we apply our framework to two different epochs of the Saunalahti dataset and check the difference of interpretation on the non-changed attachments. The detection of non-changed attachments in both epochs of Saunalahti test site is explained in **Section 5.4.3**. The threshold of the distance between centre points of a pair of corresponding attachments in two epochs is set to be 0.8m. There are 68 non-changed attachments, among which 60 are given the same label. The similarity rate of the prediction of the same attachments is 88.2%. Therefore, there is high robustness with our model. The interpretation of non-changed road furniture is as shown in **Figure 5.13a, b**. The 8 objects are predicted with different labels because of incorrect decomposition (as illustrated in **Figure 5.14a**) and stray points (as shown in **Figure 5.14b**). In this research, stray points are noisy points caused by the scanning process. In **Figure 5.14a**, from left to right, the two figures on the left side are the decomposition and the interpretation of road furniture in the Saunalahti winter dataset, and the two figures on the right side are the decomposition and interpretation of road furniture in the Saunalahti spring dataset. The traffic signs in the Saunalahti winter dataset (the magenta attachments in the second left figure of **Figure 5.14a**) are separated as shown in the first left figure of **Figure 5.14a**. Because of the incorrect decomposition, the upper two attachments are connected (the sea green attachment in the third left figure of **Figure 5.14a**), which leads to the wrong prediction of these two attachments. They are mispredicted as a connected traffic sign (the maroon attachment in the fourth left figure of **Figure 5.14a**). In Fig. 15b, the connected traffic sign (the left figure) is correctly predicted in the Sauanalahti winter dataset. There are noisy points surrounded with the connected traffic signs in the right figure of **Figure 5.14b**. The calculation of size features and eigenvalue-based features is not reliable, which triggers the wrong prediction.

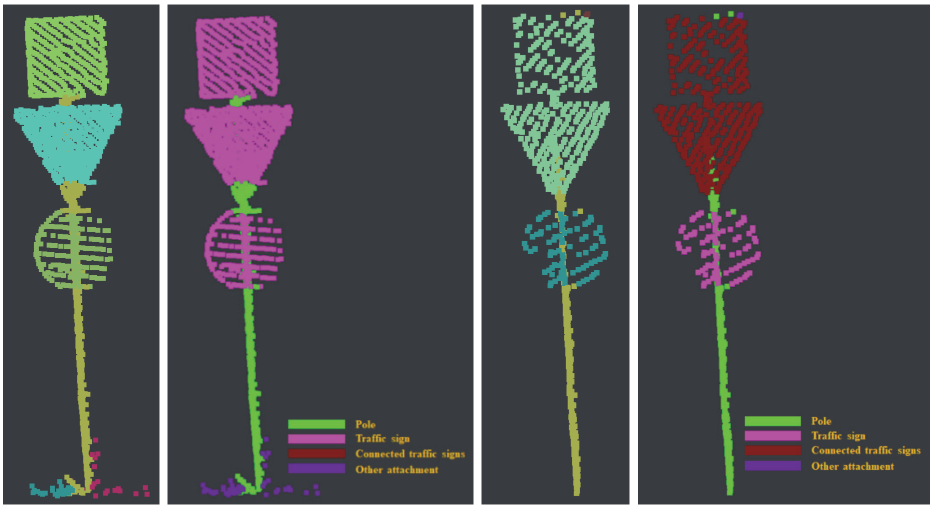


(a)

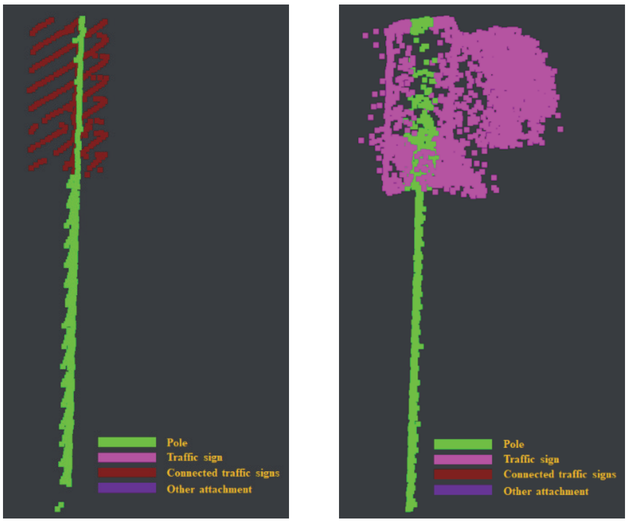


(b)

***Figure 5.13**: The interpretation of same road furniture in two epochs in Saunalahti. The interpretation of corresponding road furniture in (a) Saunalahti winter dataset and (b) Saunalahti spring dataset.*



(a)



(b)

***Figure 5.14**: (a) The decomposition and interpretation of road furniture and (b) point clouds with noisy points. The left two figures in (a) and the left one figure in (b) are from Saunalahti winter dataset. The right two figures in (a) and the right figure in (b) are from Saunalahti spring dataset.*

We also evaluate the transferability of our framework. Transferability in this research refers to two different situations. First, transferability refers to the capability of applying a model, which is trained with a data that is collected in one epoch to the predict the labels of a dataset collected in another epoch, both epochs in the same area. Second, transferability refers to the capability of applying the trained model in a new area. We evaluate the transferability by testing the performance of the trained model of the Saunalahti winter dataset on the Saunalahti spring dataset, and vice versa. As explained above in **Section 5.4.1**, we collected the data in Saunalahti test area in both winter and spring. There are also differences caused by the construction work, and therefore the two datasets are not completely the same. Consequently, they are used for model transferability analysis. The transferability of the random forest model is as shown in **Table 5.7**. When the trained model of winter dataset is applied to predict the spring dataset, the accuracy is 92.5%. When the trained model of spring dataset is used to predict the winter dataset, the accuracy is 91.2%. It demonstrates that the transferability of our random forest model is high when two datasets share the same types of road furniture. Aside from the analysis of transferability of different epochs in the same area, we also test the transferability of trained models between different test sites. We use the trained model of Enschede test site to predict the label of attachments in the Saunalahti test site. The overall accuracy of prediction is 65.8% and 69.7% in Saunalahti winter and spring dataset respectively. The recognition rate of street light is still higher than 87.0%. In Enschede test site, there are seven types of attachments. Traffic lights and other signs without traffic functionality are not included in the Saunalahti test site. The main error is from the misrecognition of traffic signs as other signs. This is because there are significant differences of radiometric attributes between the Enschede test site and Saunalahti test site.

***Table 5.7**: The performance of transferability of trained models*

Model transferring	Random forest
Saunalahti winter training + Saunalahti spring testing	92.5%
Saunalahti spring training + Saunalahti winter testing	91.2%
Enschede training + Saunalahti winter testing	65.8%
Enschede training + Saunalahti spring testing	69.7%

5.5 Analysis

We analyse the experimental results with three points. The reason why the RF classifier outperforms the SVM classifier is first investigated. We explain why the interpretation of some types of attachments only achieves low accuracy. To confirm the effectiveness of our designed features, we calculate their importance and analyse the reason.

The experimental results indicate that the random forest classifier performs the best and the GMM classifier performs the worst. SVM performs worse than random forest. Random forest is a combination of weak learners, and SVM recognises two classes by maximising the margin. In the SVM classifier, the backward elimination is performed to select the most distinctive features. Backward elimination starts with taking all features and removing each feature in turn. With each feature removed in turn, the CV accuracy is evaluated. The model with the lowest CV accuracy is discarded, and the corresponding feature is eliminated from the set of all features. This process is repeated until only one feature is left. As a result, an elimination curve is obtained in which the CV accuracy after each elimination is plotted as a function of the elimination round. The feature set with the highest CV accuracy is chosen as optimal. The starting point of backward elimination is actually in the right side of **Table 5.8**, where all features are used. In the three test datasets, within the first eleven most important features all the five types of features (size features S, eigenvalue-based features E, radiometry features R, angle features A, relative height features H) are present. **Table 5.8** illustrates the maximum performance of

SVM can be reached with only eleven features. The performance of SVM classifier is not improved by adding extra features. This indicates that the performance of SVM classifier converge after using the most distinctive features (the first eleven features), that is, it is difficult for SVM to use designed features effectively. The reason that the accuracy does not improve after the first eleven can be that the rest features (or at least part of them) correlate strongly with (some of) the first eleven ones. Then they are redundant and may not improve the accuracy (Guyon and Elisseeff, 2003). SVM is more feasible for solving the problem in case there are many features with less correlation.

Compared to the random forest model, the knowledge-driven method performs much worse. The knowledge-driven method was based on generic rules in combination with a few features, whereas random forest took advantage of the distribution of features to construct many weak learners. The reason is that it is difficult to use a few generic rules with a rather limited combination of features to differentiate similar objects. As shown in **Figure 5.15**, traffic signs are not correctly classified by the knowledge-driven method, whereas random forest classifier predicts the label of traffic sign correctly by combining the weak constraints of designed features. The combination of weak learners is similar to a combination of many manually designed rules. Via these weak learners, the excellent performance of the classification of attachments is achieved. However, it is difficult for a knowledge-driven method to use as many rules or learners as random forest to perform the classification. The knowledge-driven method does not need that much training data. Therefore, it is suitable to solve classification problems with few classes and limited data, or to make the initial training data, which is manually corrected and improved.

