CLASSIFICATION AND CHANGE DETECTION IN MULTI-EPOCH AIRBORNE LASER SCANNING POINT CLOUDS

Sudan Xu

Examining committee:

Prof.dr.A.K. Skidmore	University of Twente, ITC
Dr.ir. T. Hartmann	University of Twente, CTW
Prof.dr. J. Stoter	Delft University of Technology
Prof.dr.-ing. F. Rottensteiner	Leibniz Universität Hannover

ITC dissertation number 266
ITC, P.O. Box 217, 7500 AE Enschede, The Netherlands

ISBN 978-90-365-3835-0
DOI 10.3990/1.9789036538350

Cover designed by Benno Masselink
Printed by ITC Printing Department

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

CLASSIFICATION AND CHANGE DETECTION IN MULTI-EPOCH AIRBORNE LASER SCANNING POINT CLOUDS

DISSERTATION

to obtain
the degree of doctor at the University of Twente,
on the authority of the rector magnificus,
prof.dr. H. Brinksma,
on account of the decision of the graduation committee,
to be publicly defended
on Wednesday 1st July 2015 at 16.45 hrs

by

Sudan Xu

born on 1st December 1982

in Hubei, China

This thesis is approved by
Prof.dr.ir. M.G. Vosselman, promoter
Dr. S. Oude Elberink, co-promoter

Acknowledgements

This book is about the classification and change detection of the airborne laser scanning (ALS) point clouds. That was my first time dealing with the ALS data since 2010. The topic originates from the interests of the Municipality of Rotterdam who has the requirements to check the permits of buildings in Rotterdam. They offer us the ALS data and frequently communicate with our laboratory. Therefore, I would like to thank them for the data they offer.

I still remembered the first time I had the research meeting with my Promoter Professor George Vosselman, who made several lines of codes and ran the codes on some sample ALS data. It was like a magic for me to understand the problem of my research topic, to get to know my coding library and to know how to process the ALS data within just several minutes! And, his way of supervision is suitable to me and I would like to extremely thank you for your strict supervision and your help during these four year.

My supervisor Sander Oude Elberink was the first person who sent me some sample codes, and Xiao Jing and Zhou Liang are the first people who showed me how to install my computer. Their help made me quickly got familiar to my working environment and was able to get some preliminary results very quickly. I would like to thank you all, and especially thank Sander for your creative suggestions. I admire a lot in your writing skills.

During the stay in the Netherlands, I was sometimes stacked in my way of research as well as my life . Thanks to the department, where there are so many sweet people: I can discuss with Markus, Kourosh, Adam, Biao, Liang, Salma and Meisam when I had no idea what to do in my research, I can discuss with Teresa about Yoga and life feelings whenever I felt happy or blue, and I can spend my spare time with Tiejun’s family, Xiaojing, Donghai, Yijian’s family, Meisam’s family, Liang’s family and all the cute Chinese friends. Thank you for your accompanies, and especially thank my husband, who is always by my side and encourages me.

Table of Contents

Chapter 1 Introduction

Change detection is defined as the process of identifying differences in the state of an object or phenomenon by observing it at different times (Singh, 1989). In 19^{th} century change detection was closely associated with military technology and was commonly used for civilian applications in 20^{th} century as regular acquisition of digital data of earth surface was possible and suitable computer programs became available. Change detection has become quite an important topic due to increasing concerns about the overall consequences of global and local change. Over the past fifty years, remarkable progress has been made in developing the techniques for change detection. Techniques, as documented by Lu et al. (2004), can be divided into five categories: visual analysis, algebraic approaches, transformation, classification, and combination of images with geographic information systems (GISs).

Visual analysis entails the observation of changes by collating epochs of data manually. It is an original method and is most widely used in the manufacturing of physical products. Visual analysis is, however, highly dependent on the analyst's experience and is often quite time-consuming. With the introduction of algebraic processing, such as image differencing and rationing, in the 1970s, change detection underwent a significant evolution. Algebraic processing can be used to remove most of any unchanged information, thereby greatly improving the efficiency of detecting change as compared to visual analysis. Algebraic techniques are, however, sensitive to noise. Roughly a decade later, transformation methods were introduced by Byrne (1980) to monitor changes in forest areas. One of the most popular transformation methods is principle component analysis (PCA), which is often applied to detect changes in land use. Early in the 21^{st} century, with the development of computing techniques and the gradual maturation of theoretical approaches, classification methods were developed and applied to change detection. Simultaneously, these methods were combined with geographic information systems (GISs) to create a vital tool for change detection and change analysis.

In the past twenty years, light detection and ranging (lidar), a very successful remote sensing invention, has created enormous new opportunities for exploring the world in 3D. Compared to traditional remote sensing techniques, such as aerial and satellite photography, lidar

has the advantages of being able to accurately measure elevation and to penetrate forested areas. The era of change detection using lidar data began in 1999, when Murakami et al. (1999) first compared two epochs of lidar data sets to detect the changes in buildings after an earthquake.
Change detection using lidar has a wide range of applications, including the fields of archaeology, geography, geology, geomorphology, seismology, forestry and civil engineering. From the perspective of the civil engineering applications, the results from the change detection can be applied to a variety of situations according to the data acquisition approaches used. Changes extracted from airborne laser scanning (ALS) data are commonly used to update maps (e.g. Vosselman et al. (2004a) and Matikainen et al. (2004)), as well as to evaluate damage to buildings after the occurrence of physical disasters (Murakami et al., 1999). Terrestrial laser scanning data offers more local change information, which is often used to monitor changes to a variety of structural deformations (Roberts and Hirst, 2005), in particular mountain slides (Sui et al., 2008). Mobile laser scanning systems may be installed to distinguish permanent objects from temporary objects (Xiao et al., 2013), or to track the motion of vehicles or pedestrians.

1.1 Research motivation

My PhD research is centred on the use of ALS data in building change detection. Detection of changes to buildings can have many applications. For example, when a change is made to a building it may have been done without permission, which would mean the change was made illegally. To be able to quickly locate such illegal activity it is essential to first know where any changes have been made to a building. Nevertheless, in practice it will always be rather challenging to manually detect changes occurring among potentially millions of objects within a metropolitan environment, whether from images or from ALS data. Compared with aerial or satellite images, ALS data has the advantage that buildings under trees remain visible. Some municipalities are making their ALS data available so that they can be used to explore the possibility of automating the detection of changes to buildings. Among them is the Municipality of Rotterdam, which, by using change detection derived from ALS data, is interested to verify changes to buildings against building permits issued, and to check the quality of the base registration of addresses and buildings (BAG). To explore the feasibility of various applications of lidar, the Municipality of Rotterdam contracted the company Fugro Aerial Mapping to capture multi-temporal ALS data sets of the Rotterdam area (the Netherlands) for the years 2008, 2010 and 2012.

As part of my PhD research I have been exploring the use of these ALS data sets for detecting both large and small changes to buildings. Large changes would usually be a newly built or demolished building or extensions to a building. Small changes, which are also of interest to the municipality, are changes attached to building roofs, for example dormers built on top of roof gables in residential areas. Some of the new constructions are likely to have been built or extended without permits. Unless these illegal constructions are registered and taxed, the municipality's building management system will fall into disorder.

For buildings, change detection by visual analysis is often an enormous, time-consuming activity. According to the statistics, visual analysis can account for up to 40% of the time and labour costs related to inspection of changes to objects in a dense city area, with the remaining 60% being taken up by charting and updating (Champion, 2007). Yet, in general, 95% of buildings in large cities remain unchanged for periods of 5-10 years (Frontoni et al., 2008). To avoid these disadvantages, my research therefore focuses on using the ALS data provided by the Municipality of Rotterdam to automatically detect and interpret changes to buildings that are of interest to its planning office.

Automated change detection in an urban environment that comprises thousands of real-world objects, large numbers of which are undergoing a wide variety of changes every second of the day, is a challenging task. If relevant changes are to be distinguished from among the total number of changes occurring, the detection approach taken should be able to not only separate real changes from spurious ones but also to identify changes considered to be relevant from those considered irrelevant, in addition to being able to distinguish buildings from other objects in the first place! (Spurious and irrelevant changes are described in more detail in Section 1.4) Furthermore, the approach chosen should be capable of detecting these changes automatically. Once detected, change to buildings needs to be found out and interpreted in order to know what the changes are. For this reason, another focus of my research has been the automated classification of buildings and the detected changes.

1.2 Research scope and limitations

This research mainly focuses on detecting changes to buildings, including newly built and demolished walls, roofs and roof elements. Roof elements are defined as things that are attached to the roofs of buildings, such as dormers, chimneys and antennas. Changes detected in roof elements are classified as one of four types: construction, vehicle, dormer, or undefined object. Changes to building walls fall outside the scope of my research. Descriptions of other aspects of the scope and limitations of my research follow in the remainder of this section.

Data processing

ALS points were directly used in the pre-processing and subsequent processing steps. To date, most methods for detecting and classifying changes in buildings have made use of either multi-temporal aerial images (Rottensteiner, 2007; Champion et al., 2009) or DSMs derived from ALS data (Murakami et al., 1999; Rutzinger et al., 2010). Often, changes to buildings are verified in classified DSMs or classified images. Several other methods, for example that of Hebel et al. (2013), run on the occupancy grid – directly derived the ALS data. Compared against aerial images and DSMs, ALS data offer better visibility of buildings in tree-covered areas and more accurate information on heights. Utilization of solely ALS data can avoid information loss.

Registration

The ALS data used have systematic strip differences of no more than 5 cm, whether within the same epoch or between different epochs. The standard deviation of noise in the height measurements is also less than 5 cm. It was decided to track changes larger than 10 cm, because it is very unlikely that a change of less than 10 cm will be made to a building. Therefore, no improvement of data registration was needed before change detection was performed.

Data filtering

It was assumed that the input data sets for classification had already been accurately filtered to separate ground and non-ground points. Data obtained in year 2010 had been already manually filtered into ground and non-ground points by the data supplier. The DTM from the 2010 data set was taken as the reference DTM in my research. Other epochs of data sets, which had not been filtered, were first filtered using “LASground” software from the “LAStools” suite. Thereafter the filtered data sets were refined automatically based on the reference DTM of year 2010.

Classification

Besides to change detection, classification is another but equally important focus in this thesis. There are two types of classification relevant to my research. Scene classification is performed on the whole data set to distinguish points as either water, ground, vegetation, wall, roof, roof element or an undefined object. Although during scene classification the main emphasis is on extracting building points, other objects of interest to the municipality, such as vegetation (e.g. high vegetation) and water were also extracted. In this classification step, buildings are divided into wall, roof and roof elements, thus providing the rough context for any building.

The other, second type of classification dealt with in this thesis may be called the “classification of change in buildings”. Following the change detection process for buildings, any change detected in roof elements was classified as being either a dormer, a vehicle on a roof top, a construction on top of a roof or an undefined object, and changes in large construction sites are classified as being a change in a roof, a roof element or a wall. This “classification of change” step helps to interpret changes detected in roof elements. When scene classification and classification of change are mentioned in the rest of my thesis, they refer to the concepts explained above.

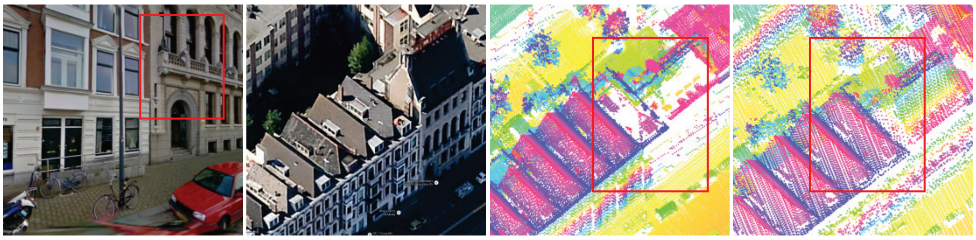
Unless specifically stated otherwise, all processes described in my research are automated, i.e. no human activity is involved in the segmentation, classification, and change detection or evaluation steps. I assume that these activities do not include data capture with sensors, filtering, making reference data for evaluation and adjusting parameters of algorithms, which will involve manual input.

1.3 Research problems and questions

The research for my thesis has focused on classification and change detection applied in an urban environment. The problems, that have arisen and their related research questions, which form the basis of my investigations, are described briefly in the following subsections.

1.3.1 Classification

Scene classification is fundamental to the process of change detection, as you have to know where buildings exist. The same holds for the "classification of change" process, since the aim is to identify which building changes are related to dormers. Classification generally includes the steps of entity definition, feature selection, classifier selection and evaluation. Entity definition entails specifying the grouping rules of points. Before generating features for classification, one should consider which entity should be adopted for calculating its features. Most literature makes efforts to discuss the importance of feature and classifier selection, in doing so raising such questions as: "Which features can be selected from a long list of potential features?"; "What method can be used for feature selection?"; and "Which classifier performs well in certain situations?". Before addressing these questions, the characteristics of the target classes need to be analysed.



(a) 2009 (b) 2013 (c) 2008 (d) 2010

Figure 1.1. (a) and 1(b) are two images of the same building obtained in 2009 and 2013 from Google Earth. The image in (b) is a street view of the wall in the rectangle in image (a). These two images indicate that there is no change on the roof and wall from year 2009 to 2013. (c) and 1(d) are the corresponding laser points scanned in years 2008 and 2010. In (c) there is a lack of points for the roof and the wall in the rectangular area, whereas in (d) the points are adequate for identifying the roof and the wall. Problems caused by lack of data in one epoch of data set. The lack of points in (c) yields two problems. First, in (c) it is difficult to classify the roof and the wall with common features. Second, a change will be found as soon as the laser points in (c) and (d) are compared. However, as shown in (a) and (b), these changes are not real.

Building

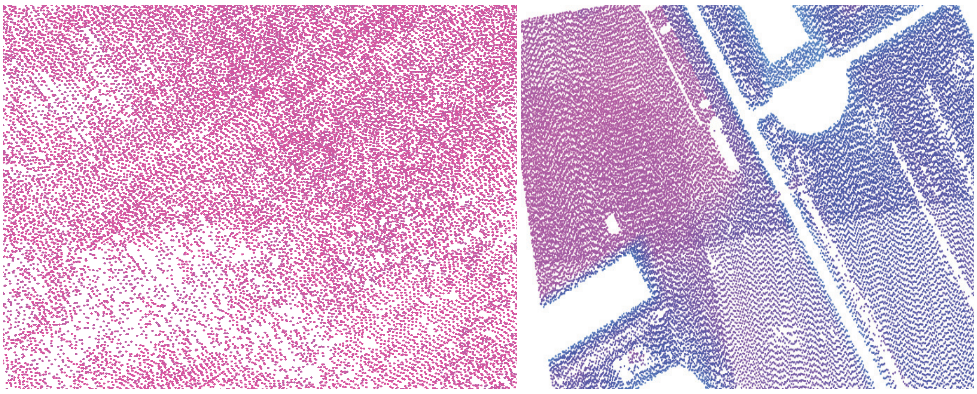
As the municipality is specifically interested in dormers on rooftops, all buildings need to be classified into walls, roofs and roof elements in order to provide better context information about dormers. There are potentially thousands of types of building walls, roofs and dormers in the urban scene. For this reason there is little chance of drawing up a uniform list of defining features for the simultaneous identification of all types of these elements, even if the list of features is long and all these features are uncorrelated with each other. In addition, many roofs lack corresponding points in the data (e.g. Figure 1.1), probably caused by absorption of the laser signal or an inappropriate laser scanner angle. This only makes definition of features more difficult.

Vegetation

Unlike man-made buildings, vegetation is a seasonal object involving a variety of species. Multiple-pulse information in ALS data is usually considered to be an important indicator of vegetation: in areas of vegetation, multiple reflectance of ALS laser beams is common. Other features, for instance "surface roughness" (or "sphericity" (Hebel et al., 2013), "plane fitting residual" (Niemeyer et al., 2012)) and "intensity", have also been found helpful in recognising the presence of vegetation. However, the expression of these features will differ in data from different seasons and for various tree species. In seasons when trees have their leaves, especially in forested areas, reflectance from vegetation is similar to that of flat roofs: the leaf canopy may be too dense to allow the emitted laser beams to generate multiple-pulse returns. The use of these features may not allow classification of all types of vegetation, even if an exhaustive list is incorporated in the classifier.

Water and ground

Sometimes, if laser beams strike water bodies they will not be reflected back to the sensor. On average this will result in a relatively low point density for a water surface as compared to a ground surface. Nevertheless, this is not always the case. The point density for a water surface may occasionally increase, due to a high specular reflectance from directly below the aircraft such that the point density of that water surface is approximately as high as that for the ground surface (Figure 1.2a). Another factor that may influence point density is strip overlap, which occurs when the same area is scanned twice during the flight (e.g. Figure 1.2b).



(a) (b)

Figure 1.2. Examples of varying point density. Different colours indicate different height values. (a) High point density attributed to specular reflectance (point density varies in the image). (b) Varying point density caused by strip overlap on the ground surface.

The problems associated with the defined classes mentioned above show that a long list of features is not always a good solution for classification. In addition, it is often quite expensive to calculate a large number of point features in large ALS data sets. Even when feature selection allows us to reduce the points needed, it will still be a time-consuming step. For example, it takes more than 50 minutes to generate nine features from point clouds of an area of 50 m x 50 m. Therefore, the questions for the classification are: As generation of large number of features is infeasible for the classification of all the objects represented in large data sets, is there a strategy that makes it possible to obtain promising classification accuracy by applying only a few commonly used features? If the answer is “yes”, then what results can be expected from using different entities with a few features and how can a compromise between accuracy and computation speed be achieved with the utilization of different entities?

Entity

The term “entity” has various definitions depending on the specialization under discussion. In classification, the term is introduced to indicate a certain division of ALS points. In the current literature, commonly used entities with ALS data are “point”, “segment”, and “voxel”. With differently defined entities, we can generate multiple divisions of points that are adaptive for the shapes of different objects. Different entities can represent different objects (Figure 1.3). The same feature calculated for different entities may yield different values, which allows features to behave more distinctly among different objects. If multiple entities can handle the classification problem, even when only a few features have been derived and when the data sets are extremely large, the main problem then becomes what entities and their corresponding features can be used for classification?

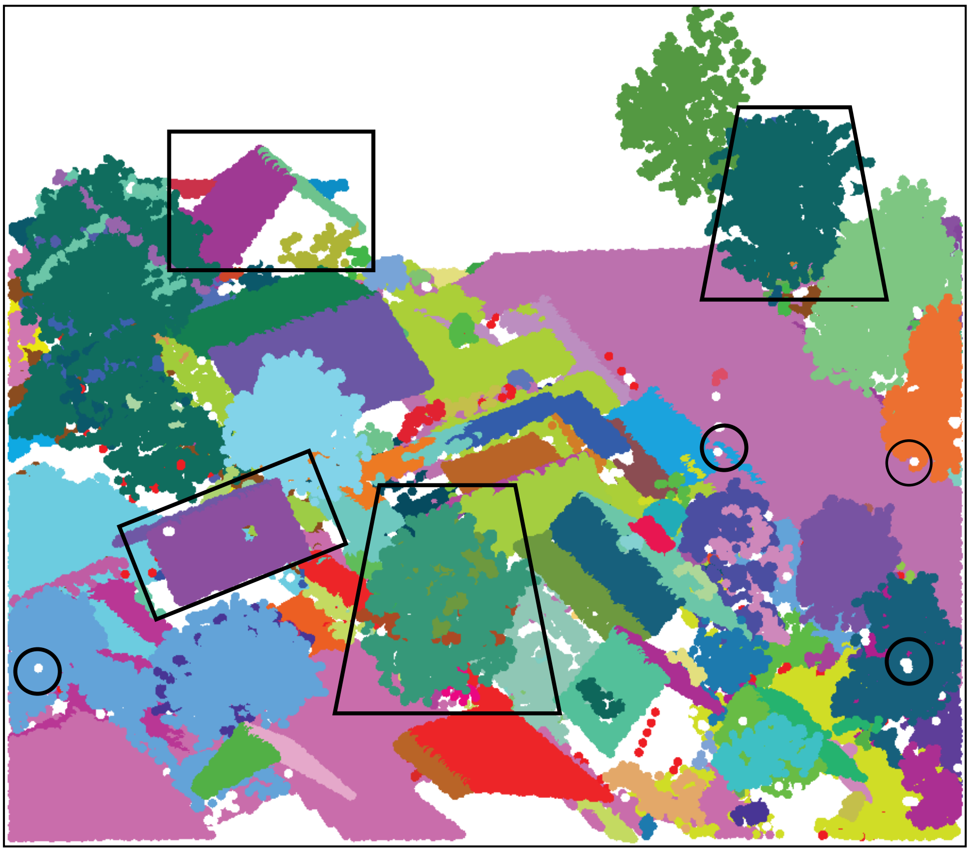


Figure 1.3. An example of different entities, segmented using the method published by Vosselman (2013). Different colours indicate different segments. There are points in circles, planes in rectangles and a tree as one segment in the keystones.