Table 5.8: Feature selection of SVM in three datasets

Test datasets	The first eleven features					Convergence accuracy	The rest of features	Accuracy with all features
	H	S	E	A	R		H, S, E, A, R	
Enschede	H_{CT}, H_{BT}, H_{TT},	S_H	E_E, E_{LP}, E_L, e_3,	A_{VA}	R_{MA}, R_{MD}	75.6%	E_S, E_P, E_O, S_{D-H}, E_A, S_W, H_{TB}, H_{BB}, e_1, S_V, S_D, S_L, H_{CB}, e_2, A_{PA}	75.6%
Saunalahti winter	H_{TB}, H_{CB}, H_{BT}	S_H	e_1,E_O , e_2, e_3, E_{LP}	A_{VA}	R_{MA}	82.0%	H_{BB}, E_S, E_E, H_{TT}, R_{MD}, E_P, S_W, H_{CT}, E_A, A_{PA}, S_V, S_{D-H}, S_D, S_L, E_L	83.3%
Saunalahti spring	H_{BT}, H_{BB}, H_{TB},	S_{D-H}, S_W	E_E, E_O, E_{LP}, E_L	A_{VA}	R_{MD}	83.2%	R_{MA}, E_S, E_P, S_D, e_2, H_{CT}, H_{TT}, H_{CB}, S_H, e_3, E_A, e_1, S_V, S_L, A_{PA}	83.8%

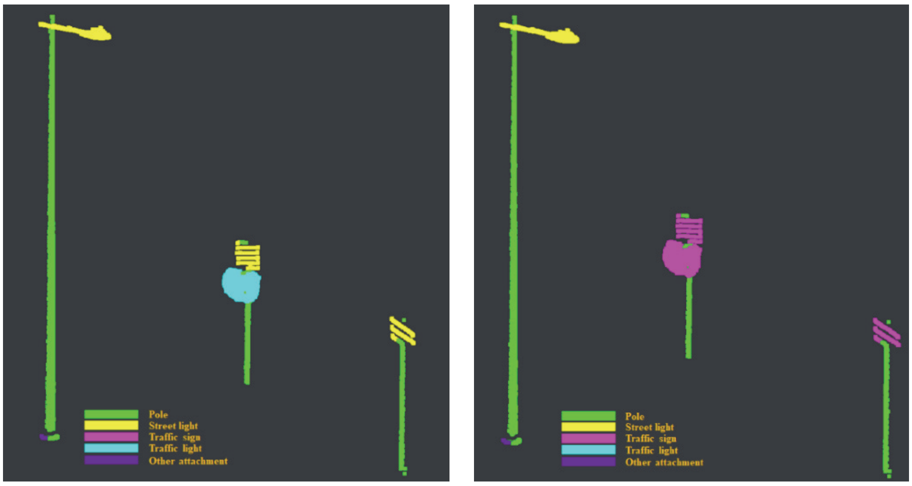
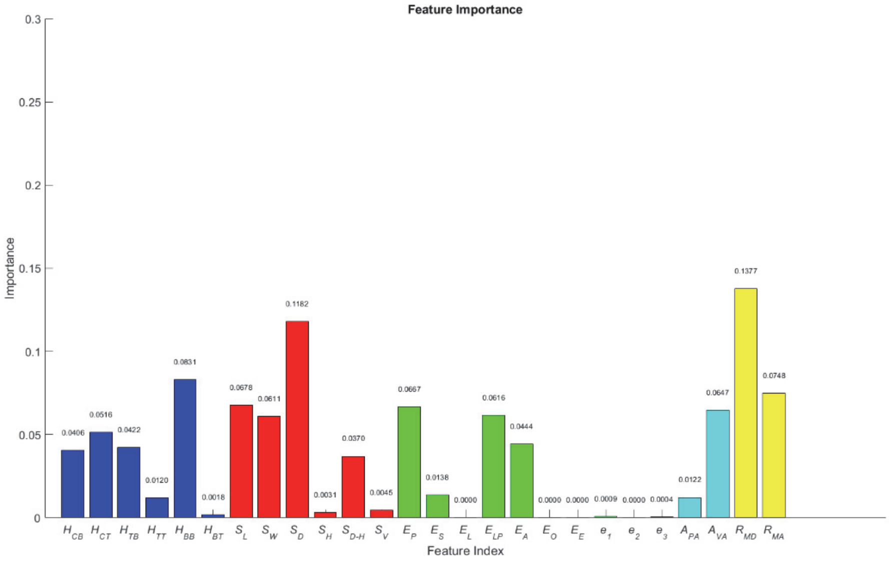


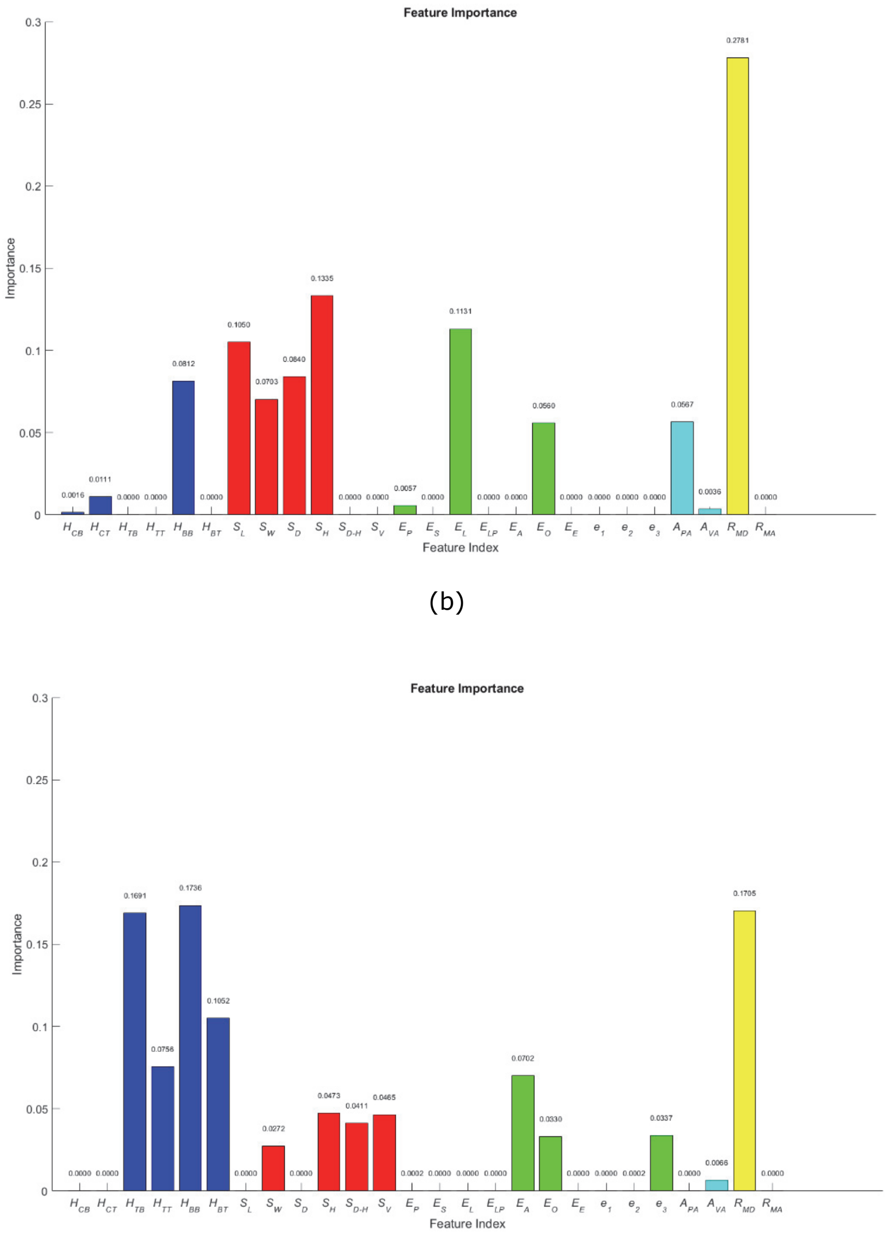
***Figure 5.15**: The result of the interpretation of road furniture from the knowledge-driven method (left) and the random forest classifier (right).*

In the classification of attachments, the accuracy of the recognition of street light is the highest, higher than 90%. In contrast, the recognition rate of street signs, connected traffic signs and other signs is low, less than 80% in Saunalahti test site. One reason is that there are not many training samples and testing samples. For instance, in Saunalahti spring dataset, there are only 5 street signs in the training dataset and 17 street signs in testing dataset. By contrast, there are 58 street lights in the training dataset and 121 street lights in the testing dataset. The classifier can be biased to the class with more training samples. Another reason is that the designed features of other signs and traffic signs are quite similar in structural view. For example, both bus schedule plates and traffic signs are in drivers' view and perpendicular to the principal direction of poles. When manually making ground truth dataset, colour imagery was used to aid the labelling. It is difficult to distinguish the classes well from point clouds only.

To verify the effectiveness of the designed features, we also explore the feature importance in the random forest model. The importance of features in three test sites is as shown in **Figure 5.16a-c**. In **Figure 5.16a**, these bins with respective colour represent relative height features (blue), size features (red), eigenvalue-based features (green), relative angle features (cyan) and radiometric features (yellow). Radiometric features play the most significant role in all three datasets because they discriminate traffic functional signs and attachments. Relative height and size features play an important role in the interpretation of road furniture in all three datasets. This is because there are significant differences in relative height and size between different classes. Relative height features play the most significant role in the Saunalahti spring dataset. Point clouds of signs are rather noisy as aforementioned, and the calculation of other features is not reliable, especially the calculation of the normal direction of signs. That is the reason why relative height features are so important and the importance of relative angles features is low. For instance, street lights are often mounted above a certain height, and signs should be within drivers' view field. The calculation of relative height features is less affected by stray points in the point clouds of attachments. For instance, the most important relative height feature is relative height between the bottom position of an attachment and the bottom position of its connected pole. The calculation of this feature is relevant to the extraction of poles, which is robust and less affected by the stray points around attachments. In contrast, the calculation of other features is more relevant to the noise level of point clouds of attachments. When noisy points surround them, the calculation of these features is unreliable. For instance, the size features are not authentic because of the surrounding noisy points of attachments. Therefore, the calculation of designed relative height features is robust even in noisy point clouds. Compared to other features, the angle features are not important because all signs are mounted under the same angle. Hence, the angle features cannot discriminate different types of signs. In Saunalahti winter dataset, there is a lower level of noisy points. The calculation of size and geometric features is reliable, which contributes more to the random forest classifier. It leads to the less importance of other features such as relative height features. Compared to the importance of relative height features in other datasets, relative height features in the Saunalahti winter dataset are less important (**Figure 5.16b**).



(a)



(b)

(c)

Figure 5.16*: Feature importance in random forest in (a) Enschede dataset, (b) Saunalahti winter dataset and (c) Saunalahti spring dataset, blue bars represent relative height features, size features are denoted by red bins, green bins indicate eigenvalue-based features, cyan bins are relative angle features, radiometric features are represented by yellow bars.*

5.6 Conclusion

We draw the conclusions as four parts: the contribution of this paper, the comparison between different machine learning classifiers, the impact of feature importance and the reliability and transferability of our framework.

To conclude, in this chapter, we first decompose a point cloud of detected road furniture to separate poles and their attachments. Secondly, we generate features on attachments, pole and between attachment and poles in order to interpret road furniture. With the decomposition processing, our framework is able to interpret pole-like road furniture at a detailed level. Our framework represents a detailed interpretation of pole-like road furniture and presents the state of the art of pole-like road furniture interpretation at a functionality level, which is of great significance for precise mapping and autonomous driving. The research shows great promising for 3D precise mapping in the urban road environments.

Of the four machine learning classifiers, random forest outperforms the rest of the classifiers. When there are attachments with features with small differences but with potential rules, it is difficult to distinguish them by constructing hyperplanes. Even though a non-linear kernel function is able to enhance the performance of SVM classifier, a limited number of complex hyperplanes are still not able to cover a large number of weak rules, which can distinguish different classes. However, with weak learners which are trained with the bagging strategy, these attachments can be differentiated. When the difference between features is not distinctive but with potential weak rules, the random forests is a good choice. The recognition accuracy of street light heads is higher than that of other types of furniture components because of its rich samples in the training data.

The analysis of feature importance indicates that the designed contextual features play an important role in the interpretation of road furniture. The relative height features play the most important role in the point clouds which contains numerous stray points. It demonstrates our framework is less affected by stray points because of the combination of designed features. The decomposition significantly affects the interpretation stage. The decomposition stage decides the generation of attachments. If there is no decomposition, complex pole-like road furniture with multiple classes cannot be discriminated reliably. Thus, features of attachments cannot be reliably extracted, and these attachments cannot be classified correctly. The decomposition significantly affects the interpretation stage. A decomposition with reliable boundary preserved and correct number of attachments is of great significance to the extraction of contextual features. Therefore, the decomposition also plays an important role in the interpretation of road furniture.

The difference of performance metrics of the interpretation of corresponding road furniture between two temporal datasets is not large, which confirms the reliability of our proposed framework. When applying the trained model of one epoch dataset to another epoch dataset, the achieved accuracy is higher than 90%. The transferability of our proposed framework between temporal datasets collected in the same area therefore is high, which proves the great potentiality of our framework applied to change detection and map updating.

Chapter 6 – Conclusions and Recommendations

6.1 Conclusions and research questions answered

The primary objective of this research is to develop a framework to interpret pole-like road furniture based on their functionalities. In this research, we have presented a framework to semantically segment pole-like road furniture based on their functions. Our contribution is made on three aspects:

1. We present a novel method to detect pole-like road furniture in unevenly and sparsely distributed MLS data. The proposed method does not require high and even point density. In addition, we remove objects behind building facades, which are incorrectly detected as pole-like road furniture.

2. We propose a method to decompose detected pole-like road furniture into poles and attachments. In this method, we innovatively introduce three different algorithms to extract poles from three types of pole-like road furniture. Merging and splitting rules are proposed to refine the initial separation of attachments. These rules significantly improve the performance of initial decomposition. The decomposition processing separates poles and their attachments and provides essential information for the detailed interpretation of pole-like road furniture.

3. We propose an approach to interpret decomposed pole-like road furniture based on their functionalities. Unary features and contextual features are designed to interpret pole-like road furniture. We compare the performance of a knowledge driven method and four machine learning classifiers, of which the random forests classifier performed the best. The designed features play important roles in the interpretation of road furniture. Our method has pushed the classification of road furniture to a more detailed level, the semantic segmentation of road furniture based on their functionalities.

The conclusions and the research questions related to each sub-objective are drawn and answered in this section.

Pole-like road furniture detection

With respect to the first sub-objective, we propose a method to detect pole-like road furniture in sparsely and unevenly distributed mobile laser scanning data.

To conclude, our method is stable and consistently performs well on three different datasets representing different laser ranging techniques and different scanning geometry patterns of a mobile laser scanning system. Objects behind building façades are excluded by an occlusion analysis. The calculation of the designed features in this method does not rely on high point density or evenly distributed points. This is because the 2D connected component analysis was used to detect slices which belong to poles and the number of detected slices to identify pole-like road furniture. The calculation these two features is robust and does not strongly rely on the point density and the evenness of point distribution. This method works well not only in sparsely and unevenly distributed mobile laser scanning data. The performance of the detection of pole-like road furniture in densely mobile laser scanning data is competitive to state of the art as well. Our algorithm could potentially be of practical use in the pole-like road furniture extraction in sparse point clouds. An example of such sparse data is MLS data collected using Velodyne laser scanners, also widely used in autonomously driving vehicles.

Related research questions

1. Which features can be used and how can they be used to remove objects that do not belong to street furniture?

We use local height difference and elevation of the trajectory line to detect and remove ground points. The detection of ground points is on the basis of that ground points have a low local height difference, and they are lower than the elevation of corresponding trajectory points.

Area, orientation, height and width of a detected planar patch are used to detect and remove buildings. The area of a building façade should be large, and its orientation should be vertical. The height and width of a building façade should not be too small. Based on these constraints, buildings are detected. The designed features are robust descriptors of building facades in MLS data. The detected building facades in combination with trajectory lines are utilised to eliminate objects behind building façades in the occlusion analysis.

The ratio of points with the first return number in above ground objects is utilised to detect and remove trees.

2. How to detect complex street furniture with incomplete, unevenly distributed and low-resolution data, or connected with other objects?

We propose a method which combines the slice cutting algorithm and the cylinder fitting algorithm to detect pole-like road furniture. The combination of these two algorithms successfully excludes trees even without using the number of pulse count. The performance of pole-like road furniture detection in unevenly and sparsely distributed MLS data outperforms other known methods.

3. How to use the result of road furniture decomposition to refine the detection result?

We use the feedback of the decomposition of road furniture to recheck the detection result. If there are no poles extracted from one piece of pole-like road furniture in the decomposition process, we then remove this piece of road furniture from detected pole-like road furniture. The correctness of pole-like road furniture detection is thereby improved to over 90%.