1.3.2 Change detection

The research problems associated with change detection do not lie in the characteristics of objects or defined classes. Rather, they lie in the ability to separate relevant changes from spurious and irrelevant changes. Spurious changes, which are the result of shortcomings of the change detection algorithm or a lack of data in one epoch of data, are the most common problem. An example of a spurious change caused by failure of the algorithm is error propagated from classification. In this type of algorithm, the target object is extracted after classification so that comparisons to identify change can be made. If the target object is correctly classified in one data set but is wrongly classified in another, a spurious change will be detected upon comparison. Spurious changes may also be caused by occlusion: a spurious change is detected when there is no point in one epoch of data because of occlusion but points for the same location are available in another epoch of data.

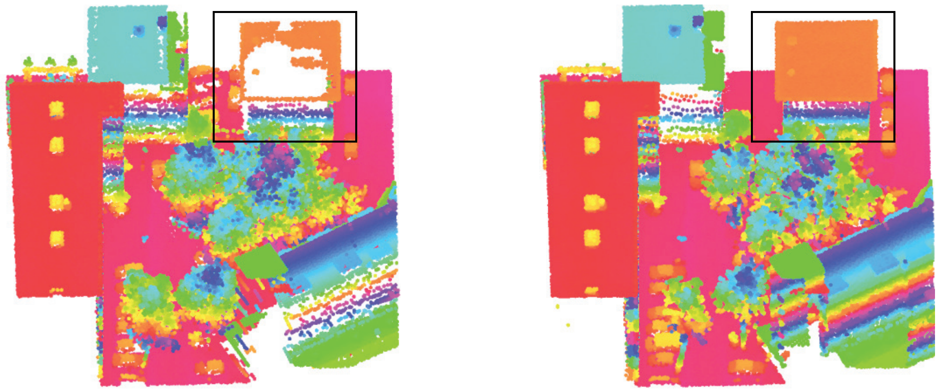
Besides spurious changes, there are some real changes that indeed occur in the real world and are detected as change in the data. Nevertheless, these real changes are not always changes we are interested in. Such irrelevant real changes also present a challenge for change detection processing. An obvious research problem is, then, how to separate spurious changes from real changes and how, simultaneously, to distinguish irrelevant changes from relevant ones.

Spurious change and irrelevant change

Spurious change: A spurious change is defined as a change detected in the data while in reality no change has occurred. Often, spurious change is caused either by incorrect classification or by lack of data points, the latter commonly the result of either occlusion or absorption of the laser beams by water after rainy days. Despite the fact that many researchers tried to solve the problem of occlusion by using occupancy grids (Hebel et al., 2013), there are still a significant number of problems to solve.

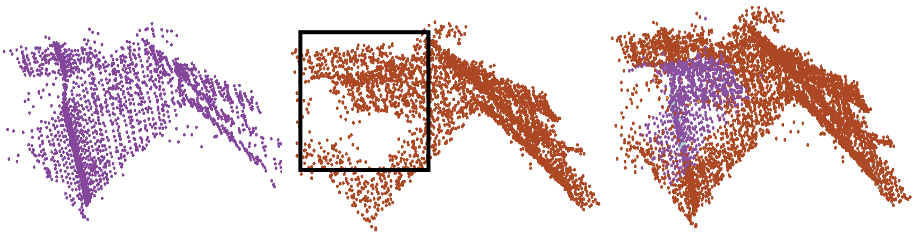
Figure 4 shows an example of spurious change that is caused by laser signal absorption on a building roof that cannot be solved using an occupancy grid. In Figure 4, we can visualize a “change” on the roof if we compare the two epochs of data in 1.4(a) and 1.4(b). However if one inspects the attached walls in the two data epochs no change to the roof would be expected: no change appears to have occurred on the walls, which implies that very probably the roof was in the same condition for both data sets. Nevertheless, in this case an occupancy grid would have detected a change. Another case is the newly built dormer on top of a gable roof shown in Figure 1.5. One would expect changes to be detected both for the dormer and for the roof below the dormer. Most existing methods would only detect changes on the dormer, however; the part on the roof below the dormer would be detected as occluded.

Irrelevant change: Elements attached to a building are quite dynamic and many small changes are commonplace. Vehicles are parking and leaving the flat roofs of buildings (parking places are sometimes created on the roofs of buildings), flags are flapping in the wind, sun shade panels and windows are open and closed from time to time, etc. Such changes, which are frequent and temporary, are irrelevant. To be useful, any algorithm used should be capable of distinguishing these from the long-term activities such as building construction or demolition.



(a) 2008 (b) 2010

Figure 1.4. Spurious change caused by lack of data for the roof in epoch 2008.



(a) (b) (c)

Figure 1.5. An example of a newly built dormer. Points in (a) and (b) are from different epochs. There is a newly built dormer located within the rectangle in (c). The dormer in (b) is newly built and part of the roof under the dormer has been demolished, i.e. changes have been made to both the roof and the dormer.

Challenges for the algorithm

Next to the challenges arising from detecting spurious and irrelevant changes, still further challenges are presented by gradual increases requirements such as high level modeling for the changes in dormers, chimneys, tree growth, etc. The aim is to be able to detect changes in roof elements such as dormers and vehicles on roof tops: to be satisfactory, the algorithm should be capable of classifying these roof elements. Previous researchers detected changes by comparing classified DSMs from different epochs. Unfortunately this leads to propagation of errors occurring from classification to change detection. The algorithm should, therefore, be capable of avoiding error propagation arising from classification.

1.4 Goals and objectives

The general goals of my research are: (1) to present an automated approach for explicitly classifying ALS point clouds into the categories water, ground, vegetation, building wall, building roof, roof element and undefined object; and (2) to automatically detect changes to buildings and to distinguish and quantify changes for dormers using the "classification of change" approach.

These general goals can be subdivided into the following specific goals:

- to design, implement and analyse an algorithm that can achieve a highly accurate classification result for each object class. Especially since a long list of object features is not suitable for large data sets, there is a practical advantage in exploring the potential of using "entity definition" rather than "feature selection" and "classifier selection" during classification. The results of this approach need to be evaluated and analysed.
- to design, implement and analyse a change detection algorithm that is capable of distinguishing real changes from spurious and irrelevant changes. The results should be visualized in an object-based manner so that customers are able to access each changed object.
- to design, implement and analyse a method for extracting dormers from the changed objects. The number of changed dormers also needs to be calculated.
- to statistically evaluate change detection results.

1.5 Thesis outline

This thesis is composed of seven chapters, of which the middle four chapters set out the main contributions to the methodologies and experiment results.

Chapter 1 places the research described in this thesis in its broad context: the reasons for the research, its scope, the problems to be addressed and the goals set for solving them.

As the study involves large data sets, the information in the data sets, such as data organization and pre-processing procedures, including pyramid generation and data filtering, are introduced in Chapter 2.
Chapter 3 includes a literature review of algorithms suitable for the classification of lidar data and explains how the data sets used in this PhD research are classified into water, ground, vegetation, wall, roof, roof element and undefined objects. By estimating that promising results can be achieved with multiple entities, even if only a few features are used, the data sets have been sorted into planar segments and a planar-segment-based classification performed. For each planar segment, values of five commonly used features from the literature were calculated, as well as the value of another feature especially for vegetation if multiple returns are unavailable in the data. After this step is carried out, most points for roofs, vegetation, water, ground and undefined objects are well separated. Classifying points from wall and roof elements is challenging because of their diversity and, as well, because occasionally points are missing. To compensate for this, contextual information and multiple entities are included in the strategy. Chapter 3 explains why they are included and how the choice of entity is made and combined with the method. Examples of classified data sets for both epochs are shown in Chapter 4. The chapter also includes an evaluation and analysis of the classification, including accuracy, confusion matrices, impact of features, and the impact of multiple entities.

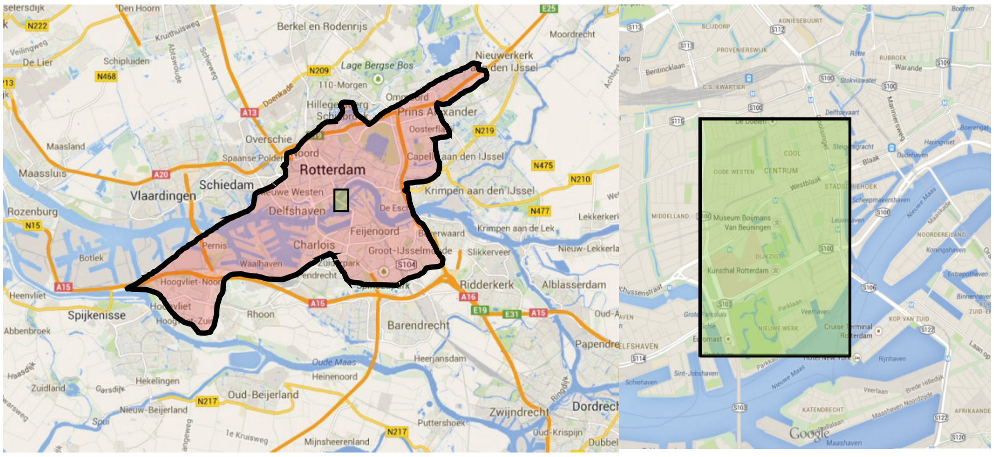
Chapter 5 reviews the state-of-the-art of change detection with lidar, and covers the change detection methodology used in my research. The classification result from Chapter 3 is not immediately used for comparison to identify change. Instead, a point-wise search for possible areas of change is made by merging two epochs of unclassified data. After identifying possible areas of change, areas that belong to a building in either epoch of data are extracted as changes in buildings. These areas are organized as object-based components and put into the "classification of change" step in order to extract dormers. The results and analysis in Chapter 6 involves examples of all types of change in buildings and a list of errors and the reasons for their occurrence. Examples of dormer area and volume calculation are also described in Chapter 6. To conclude this thesis, I present the conclusions and recommendations arising from my research in Chapter 7.

Chapter 2 Data sets

The lidar data used, its quality and the pre-processing procedures are explained in this chapter. The study area and the data sets are introduced in Section 2.1. Section 2.2 describes the data quality, in particular the point density and strip differences that occur. These latter two factors have the greatest impact on the classification and change detection results. Point density is considered during the feature extraction procedure and strip difference determines how well the data sets register with each other and thus affect the quality of change detection. Although they lead to uncertainties in X, Y and Z coordinates, other data quality indices such as errors due to sensor position, sensor and aircraft alignment, and time measurement are not discussed because strip differences already incorporate the effects of these errors. Moreover, if the expected change is larger than the offset between two strips, the errors mentioned above will not affect the change detection result. Data pre-processing procedures, which include data organization and data filtering, are introduced in Sections 2.3 and 2.4.

2.1 The study area

Three epochs of data of Rotterdam, the Netherlands, were made available by the Municipality of Rotterdam. Two epochs of the data, which were obtained in 2008 and in 2010, cover most of Rotterdam (Figure 2.1 (a)), stretching over approximately 120 km^2. The other epoch, which was obtained in 2012, covers 0.8 km^2 and is located in the northeast of Rotterdam (Figure 2.1 (b)). The study area is generally flat with only slight variations in relief; canals can be found throughout the whole area.



(a) (b)

Figure 2.1. The study area (from Google Maps). (a) Boundaries of data sets 2008 , 2010 and 2012(largest boundary for 2008 and 2010 data sets) (b) Boundary of data set 2012

2.2 Data Quality

The average point densities for the data sets are 20-30 points/m^2 for 2008, 30-40 points/m^2 for 2010, and 40-50 points/m^2 for 2012. As the data sets only have minor strip differences, no further registration was required. To verify that the registration accuracy is better than the maximum of 10 cm, several randomly chosen roofs were checked by comparing the planar surfaces in the different strips, either from within the same epoch or between different epochs: the systematic discrepancies were less than 10 cm. Furthermore, checking of the distances between ridgelines of building roofs in the overlapping strips showed that the differences were also less than 10 cm. Figure 2.2 is a sample of a randomly selected roof.

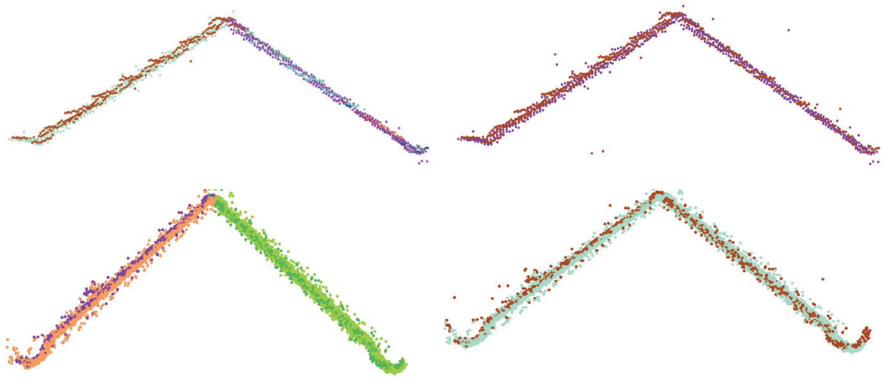
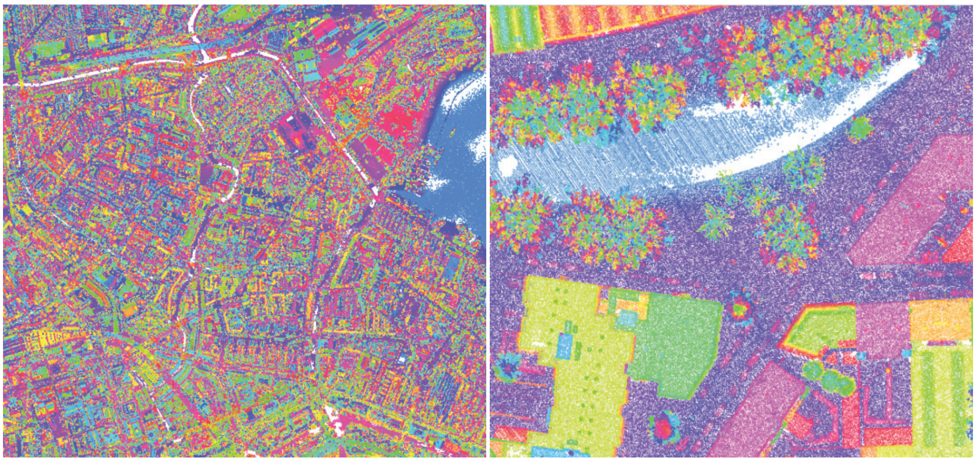


Figure 2.2 A sample of a randomly selected roof showing strip differences between different strips in the same epoch (top row); and different strips from different epochs (bottom row).(a) Colour stands for segment number; (b) Colour denotes different epochs

As the data sets are registered well enough to meet requirements for detecting change in buildings, pre-processing of the data could proceed, including: (1) organization of the large data sets for process and display; and (2) filtering (separating terrain points from non-terrain points).

2.3 Data organization

The data sets are stored in LAS file format, which is a compact binary file format specific to lidar data. It is a public file format for the interchange of 3-dimensional point-cloud data between data users (http://www.asprs.org/Committee-General/LASer-LAS-File-Format-Exchange-Activities.html). These large data sets are processed by cutting the data sets into small pieces (tiles) and visualising them by generating lidar data pyramids. Lidar data pyramids, which are similar to image pyramids, are composed of several levels. In each level, the point density decreases by selecting one point from a set of points in the previous level. Figure 2.3 (a and b) shows different levels in the data pyramids. The same tile boundaries are used in the data sets of all three epochs facilitating tile-wise change detection.



(a) (b)

Figure 2.3 Data view in different levels of the pyramid. (a) level 3, (b) level 0. Colours indicate the altitude.

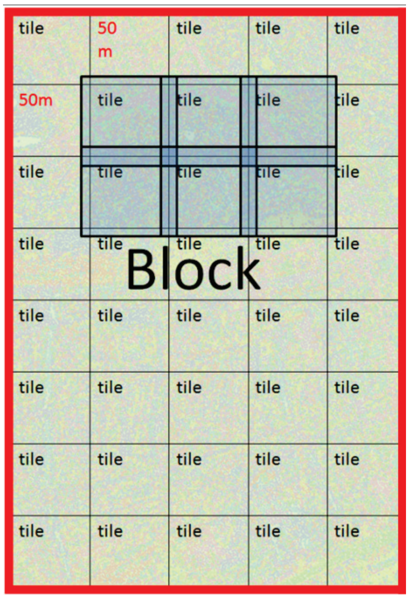


Figure 2.4 Example of a block with overlapping tiles.

Blocks and tiles

Figure 2.4 shows how a block is divided into many tiles. A tile comprises the ALS points within a 50 m x 50 m area; all processing is done tile by tile. In the data set, one block was generated for each epoch of data. The blocks for 2008 and 2010 each have 39,953 tiles, while the block for 2012 has 320 tiles. In total the blocks contain some 3 billion points.

Tile overlap

The tile overlaps are shown in Figure 2.5. They are used in scene classification in order to provide as much contextual information as possible. Usually, if an object such as a building or a tall tree by chance appears on tile boundaries, this object will be cut into different parts and stored separately in different tiles. In some of these tiles, only a small part of the object will be stored (e.g. Figure 2.5 (a)). If the tiles are processed one by one, it will be very difficult for the algorithm to interpret the object from the small piece of it occurring in the tile. In such cases, tile overlaps with its neighbours offer larger parts of the object and therefore can help to interpret these objects accurately.

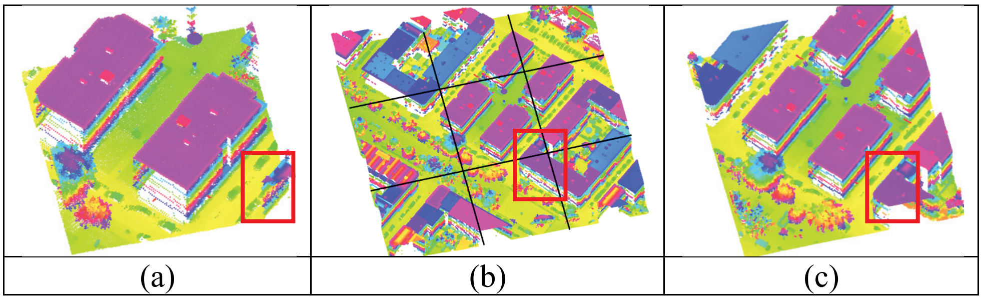


Figure 2.5 An example of tile overlap. Overlapping of tiles helps provide more contextual information about objects along tile boundaries. (a) Tile A ; (b) Combination with its eight neighbours; (c) Tile A with overlap.

Block bound

The block bound, shown as the large red rectangle in Figure 2.4, is a rectangular range that determines the positions of the first and last tiles in each block. For the purpose of generating correspondences for change detection, the same block bound is required to ensure that the same tile number represents the same location in two epochs of data sets.

2.4 Filtering

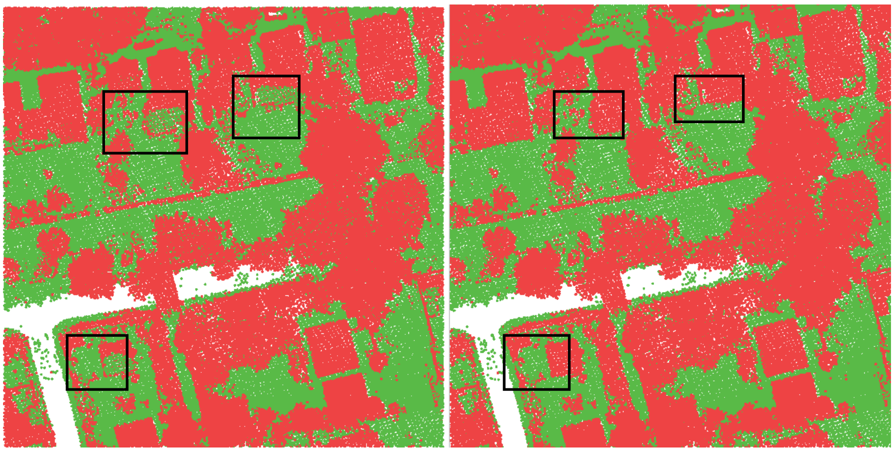
Besides the pre-processing explained above, "filtering" of lidar data is also required for classification since we want to calculate the height of points above the ground (represented by the Digital Terrain Model (DTM)). The term "filtering" usually indicates the process of bare-earth extraction of laser scanning point clouds.

In general, filtering methods belong to one of two categories: interpolation-based methods; or morphological methods (Chen et al., 2007). Lindenberger (1993) described how mathematical morphology can be used for filtering data recorded by a laser profiler (Vosselman, 2000); morphology opening was then improved by researchers to become a classic method for filtering. Chen et al. (2007) describes in detail about how they choose window size and how they cope with problems caused by tree areas, small outliers and built areas, as well as missing data. Sithole and Vosselman (2004) assessed eight different filtering methods; outliers, object complexity, detached objects, vegetation and discontinuity are the qualitative factors found in their report. In addition, they analysed the frequency of Type I (rejection of bare-earth points) and Type II errors (acceptance of object points as bare earth) of the various methods.

Filtering algorithms are a research topic in their own right, and I have not attempted to develop any filtering algorithms for my study. Nevertheless, the bare earth (or DTM) needs to be extracted to estimate the height of objects above the ground in the classification.