Pole-like road furniture decomposition

We propose a method to decompose pole-like road furniture into poles and attachments in MLS data. With respect to three types of pole-like road furniture, three corresponding methods are proposed. We detect the type of road furniture and pick up the corresponding method to extract poles automatically. We extract centre lines of poles by the following a two-step procedure: (1) the initial centre line detection by these three methods and (2) the optimisation of centre lines based on the initially extracted poles. We use the centre line of the pole and the distance of points to this centre line to extract points which belong to this pole. The extraction of poles does not rely on numerous constraints of features, which makes the method strongly generic. The optimised pole extraction effectively refines the initial extracted centre lines of poles, and it leads to a better pole extraction. After the pole extraction, a connected component analysis is carried out to separate attachments and obtain the initial decomposition. Then merging and splitting rules are performed to optimise the decomposition. The completeness of decomposition is improved to over 80%. The merging and splitting rules mitigate the over- and under-decomposition. All in all, the optimised pole extraction and merging and splitting rules improve the performance of the initial decomposition of pole-like road furniture in MLS data. The decomposition of pole-like road furniture provides the knowledge of instance segmentation for the interpretation stage.

Related research questions

1. Which multi-scale features and contextual features can be used and how can they be used to decompose street furniture into meaningful components?

In the first pole extraction method, we use the 2D point density feature to initially extract poles. Multiple-scale eigenvalue-based features are used in the second pole extraction method to extract points with a highly linear neighbourhood, so that centre lines of poles can be extracted from these points with high linearity. Multiple-scale eigenvalue features are used to extract points with high linearity and the RANSAC algorithm was used to extract centre lines of poles. In the third method, we propose a slice cutting algorithm to extract poles. The extracted centre lines of poles and the distance of points to the extracted centre lines are the two main clues in the slice cutting algorithm.

In the merging and splitting rules, the diameter of the cylinder used to extract poles is changed, and the spatial relations between neighbouring attachments are obtained. The spatial relations between neighbouring attachments are the main clues to decide on merging or splitting components.

2. How to decompose street furniture with incomplete, unevenly distributed and sparse data? How to separate street furniture connected with other objects?

Instead of using complex feature constraints, we use the centre lines of poles and the distance of points to these centre lines to extract poles. The rule is generic and robust. Once poles are extracted, objects connected to poles, such as bicycles and trees, can also be separated from road furniture as long as they are not connected to attachments of road furniture, such as traffic signs and street lights.

3. How to decompose a wide range of complex street furniture with many irregular elements?

Poles are the connection between different attachments. This clue indicates that we can separate different attachments by removing poles. Based on this clue, we propose generic rules to extract poles and then separate the rest of the attachments.

4. How to address the over-decomposition and the under-segmentation of road furniture?

Due to the limited number of training samples, we decide to utilise generic rules to mitigate the over-decomposition and under-decomposition instead of using machine learning algorithms. The optimisation of pole extraction and merging and splitting rules are proposed to mitigate the over-decomposition and under-decomposition. The completeness of decomposition is improved to over 80%.

Pole-like road furniture interpretation

We propose a knowledge-driven method and applied four machine learning classifiers to interpret pole-like road furniture in MLS data based on their functionalities. We analyse the difference in performance of these five methods, among which the random forests classifier outperforms the knowledge-driven method and the other classifiers. It proves that the random forest classifier is well suited for the classification task with potential rules. We design unary features and contextual features as the input for the knowledge-driven method and machine learning classifiers. We validate the effectiveness of our designed features by computing the importance of designed features. There is no significant difference of interpretation between two epochs dataset in Saunalahti test site, which indicates the stability of our trained model. The transferability of the learned random forests model to different temporal datasets is good. This potentially enables the use for change detection and map updating. Although the performance of our knowledge-driven method is not the best, it is able to provide the initial semantically labelled MLS data to create training data. Most errors come from the over-decomposition and under-decomposition. The decomposition stage strongly affects the interpretation because the result of decomposition is the input of interpretation. Altogether, our framework represents the state of the art of pole-like road furniture interpretation at a functionality level, which is of great significance for 3D precise mapping and autonomous driving.

Related research questions

1. Which (local) features of components can be selected to describe specific street furniture?

In total, five types of features are designed to interpret road furniture. They are size, eigenvalue-based features and the radiometric features of attachments, the relative angle features and relative height features. The last two types of features are particular meant to describe the relation between poles and their attachments. These relations are only possible after the decomposition.

2. How to use these local features (features of components) to interpret street furniture?

A knowledge-driven method is proposed, and four machine learning classifiers are applied to interpret pole-like road furniture in MLS data. We also test the importance of designed features with the machine learning classifier with the best performance. Experimental results indicate that our designed features play important roles in the interpretation of pole-like road furniture.

3. Which contextual features of street furniture can be used for street furniture interpretation?

The relative angle features and the relative height features are designed to improve the interpretation of road furniture. The relative angle features consist of the angle between the principal direction of poles and the normal direction of their attachments and the angle between the vertical direction and the normal direction of attachments. The relative height features are the differences of height between the position of attachments and the position of poles. These designed contextual features play significant roles in the interpretation of pole-like road furniture.

4. How to use primitive fitting and part template fitting to label various kinds of street furniture components?

We design primitive models to describe different types of attachments of road furniture. These models are constructed based on feature constraints which are obtained from numerous observations. These features are the sizes of attachments, eigenvalue-based feature of attachments, relative angles and relative height.

5. Which machine learning classifiers can be used to improve the performance of road furniture interpretation if the proposed template fitting method does not work decently?

We have tested four machine learning classifiers on pole-like road furniture interpretation. They are random forests, support vector machine, Gaussian mixture model and naïve Bayes. The random forests classifier currently is best suited to our interpretation problem.

6.2 Recommendations

Existing problems and future work

We have proposed the decomposition method to separate pole-like road furniture connected to other objects. However, it is still difficult to separate pole-like road furniture connected to other objects such as trees and building façades. The use of imagery or multispectral LiDAR could further help in distinguishing between the objects, i.e. a pole close to a facade should be detectable even with a single channel intensity when there is a large enough intensity gradient between the two objects. However, the use of colour information may lead to over-segmentation of single objects. How to separate objects and avoid over-segmentation by adding colour information remains to be explored in the future.

When attachments are connected, it is still difficult to decompose road furniture correctly. For example, it is difficult to separate traffic lights connected to each other. Some road furniture is scanned with extremely low point density. It is difficult to cluster points together to construct attachments. The over- and under-decomposition obstruct the interpretation processing. They lead to the main errors in the interpretation. Therefore, the over- and under-decomposition still need to be mitigated by more investigation on pole-like road furniture decomposition.

Recommendations

Because of the limitation of data and complex scene, our framework is not able to cope with research problems perfectly. Therefore, there are recommendations given as follows,

Pole-like road furniture detection: Before detecting pole-like road furniture, we simply conduct the connected component analysis which is merely operated based on the distance and the number of nearest neighbouring points. It is challenging for the connected component analysis to separate connected above ground objects. Therefore, more features can be introduced as criteria of growing to in connected component analysis, and this may cope with such a complex situation.

Pole-like road furniture decomposition: There are other types of features such as convexity and concave information not used in the decomposition of pole-like road furniture. The constraints of these features may improve the performance of pole-like road furniture decomposition.

Pole-like road furniture interpretation: Although contextual features between a pole and its attachments are encoded, the relations between multiple nearby poles and the relations between attachments are still not utilised in this research. Future work will be carried out on pole-like road furniture interpretation by adding further contextual information. Another recommendation is that deep learning networks such as PointNet can be applied to our research as long as we have enough labelled training data with high quality.

Bibliography

Adesiyun, A., Avenoso, A., Dionelis, K., Cela, L., Nicodème, C., Goger, T. and Polidori, C., 2016. Effective and coordinated road infrastructures safety operations. *Transportation research procedia*, 14, pp. 3304-3311.

Aijazi, A.K., Checchin, P. and Trassoudaine, L., 2013. Segmentation based classification of 3D urban point clouds: A super-voxel based approach with evaluation. *Remote Sensing*, *5*(4), pp. 1624-1650.

Aleotti, J. and Caselli, S., 2012. A 3D shape segmentation approach for robot grasping by parts. *Robotics and Autonomous Systems*, 60(3), pp. 358-366.

Attene, M., Falcidieno, B. and Spagnuolo, M., 2006. Hierarchical mesh segmentation based on fitting primitives. *The Visual Computer*, 22(3), pp. 181-193.

Attene, M., Robbiano, F., Spagnuolo, M. and Falcidieno, B., 2009. Characterization of 3D shape parts for semantic annotation. *Computer-Aided Design*, 41(10), pp. 756-763.

Bauer, R., Steiner, M., Kühnelt-Leddhin, A., Lyons, R., Turner, S., Walters, W., Larsen, B., Valkenberg, H., Bejko, D., Ellsaesser, G. and Rogmans, W., 2017. Scope and patterns of under-reporting of vulnerable road users in official road accident statistics Monica Steiner. *European Journal of Public Health*, 27(suppl_3).

Belton, D., Moncrieff, S. and Chapman, J., 2013. Processing tree point clouds using Gaussian mixture models. *Proceedings of the ISPRS annals of the photogrammetry, remote sensing and spatial information sciences,* pp. 11-13.