To quickly filter the data sets, use was made of LAStools, free and commercial versions of which are available, to process lidar point clouds. After filtering, the output was visualized. Most DTM points were correctly extracted, although some errors were observed for roofs (Figure 2.5): some roof points were wrongly assigned as terrain, due to the small height differences between some low buildings and the surrounding terrain. Incorrect assignment has an effect later for the scene classification process, since it was assumed that all the input data sets are correctly filtered into ground and non-ground elements.

Fortunately, such filtering errors can be corrected. The data set obtained in 2010 was already interactively filtered before it was made available for my study. Hence, the DTM for 2010 is more accurate. Under the assumption that the urban topography of the study area will have remained virtually unchanged over a period of 5 years, the DTM in 2010 was selected as a reference for correcting filtering errors in other epochs. Figure 2.6 shows an example of the correction of filtering error in the 2008 data set. In the left-hand image there are some green points in the rectangles that are actually roof points but have been filtered as ground. This error was corrected by: (1) extracting the nearby DTM from 2010 datasets which is taken as a reference DTM; (2) comparing the filtered terrain points to the reference DTM, and; (3) checking their fitness per point by calculating the vertical height difference. If the height difference is greater than 1 m, the terrain points will be re-labelled as "non-terrain" points, as is shown in the rectangles in the right-hand image.



(a) (b)

Figure 2.6 Correction of filtering errors from "LAStools" (red = above ground; green = ground points). (a) Filtering with LAStools;(b) Correction against the reference DTM in 2010.

As correction has been performed under the assumption that there has been no change to the terrain, the correction method may still result in errors if there have been great changes to the terrain, e.g. on large-scale construction sites. After correction, all buildings are labelled as non-terrain points. The data sets are then fully prepared for the scene classification process.

Chapter 3 Methodology for the scene classification

This chapter is an introduction to the methodology of the scene classification, by which we intend to classify all the data sets into water, ground, vegetation, wall, roof, roof element and undefined object. The "vegetation" is defined as trees taller than 2.0m. Low vegetation such as meadow and bush are taken as the "undefined objects". The "Building roof" is defined as a structure larger than 3m in size and larger than 4.0m^2 in area. The "Roof element" is attached to and on the top of the "building roof", such as dormers, chimneys, sun panels etc. The "Building wall" refers to facade as well as the attachments to the facades, such as sun shades, balconies, windows etc. All other objects belong to "undefined objects". These rules, which are the basic rules for the scene classification, are also the basis for making reference data for training and evaluation.

3.1 Introduction

In the past few years, ALS data have been widely used in urban applications, such as building extraction (Sithole and Vosselman, 2004), building reconstruction (Oude Elberink and Vosselman, 2009a), building change detection (Murakami et al. 1999), bridge detection (Sithole and Vosselman, 2006), road extraction (Samadzadegan, 2009) and road reconstruction (Oude Elberink and Vosselman, 2009b). With the improvement of ALS data point density and accuracy, there are more and more high-level requirements for updating maps, disaster evaluation, illegal activity detection and detailed modelling, etc.. Fine pre-processing of ALS data, such as high-level classification with high accuracy, is necessary for these requirements.

Most ALS data classifications are two-class problems, in which researchers focus on extracting one certain object. In most of the scene classification cases, three classes are frequently used, namely: vegetation, building and ground (Samadzadegan et al. 2010), these are the elementary objects in urban areas. Although multiple classes are usually defined and classified in terrestrial laser scanning points, a multiple-class system with seven classes is rarely seen in airborne lidar data. As we will classify the urban scene into: water, ground, building, vegetation and undefined object, in which "building" is further divided into roof, wall (including windows, balconies and all objects attached to walls), and roof element. We face two difficulties to classify the targeted classes: (1) Walls and roof elements are quite irregular in geometry, due to either various construction styles or a lack of data for walls and roof elements (mainly caused by occlusion or rainfall). It is hard to define proper features for them. (2) Roofs partly or largely covered by vegetation add another complication to the classification. When vegetation is both above roofs and near roof elements, the classifier cannot distinguish vegetation from a roof element because they are both irregular in geometry and are above the roof in space (shown in Figure 3.1). In these areas, if we can ensure different objects belong to different segments, the classification will be much easier.

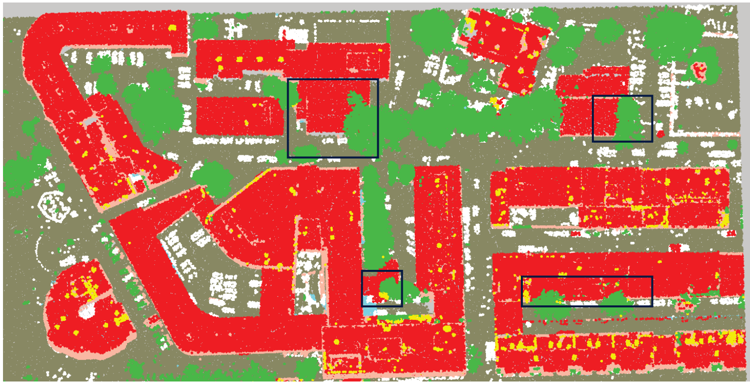


Figure 3.1. Example of an urban scene. Many roofs are covered by vegetation in this area, roof elements and vegetation are not easily distinguished as they both contain multiple pulse returns and have a wide variety of shapes and sizes.

Current approaches to complex urban “scene classification” using ALS data are: Lafarge and Mallet (2011) use an energy function and Potts model (based on Li., 2001) to combine local features and local context. Chehata et al. (2009) select several best performance features from a large amount of possible features generated from both discrete-return and full-wave ALS data. Rottensteiner et al. (2007) fuse multi-spectral images and ALS data.

Most of the above approaches use only a single entity to calculate features. The combination of multiple entities, however, has been seldom discussed in previous literatures. A portion of the existing classifications is implemented with point- (or pixel-) based features. These methods can obtain accurate classification results, but are time consuming for large data sets. Other works use segment or voxel-based features, which attempt to speed up the computation. Multiple-entity features are utilized in several literatures (Lim and Suter, 2009), (Kim and Sohn, 2011). However, Lim and Suter (2009) did not study the impact of the entity on classification accuracy. Although Kim and Sohn (2011) generated point-based and object-based features, to achieve a classifier fusion, they used these two entities separately to generate two classifiers.

Therefore, we put forward the assumption is that if points are organized in multiple ways to be adaptive to the shapes of different classes of interest, a few features will obtain high accuracy and a multiple-entity strategy will achieve higher accuracy than a single-entity method.

To verify this assumption, we propose a strategy to classify ALS data which combines three types of entities. The impact of these entities on the overall accuracy, and the accuracy for each class of interest is also studied. The particularities of the classification strategy are:

1. Seven classes are classified simultaneously and directly on 3D lidar points. These classes are basic elements (ground, water surface, building, vegetation and undefined object) in an urban scene. To enhance the capability to perform building analysis and modelling, the building class is further decomposed into roof, wall and roof element.

2. A multiple-entity strategy is employed and researched. By analysing the challenges in the urban scenes in Figure 3.1, we generate three entities (planar segments, segments obtained by mean shift segmentation, and individual points) according to the characteristics of different classes.

3. The contextual information of a building is employed to separate walls and roof elements, and is proved to be a good feature for all types of walls and roof elements.

4. We define a new feature, namely, “average point spacing” in planar segments for classifying vegetation.

The remainder of this chapter is organized as follows: Section 3.2 contains a literature review on how various entities, features and classifiers are currently being used in ALS data classification methods.

Our proposed entities, features for the entities are described in Section 3.3, and the classification strategy is discussed in Section 3.4.

3.2 Literature Review

The classification of ALS data in urban areas has long been studied and discussed in many literatures. Lidar-data classification methods on DSMs can be found at (Matikainen et al. 2007; Rutzinger et al. 2006; Rottensteiner et al. 2004). Matikainen et al. (2007) use a list of features calculated from first pulse, last pulse DSM segments and image segments, and they run a classification tree on four different combinations of these features to extract buildings in urban area. They explain that the difference in accuracy between the four different groups of feature combinations is very small, and that the overall mean accuracy is about 90%. Rutzinger et al. (2006) choose crisp thresholds for features derived from DSM segments to detect buildings. Different from these two segment-based methods, Rottensteiner et al. (2004) initially achieve a per-pixel classification with Dempster-Shafer Fusion on first and last DSM and NDVI derived from multi-spectral image, and then use morphologic opening to eliminate single building pixels. Finally, they run the Dempster-Shafer for the second time to improve the results, but still there are errors caused by shadows in the image and the resolution of the lidar data. Khoshelham et al. (2010) evaluate these automated approaches to building detection on both pixel level and object level and draw the conclusion that Dempster-Shafer and AdaBoost work best. They state that most errors occur at building boundaries and in areas where dense trees are present.

Chehata et al. (2009) explicitly select, analyse, and compare Lidar point features. They propose 21 features which are grouped into height-based features, echo-based features, eigenvalue-based features, local plane-based features and full-wave lidar features. Feature selection is accomplished by iteratively fitting the Random Forests after eliminating a small fraction of features according to the importance of features ranked in Random Forests. Finally, six features from height-based, local plane-based and full-wave features are kept for the final classification, which keeps the error of the classification stable and below 6%.

Classifiers that are usually seen in lidar data classifications are Random Forests (RF) (Chehata et al. 2009).), AdaBoost (Lodha et al., 2007), Artificial Neural Networks (ANN) (Priestnall et al., 2000), Supported

Vector Machine (SVM) (Lodha et al., 2006), Expectation-Maximum (EM) (Lodha et al., 2007) and Dempster-Shafer (Rottensteiner et al. 2004). These classifiers are irrespective of the contextual information and only consider local features. However, they are automatic and work stable. Lodha et al. (2007) claim that the classification accuracies obtained by EM, SVM and AdaBoost appear to be quite similar. Popular classifiers which take contextual knowledge into account are Markov Random Field (MRF) and Conditional Random Field (CRF), and these techniques are increasingly applied to lidar data classification. Anguelov et al. (2005) use the associate MRF classifier, and Munoz et al. (2008) try to improve the results of the associate MRF by using more accurate pair wise potentials. Munoz et al. (2009) and Lim and Suter (2009) separately use CRF on points of a clique, and on voxels, to speed up the computation. The limitations of these classifiers are that they are slow, and both small and big objects are easily wrongly classified due to over smoothing. If a small region in particular is wrongly classified, it is possible that a large region near it will be wrongly classified due to the error propagation (Shapovalov et al., 2010).

For classification methods that directly run on 3D point clouds, there are typically four kinds of entities mentioned in current literatures: point, segment, voxel and object.

3.2.1 Classification using point features

Methods using point-based features can be found at (Lafarge and Mallet 2001; Niemeyer et al. 2011, Brodu and Lague, 2012). Lafarge and Mallet (2011) use discrete features generated from points with neighbours. These features are normalized and formalized in an energy function, which is composed of a sum of partial data term and pairwise interaction as defined by the Potts model (Li, 2001). The class label choices are made by minimizing the energy function. Lafarge and Mallet concludes that some local errors which have no consequence on the final results. Niemeyer et al. (2011) combine features with contextual information using conditional random fields in point-wise classification. They use features derived from full-wave form lidar data, which results in an overall accuracy of 94% with a lower correctness of only 87.7% for building compared with the other two classes (ground and vegetation). Brodu and Lague (2012) introduce a multi-scale dimensionality criterion for point features to classify complex natural scenes. They use local dimensional properties at each point, and at different scales, as input features for classification. These local dimensional properties are expressed by eigenvalues calculated from points in stepwise increasing neighbourhoods.

3.2.2 Classification using segment/voxel features

Classification approaches that work on segments are (Wei et al. 2009; Darmawati, 2008). In Darmawati's (2008) method, a rule-based classification method is applied to planar segments derived by surface growing (Vosselman et al. 2004). Pulse count information is used to separate buildings and vegetation in urban areas. A normalized cut approach is used to classify segments derived by mean shift in Wei's (2009) work in order to extract flyovers and vehicles in complex urban areas. Results from the mean shift segmentation are further classified based on geometric features such as the horizontal and vertical similarity within each mean shift segment. The flyovers are 100% extracted, however, vehicles have a much lower extraction rate at about 50-70%, which happens in areas such as parking places with many vehicles.

Examples of methods using voxel features can be found in (Lim and Suter, 2008; Lim and Suter, 2009). Lim and Suter (2008) over-segment the 3D data using 3D scale theory to form super-voxels. Multi-scale Conditional Random Field (mCRF) is run regionally on super-voxels, and locally within each voxel on the points. Lim and Suter (2008) conclude that mCRF improves the classification accuracy by 5% - 10%. Based on this work, Lim and Suter (2009) discuss the advantages of using super-voxels instead of points, and conclude that, with super-voxels, the total amount of data is reduced to 5% (in most cases) of the original data.

3.2.3 Classification using multiple entity features

The work of Kim and Sohn (2011) is an example of using both point-based and object-based features. They propose a method which uses multiple classifiers (Random Forests) to classify an urban scene into: building, vegetated, wire and pylon. Line and plane segments have been obtained by RANSAC and Minimum Description Length (MDL) based on voxel-segmentation results. These line and plane segments are further examined with shape criteria to correct erroneous segments. Point-based features and object-based features are extracted from these segments. Point features and object features are used as separate inputs for two

Random Forests-classifier generations. They conclude that the classifier fusion improved the accuracy by 10%. Xiong et al. (2011) use both bottom level (point) and top level (segment) features to classify lidar point clouds. They incorporate the context information by using the k-class logistic regression model to train the weights of both the geometry and contextual features in a stacking method. These weights indicate the up-middle-down relationship possibilities between the classes. They conclude that by using both bottom and top level features, both the classification accuracy and computation cost are improved.

Unlike the classifiers that only consider local features, there are one kind of classifier which takes contextual knowledge into account by checking the homogeneity and configuration of the whole classified points. Representative ones that belong to this kind are Markov Random Field (MRF) and Conditional Random Field (CRF), and these techniques are gradually extended from 2-D images to 3-D lidar data classification. Anguelov et al. (2005) use the associate MRF classifier, and Munoz et al. (2008) try to improve the results of the associate MRF by using more accurate pair wise potentials. Munoz et al. (2009) and Lim and Suter (2009) separately use CRF on points of a clique, and on voxels, to speed up the computation. The limitations of these classifiers are “over smoothing” and “error propagation”. The over smoothing problem will happen when for example a small tree is standing closely to a tall building. In this case, if a small part of the tree is incorrectly classified as building, it will be very likely that whole tree will be classified as a building by using the homogeneity theory. If a small region in particular is wrongly classified, it is possible that a large region near it will be wrongly classified due to the error propagation (Shapovalov et al., 2010). Unless stated above, these classifiers are also very slow in computation. Therefore, we will not choose this kind of classifier for large data sets.

3.3 Entities for classification and their features

Based on the analysis of the complexity of the scene in Figure 3.1 and the overview of the previous works, we start our classification by first considering different entities. Because different entities can form different divisions of points that are adaptive to the objects. After considering the entities and their features, we designed a four-step classification strategy to use different entities and their features in different steps. Definitions of entities and their features are introduced in Section 3.3.1 and 3.3.2, and the strategy is illustrated in Section 3.4.

3.3.1 Three types of entities

3.3.1.1 Introduction to three entities

There are seven classes for the scene classification, and it is difficult to choose one type of entity suitable for all classes. In our observations, we noticed that water, ground and roof are mostly planes, and objects from other classes are quite irregular in geometry. Therefore, we defined three kinds of entities (Figure 3.2), namely, (a) single points, (b) planar segments and (c) segments obtained by mean shift segmentation according to the characteristics of each class and errors that may occur during the classification (introduced in Section 3.4). Features of single points are derived from either a point itself or from a set of neighbouring points within a radius. This entity represents the local characteristics of an object, and it is sensitive to outliers and noise. Planar segments are obtained through surface-growing segmentation (Vosselman et al. 2004b). Segment features are derived from properties of the segment or by aggregating features of the points within a segment. Every point in a segment has the same feature values. The same holds for the segments derived by mean shift segmentation (Comaniciu and Meer, 2002; Ferraz et al., 2010). Because of different segmentation algorithms, the grouping ways of points into segments, as well as the segment feature values, are different. Both planar segments and segments derived by mean shift are stable in the case of data noise, and represent the uniform characteristics of a segment.

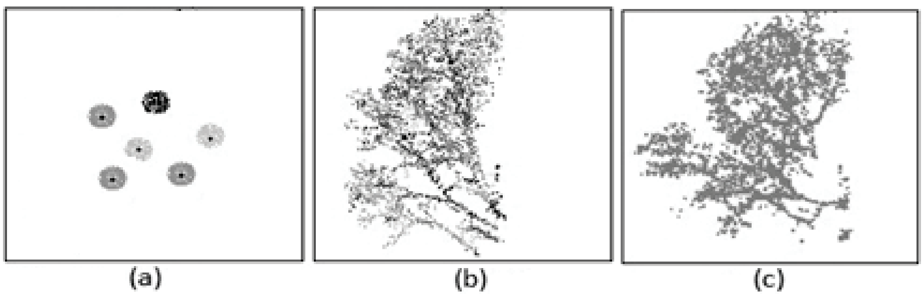


Figure 3.2. Three types of entities used. (a) Point features, the central points are in black in the middle. (b) Planar segments in vegetation, vegetation is cut into different planes (grey and dark points belong to different segments). (c) One mean shift segment from part of a tree (all points belong to one segment).

In our workflow, we first choose planar segments to obtain a rough classification of the following: ground, water, vegetation, roofs and undefined objects. We subsequently use this classification result as context for point-wise classification of walls and roof elements. Finally, errors caused by the contextual information are re-segmented with mean shift segmentation and are reclassified.

The reason for using three entities, rather than a single one, is because a single entity cannot represent various complicated objects. For example, building walls do not have any unique characteristics because walls have different types, and because laser points have different point density on walls, which is caused by the scanning angle of a scanner. A unique property of walls is that they are always below the nearest building roofs. Similar reasons also hold for roof elements. However, areas with vegetation above the roof are often present in urban scenes (Figure 3.3 (a)). In these areas, vegetation is also above and near the building roof and is normally classified as roof element (Figure 3.3 (b)) because it has the same property as roof element. To locate these areas, points that share the same topological relationship with the same building roofs are grouped into one component (Figure 3.3 (c)). These components are the areas in which errors may occur.

Mean shift is a non-parametric mode-seeking algorithm, and through mean shift we can obtain some genuine clusters of 3D point clouds without enforcing any model assumptions (Wei et al. 2009). Therefore, mean shift is performed on components that are near and above a building to re-cluster points according to their locations and features (Figure 3.3 (d)). A ratio of distances from a point to the nearest roof and vegetation, and the X-, Y- and Z-coordinates are the input features for mean shift; the band width is adaptive to the range (difference between the largest value and the smallest value) of each feature value within one component. This ensures that most of the vegetation and roof elements belong to different segments.

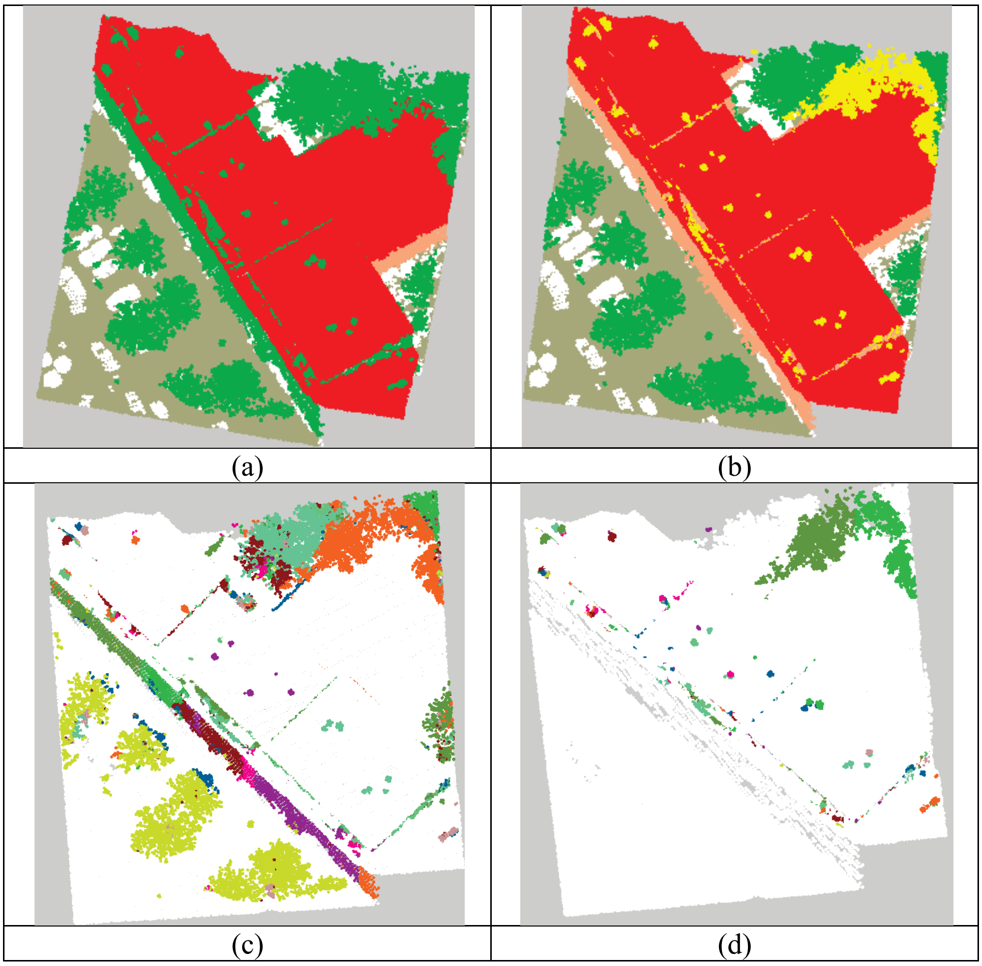


Figure 3.3. (a) Building roofs with vegetation and roof elements are misclassified as vegetation; (b) All points above the roof are classified as roof element after using the contextual information for the building; (c) Potentially wrongly-classified points are grouped into components, and different colours represent different components; (d) Potentially wrongly-classified points near roofs and above roofs. These points are re-clustered (different colours represent different MS segments) by mean shift, which separates the vegetation and roof element points from each other.