Bishop, C. M., 2006. *Pattern Recognition and Machine Learning.* New York, NY, USA: Springer.

Boulch, A., Le Saux, B. and Audebert, N., 2017, April. Unstructured Point Cloud Semantic Labeling Using Deep Segmentation Networks. In *3DOR*.

Breiman, L., 2001. Random forests. *Machine learning*, *45*(1), pp. 5-32.

Bremer, M., Wichmann, V. and Rutzinger, M., 2013. Eigenvalue and graph-based object extraction from mobile laser scanning point clouds. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 55–60.

Brenner, C., 2009. Extraction of features from mobile laser scanning data for future driver assistance systems. In *Advances in GIScience*. Springer, Berlin, Heidelberg, pp. 25-42

Brodu, N. and Lague, D., 2012. 3D terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: Applications in geomorphology. *ISPRS Journal of Photogrammetry and Remote Sensing*, *68*, pp. 121-134.

Cabo, C., Ordoñez, C., García-Cortés, S. and Martínez, J., 2014. An algorithm for automatic detection of pole-like street furniture objects from Mobile

Laser Scanner point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87, pp. 47-56.

Chang, C.C. and Lin, C.J., 2011. LIBSVM: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3), p. 27.

Chehata, N., Guo, L. and Mallet, C., 2009, November. Contribution of airborne full-waveform lidar and image data for urban scene classification. In *Image Processing (ICIP), 2009 16th IEEE International Conference on.* IEEE, pp. 1669-1672.

Cornea, N.D., Silver, D., Yuan, X. and Balasubramanian, R., 2005. Computing hierarchical curve-skeletons of 3D objects. *The Visual Computer*, 21(11), pp. 945-955.

Criminisi, A., Shotton, J. and Konukoglu, E., 2012. Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Foundations and Trends® in Computer Graphics and Vision*, 7(2–3), pp.81-227

Demantke, J., Mallet, C., David, N. and Vallet, B., 2011. Dimensionality based scale selection in 3D lidar point clouds. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 97-102.

Demantké, J., Vallet, B. and Paparoditis, N. Streamed vertical rectangle detection in terrestrial laser scans for facade database production. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*. 2012, *1*, pp.99–104.

El-Halawany, S.I. and Lichti, D.D., 2013. Detecting road poles from mobile terrestrial laser scanning data. *GIScience & Remote Sensing,* 50(6), pp. 704-722.

Fritsch, J., Kuehnl, T. and Geiger, A., 2013, October. A new performance measure and evaluation benchmark for road detection algorithms. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, pp. 1693-1700.

Fukano, K. and Masuda, H., 2015. Detection and Classification of Pole-Like Objects from Mobile Mapping Data. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences,* 1, pp. 57-64.

Geiger, A., Lenz, P. and Urtasun, R., 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on.* IEEE, pp. 3354-3361.

Golovinskiy, A. and Funkhouser, T., 2008. Randomized cuts for 3D mesh analysis. *ACM transactions on graphics (TOG).* 27(5), p. 145.

Golovinskiy, A. and Funkhouser, T., 2009. Min-cut based segmentation of point clouds. *IEEE Workshop on Search in 3D and Video (S3DV), IEEE International Conference on Computer Vision*, pp. 39–46.

Golovinskiy, A., Kim, V. G. and Funkhouser, T., 2009. Shape-based recognition of 3D point clouds in urban environments. *IEEE International Conference on Computer Vision*, pp. 2154–2161.

Guyon, I. and Elisseeff, A., 2003. An introduction to variable and feature selection. *Journal of machine learning research*, *3*, pp.1157-1182.

Hackel, T., Wegner, J.D. and Schindler, K., 2016. Fast semantic segmentation of 3D point clouds with strongly varying density. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3(3), pp. 177-184.

Hofmann, S. and Brenner, C., 2009. Quality assessment of automatically generated feature maps for future driver assistance systems. *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems*, pp. 500-503.

Hsu, C.W., Chang, C.C. and Lin, C.J., 2003. *A Practical Guide to Support Vector Classification*. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

Hsu, C.W. and Lin, C.J., 2002. A comparison of methods for multiclass support vector machines. *IEEE transactions on Neural Networks*, *13*(2), pp. 415-425.

Huang, Q.X., Wicke, M., Adams, B. and Guibas, L., 2009. Shape decomposition using modal analysis. In *Computer Graphics Forum.* Oxford, UK: Blackwell Publishing Ltd, 28(2) , pp. 407-416.

Huang, J. and You, S., 2015. Pole-like object detection and classification from urban point clouds. *IEEE International Conference on Robotics and Automation*, pp. 3032-3038.

Huang, J. and You, S., 2016. Point cloud labeling using 3d convolutional neural network. In *Pattern Recognition (ICPR), 2016 23rd International Conference on*. IEEE, pp. 2670-2675

Ibañez-Guzman, J., Laugier, C., Yoder, J.D. and Thrun, S., 2012. Autonomous driving: Context and state-of-the-art. *Handbook of Intelligent Vehicles*, pp.1271-1310.

IGN, 2013. http://data.ign.fr/benchmarks/UrbanAnalysis/.

Jaakkola, A., Hyyppä, J., Hyyppä, H. and Kukko, A., 2008. Retrieval algorithms for road surface modelling using laser-based mobile mapping. *Sensors*, 8, pp. 5238–5249.

Kaartinen, H., Hyyppä, J., Kukko, A., Jaakkola, A. and Hyyppä, H., 2012. Benchmarking the performance of mobile laser scanning systems using a permanent test field. *Sensors*, 12(9), pp. 12814-12835.

Kaick, O.V., Fish, N., Kleiman, Y., Asafi, S. and Cohen-Or, D., 2014. Shape segmentation by approximate convexity analysis. *ACM Transactions on Graphics (TOG)*, 34(1), p. 4.

Kalogerakis, E., Hertzmann, A. and Singh, K., 2010. Learning 3D mesh segmentation and labeling. *ACM Transactions on Graphics (TOG)*, 29(4), p.102.

Kalogerakis, E., Averkiou, M., Maji, S. and Chaudhuri, S., 2017. 3D shape segmentation with projective convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* pp. 3779-3788.

Katz, S., Leifman, G. and Tal, A., 2005. Mesh segmentation using feature point and core extraction. *The Visual Computer*, 21(8-10), pp. 649-658.

Katz, S. and Tal, A., 2003. Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM Transactions on Graphics (TOG)*, 22(3), pp. 954-961.

Klokov, R. and Lempitsky, V., 2017. Escape from cells: Deep kd-networks for the recognition of 3d point cloud models. *IEEE International Conference on Computer Vision*, pp. 863-872.

Koppula, H.S., Anand, A., Joachims, T. and Saxena, A., 2011. Semantic labeling of 3d point clouds for indoor scenes. In *Advances in neural information processing systems,* pp. 244-252.

Kukko, A., 2013. *Mobile laser scanning–system development, performance and applications*. Finnish Geodetic Institute.

Kukko, A., Andrei, C.O., Salminen, V.M., Kaartinen, H., Chen, Y., Rönnholm, P., Hyyppä, H., Hyyppä, J., Chen, R., Haggrén, H., Kosonen, I. and Čapek, K., 2007. Road environment mapping system of the Finnish Geodetic Institute - FGI ROAMER. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXVI, pp.241-247.

Kukko, A., Kaartinen, H., Hyyppä, J. and Chen, Y., 2012. Multiplatform Mobile Laser Scanning: Usability and Performance. *Sensors,* 12(9), pp.11712-11733.

Lai, K. and Fox, D., 2010. Object recognition in 3D point clouds using web data and domain adaptation. *The International Journal of Robotics Research*, 29(8), pp.1019-1037.

Landrieu, L. and Simonovsky, M., 2017. Large-scale point cloud semantic segmentation with superpoint graphs. arXiv preprint arXiv:1711.09869.

Lehtomäki, M., Jaakkola, A., Hyyppä, J., Kukko, A. and Kaartinen, H., 2010. Detection of Vertical Pole-Like Objects in a Road Environment Using Vehicle-Based Laser Scanning Data. *Remote Sensing*, 2(3), pp. 641-664.

Lehtomäki, M., Jaakkola, A., Hyyppä, J., Lampinen, J., Kaartinen, H., Kukko, A., Puttonen, E. and Hyyppä, H., 2016. Object classification and recognition from mobile laser scanning point clouds in a road environment. *IEEE Transactions on Geoscience and Remote Sensing*, 54(2), pp. 1226-1239.

Li, D. and Oude Elberink, S., 2013. Optimizing detection of road furniture (pole-like objects) in mobile laser scanner data. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 163-168.

Li, F., Oude Elberink, S. and Vosselman, G., 2016. Pole-Like Street Furniture Decomposition in Mobile Laser Scanning Data. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3(3), pp. 193-200.

Li, F., Oude Elberink, S. and Vosselman, G., 2017. Semantic labelling of road furniture in mobile laser scanning data. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42.