3.3.1.2. Parameters for segmentation

Two segmentation methods, namely, *surface growing* and *mean shift*, are used to form different entities. There are some important parameters used in our experiments which should be mentioned (Table 3.1). The values of these parameters are determined by repeated adjustments and tests.

Table 3.1 Parameters for segmentation.

Surface Growing				**Mean shift**
Parameters for seed selection		Parameters for growing		Window size
Seed neighbourhood radius	1.0m	Growing radius	1.0m	
Minimum number of seed points	10 pts	Maximum distance of a point to the plane	0.2m	$(Value_{max} - Value_{min})/4$

In Table 3.1, the "seed neighbourhood radius" and the "growing radius" indicate the maximum radius for allowing a point participating in the seed plane extraction and the plane growing phase. The "minimum number of seed points" means the minimum amount of points that are required for initialization of a plane. The "maximum distance of a point to the plane" ensures that only nearly co-planar points are added to the seed plane.

3.3.2 Features for classification

Various features are computed for these three entities. These features are listed in Table 3.2 and feature behaviours are analysed in this section.

Table 3.2. List of entities and features.

Entity name	Point	Planar segment	Segment derived by mean shift
Feature name	Height variance	Segment size	Average distance ratio
		Maximum height to DTM	
	Distance ratio	Average point spacing (APS)	Segment size
		Percentage of points with multiple pulse count	Average height-variance
	Local plane fitting residual	Plane fitting residual	Normal of planes
		Normal	

3.3.2.1 Point features

Point features are employed either as a feature for classification of a single point or as a feature that is aggregated to a segment feature for a segment-based classification. We calculated three point features.

Height-variance is the variance of the Z-coordinates of the neighbourhood points; it has been introduced mainly to separate the roof elements from vegetation above roofs. Dormers above the roofs have a smaller Height-variance than vegetation.

Distance ratio is calculated based on the results of the classification for planar segments (as introduced in 3.3.1). Assuming that all points are labelled correctly, the points located nearer to vegetation, and further away from roofs, are more likely to be vegetation points. This feature is used as an input feature for mean shift, and is also used as a feature for classifying the resulting segments.

Local plane fitting residuals represent the roughness of a plane fitted to a point and its neighbours. Planar objects like roofs tend to have a smaller value than irregular objects like vegetation.

3.3.2.2 Features for planar segments

Planar segments are derived by a surface-growing algorithm. Their features are used in a rule-based classifier to obtain a rough classification result.

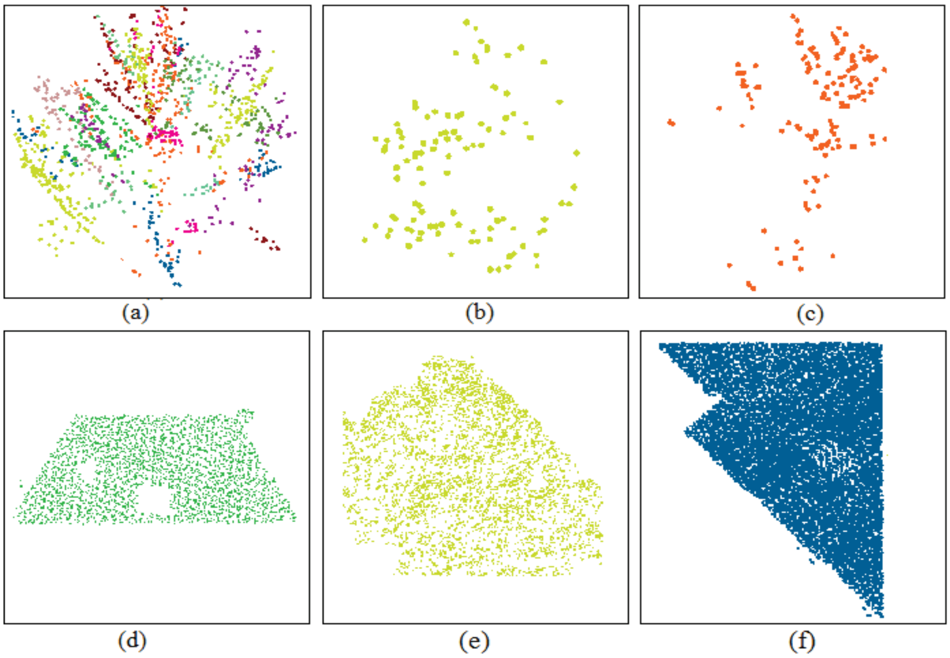


Figure 3.4. Planar segments derived by surface growing (a) Surface-growing segmentation on vegetation (different colours show different segments); (b) and (c) are two segments from (a); (d) one segment from roof; (e) one segment from water surface; (f) one segment from ground surface. The average point spacing in (d) is smaller than (b) and (c), and the average point spacing in (f) is smaller than in (e).

The size of each segment is expressed by the number of points in the segment and is an approximate indication of the area of the segment. The maximum height to the nearest DTM points allows for a distinction between high objects (roofs and vegetation) and low objects. Average point spacing (APS) is introduced to separate water from ground surface (Figure 3.4 (e)(f)). It also shows a significant difference in value for planar segments of vegetation and building roofs (Figure 3.4 (b~d)). As the point density within a segment in vegetation is low, the APSs for these segments are larger than for roof segments. This provides an alternative feature for ALS data when pulse counts are not available. If pulse count information is available, the percentage value of points with multiple echoes differs between vegetation and roof segments. Combinations of APS and pulse count information may improve the detection rate of vegetation, as will be identified later in the results. Different behaviours of these features are shown in various colours in Figure 3.5.

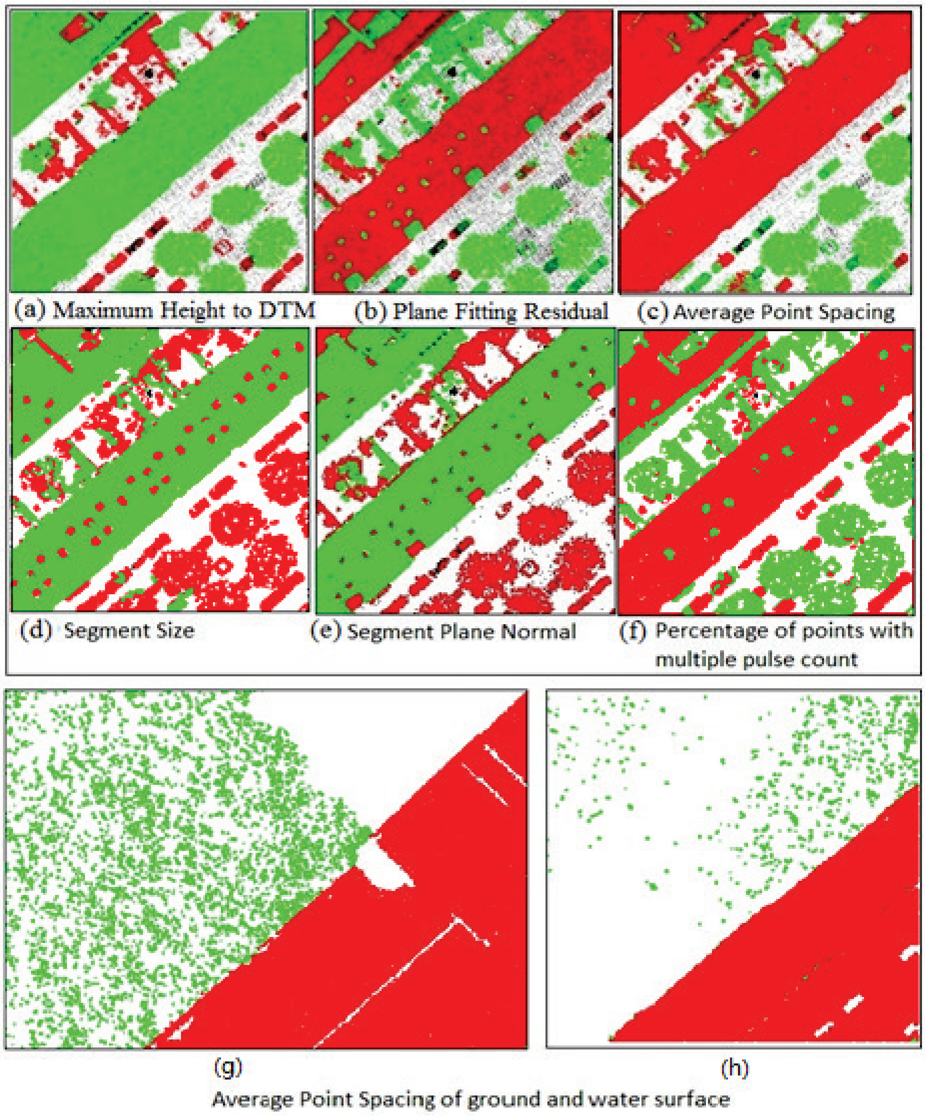


Figure 3.5. Behaviours of planar-segment features. (The colour red represents low feature values; the colour green represents high feature values). All the behaviours of features for above ground objects are shown in (a)-(f), white points are DTM points; feature behaviour for ground and water is shown in (g)-(h), white represents no points.

3.3.2.3 Features for mean shift segments

For the segments derived by mean shift, four features (Listed in Table 3.1) are calculated. Mean shift segments are irregular in space; therefore, we generate the features by aggregating the point feature values within the segment. Segment size is useful to distinguish vegetation from a roof element when a large area of roof is covered by vegetation. For lighter vegetation coverage, average height-variance and number of non-vertical and non-horizontal planes within the segment are used, and greater values for these features indicate a high probability of being vegetation. Examples of mean shift segments can be found in Figure 3.3(d).

3.4 Classification Strategy

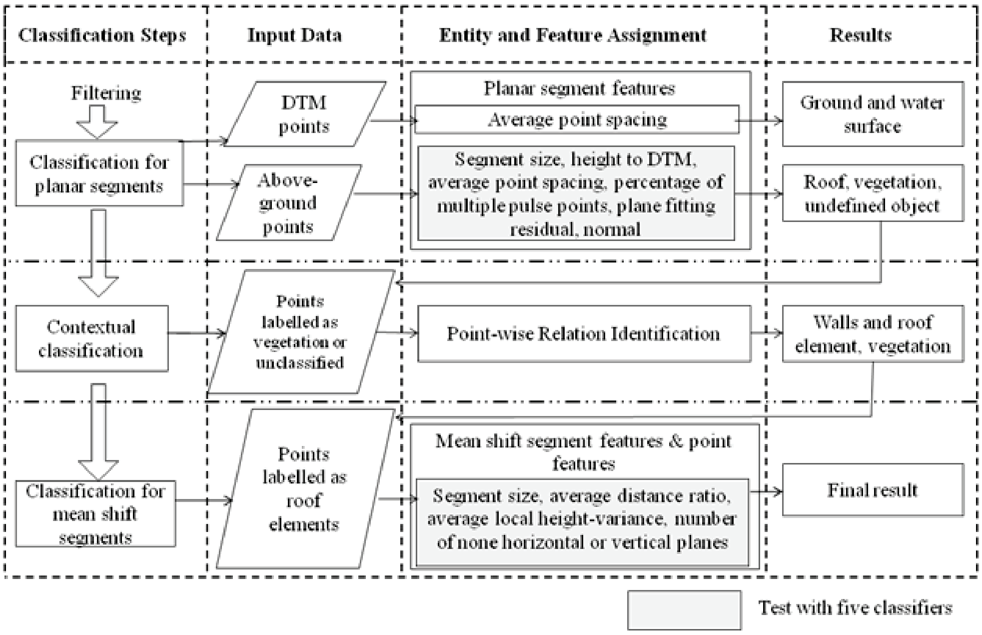


Figure 3.6. Classification strategy: the classification is composed of four steps as listed in the left side in the figure, and the corresponding input data, entity and feature used, and final output, are listed in columns.

The classification entails four steps. The data is first filtered into ground points and non-ground points. This is done by the data supplier. The non-ground points are segmented into planar segments using the surface-growing method. With the features from Table 3.1, planar segments are classified into: water surface, ground surface, building roof, vegetation and undefined objects. These classified points offer general contextual information for the next step. Points which are not classified as roof are point-wise classified as either a wall or roof element based on the contextual information which indicates the relationship between these points and the nearby roof points. This step leads, in the meantime, to the misclassification of vegetation to roof elements in cases where the vegetation covers the roof. The possible areas for these cases are searched for and found. In these areas, neither points nor planar segments are suitable entities to distinguish roof elements from parts of vegetation. Therefore, mean shift is used to re-segment these areas to separate vegetation from roof elements. The flow of the classification and the entity and features used are shown in Figure 3.6.

In order to find out whether the classification strategy performs well with different classifiers, both planar and mean shift segments are classified with five different classifiers. These classifiers are frequently used and are introduced in Section 3.4.1. The contextual-rule classification step (in Section 3.4.2), for which wall and roof element are point-wisely classified, is performed equally for all classifiers.

3.4.1 Classification for Planar Segments

The data set is filtered into terrain and above terrain by the data supplier. Our first step is to segment the data into planar segments using the surface-growing method. Planar segment classification is decomposed into a two-class problem (water and ground) for ground points and a three-class problem (roof, vegetation and undefined objects) for the non-ground points. For water and ground, the distinguishing characteristic is that the point density on water is lower than that on ground because of the water absorption and mirror reflection of the laser beam. Within each planar segment, the average point spacing (APS) is calculated. 737 training samples of planar segment are generated and the threshold for water and ground can be found with the histograms shown in Figure 3.7. Water and ground is classified with only one feature: average point spacing. Because Random Tree cannot be generalized with only one feature, we only used the other four classifiers (as introduced in the following paragraph) to classify water and ground.

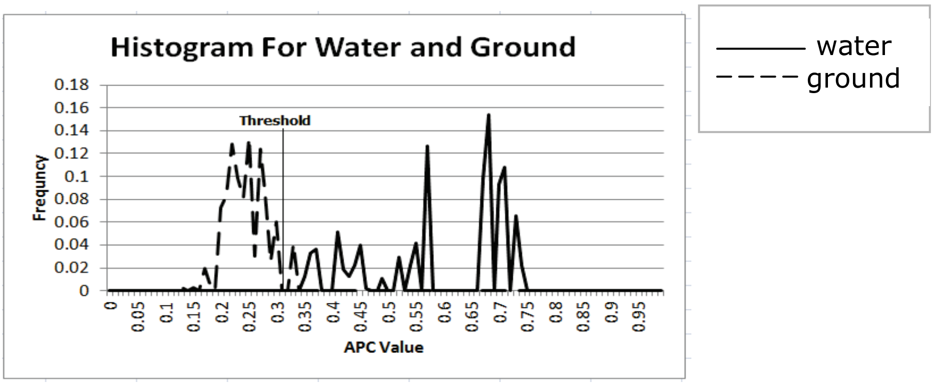


Figure 3.7. Histogram of average point spacing value of water and ground from the training data

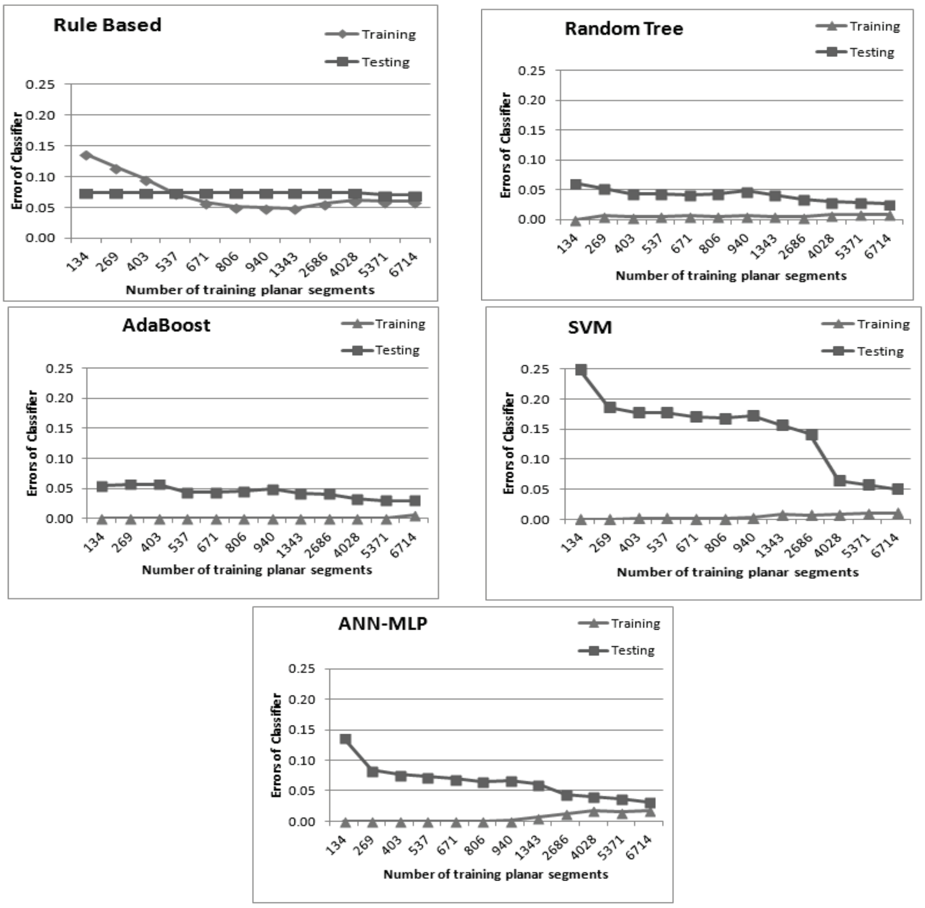


Figure 3.8. Learning curves for planar segments with five different classifiers. All accuracies are calculated with a 2-fold cross validation method (holdout method).

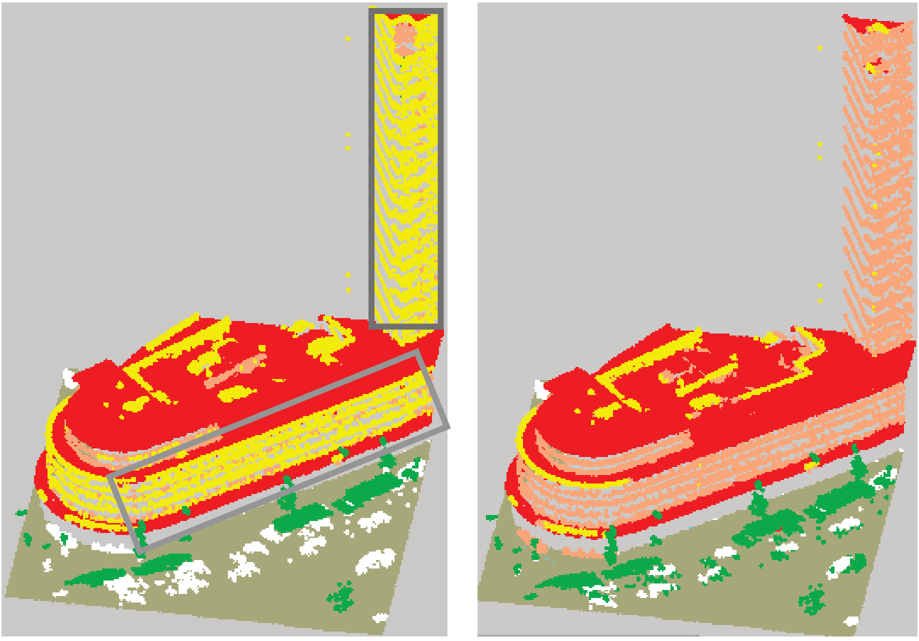
For the three classes above ground, five classifiers are used, namely: Rule-based classifier (RB), Random Tree (RT), Extended AdaBoost for Multiple Classes (ADB), Artificial Neural Networks–Multiple Layer Perceptrons (ANN_MLP) and Supported Vector Machine (SVM). RB is similar to a classification tree, and we use the RB in (Xu et al. 2013). RT is a variant of bagging proposed by Breiman (2001). It is a decision tree-based ensemble classifier which performs excellently in classification tasks, comparable to that of boosting (Breiman, 2001), even SVMs (Pal, 2005) (Chehata, 2009). ADB is an elegant algorithm which automatically combines rough guesses or weak hypotheses into a stronger, more general classifier by training on some data set with a known classification. ADB is at heart a binary (2-class) algorithm; however, there are several extensions (such as AdaBoost.M2, AdaBoost.MH, or error-correcting codes) which allow for multi-class categorization (Lodha et al. 2007). ANN_MLP is the most commonly used type of artificial neural networks (from: http://opencv.willowgarage.com/). SVM is a kernel-based method, it maps feature vectors into higher-dimensional space using a kernel function, and then it builds an optimal linear discriminating function in this space (from http://opencv.willowgarage.com/). RT, ADB, ANN_MLP and SVM are implemented in the OpenCV library, and all of the parameters for these classifiers are the suggested parameters, which are optimized in OpenCV.