Li, F., Oude Elberink, S. and Vosselman, G., 2018a. Pole-Like Road Furniture Detection and Decomposition in Mobile Laser Scanning Data Based on Spatial Relations. *Remote sensing*, 10(4), p. 531.

Li, F., Lehtomäki, M., Oude Elberink, S., Vosselman, G., Puttonen, E., Kukko, A. and Hyyppä, J., 2018b. Pole-like road furniture detection in sparse and unevenly distributed mobile laser scanning data. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 4(2).

Li, Y., Bu, R., Sun, M. and Chen, B., 2018c. PointCNN. arXiv preprint arXiv:1801.07791.

Liberge, S., Soheilian, B., Chehata, N. and Paparoditis, N., 2010. Extraction of vertical posts in 3D laser point clouds acquired in dense urban areas by a mobile mapping system. *International Archives of Photogrammetry Remote Sensing and Spatial Information*, 38, pp.126-130.

Lien, J.M., Keyser, J. and Amato, N.M., 2006. Simultaneous shape decomposition and skeletonization. In *Proceedings of the 2006 ACM symposium on Solid and physical modelling*. ACM, pp. 219-228.

Lien, J.M. and Amato, N.M., 2007. Approximate convex decomposition of polyhedra. In *Proceedings of the 2007 ACM symposium on Solid and physical modelling*. ACM, pp. 121-131.

Lim, E.H. and Suter, D., 2008. Multi-scale conditional random fields for over-segmented irregular 3D point clouds classification. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on.* IEEE, pp. 1-7.

Liu, R. and Zhang, H., 2007. Mesh segmentation via spectral embedding and contour analysis. In *Computer Graphics Forum*. Oxford, UK: Blackwell Publishing Ltd, 26(3), pp. 385-394.

Louppe, G., Wehenkel, L., Sutera, A. and Geurts, P., 2013. Understanding variable importances in forests of randomized trees. In *Advances in neural information processing systems,* pp. 431-439.

Ma, L., Li, Y., Li, J., Wang, C., Wang, R. and Chapman, M., 2018. Mobile laser scanned point-clouds for road object detection and extraction: A review. *Remote Sensing*, 10(10), p. 1531.

Maturana, D. and Scherer, S., 2015, September. Voxnet: A 3d convolutional neural network for real-time object recognition. *IEEE/ RSJ International Conference on Intelligent Robots and Systems*, pp. 922-928.

Mortara, M., Patané, G., Spagnuolo, M., Falcidieno, B. and Rossignac, J., 2004. Plumber: a method for a multi-scale decomposition of 3D shapes into tubular primitives and bodies. In *Proceedings of the ninth ACM symposium on Solid modeling and applications*. Eurographics Association, pp. 339-344.

Müller, A.C. and Behnke, S., 2014, May. Learning depth-sensitive conditional random fields for semantic segmentation of rgb-d images. *IEEE International Conference on Robotics and Automation*, pp. 6232-6237.

Munoz, D., Bagnell, J. A., Vandapel, N. and Hebert, M., 2009. Contextual classification with functional max-margin Markov networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 975–982.

Munoz, D., Vandapel, N. and Hebert, M., 2008. Directional associative markov network for 3D point cloud classification. In *Proceedings of the Fourth International Symposium on 3D Data Processing, Visualization and Transmission*, Atlanta, GA, USA.

Murphy, K.P., 2012. *Machine learning: a probabilistic perspective*. MIT press.

Niemeyer, J., Rottensteiner, F. and Soergel, U., 2013. Classification of urban LiDAR data using conditional random field and random forests. In *Urban Remote Sensing Event (JURSE), 2013 Joint.* IEEE, pp. 139-142.

Niemeyer, J., Rottensteiner, F. and Soergel, U., 2014. Contextual classification of lidar data and building object detection in urban areas. *ISPRS Journal of Photogrammetry and Remote Sensing*, *87*, pp. 152–165.

Nikoohemat, S., Peter, M., Elberink, S.O. and Vosselman, G., 2017. Exploiting Indoor Mobile Laser Scanner Trajectories for Semantic Interpretation of Point Clouds. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *4*.

Nurunnabi, A., Belton, D. and West, G., 2012. Robust segmentation in laser scanning 3d point cloud data. *IEEE International Conference on Digital Image Computing Techniques and Applications*, pp. 1-8.

Nurunnabi, A., Belton, D. and West, G., 2016. Robust Segmentation for Large Volumes of Laser Scanning Three-Dimensional Point Cloud Data. *IEEE Transactions on Geoscience and Remote Sensing*, 54(8), pp.4790-4805.

Oude Elberink, S. and Kemboi, B., 2014. User-assisted object detection by segment based similarity measures in mobile laser scanner data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 239-246.

Oude Elberink, S., Khoshelhama, K., Arastouniab, M. and Benitoc, D.D., 2013. Rail track detection and modelling in mobile laser scanner data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2, pp. 223–228.

Oude Elberink, S. and Vosselman, G., 2009. 3D information extraction from laser point clouds covering complex road junctions. *Photogramm. Rec*. 24, 23–36.

Podolak, J., Shilane, P., Golovinskiy, A., Rusinkiewicz, S. and Funkhouser, T., 2006, July. A planar-reflective symmetry transform for 3D shapes. *ACM Transactions on Graphics (TOG)*, 25(3), pp. 549-559.

Pu, S., Rutzinger, M., Vosselman, G. and Oude Elberink, S., 2011. Recognizing basic structures from mobile laser scanning data for road inventory studies. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(6), pp. S28-S39.

Puente, I., González-Jorge, H., Martínez-Sánchez, J. and Arias, P., 2013. Review of mobile mapping and surveying technologies. *Measurement*, 46(7), pp. 2127-2145.

Puttonen, E., Jaakkola, A., Litkey, P. and Hyyppä, J., 2011. Tree classification with fused mobile laser scanning and hyperspectral data. *Sensors*, *11*, pp. 5158–5182.

Qi, C.R., Su, H., Mo, K. and Guibas, L.J., 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, *1*(2), p. 4.

Rabbani, T., Van Den Heuvel, F. and Vosselmann, G., 2006. Segmentation of point clouds using smoothness constraint. *International archives of photogrammetry, remote sensing and spatial information sciences*, 36(5), pp.248-253.

Riegl, 2017. Available online: http://www.riegl.com/uploads/tx_pxpriegldownloads/RIEGL_VMX-1HA_at-a-glance_2017-05-04.pdf (accessed on 28 January 2018).

Riegler, G., Osman Ulusoy, A. and Geiger, A., 2017. Octnet: Learning deep 3d representations at high resolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3577-3586.

Riveiro, B., Díaz-Vilariño, L., Conde-Carnero, B., Soilán, M. and Arias, P., 2016. Automatic segmentation and shape-based classification of retro-reflective traffic signs from mobile lidar data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(1), pp. 295-303.

Kohavi, R., 1995. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection, in *Proc. 14th International Joint Conference on Artificial Intelligence*, pp. 1137–1143.

Ross, S., Munoz, D., Hebert, M. and Bagnell, J.A., 2011. Learning message-passing inference machines for structured prediction. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2737-2744.

Rusu, R.B., Holzbach, A., Blodow, N. and Beetz, M., 2009, October. Fast geometric point labeling using conditional random fields. *IEEE/ RSJ International Conference on Intelligent Robots and Systems*, pp. 7-12.

Rutzinger, M., Elberink, S.O., Pu, S. and Vosselman, G., 2009. Automatic extraction of vertical walls from mobile and airborne laser scanning data.

International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 38(W8), pp.7-11.

Rutzinger, M., Höfle, B., Oude Elbernik, S. and Vosselman, G., 2011. Feasibility of facade extraction from mobile laser scanning data. *Photogramm. Fernerkund. Geoinf*, *3*, 97–107.

Schnabel, R., Wahl, R. and Klein, R., 2007. Efficient RANSAC for Point-Cloud Shape Detection. *Computer graphics forum*, 26 (2), pp. 214-226.

Sedlacek, D. and Zara, J., 2009, November. Graph cut based point-cloud segmentation for polygonal reconstruction. In *International Symposium on Visual Computing*. Springer, Berlin, Heidelberg, pp. 218-227

Serna, A. and Marcotegui, B., 2014. Detection, segmentation and classification of 3D urban objects using mathematical morphology and supervised learning. *ISPRS Journal of Photogrammetry and Remote Sensing,* 93, pp.243-255.

Strom, J., Richardson, A. and Olson, E., 2010. Graph-based segmentation for coloured 3D laser point clouds. *IEEE/ RSJ International Conference on Intelligent Robots and Systems*, pp. 2131-2136.

Soilán, M., Riveiro, B., Martínez-Sánchez, J. and Arias, P., 2016. Traffic sign detection in MLS acquired point clouds for geometric and image-based semantic inventory. *ISPRS Journal of Photogrammetry and Remote Sensing*, 114, pp. 92-101.