Learning curves for these classifiers are graphed for planar segments (Figure 3.8), and all of the curves can converge when both errors from training and testing data do not have any big changes. Although a few training samples are sufficient to obtain high accuracy in RT and AdaBoost, other classifiers, such as SVM, still require more training samples to converge. To treat every classifier equally, we chose the largest training sample number (4,028 planar segments) from SVM for all the classifiers.

3.4.2 Contextual classification for walls and roof elements

Assuming that all roof points classified in the planar segment classification are correct, walls and roof elements are classified according to the contextual information of a building. The defined contextual information is that "wall" is always near and below a roof, and "roof element" is always near and in a higher location than a roof. Following this rule, we associate each point with the nearest roof point (check its planimetric (2D) distance within 1.0m) and check their spatial relations.

For complicated buildings, such as tall buildings connected to low buildings, the classification of wall points will be incorrect if the lower roof is nearer than the higher roof (light points in Figure 3.9 (a)). Therefore, we always search for the nearest highest roof point within a 1.0m planimetric neighbourhood. The defined context is treated as a point feature to classify wall and roof element. (Figure 3.9 (b)).



(a) (b)

Figure 3.9. Example of a complex building. (a) Wall points are wrongly classified as roof element (light) in the blue rectangles, which is caused by the wrong choice of the nearest roof points; (b) Points in rectangles in (a) are corrected by using the "nearest, highest roof point" constraint (the darkest points are roof points, lighter points connected to the roof are roof elements, slightly darker points are wall points.)

3.4.3 Classification for mean shift segments

After the planar-segment classification process and the contextual-rule classification, two categories of errors emerge. The first kind of error occurs if the features used are insufficient to distinguish undefined classes. The second kind of error is caused by the context-rule classification of wall and roof element. Vegetation above the roof is wrongly classified as roof element. This error can be corrected. To correct this error, 4,981 mean shift segments in total are derived in areas where vegetation covers the roof. (These areas are near building roofs, and the generation of mean shift segments is explained in 3.3.2). Learning curves show that 10% (498 from all 4,981 segments) of the reference data can train high-accurate classifiers for all five situations. (Figure 3.10)

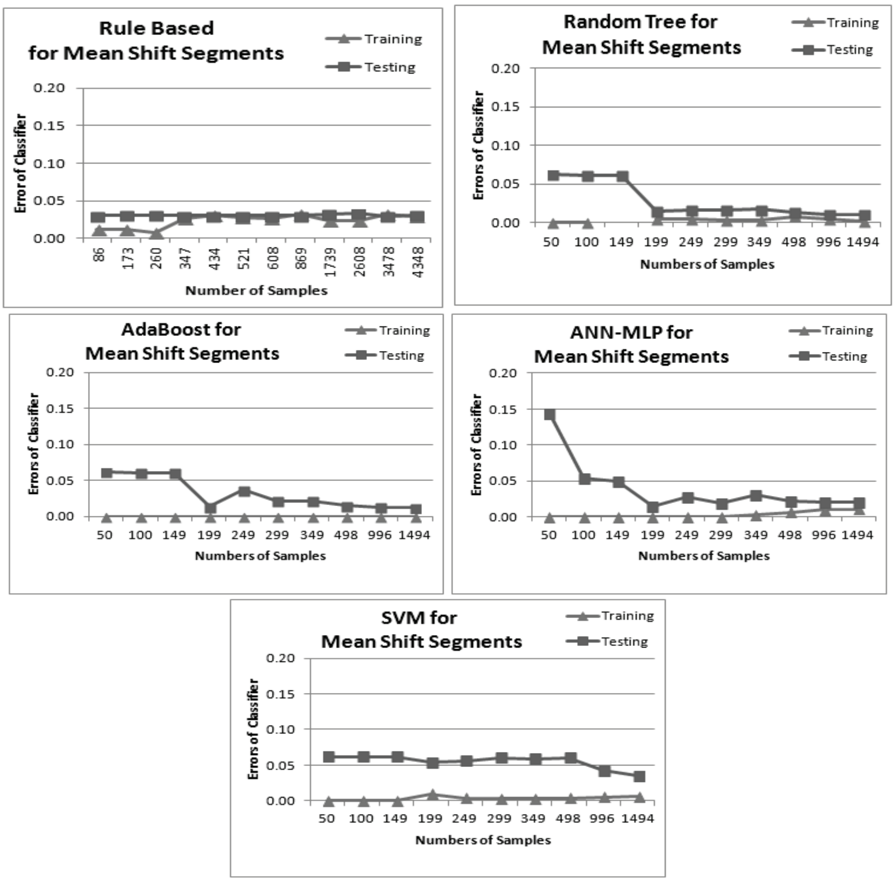


Figure 3.10. Learning curves for mean shift segments with different classifiers

With the defined features, the learning curves from the five classifiers show that the overall accuracy of the training data can reach over 95%. By visualizing the results in the training area, we found that: (1) the defined seven classes are separated; (2) most walls are corrected classified using the contextual reasoning; (3) vegetation, even when covering part of the roof, is well separated from the roof elements using the multiple entity based method. Chapter 4 reports a numerical evaluation of the classification results.

Chapter 4 Evaluation of the scene classification

Using the trained “rule based” classifier and the proposed strategy, we classified the whole Rotterdam data set for the year 2008 and the year 2010. An overall view of the classification results is shown in Figure 4.1. To analyse and evaluate the accuracy of the classification results, we extracted 30 tiles (not from the training area) and did the following experiments:

1. Impact of classifiers: five different supervised classifiers are used for our classification strategy, and the overall accuracy and the accuracy for each individual class are calculated and compared for each classifier.

2. Impact of features: the performance of the newly generated feature "average point spacing" is compared to feature generated from “multiple pulse count” information, and testing demonstrates if it is a good feature for separating vegetation from other objects when the vegetation is not too dense. Conclusions on how well average point spacing performs in comparison with multiple pulse count are drawn.

3. Impact of entity: the performances of multiple-entity classification is compared to the single-entity classification results, these single-entity classifications are only point based and only planar segment based methods.

The introduction to the extracted area and their classification results are in Section 4.1, followed by the three impacts analysis mentioned above. Impact of different classifiers, features and entities are elaborated in Section 4.2, 4.3 and 4.4. Besides the above stated impacts, there are some other discussions on the classification results. For instance, part of the classification errors will not influence our purpose of change detection and are therefore not important errors etc. All these are finally explained in Section 4.5.

(a)

(b)

(c)

Figure 4.1 Classification of part of the data sets in (a) 2008; (b) 2010; (c) 2012.

4.1 Results analysis

To analyse the classification result, we extracted a small area (Figure 4.2) from data set of 2010 which was obtained in March with a point density of 30-40 pts/m^2 on average. Pulse return was recorded up to a maximum of four echoes. This area is located in the centre of Rotterdam, and is almost flat with occasional topographic relief. It contains canals, tall, large, and irregular building groups and residential buildings, buildings under tall vegetation, vegetation, construction sites, vehicles and other objects. The data was already filtered into ground points and above ground points in the original data set. The area is about 2.4km^2, and it has been organized into 960 50m×50m tiles for processing the data.

Figure 4.2. Test area (left), image from Google; (Right) ALS data in test area (a black area means no data; the lighter the shade, the higher the point).

4.1.1 Reference data

For training, we randomly chose 113 tiles as the subset of the data 2010, and manually classified them into the classes of interest. A three-person group was organized to unify the standards for labelling each class of interest, and to specify the rules for ambiguous cases. The labelled data subset was taken as reference data for training and testing. Finally, the whole area was classified with the classifiers generated from the subset.

13,428 planar segments were derived for objects above terrain, 737 planar segments for terrain and 4,981 mean shift segments were derived from the reference data. Point features were calculated with a neighbourhood radius of 0.5m, 1.0m and 1.5m. It is notable that the point feature number increased with the increase of the radius. This is because point features were calculated with points in a neighbourhood. If there are less than two points in the neighbourhood, the feature cannot be computed because three or more points are required to fit a plane. Therefore, by increasing the radius, the number of points which can generate features increases. Features on each entity are listed in the subset in Table 4.1.

Table 4.1. Feature numbers generated from data

<table>
<tr><td></td><td colspan="5">Planar segments</td><td colspan="2">Mean shift segments</td><td colspan="3">Points with a feature</td></tr>
<tr><td rowspan="2">Total</td><td colspan="3">Above terrain</td><td colspan="2">Terrain</td><td colspan="2" rowspan="2">4,981</td><td rowspan="2">0.5m</td><td rowspan="2">1.0m</td><td rowspan="2">1.5m</td></tr>
<tr><td colspan="3">13,428</td><td colspan="2">737</td></tr>
<tr><td rowspan="2">Indivi-dual</td><td>Roof</td><td>Veg.</td><td>Undefined objects</td><td>Ground</td><td>Water</td><td>Veg.</td><td>Roof element</td><td rowspan="2">141,602</td><td rowspan="2">147,622</td><td rowspan="2">147,946</td></tr>
<tr><td>1,659</td><td>8,361</td><td>3,408</td><td>574</td><td>163</td><td>310</td><td>4,671</td></tr>
</table>

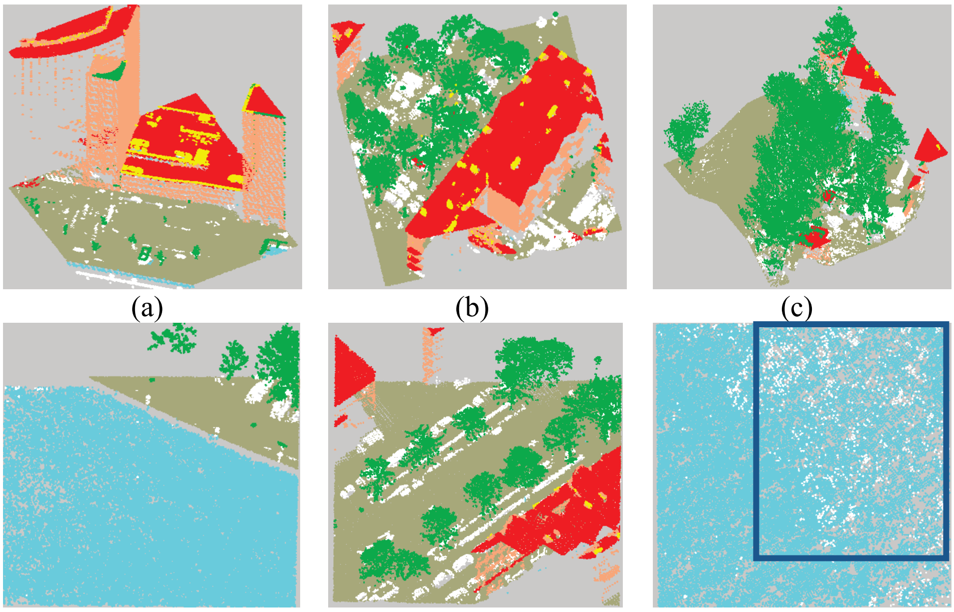
4.1.2 Classification results

We tested the multiple-entity classification strategy with five classifiers on planar segments and mean shift segments, and computed the accuracy on 30 tiles with a total of 3,142,237 points which were randomly selected from the reference data but not from the training data. Special objects such as cranes, boats and bridges were not included in the accuracy calculation, as these objects have not been considered.

Some results from the rule-based method are shown in Figure 4.3. We select several successful instances such as connected buildings with different heights in (a), building walls with different shape of balconies in (b), a building roof covered by vegetation in (c), ground surface and dense water surface in (d) and a building roof cut by a tile boundary in (e). Failures in classification of water, roof, vegetation and wall are shown in Figure 4.3(f-i), and wrong parts are emphasized with blue rectangles.

The incorrect assignment of water to undefined objects is mainly caused by a filtering error in which water points are regarded as objects above ground, as indicated in confusion matrix in Table 4.3. A large area with this incorrect assignment can be redressed by defining a rule that large horizontal planar segments with small height to water are corrected as water points. However, there are still some small incorrect areas as shown in Figure 4.3 (f).

A lack of data on roofs (scan angle or water on roof may cause this problem) results in the wrong classification of "building roof" as "wall" as shown in Figure 4.3 (g). However, this is not a huge error as a building wall is still a part of a building. The case depicted in Figure 4.3 (i) shows some building wall points wrongly labelled as vegetation. These points are far away from the building roof points in 2D distance (larger than 1.0m), thus, there is insufficient contextual information to classify them as wall. Although this kind of error can be eliminated by enlarging the radius for searching roof points, this may lead to new problems such as vehicles parked near a building may be classified as wall. In Figure 4.3 (h), some vegetation is wrongly classified as either building roof, wall or roof element. Some of these vegetation points were misclassified as roofs in the planar-segments classification step, and surrounding vegetation points were classified as walls and roof elements in the contextual classification step. This error propagation was caused by the assumption that all roof points classified during the planar segment classification were correct and error free.



(a) (b) (c)

(d) (e) (f)

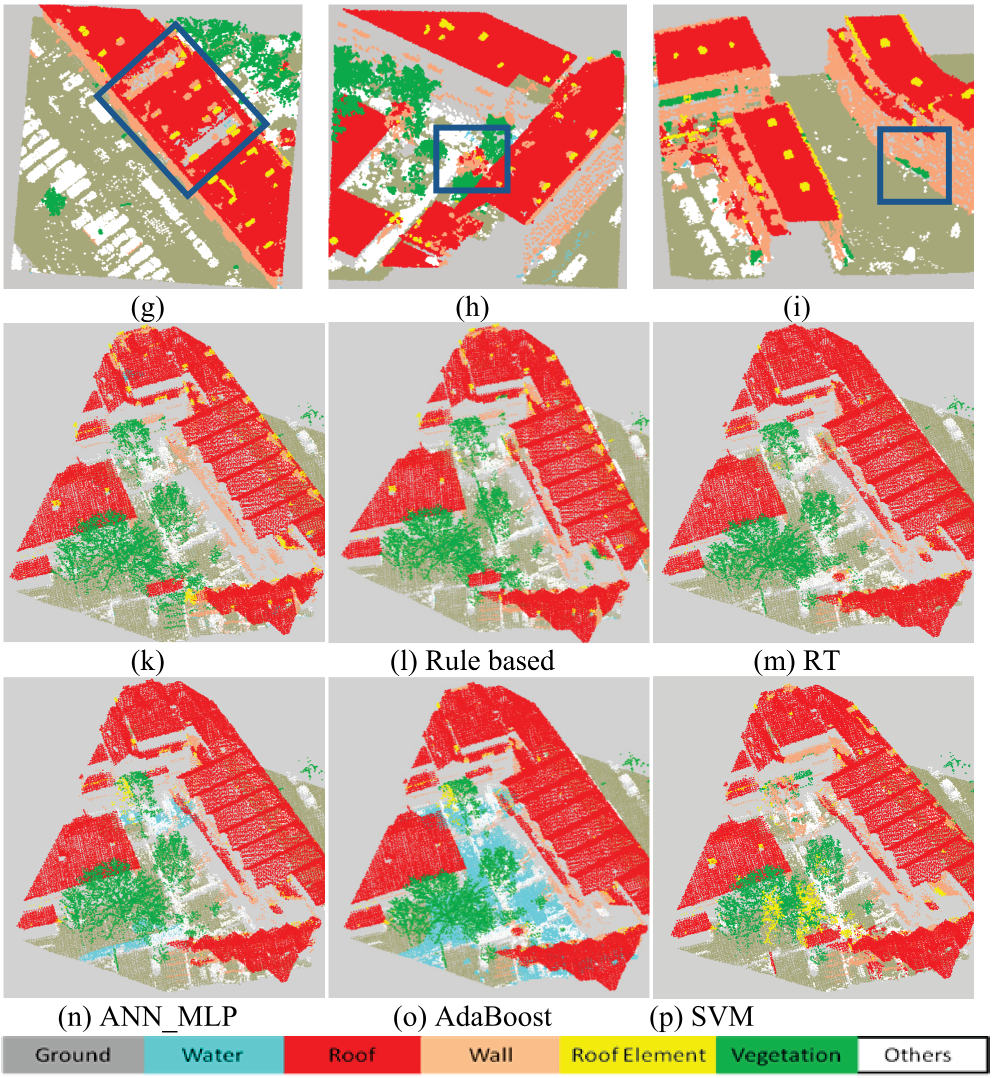


Figure 4.3. Some classification results. (a~e) are correct, (f~i) show some incorrect results within blue rectangles. (k) an example of ground truth data (l-p) are comparisons of results among five classifiers.

Figure 4.3 (k) contains the ground truth data, and (l) to (p) are the results of the rule-based, RT, AdaBoost, ANN_MLP and SVM methods. They are placed in sequence according to the similarity to the ground truth data in (k). For example, the rule-based method (l) has the highest similarity to the reference data (k), and SVM (p) the lowest.

4.2 *Impact of classifier*

We used five classifiers on planar segments in the mean shift segments classification step. The same context rule and features were applied; (Random Tree, AdaBoost, ANN_MLP and SVM were implemented with OpenCV.) 4,028 planar samples and 498 mean shift samples were used to train the classifiers. The accuracy was computed point-wise on 30 tiles, with a total of 3,142,237 points.

4.2.1 Overall accuracy

The overall accuracy, as well as the KAPPA accuracy in Table 4.2 show that the rule-based classification method performs better than the other classifiers. The other four perform nearly equally. Both two measures agree. We also calculated the significances of classifier differences using the kappa statistics and their variances. These differences are significant at a 99% confidence level.

Table 4.2. Overall accuracies and kappa statistics of the five classifiers.

	Random tree	AdaBoost	ANN_MLP	SVM	Rule based
Overall accuracy	95.5%	94.3%	95.0%	94.1%	97.0%
Kappa accuracy	93.4%	91.6%	92.6%	91.3%	95.7%

The correctness and completeness of each class were calculated and are shown in bar charts in Figure 4.4. Excepting wall, roof element and undefined object, the other classes obtained, on average, an accuracy of over 90%. The reason for 30-60% accuracy for the wall, roof element and undefined object classes is analysed in Section 4.2.2. For vegetation, all classifiers show similar completeness, except SVM. SVM also indicates a much lower completeness of “undefined object”. The classifier itself is the main reason for this. We observed many vegetation and roof points which were wrongly classified as undefined object with SVM. This occurred mainly because the features were not weighted, and were treated equally in SVM.