Soilán, M., Riveiro, B., Sánchez-Rodríguez, A. and Arias, P., 2018. Safety assessment on pedestrian crossing environments using MLS data. *Accid. Anal. Prev*, 111, 328–337.

Song, S. and Xiao, J., 2014. Sliding shapes for 3d object detection in depth images. *Proceedings of the European conference on computer vision*, pp. 634-651.

Song, S. and Xiao, J., 2016. Deep sliding shapes for Amodal 3D object detection in RGB-D images. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 808-816.

Su, H., Maji, S., Kalogerakis, E. and Learned-Miller, E., 2015. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE international conference on computer vision*, pp. 945-953.

Su, H., Jampani, V., Sun, D., Maji, S., Kalogerakis, E., Yang, M.H. and Kautz, J., 2018. Splatnet: Sparse lattice networks for point cloud processing. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2530-2539.

Tang, J., Miller, S., Singh, A. and Abbeel, P., 2012, May. A textured object recognition pipeline for color and depth image data. *IEEE International Conference on Robotics and Automation*, pp. 3467-3474.

Tombari, F., Fioraio, N., Cavallari, T., Salti, S., Petrelli, A. and Di Stefano, L., 2014. Automatic detection of pole-like structures in 3d urban environments. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4922-4929.

Triebel, R., Kersting, K. and Burgard, W., 2006. Robust 3D scan point classification using associative Markov networks. *IEEE International Conference on Robotics and Automation*, pp. 2603-2608.

U.S. Department of Transportation Federal Highway Administration. Available online: http://safety.fhwa.dot.gov/tools/data_tools/mirereport/ (accessed on 28 January 2018).

Vaaja, M., Kukko, A., Kaartinen, H., Kurkela, M., Kasvi, E., Flener, C., Hyyppä, H., Hyyppä, J., Järvelä, J. and Alho, P., 2013. Data processing and quality evaluation of a boat-based mobile laser scanning system. *Sensors*, *13*(9), pp. 12497-12515.

Vallet, B., Brédif, M., Serna, A., Marcotegui, B. and Paparoditis, N., 2015. TerraMobilita/iQmulus urban point cloud analysis benchmark. *Computers & Graphics*, 49, pp. 126-133.

Vandapel, N., Huber, D.F., Kapuria, A. and Hebert, M., 2004. Natural terrain classification using 3-D Ladar data. *IEEE International Conference on Robotics and Automation*, 5, pp. 5117–5122.

Velizhev, A., Shapovalov, R. and Schindler, K., 2012. Implicit shape models for object detection in 3D point clouds. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. I-3, pp. 179-184.

Vosselman, G., 2013. Point cloud segmentation for urban scene classification. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Antalya, Turkey, Vol. XL-7/W2, pp. 257-262.

Vosselman, G., Gorte, B.G., Sithole, G. and Rabbani, T., 2004. Recognising structure in laser scanner point clouds. *International archives of photogrammetry, remote sensing and spatial information sciences*, 46(8), pp.33-38.

Vosselman, G., Maas, H.G., 2010. Airborne and Terrestrial Laser Scanning. CRC Press.

Wang, J., Lindenbergh, R. and Menenti, M., 2017a. SigVox–A 3D feature matching algorithm for automatic street object recognition in mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, *128*, pp.111-129.

Wang, P.S., Liu, Y., Guo, Y.X., Sun, C.Y. and Tong, X., 2017b. O-cnn: Octree-based convolutional neural networks for 3d shape analysis. *ACM Transactions on Graphics (TOG)*, *36*(4), p.72

Wani, M.A. and Arabnia, H.R., 2003. Parallel edge-region-based segmentation algorithm targeted at reconfigurable multiring network. *The Journal of Supercomputing*, 25(1), pp. 43-62.

Weinmann, M., Jutzi, B. and Mallet, C., 2014. Semantic 3D scene interpretation: a framework combining optimal neighborhood size selection with relevant features. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3, pp.181-188.

Weinmann, M., Jutzi, B., Hinz, S. and Mallet, C., 2015. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing*, 105, pp. 286-304.

Wolf, D., Prankl, J. and Vincze, M., 2015. Fast semantic segmentation of 3D point clouds using a dense CRF with learned parameters. *IEEE International Conference on Robotics and Automation*, pp. 4867-4873.

World Health Organization. Global status report on road safety 2015. 2015 World Health Organization.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X. and Xiao, J., 2015. 3D ShapeNets: A deep representation for volumetric shapes. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1912-1920.

Xiong, X., Munoz, D., Bagnell, J. A. and Hebert, M., 2011. 3-D scene analysis via sequenced predictions over points and regions. *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2609–2616.

Yang, B. and Dong, Z., 2013. A shape-based segmentation method for mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 81, pp. 19-30.

Yang, B., Dong, Z., Zhao, G. and Dai, W., 2015. Hierarchical extraction of urban objects from mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 99, pp. 45-57.

Yang, B. and Fang, L., 2014. Automated extraction of 3-D railway tracks from mobile laser scanning point clouds. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(12), pp.4750-4761.

Yang, B., Liu, Y., Liang, F. and Dong, Z., 2016. Using mobile laser scanning data for features extraction of high accuracy driving maps. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 433-439.

Yang, X. and Tian, Y., 2014. Effective 3d action recognition using eigenjoints. *Journal of Visual Communication and Image Representation*, *25*(1), pp.2-11.

Yi, L., Kim, V.G., Ceylan, D., Shen, I., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A. and Guibas, L., 2016. A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics (TOG)*, 35(6), p.210.

Yi, L., Su, H., Guo, X. and Guibas, L.J., 2017. Syncspeccnn: Synchronized spectral cnn for 3d shape segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2282-2290.

Yokoyama, H., Date, H., Kanai, S. and Takeda, H., 2013. Detection and classification of pole-like objects from mobile laser scanning data of urban environments. *International Journal of CAD/CAM*, 13(2), pp. 31-40.

Yu, X., Hyyppä, J., Vastaranta, M., Holopainen, M. and Viitala, R., 2011. Predicting individual tree attributes from airborne laser point clouds based on the random forests technique. *ISPRS Journal of Photogrammetry and remote sensing*, *66*(1), pp.28-37.

Yu, Y., Li, J., Guan, H. and Wang, C., 2015a. Automated extraction of urban road facilities using mobile laser scanning data. *IEEE Transactions on Intelligent Transportation Systems*, 16 (4), pp. 2167-2181.

Yu, Y., Li, J., Guan, H., Wang, C. and Yu, J., 2015b. Semiautomated extraction of street light poles from mobile LiDAR point-clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 53(3), pp. 1374-1386.

Yu, Y., Li, J., Wen, C., Guan, H., Luo, H. and Wang, C., 2016. Bag-of-visual-phrases and hierarchical deep models for traffic sign detection and recognition in mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 113, pp. 106-123.

Zhang, W., 2010. Lidar-based road and road-edge detection. In *2010 IEEE Intelligent Vehicles Symposium*. IEEE, pp. 845-848.

Zhou, L. and Vosselman, G., 2012. Mapping curbstones in airborne and mobile laser scanning data. *International Journal of Applied Earth Observation and Geoinformation*, 18, pp. 293-304.

Summary

Road furniture recognition has become popular in the past few years because of its great importance for road safety analysis, smart cities, and autonomous driving. To enhance road safety, policies that encourage the road furniture inventory are prevalent in many countries. Such an inventory can be remarkably facilitated by the automatic recognition of road furniture.

Research has been carried out on the recognition of pole-like road furniture, such as traffic signs and lamp posts. So far, the proposed methods have mainly classified road furniture as individual objects. However, most road furniture consists of a combination of classes, such as a traffic sign and a street light mounted on the same pole. Due to the recent developments in mobile laser scanners, more accurate data is available that allows for the segmentation of road furniture at a detailed level. Precise interpretation of road furniture in mobile laser scanning data still remains unexplored. To tackle this problem, we propose a three-stage framework to interpret road furniture at a more detailed level. Instead of being interpreted as single objects, mobile laser scanning data of road furniture is decomposed in elements individually labelled as pole, street light, traffic sign, etc.

In the first stage, we develop a method to detect pole-like road furniture from mobile laser scanning data. The framework is carried out in four steps. The unorganised point cloud is first partitioned. Then above ground points are clustered and roughly classified after removing ground points. A slicing check in combination with cylinder masking is proposed to extract pole-like road furniture candidates. Pole-like road furniture is obtained after occlusion analysis in the last stage. The average completeness and correctness of pole-like road furniture in sparse and unevenly distributed mobile laser scanning data was above 83%. In dense mobile laser scanning data the average completeness and correctness of pole-like road furniture are above 90%. This is comparable to the state of art in the field of pole-like road furniture detection in mobile laser scanning data of good quality and is potentially of practical use in the processing of point clouds collected by autonomous driving platforms.