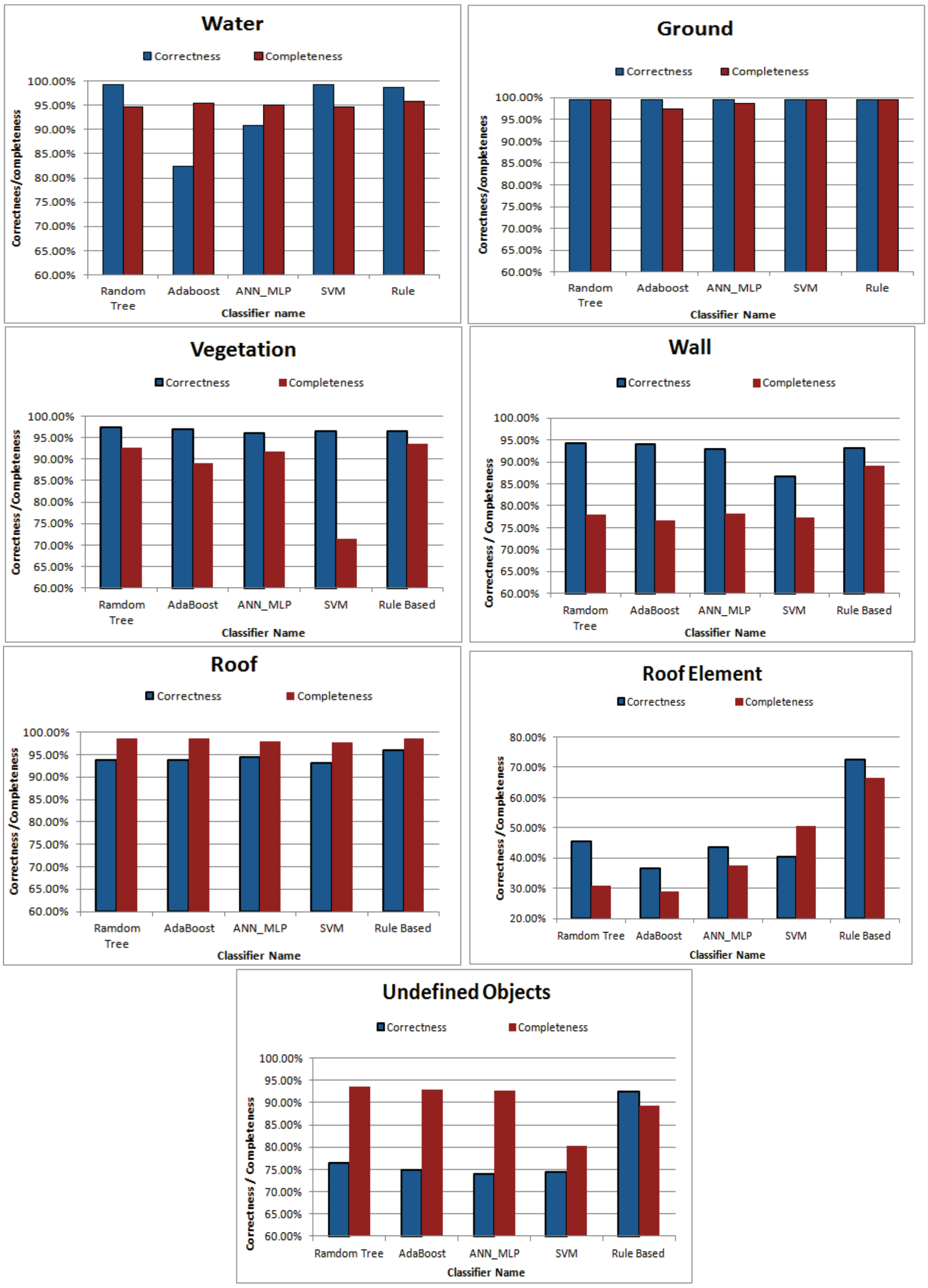


Figure 4.4. Completeness and correctness for each class of interest. In order to make the accuracy difference in each classifier dominant, we changed the bar scale for the roof element because the accuracy was low in comparison with other classes.

4.2.2 Confusion matrix analysis

In order to analyse the accuracy of the roof elements, walls and undefined objects, we generalized the confusion matrix, which is listed in Table 4.3 (confusion matrices from all five classifiers), and categorized the errors into two kinds: filtering errors (light grey in the tables) and classification errors (dark grey in the tables).

1. Analysis on confusion matrix

The confusion matrix indicates that most of the wrongly-classified vegetation and wall points are labelled as undefined object. These errors are caused by our algorithm in the planar-segment classification step. Some points at the foot of the building are wrongly classified as undefined object, and this occurs when the point density is low at the foot of the building and these points are un-segmented and are classified as undefined object. Some of the stairs which extend from the building are quite irregular and are, therefore, classified as undefined object rather than as part of a building.

Table 4.3. Confusion matrices computed from five classifiers. Left column with dark grey figures represents reference data. Top row contains classified points. The number of true positives is listed in the white cells; light grey cells represent filtering errors, and dark grey cells represent classification errors.

Random Tree	Water	Ground	Vegetation	Wall	Roof	Roof Elements	Undefined Objects
Water	146,014	6,663	0	0	1	0	1,557
Ground	1,210	1,388,065	0	12	964	0	3,297
Vegetation	0	107	127,528	3,155	1,557	957	4,228
Wall	1	9	863	220,363	35,905	15,337	10,015
Roof	0	0	81	2,693	1,005,752	1,690	9,974
Roof Element	0	0	48	5,318	27,087	15,119	1,189
Undefined Object	1	261	2,544	2,091	1,650	165	98,712
Adaboost	Water	Ground	Vegetation	Wall	Roof	Roof Elements	Undefined Objects
Water	147,354	5,323	0	0	0	0	1,558
Ground	31,577	1,357,698	1	11	0	0	4,261
Vegetation	0	107	122,303	3,548	1,752	4,962	4,860
Wall	1	9	1,029	216,400	36,677	16,796	11,581

Roof	0	0	90	3,559	1,006,091	2,770	7,680
Roof Element	0	0	43	4,836	26,741	14,155	2,986
Undefined Object	1	261	2,647	2,068	2,330	114	98,003
ANN_MLP	Water	Ground	Vegetation	Wall	Roof	Roof Elements	Undefined Objects
Water	146,687	5,990	0	0	0	0	1,558
Ground	14,848	1,374,427	1	3	964	0	3,305
Vegetation	0	107	126,225	2,829	769	2,778	4,824
Wall	1	9	1,529	220,874	30,835	16,131	13,114
Roof	0	0	173	5,665	999,318	4,499	10,535
Roof Element	0	0	136	5,608	23,670	18,262	1,085
Undefined Object	1	261	3,236	2,501	1,336	346	97,743
SVM	Water	Ground	Vegetation	Wall	Roof	Roof Elements	Undefined Objects
Water	146,014	6,663	0	0	1	0	1,557
Ground	1,210	1,388,065	1	11	2,521	0	1,740
Vegetation	0	107	98,149	13,981	4,384	14,395	6,516
Wall	1	9	852	218,599	36,118	15,722	11,192
Roof	0	0	128	10,077	997,127	5,851	7,007
Roof Element	0	0	198	7,016	15,645	24,708	1,194
Undefined Object	1	261	2,505	2,749	14,829	529	84,550
Rule based	Water	Ground	Vegetation	Wall	Roof	Roof elements	Undefined objects
Water	147,760	4,917	0	0	1,499	0	158
Ground	1,901	1,387,374	0	43	3,402	0	824
Vegetation	1	106	128,745	2,998	1,863	2,003	1,800
Wall	1	9	1,191	251,452	20,640	7,064	2,126
Roof	0	0	266	7,530	1,007,341	2,805	2,248
Roof element	0	0	494	5,386	9,894	32,347	636
Undefined object	38	224	2,608	2,797	5,334	324	94,088

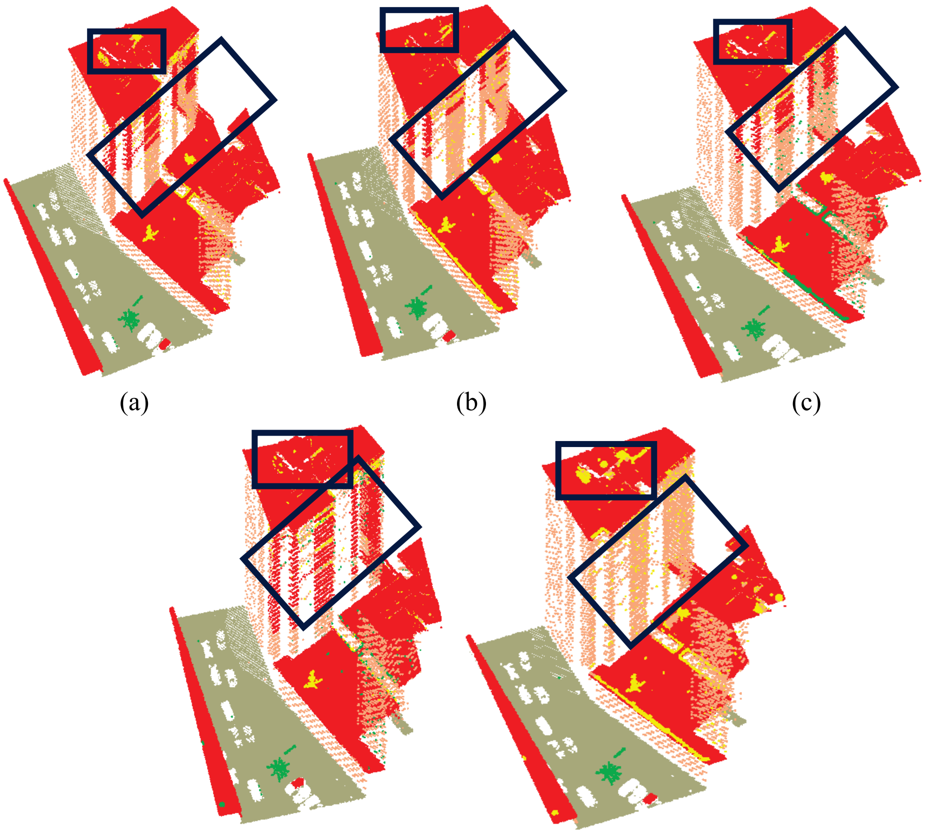
From all the five confusion matrices, we noticed that most undefined objects are classified as roof or wall. These undefined objects are mostly large vehicles such as trucks, which are easily classified as roof, and fences around gardens, which are wrongly classified as wall. Furthermore, roof and wall are frequently mixed up by the classifiers, as some long balconies in the walls are wrongly classified as roof.
The comparison of the five confusion matrices shows that the rule based classifier performs better in distinguishing wall, roof, and roof element than the other four classifiers, but poorer in classifying the undefined objects than the other classifiers except the SVM.

2. Analysis on errors

We also observed that many roof elements and walls were classified as building roofs in the planar-segment classification phase. These errors are not severe errors, as these laser points can still be considered part of a building (as shown in Figure 4.5). If we ignore these errors, the overall accuracy of classification is much higher and the results from every classifier resemble, as shown in Table 4.4

Table 4.4. Accuracy analysis. When ignoring the classification errors for roof elements and walls (white figures in the table), the errors for each classifier, except SVM, resemble each other. If the filtering errors could be eliminated, the accuracy of each classifier would improve by 0.2%.

Classifiers		Random Tree		AdaBoost		ANN_MLP		SVM		Rule based	
Filtering errors		0.20%		0.20%		0.20%		0.20%		0.20%	
Classifica-tion errors	Wrongly classified as roof element	4.28%	1.14%	5.54%	1.17%	4.85%	0.98%	5.69%	1.15%	2.76%	0.66%
	Wrongly-classified wall points		0.86%		0.85%		0.75%		0.5%		0.31%
	Not severe	2.00%		2.02%		1.73%		1.65%		0.97%	
	Severe	2.28%		3.52%		3.12%		4.04%		1.79%	



(a) (b) (c) (d) (e)

Figure 4.5. Most roof elements and wall points (light grey within building) are wrongly classified as roof points (dark) as shown in the rectangles, however, these points still belong to buildings and are, therefore, not considered to be severe errors.(a) Random Tree; (b) AdaBoost; (c) ANN_MLP; (d) SVM; (e) Rule based.

4.3 Impact of Features

Pulse count is considered a useful feature for separating points on vegetation from points on roofs. However, in summer, vegetation can be too dense to generate multiple echoes of laser pulses. If this is the case, "average point spacing" is considered a feature which can distinguish vegetation from roofs for planar segments. To make sure how well the feature "average point spacing" performs to separate the "roof" from the "vegetation", we compared it to the feature generated from pulse count information. The results are in Section 4.3.1

Meanwhile, as the point density for "water" is usually lower than that for "ground", the average point spacing can be used to distinguish the "water" and the "ground" as well, and the performance is discussed in Section 4.3.2.

4.3.1 Average point spacing for vegetation and roof

In order to compare the performances of the feature "average point spacing per segment" and the "percentage of points with the multiple pulse count per segment" on roof and vegetation, the overall accuracy and the accuracy of roof and vegetation are calculated with all five classifiers by using both and separately, and the results are shown in Table 4.5 and Figure 4.6.

Table 4.5. Overall accuracy and accuracy on vegetation comparisons on performance of average point spacing and the percentage of points with multiple pulse count.

	With only Average Point Spacing (%)			With only Pulse Count (%)			With both (%)			Without both (%)		
Classifiers	Overall	Veg	Roof	Overall	Veg	Roof	Overall	Veg	Roof	Overall	Veg	Roof
Random Tree	95.36%	83.71%	92.98%	95.24%	90.51%	92.26%	95.52%	90.39%	92.49%	94.66%	87.66%	90.59%
AdaBoost	94.10%	81.17%	92.67%	94.22%	85.90%	92.40%	94.26%	86.52%	92.50%	93.18%	77.60%	89.76%
ANN_MLP	93.02%	75.87%	90.28%	94.05%	82.26%	91.51%	94.95%	88.50%	92.72%	94.63%	89.65%	91.58%
SVM	93.79%	61.07%	91.16%	94.11%	69.55%	91.18%	94.11%	69.50%	91.17%	94.18%	68.63%	91.42%
Rule based	97.07%	85.47%	95.70%	97.32%	91.36%	95.75%	97.03%	90.06%	94.72%	96.33%	79.95%	94.23%

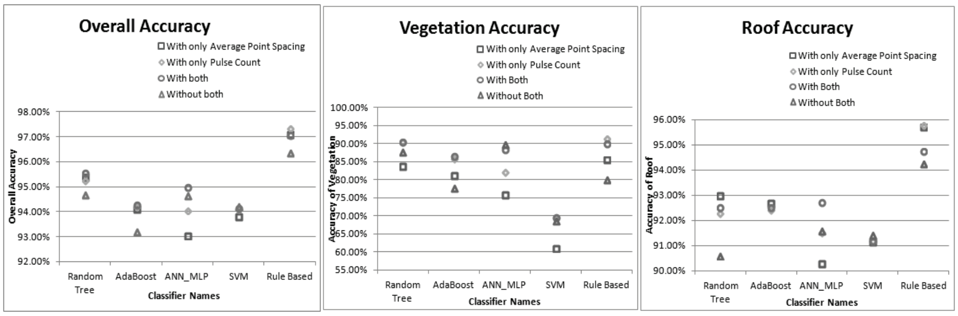


Figure 4.6. Comparison of overall accuracy and the accuracy for vegetation and roof classes

For the overall accuracy, except in the case of ANN_MLP and SVM, the three other methods demonstrated that using both features obtains results similar to using only pulse count information, or only APS, and classification with neither of the features achieved lower accuracy. ANN_MLP and SVM showed that, when only APS was used, the overall accuracy was lower.

For vegetation and roof, using only pulse count information obtained similar results to using both. However, APS did not always improve the accuracy of the vegetation or roof class. With the ANN_MLP classifier, using only APS achieved the lowest accuracy.

4.3.2Average point spacing for water and ground

There are many literatures on classifying water in ALS data, in which a list of features are derived and researched such as height, point density, slope difference along scan line, eigenvalues etc. (Brzank and Heipke, 2006, Smeeckaerta et al., 2013). However, as no information on scan lines is stored in our data sets, we use the feature “average point density” and only this feature to distinguish “water” from “ground”. Except the Random forest which is not suitable for the classification with only one feature and therefore not used, other four classifiers are tested, and the results are in Table 4.6.

Tabel 4.6 Completeness and correctness of classification of “water” and “ground”.

	AdaBoost		ANN_MLP		SVM		Rule Based	
	Comp	Correct	Comp	Correct	Comp	Correct	Comp	Correct
Water	95.79%	99.50%	95.79%	99.50%	95.79%	99.50%	96.09%	99.86%
Ground	99.68%	99.87%	99.68%	99.87%	99.68%	99.87%	99.83%	99.77%

Except that the completeness of the “water” is slightly lower than the “ground”, the results are promising. The low completeness is caused by either wrong filtering or wrong classification. As shown in Figure 4.3 (j), some points in water are classified as “undefined objects” (caused by filtering error) and some are assigned to “ground” (classification error).

4.4 Impact of Entity

The multiple-entity based strategy is designed to compromise between the computation time and the classification accuracy. To see the performance of multiple-entity strategy, we compare it to the single-entity strategy: point only and planar segment only, by using exactly the same features and contextual reasoning for classification. Solely for point-based classification, features were computed using the neighbouring points of the central point. We calculated the classification error with different neighbourhood sizes of 0.5m, 1.0m and 1.5m for each classifier. The classifier errors became stable when we increased the radius from 1.0m to 1.5m (Figure 4.7). Therefore, the radius of the neighbourhood was a fixed 1.0m. No mean shift or point-based features were used for only planar-segment based classification. It is very slow to run mean shift over all the points; therefore, we did not add the accuracy of only mean shift segment based classification.

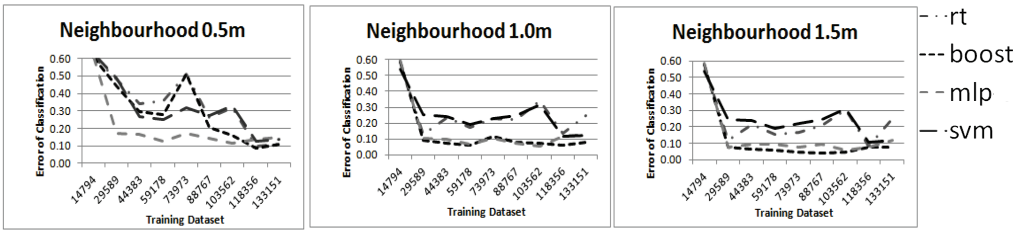


Figure 4.7. Learning curves generated from different features derived with neighbourhood different sizes

4.4.1 Comparison on computation cost

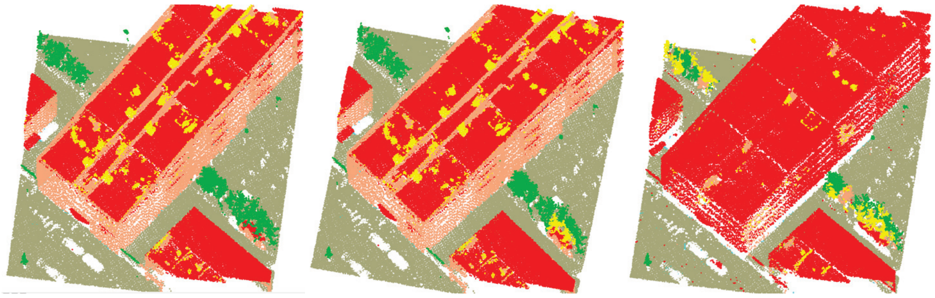
The computation costs for single entity based and multiple entity based method are shown in Table 4.7. Apparently, point based classification method is the slowest. Multiple based classification is several minutes slower than planar based classification, which is caused by the mean shift segmentation procedure, we however expect an improvement in the accuracy and especially an increase in accuracy of the "vegetation" in areas where buildings are covered by tall vegetation.

Table 4.7. Approximate computation time per tile. These computation times are listed here to give an

Multiple entity based	Planar segment based	Point based
5 minutes	3 minutes	60 minutes

4.4.2 Comparison on Accuracy

One example of the classification result is shown in Figure 4.8. The example is from the rule based classifier and all three results are generated from the same strategy with different entities. From a visual point of view, the results in vegetation are improved with multiple-entity based method, comparing to the other two single entity methods.



(a) (b) (c)

Figure 4.8. an example from rule based classifier. (a) Multiple-entity based; (b) Planar-segment based; (c) Point based

We also calculated the results in statistics. As shown in the results in Table 4.8, the overall accuracy of classification with multiple entities is clearly higher than the accuracy of the point-based classification, as expected but nearly the same as the accuracy of the classification with only planar segments. The accuracy of vegetation, in particular, is improved when using multiple entities instead of single entities (except for ANN_MLP and SVM).

Table 4.8. Comparisons of overall accuracy and accuracy on vegetation with different entity-based experiments

	Multiple entity		Planar segment		Point	
	Overall	Veg.	Overall	Veg.	Overall	Veg.
Random Tree	95.5%	90.4%	95.4%	88.1%	84.8%	52.0%
AdaBoost	94.3%	86.5%	94.3%	86.5%	89.1%	54.2%
ANN_MLP	95.0%	88.5%	95.0%	88.8%	88.5%	57.7%
SVM	94.1%	69.5%	94.1%	69.5%	87.0%	65.0%
Rule based	97.0%	90.1%	96.9%	86.7%	87.8%	60.3%

The reason why the overall accuracy was not greatly improved for the multiple-entity and planar segment-based methods is mainly that the number of vegetation points is relatively small compared to the number of building points. Hence, the vegetation points contributed little to the overall accuracy.

4.5 Discussion

A multiple-entity based classification strategy to classify seven classes in urban areas with lidar data was proposed. Promising results have been achieved with different classifiers using the multiple-entity based classification strategy. The rule-based classifier achieves an overall classification accuracy of 97.0%. The completeness and correctness of each class of interest is over 90%. A correctness of 85%, on average, for all classifiers for undefined objects indicates incorrect classification of "undefined objects" in the other classes. This is mainly because that undefined object belongs to a collection of objects with different shapes, sizes, appearances, etc. The confusion matrix (Table 4.3) indicates that most of the wrongly-classified undefined objects are classified as roof, wall or vegetation. The misclassification of undefined object as wall was mainly caused by the lack of a class definition for poles and wires in urban areas. The difficulty of separating big vehicles from roofs resulted in the misclassification of undefined objects and roofs. The roof element has a relatively low accuracy. Most roof element points are classified as roof or wall, as indicated in the confusion matrix. However, this is not a vital error because roofs and walls are still parts of a building.