In the second stage, we propose an approach to decompose detected pole-like road furniture into different components based on their spatial relations in a three-step procedure: (i) pre-identification, (ii) pole extraction, (iii) decomposition into poles and attachments. For the pole extraction, a novel global pole extraction approach is proposed to handle three different cases of street furniture. The result of the decomposition is taken as a feedback to remove spurious pole-like road furniture as a following step. If there are no poles or attached objects extracted in the decomposition stage, these incorrectly detected pole-like road furniture—such as the pillars of buildings—

will be removed from the detection list. We further introduce a method to evaluate the results of the decomposition.

Detected road furniture is decomposed into poles and attachments (e.g. traffic signs). In the third stage, we extract a set of features to classify the attachments by utilising a knowledge-driven method and four representative types of machine learning classifiers (random forests, support vector machines, Gaussian mixture model, and naïve Bayes) to explore the optimal method. These designed features are the unary features of attachments and the spatial relations between poles and their attachments. Two experimental test sites in Enschede and Saunalahti were used. The data in Saunalahti was collected in two different epochs. The random forest classifier outperforms the other methods and the overall accuracy acquired is higher than 80% for the Enschede test site and higher than 90% in both Saunalahti epochs. The designed contextual features, which depict the relation between attachments and poles, play an important role in the interpretation of road furniture. The results of two epochs in the same area prove the high robustness of our framework and demonstrate that the transferability of our method achieves good transferability with an accuracy over 90% through employing the training data of one epoch to test the data in another epoch.

To conclude, we demonstrate that our framework for semantic segmentation of pole-like road furniture with respect to their functionalities enables a more detailed and significantly more accurate labelling of pole-like road furniture. The research results show great potential for precise mapping of urban road objects.

Samenvatting

Het herkennen van verkeersmeubilair, zoals verkeersborden en –lichten en lantaarnpalen, is de afgelopen jaren in populariteit gestegen vanwege het grote belang voor verkeersveiligheidsanalyses, voor toepassing in Smart Cities en om autonoom te kunnen rijden. Om de verkeersveiligheid te vergroten, wordt er in veel landen gekeken naar hoe er een inventarisatie kan worden gemaakt van al het verkeersmeubilair. Een dergelijke inventarisatie kan worden vergemakkelijkt door de automatische herkenning van straatmeubilair.

In dit promotie traject is onderzoek gedaan naar de herkenning van paal vormige objecten zoals verkeersborden, verkeerlichten en lantaarnpalen. Tot nu toe zijn de bestaande methoden in staat om deze objecten als één geheel te herkennen. Echter, vaak bestaat zo'n object uit een combinatie van klassen, zoals een verkeersbord en een straatlantaarn die op dezelfde paal zijn gemonteerd. Vanwege de recente ontwikkelingen in mobiele laser scanners, zijn er betere meetgegevens beschikbaar die de herkenning van verkeersmeubilair op een gedetailleerd niveau mogelijk maken.

In dit onderzoek stellen we een methode voor om verkeersmeubilair op een meer gedetailleerd niveau te interpreteren, opgedeeld in drie fases.

In de eerste fase ontwikkelen we een methode om paal vormig verkeersmeubilair te detecteren uit mobiele laserscanningsgegevens. Dit gebeurt in een aantal deelstappen. De laserscanning gegevens, ook wel punten in een puntenwolk genoemd, worden eerst opgesplitst in kleine gebieden. Vervolgens worden de punten die tot de grond behoren verwijderd, en de punten die erboven liggen worden geclusterd. Door middel van het herkennen van cilinders op verschillende hoogten worden de paalvormige objecten herkend. Dit kan met een nauwkeurigheid van 83% in een puntenwolk met een middelmatige kwaliteit en meer dan 90% in een nauwkeurige dataset. In deze fase is nog niet bekend wat voor een soort paalvormig object het is, en of er meerdere onderdelen aan vast hangen.

In de tweede fase gaan we gedetailleerder vaststellen wat de verschillende componenten zijn. Dit gebeurt op basis van ruimtelijke relaties tussen de paal en de onderdelen die eraan vast zitten. Eerst wordt er gecheckt of er überhaupt wel onderdelen aan vast zitten of dat de paal eigenlijk een ander object is, zoals een pilaar van een gebouw. Daarna worden de punten behorend tot de paal en punten behorend tot de aanhangende onderdelen apart geclusterd.

In de derde fase worden de kenmerken van aanhangende onderdelen, zoals grootte en hoogte, en contextuele kenmerken tussen deze onderdelen en de palen (hoogte en orientatie van het object ten opzichte van de paal) berekend.

Vijf verschillende classificatietechnieken zijn onderzocht die automatisch de gedetailleerde onderdelen kunnen herkennen: een beslissingsboom op basis van onze kennis over verkeersmeubilair, Random Forest, Support Vector Machines, Gauss-Mixture models en Naive Bayes.

De technieken zijn toegepast op twee test datasets, één uit Enschede en de ander uit Saunalahti, Finland. De gegevens in Saunalahti zijn opgenomen op twee verschillende tijdstippen. De Random Forest classificatie presteert beter dan de andere methoden en de nauwkeurigheid is hoger dan 80% voor de Enschedese testsite en hoger dan 90% in beide datasets van Saunalahti. De contextuele kenmerken, die de relatie tussen aanhangende objecten en palen weergeven, spelen een belangrijke rol bij de interpretatie van verkeersmeubilair. De resultaten van de twee datasets in hetzelfde gebied maar opgenomen op verschillende data, bewijzen de hoge robuustheid van onze methode. We tonen aan dat onze methode goed toe te passen is in een ander gebied of ander tijdvak door de methode te trainen met data uit 1 dataset en toe te passen in een andere dataset, met een nauwkeurigheid van meer dan 90%.

Concluderend stellen we dat onze methode een gedetailleerde herkenning van verkeersmeubilair mogelijk maakt. De onderzoeksresultaten laten een groot potentieel zien voor het nauwkeurig in kaart brengen van stedelijke verkeersobjecten.

Biography

Fashuai Li was born in Jinan, Shandong, China in 1988. He received a BSc. Degree in Geodesy at the China University of Mining and Technology (Beijing) in 2011. He then completed an MSc. Degree in Geographical Information System at the China University of Mining and Technology (Beijing) in 2014. In the September of 2014, he started as a Ph.D. candidate at the Earth Observation Science department at the Faculty of Geo-Information Science and Earth Observation Science (ITC), University of Twente, the Netherlands. In 2017, he visited Finnish Geospatial Research Institute as a research scientist for half a year. He focused on pole-like road furniture interpretation in mobile laser scanning data. He also shows great interest in computer vision, machine learning and robotics. He has published various papers in the leading remote sensing journals and ISPRS events.

List of Publications:

Li, F., Oude Elberink, S.O.E. and Vosselman, M.G., 2016. Automatic interpretation of pole-like street furniture. In *NCG Symposium 2016: Nederlands Centrum voor Geodesie en Geo-Informatica*. (Abstract & presentation)

Li, F., Oude Elberink, S. and Vosselman, G., 2016. Pole-like street furniture decomposition in mobile laser scanning data. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 3(3).

Li, F., Oude Elberink, S. and Vosselman, G., 2017. Semantic labelling of road furniture in mobile laser scanning data. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42.

Li, F., Oude Elberink, S. and Vosselman, G., 2017. Semantic labelling of pole-like street furniture in mobile laser scanning data. In *Annual Symposium of the Netherlands Centre for Geodesy and Geo-Informatics (NCG)* 2017. (Abstract & presentation)

Li, F., Lehtomäki, M., Oude Elberink, S., Vosselman, G., Puttonen, E., Kukko, A. and Hyyppä, J., 2018. Pole-like road furniture detection in sparse and unevenly distributed mobile laser scanning data. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 4(2).

Li, F., Oude Elberink, S. and Vosselman, G., 2018. Pole-Like Road Furniture Detection and Decomposition in Mobile Laser Scanning Data Based on Spatial Relations. *Remote sensing*, 10(4), p.531. (**Featured article**)

Li, F., Oude Elberink, S. and Vosselman, G., 2018. The detailed interpretation of pole-like road furniture in mobile laser scanning data. In *Annual Symposium of the Netherlands Centre for Geodesy and Geo-Informatics (NCG) 2018*. (Abstract & presentation)

Li, F., Lehtomäki, M., Oude Elberink, S., Vosselman, G., Kukko, A., Puttonen, E., Chen, Y. and Hyyppä, J., 2019. Semantic segmentation of road furniture in mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*

Zhou, H., Chen, Y., Feng, Z., **Li, F.**, Hyyppä, J., Hakala, T., Karjalainen, M., Jiang, C. and Pei, L., 2018. The Comparison of Canopy Height Profiles Extracted from Ku-band Profile Radar Waveforms and LiDAR Data. *Remote Sensing*, 10(5), p.701.

Zhou, H., Chen, Y., Hyyppä, J., Feng, Z., **Li, F.**, Hakala, T., Xu, X. and Zhu, X., 2018. Estimation of Canopy Height Using an Airborne Ku-Band Frequency-Modulated Continuous Waveform Profiling Radar. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, (99), pp.1-8.