The multiple-entity method can improve the classification accuracy for vegetation by 3.3% with a rule-based classifier, in comparison to the planar-segment based method. Compared to the point-based method, the multiple-entity method can achieve a 9.21% improvement in overall accuracy and a nearly 30% increase for vegetation. It is notable that the overall accuracies for the multiple-entity method and for the planar-segment method are only slightly different. Vegetation points account for only a small proportion of all the points, and mean shift segments are employed to correct the wrongly-classified vegetation. Therefore, the overall accuracy is nearly the same.

Contextual information for the building is identified as working well in classifying the walls and roof elements. Although the accuracies are lower on these two classes, compared to the other classes, most of the wrongly-classified roof element and wall points still belong to buildings. In addition, we can visualize some roof, wall and roof element points among vegetation in the results (Figure 4.3 (h)). These errors are caused by vegetation being wrongly-classified as roof during the planar segment classification phase. The surrounding points are, therefore, wrongly classified as walls and roof element during the context-rule classification. Among the five classifiers, the rule-based classification method works better than the other classifiers and SVM obtains the lowest accuracy. The rule-based classifier is a supervised classifier. Experienced experts can make complex rules and organize a decision tree with rule-based classification. In this process, the tree needs to be adjusted by adding or deleting rules in the tree and the results need to be repeatedly observed. Therefore, it can obtain higher accuracy than the other four methods. For inexperienced operators, the other four methods are recommended; these can also obtain high accurate results.

Average point spacing can be employed as a new feature to classify vegetation and building roofs when multiple echoes are not dominant in a dense vegetation area. However, the combination of APS and MPC cannot greatly improve the accuracy of vegetation. Compared to MPC, the combination performs worse in detecting vegetation points. However, except with SVM, an average accuracy of 90% for roof and 80% for vegetation when using APS is still acceptable, especially when pulse count information is unavailable.

The multiple-entity based classification strategy performs better on large point clouds in complex urban scenes than point-based method. Compared with planar-based method, multiple-entity based classification strategy nearly has the same overall accuracy, but a slight accuracy improvement in vegetation can be observed. Data with pulse count information are classified with higher accuracy than data without pulse count information. The rule-based method slightly outperformed other supervised methods. With the same classifier, higher accuracy can be obtained by using a multiple-entity based strategy than using a single entity. These high-accuracy classified data sets will be used for the change detection.

Chapter 5 Methodology for the change detection

5.1 *Introduction*

In urban areas, changes to buildings may be caused by natural disasters or geological deformation, but more often they are the result of human activities. These activities may lead to temporary or permanent changes, as, for example, discussed by Xiao et al. (2013). Detection of structural changes to urban objects, e.g. renovation of infrastructural objects and buildings, is important for municipalities, which need to keep their topographic object databases up-to-date.

Analysis of the geometric differences between point clouds from different epochs of data provides insight into how scenes change over time. The general approach in using point clouds for change detection is to focus on areas where points are present in one epoch of data and absent – at least in the close vicinity – in the other. As the need for detecting changes at higher levels of detail is growing, demand for more detailed interpretation of differences between epochs of data has also grown. There can be several reasons for absence of data of points for a certain location in one epoch of data while data points for that same location are present in another epoch:

1. the area around the location may have been occluded in one of the data epochs;
2. the surface around the location may have absorbed the ALS laser pulses;
3. outliers (either on the ground or inside buildings) may be present; or
4. the object may have undergone change.

In the first three cases, differences in the data do not correspond with real changes. In the fourth (last) case, some of the changes may not be relevant for a municipality's databases, e.g. a parking lot changes due to the temporary presence or absence of cars, which results in different arrangements of point clouds. The main problem is that it is not known whether a difference between two data sets is because of differences in scanning geometry, surface properties or changes – relevant or irrelevant – to an object. A better understanding of differences between epochs of data would allow us to precisely determine whether there has been a change in an area and, if so, what kind of change it is.

Our study aims to develop a method for detecting relevant changes occurring in urban objects using point clouds from ALS data. Our focus is on changes to buildings, including changes to roof elements and those associated with car parking lots on top of buildings. The main challenge is how to separate irrelevant differences from relevant object changes.

With the above aims in mind, we set up the following change detection procedure: first, in each point cloud, objects are classified as “ground”, “water”, “vegetation”, “building roof”, “roof element”, “building wall” or “undefined object” as described in Xu et al. (2013). This classification step is referred to as “scene classification” in the remainder of this thesis. Next, a surface difference map is generated by calculating the point-to-plane distances between the points in one epoch of data to their nearest planes in the other epoch. By combining scene classification information with the surface difference map, changes to building objects can be detected. Points on these changed objects are grouped and further analysed in a rule-based context to classify them as changes related to a roof, a wall, a dormer, cars on flat roofs, a construction on top of a roof, or undefined objects. Reference data was collected manually to determine the accuracy of this approach.

To begin, we describe the state-of-the-art of the change detection techniques in Section 2. The explanation of the method in Section 3 looks at our use of surface difference maps (Section 3.1), our change detection algorithm (3.2), the classification of changes (3.3), and our analysis of object-based changes (3.4). The data sets used are described in Section 4, followed by presentation of results and their analysis in Section 5. We offer our conclusions in Section 6.

5.2 Related research

Research related to change detection is discussed in two parts. First, an overview is given of previous research done on change detection from imagery. In the second part, we discuss techniques using lidar point clouds for detecting changes in buildings .

5.2.1 Change detection from imagery

Originally, change detection was usually done by visual interpretation. Visual interpretation, in which all changes are found manually by an analyst, was the most common method. It is, however, time consuming, in particular when frequent updates of changes are needed. Automated approaches for change detection began with image differencing and image regression. Later, transformation methods such as principle component analysis (Byrne et al., 1980) and RANSAC (Sharma et al., 2006) were introduced to detect change from images. RANSAC can be used to identify a linear or higher order relationship between two epochs of data that indicates an unchanged area. When a change does occur this will result in a mismatch with the relationship. Given a probability equation and an assumption on the radiometric variation, changes are then found. For example, Sharma et al. (2006) introduced the Lambertian assumption and the Gaussian function to determine change using RANSAC.

There are two different approaches to detecting change from imageries: (1) one based on differences being distinguished between two classified images. Both images are classified independently and differences in pixel labels are assumed to be caused by changes in, for example, land cover (Di et al., 2009). Obviously, the quality of the change detection depends on the quality of the classification. (2) the other approach is based on pixels being directly classified as changed or unchanged according to differences distinguished between features of the pixels and/or those of their neighbourhoods (Yang and Zhang, 2006). Classification methods can be supervised and unsupervised. Di et al. (2009) and Yang and Zhang (2006) both used the Supported Vector Machine(SVM) as a supervised classifier. Unsupervised methods have been described by Tanathong et al. (2009), who defined a classifier agent and an object agent, and Kasetkasem and Varshney (2002), who introduced Markov Random Field Models for change detection.

5.2.2 Change detection in lidar point clouds

Similar to image differencing, the first approach used for detecting change in multi-temporal lidar point clouds involved the subtraction of two DSMs (Murakami et al., 1999) from each other. Classification is also used to directly distinguish changed objects from unchanged ones; this is usually applied in disaster assessment. For an example of this method of classification, see Khoshelham et al. (2013), who, using an ALS data set, performed several supervised classifiers on a small number of training samples to distinguish damaged roofs from undamaged roofs.

In cases of 3D change detection in buildings, depending on the nature of the application and the availability of data sets, comparisons can be made between data sets of multi-temporal images, between multi-epoch lidar point clouds, and between image and lidar point clouds. With improvements in the accuracy of obtained images, as well as improvements in processing skills, 3D point clouds can also be derived from images by, for example, dense matching algorithms (Gerke, 2009).

Whatever the data source, there are two basic approaches to the problem of change detection, each determined by the availability of original data: (1) original data is available from both epochs to be used for detecting change in cases of disasters or geological deformation; (2) original data is only available for one epoch, while the other epoch of data is an existing map or database (Vosselman et al., 2004a). The literature of these two approaches is reviewed:

Approach 1
In 1999, Murakami et al. (1999) used two ALS data sets, one acquired in 1998 and the other in 1996, to detect changes to buildings after an earthquake in a dense urban area in Japan. A difference map was obtained by subtracting one DSM from the other. This difference map was laid over an ortho-image and an existing GIS database to identify changes to buildings. Vögtle and Steinle (2004) presented their research as a part of a project using DSMs from two ALS data sets to detect changes after strong earthquakes. Segmentation was run to find all the buildings. The changes were identified by the overlay rate of all the buildings in both epochs of data. Rutzinger et al. (2010) extracted buildings from DSMs of two data epochs using a classification tree. Shape indices and mean height difference of the building segments were compared. Differences between classified DSMs derived from two epochs of ALS data have also been used by Choi et al. (2009) to detect changes.

Approach 2
Vosselman et al. (2004a) compared ALS data to an existing medium-scale map. Change detection was done by segmentation, classification and the implementation of mapping rules. Pixel overlay rates on a classified DSM and a raster map were used to finally identify changes. This method was then improved by using aerial images to refine the classification results (Matikainen et al., 2004). Rottensteiner (2007)

employed data fusion with the Dempster-Shafer theory for building detection. He then improved his method by adding one more feature to the data fusion, which makes the classification suitable for building-change detection. Champion et al. (2009) detected building changes by comparing a DSM with a vector map. The similarity measure between building outlines in the vector map and the DSM contours were used to identify the demolished, modified and unchanged buildings. New buildings were detected separately in the DSM. Chen, et al. (2010) used lidar data and an aerial image to update old building models. After registration of the data, the area of change was detected and height differences were calculated between the roof planes in the lidar data and planes in the old building models. A double-thresholding strategy was used to identify main-structure changed and unchanged areas, while uncertain parts were identified from the line comparisons between building boundaries extracted from an aerial image and the projection of the old building models. The double-thresholding method was reported to improve overall accuracy from 93.1% to 95.9%.

Both approaches described above enable changes to be detected in 2D (maps) or 2.5D (DSMs). In 2011, Hebel et al. (2013) introduced an occupancy grid to track changes explicitly in multi-temporal 3D ALS data. In order to analyse all the laser beams passing through the same grid, they defined belief masses {empty, occupied, unknown} for each voxel in the object space. For the data sets they compared, they computed belief masses resulting from all the laser beams, with conflicts in belief masses denoting a change. By adding an extra attribute to indicate the smoothness and continuity of a surface, they were able to achieve reliable change detection results even when occlusion had occurred in either of the data epochs. Later on, Xiao et al. (2013) introduced an occupancy grid to the mobile laser scanning system to detect permanent objects in street scenes, where occlusions frequently occur. They too reported that occupancy grids are resistant to occlusion. However, when point density is too low, for example on walls, and the size of the occupancy grid is small, parts of walls were identified as occluded since no occupied points were included in the grid because of the low point density. The size of the occupancy grid may, therefore, influence the detection result when the point density of the lidar data varies from place to place.

5.3 Methodology

To interpret changes in a scene, we start by calculating the differences between two data sets. A surface difference map is generated by calculating the point-to-plane distance between the points from one data epoch to their nearest planes in the other (see Section 3.1 for more details). As the points on building roofs, walls and roof elements are considered to be extracted in the scene classification step, it is possible to combine the scene classification result with surface difference information. This combination enables us to not only detect changes on building objects but also to detect occluded areas, where it is not known whether there has been a change or not (see Section 3.2). Points on the changed objects are grouped and analysed according to a rule-based context; besides changes to roofs and walls (see Section 3.3), changes to roof elements are further classified as cars, construction, dormer or undefined objects,. The classified changes are finally grouped into building objects and analysed (see Section 3.4). Reference data were collected manually at random locations to determine the accuracy of our method.

5.3.1 Generating the 3D surface difference map

The 3D surface difference map records the disparities of points between two epochs of ALS data. The disparity per point is computed as the distance from a point to its nearest fitted plane from another epoch. Surface difference was employed by Vosselman (2012), who evaluated the quality of data using overlapping strips. In our methodoloy the surface difference map is used to indicate 3D differences between two lidar data sets.

For every point in one epoch of data, we search within its 1.0 m (radius) 3D neighbourhood to check whether there is a point from the other epoch. No point from the other epoch implies a difference greater than 1.0 m and we record the difference value as being greater than 1.0 m. If there are points from the other epoch of data, we define the surface difference as the distance from the selected point to the nearest fitted plane in the point cloud of the other epoch.

A 3D neighbourhood of 1.0 m radius is chosen because most data sets have a point density that is greater than 1 point/m2, which ensures that the comparison is not affected by the point density. Although we have two data sets that have similar point densities, in order to make sure that a 1.0 m neighbourhood is suitable for comparisons between different point-density data sets, we simulated several data sets with our test data by reducing the point density of one of the epochs (see Figure 5.1).

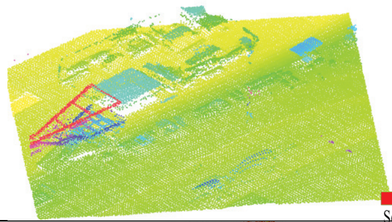
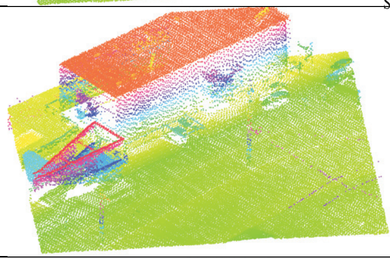
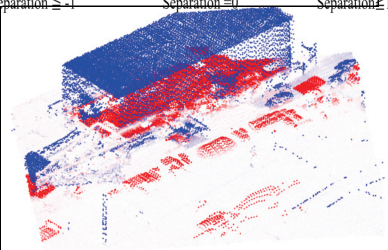
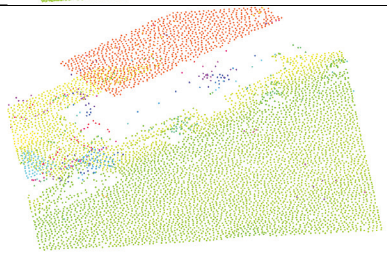
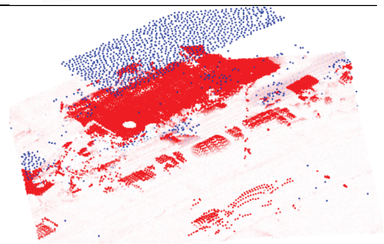
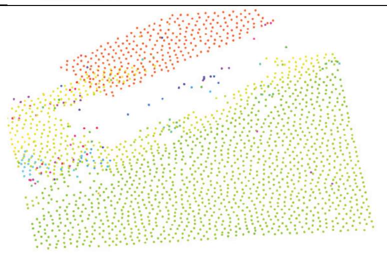
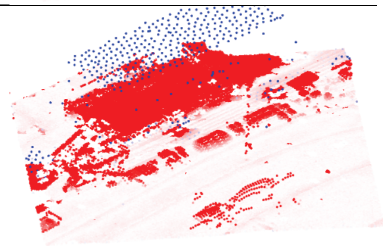
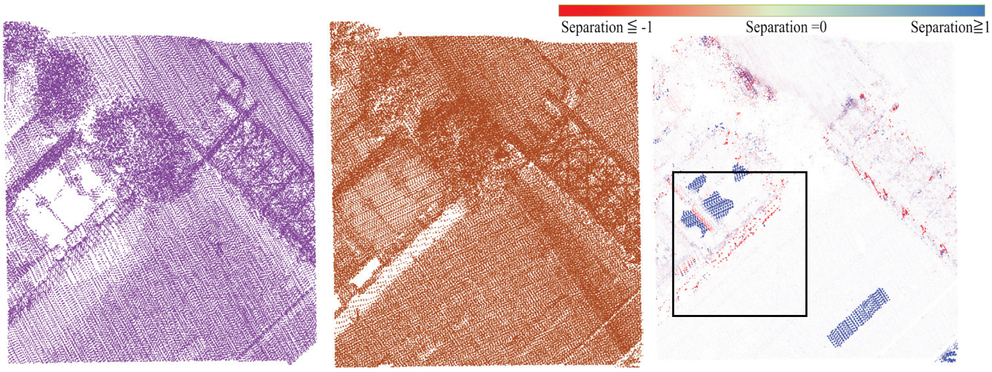
	Compared data sets	Surface difference map (All compared to epoch 2008)
Epoch 2008 *With point density of 40 points/m^2*		
Epoch 2010 *With point density of 40 points/m^2*		
Epoch 2010 *With point density of 4 points/m^2*		
Epoch 2010 *With point density of 1 point/m^2*		

Figure 5.1. Examples of the surface difference map for various point densities. Comparison of data sets with different densities from year 2010 with the data year 2008 showed that the locations of the changes are the same in all the difference maps. This indicates that the surface difference map generated with a neighbourhood of 1.0 m is suitable for all the data sets that have a point density larger than 1 point/m2.

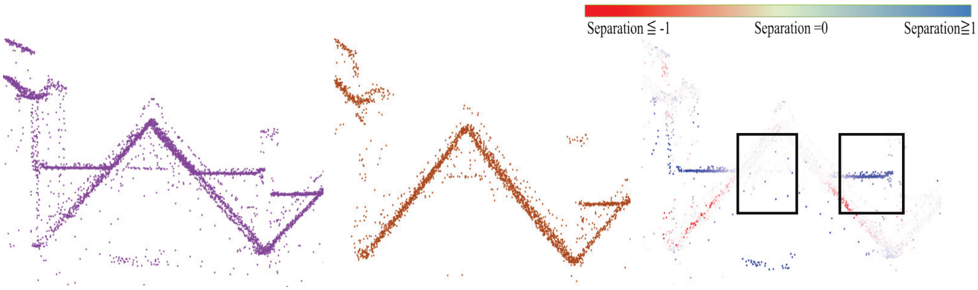
The difference map contains the geometric indication of whether there is a change or not. However, not every difference is a change (Figure 5.2), and sometimes part of a change is not represented as a large value in the difference map (Figure 5.3). Figure 5.2(c) shows the difference map containing several points with a surface difference greater than 10 cm. However, these differences are caused by lack of data in the other epoch and cannot therefore be considered as a change in an object. Conversely, points with a difference value less than 10 cm may belong to a changed object. Figure 5.3 is an example that displays points with a surface difference of less than 10 cm in the connections between a dormer and a roof; the change was detected because the dormer was newly built.

Our task is to assign the labels of "changed", "unchanged" and "unknown" to each point according to the surface difference map.



(a) 2008 (b) 2010 (c) Difference map (merged)

Figure 5.2. (a) and (b): Different colours represent different epochs of data. (c): Surface difference map derived from the two epochs. Points with a great difference do not always denote a change. For example, in (c) blue points in the rectangle are unknown points because there is no data for the roof in 2008 epoch data (perhaps caused by water on the roof), although they have high difference values.



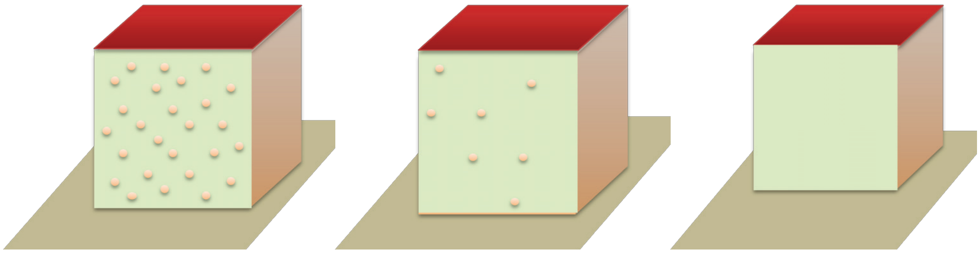
(a) 2012 (b) 2010 (c) Difference map (merged)

Figure 5.3. (a) and (b): Different colours represent different epochs of data. (c): Surface difference map. The points in the rectangle have a distance difference ≤ 0.08 m, although there is indeed a change. The difference value is small because the points on the roof from the other epoch are in the neighbourhood of the points on the dormer and they are quite close to each other. The problem is to identify the areas in the rectangle as being changed when the difference value is low.

5.3.2 Detecting a change

The strategy for interpretation of the surface difference map is as follows: points classified as building (wall, roof or roof element) in the scene classification are selected. All building points are assumed to be unchanged except for those distinguished as “unknown” or “changed”.

Points with a difference value greater than 1 m in one epoch for which, even in 2D, no nearby points can be found in the other epoch are considered “unknown”(see Figure 5.4) . This occurs in areas where there is a lack of data in one epoch because of occlusions or pulse absorption by the surface. For walls, large surface difference values may occur. However, our algorithm will label the points for the wall as “unknown” instead of “changed” if there has been no change to the roof of that same building. Due to a lack of evidence as to what happened with the wall, the points are labelled as “unknown”.



(a) (b) (c)

Figure 5.4. Points labelled as “unknown” for walls. (a) a wall scanned with dense points; (b) the same wall scanned with sparse or (c) no points in another epoch.

Generally, “changed” points have a high difference value, although they cannot be selected using a simple threshold in the difference map. Therefore, we derived a rule-based decision-tree to detect “changed” points in surface difference maps. The decision-tree is shown in Figure 5.5. The “unknown” points are first excluded from the data sets. The remaining points are grouped into planar segments using the surface growing method (Vosselman et al., 2004b). Within each segment, points are further separated into connected components with a smaller radius than during the planar surface growing step. This separates two nearby objects that are located in same plane. For example, Figure 5.6(a) shows two dormers belonging to the same planar segment, and after deriving connected components, they are separated into two components. These components are used as the basic units for identifying changed points.

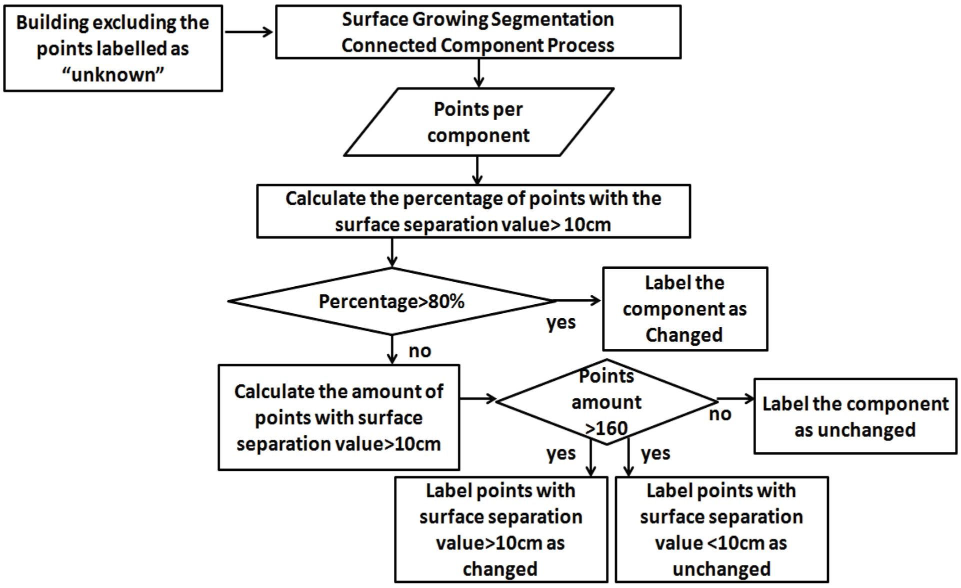
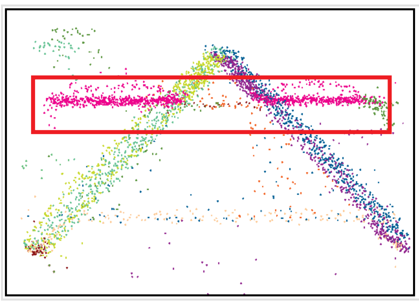
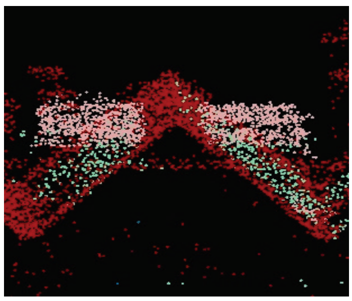


Figure 5.5. Identification of changed and unchanged points using a decision-tree

If for each connected component the vast majority of points have surface differences greater than 10 cm, the whole component is very likely to be a change and will be labelled as "changed". Otherwise, the component may be unchanged or only partly changed (if the number of the points with a surface separation value larger than 10cm is larger than 160, these points are labelled as changed. The 160 is set, because we assume that the point density is 40pts/m^2 on average and an area of 4 m^2 is roughly 160 points).After testing on several tiles, the definition of 'vast majority' was set at 80%.



(a)

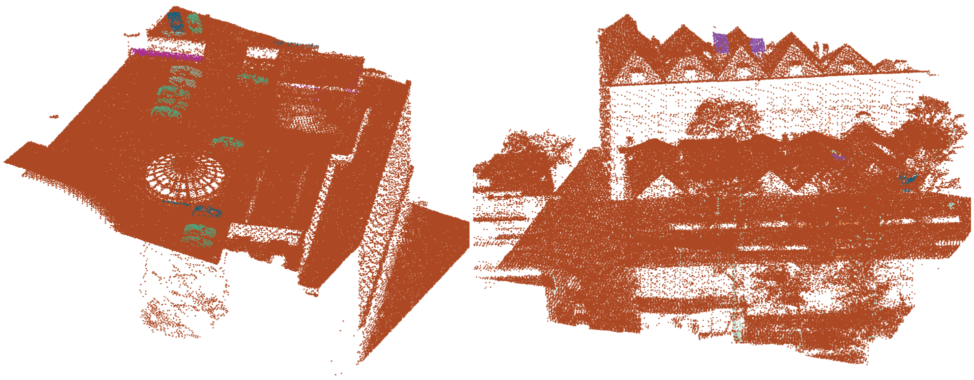


(b)

Figure 5.6. (a): The two dormers in the red rectangle belong to the same planar segment and therefore need connected components analysis to separate them (different colours indicate different segments); (b): Both changes to roofs and dormers are detected (two data epochs are merged).

5.3.3 Classification of building change

Points detected as changes in a building are extracted and a second classification step is performed on these changes. This second classification step is required to understand the activities that could have possibly led to the changes detected for a building. For example, by identifying changes to roof elements on top of a building (Figure 5.7(a)), we can infer that these elements may be cars, and that there is a parking lot on top of the building.



(a) 2008 vs. 2010 (b) 2010 vs. 2012

Figure 5.7. Connected components of changed objects are classified based on their context.

There are various types of objects that may be present on building roofs or attached to walls. Most changes to walls are caused by sun shades, stairs, flags, or vegetation near walls. Compared to the changes to roofs (e.g. a new dormer or the addition of a floor), changes near walls are less likely to be related to construction activities. For this reason, the second classification step is performed only for changes concerning roofs and roof elements.

Based on the size of changes and the underlying building structure, changes on a roof top are classified as dormers, cars or “other constructions”. The difference between a dormer and “other constructions” above the roof line is that the latter involves relatively large changes compared to a dormer. The attributes used for the classification are area, height to the nearest roof, the normal vector direction of the nearest roof (which indicates a pitched or a flat roof) and the labels from the scene classification results.

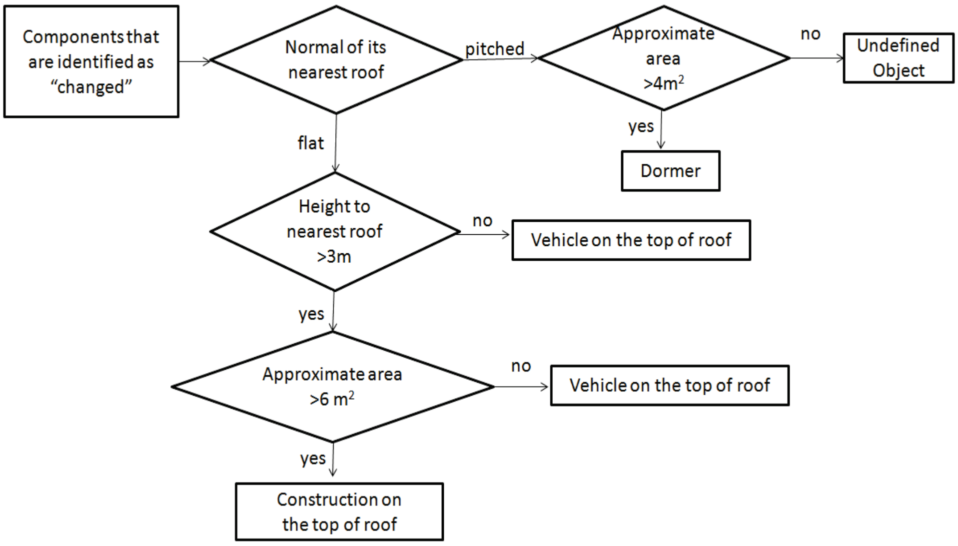


Figure 5.8. Rule-based classification of changes.

Rules are defined to classify these components. Large areas of points labelled as changed indicate a changed to a roof. Small changes occurring on a pitched roof are more likely to be newly built or removed dormers or chimneys. If changes occur on a flat roof, they are probably constructions above the roof line or cars. These rules are used in a decision-tree, and have been established as a rule-based classifier (see Figure 5.8). All thresholds used, such as the area of change being greater than 4 m2 or a height greater than 3 m, are chosen based on the authors' knowledge.

5.3.4 Analysis of changed building objects

After the points are identified as "changed" and classified, the results are points organized as connected components. These components are not building objects yet, and they may represent only a small part of an object. How points are organized as building objects and how the changed objects are analysed is described in the following subsections.

5.3.4.1 Changed building objects

The points that are identified as changed can be organised into building objects once we have labelled all the points with: (1) the type of object class from the change classification; (2) the labels "changed", "unchanged", or "unknown"; and (3) the signs from a difference map indicating a newly built or demolished structure . We distinguish buildings as being entirely changed or partly changed.

If there is a change to the roof of a building, all points that group together in a 2D connected component step will be labelled as an entirely changed building. If changes are made to building elements, the local 3D connected points will be only labelled as partly changed.

5.3.4.2 Merging objects at tile boundaries

So far, changes are detected within tiles of, for example, 50 m x 50 m. Typically, buildings are stored in different tiles, as shown in Figures 5.9 (a), (b) and (c). As changed objects may occur on tile boundaries, a merging step is needed to detect a single object instead of two (i.e. one in each neighbouring tile). To solve this problem, adjacent components are merged across tile boundaries as described by Vosselman (2013). Figure 5.9(d) shows the components after they have been merged.

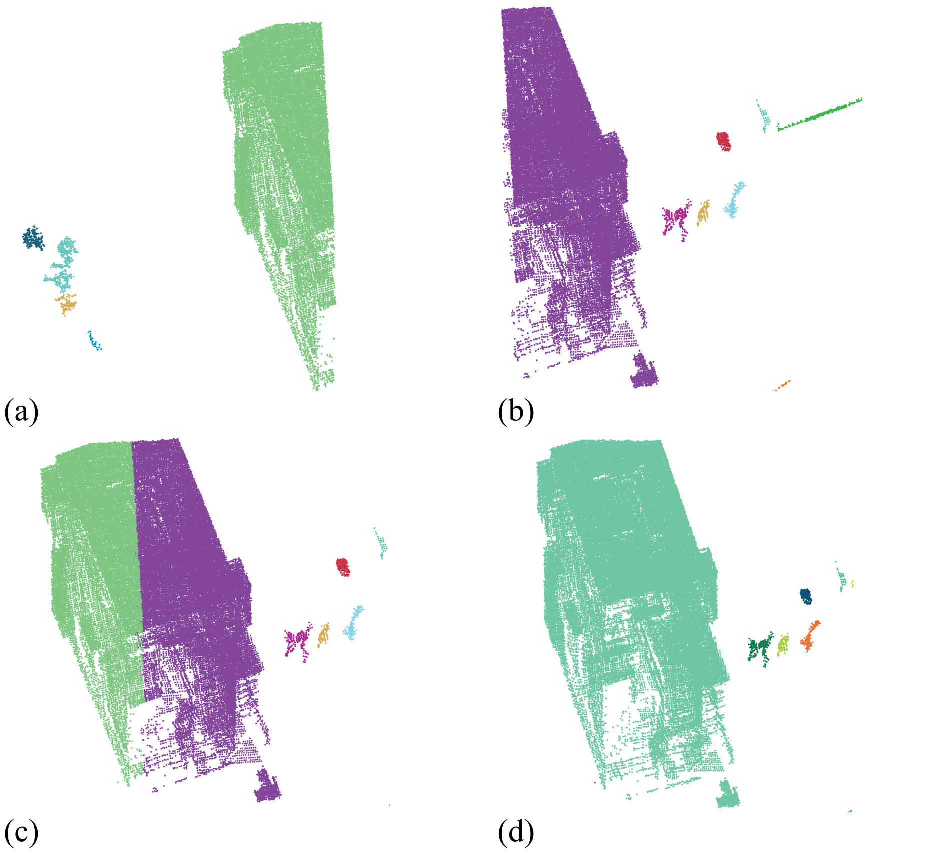


Figure 5.9. Objects are cut and stored in different tiles during the processing of large data sets. Components have been merged in (d).

When building objects have been formed, minimum 3D bounding boxes are calculated for these objects. Their width, length, height, area, volume and location of their centre point are determined from their 3D bounding boxes.

Among all the changes near buildings we found, there are some that are irrelevant for the municipality. Common examples are changes to extensive areas of flowers and shrubs, to fences and railings in gardens on rooftops of buildings. Bounding boxes have been used to measure more precisely the size of a change. This measure of size is used to separate irrelevant changes (< 4 m2) from relevant ones. For changes < 4 m2 and a length: width ratio greater than 10, no 3D bounding boxes are calculated.

Chapter 6 Evaluation for the change detection

Our change detection procedure comprises change detection, classification of any change detected, and object-based analysis. This procedure includes generation of surface difference maps and change identification. The results for each step in the procedure are described below. As the compared data sets are merged into one with different epoch numbers, all the results will be shown in a merged version.

6.1 Detecting Change

Results of the change detection step are shown and the causes of errors in detecting change to buildings are discussed here.

6.1.1 Surface difference mapping

A surface difference map is generated from original data sets, regardless of the scene classification results. Some difference maps have already been shown in Figure 5.1 (Section 5.3.1) and, as discussed, the point density does not affect the surface difference map. More difference maps can be seen in Figure 6.1. The surface difference map gives only clues as to where changes may have occurred in the data sets. To improve their display, all the data sets have been merged into one. Different colours show different values of the differences between the compared data sets.

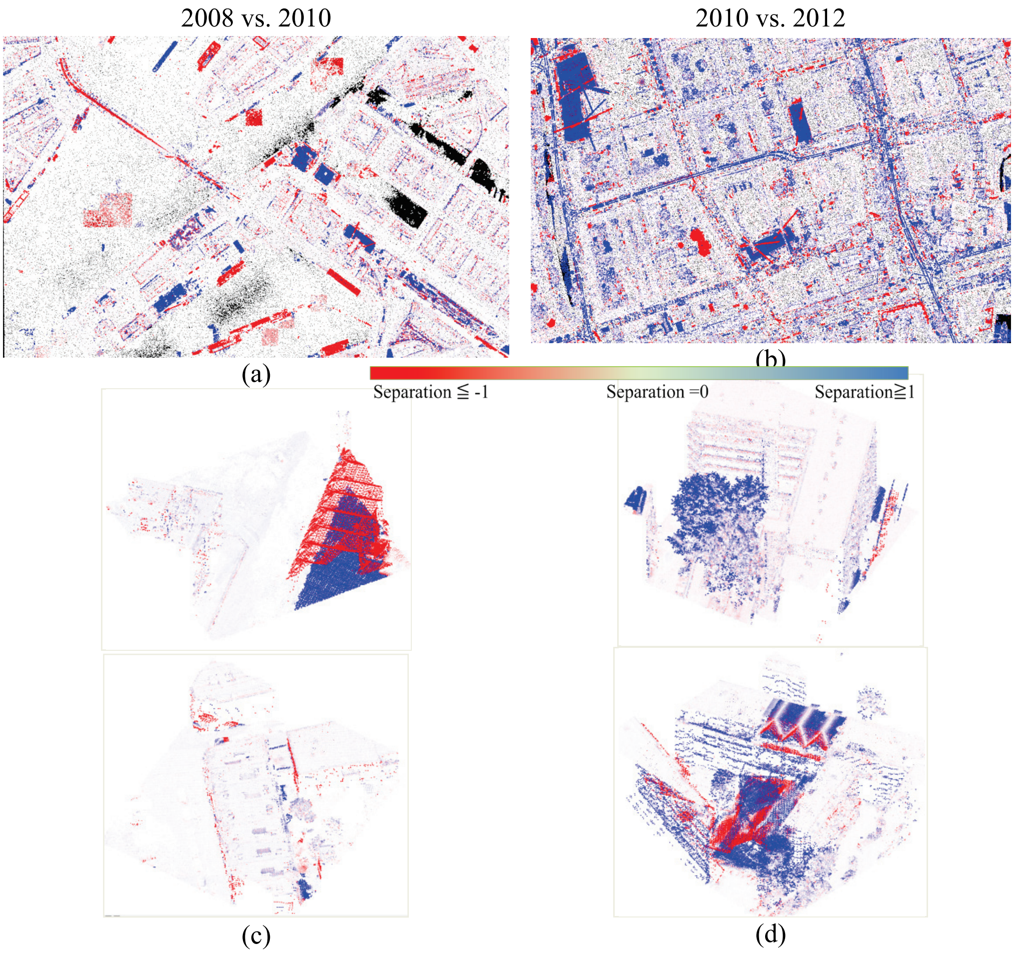
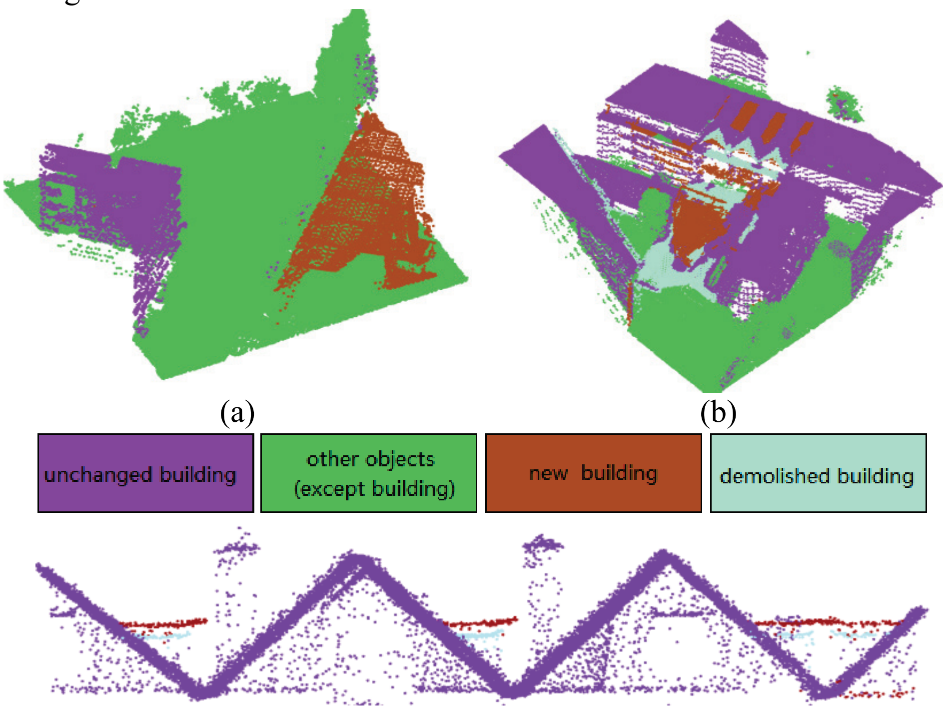


Figure 6.1. Surface difference maps for (a) 2008 vs. 2010 and (b) 2010 vs. 2012.

In Figure 6.1, significant differences can be directly observed on the surface difference maps where the colours become deeper. Red points appearing in the middle of water in 6.1(a) indicate large differences in water surfaces. This large difference is caused by lack of points on the water in one of the epochs, while points were recorded in the other epoch. These difference maps are the inputs for the change identification step.

6.1.2 Change detection results

Using the surface difference map, we labelled the data sets as "changed", "unchanged" and "unknown"}. Figure 6.2 shows some results of both types of change (an entire building change (a), as well as changes to building elements, e.g. changes (b) & (c)). The interpretations "demolished" and "new" can be decided from the sign of the surface difference value in the difference map. For example, the points seen in epoch 2008 but not in epoch 2010,are "demolished" objects. A small change is shown horizontally for clearer visualisation in Figure 6.2 (c). As there are new points in epoch 2010, as well as demolished points beneath them in epoch 2008, we can infer that there are some extensions of the objects. To understand which objects caused the extension, the changes need to be classified.



(c)

Figure 6.2. (a) & (b): Examples of large changes in buildings. Figure 6.2 (c): changes to dormer sizes in the merged data sets (removed dormers and extensions to dormers in height and length).

We selected 20 tiles randomly from each test area to evaluate the performance of the change detection method. There are more false positives than false negatives, so completeness is greater than accuracy, i.e. 50% of identified changes are not relevant. However, the vast majority of changes to buildings are actually detected, making it convenient to only look at the detected changes and ignore the irrelevant ones.

Table 6.1. The completeness and correctness of detected change.

	True positive	False positive	False negative	Correctness	Completeness
2008 vs. 2010	34	34	3	50.00%	91.89%
2010 vs. 2012	35	30	6	53.86%	83.33%

Table 6.2. Percentage of errors due to scene classification and errors due to the method of change detection used.

<table>
<tr><td></td><td colspan="2">False positive</td><td colspan="2">False negative</td><td></td></tr>
<tr><td rowspan="2">2008 vs. 2010
(Test area 1)</td><td rowspan="2">34</td><td>14</td><td rowspan="2">3</td><td>1</td><td>Errors due to the scene classification error
40%</td></tr>
<tr><td>20</td><td>2</td><td>Errors due to limitations of the change detection method
60%</td></tr>
<tr><td colspan="6"></td></tr>
<tr><td rowspan="2">2010 vs. 2012
(Test area 2)</td><td rowspan="2">30</td><td>18</td><td rowspan="2">6</td><td>6</td><td>Errors due to the scene classification error
67%</td></tr>
<tr><td>12</td><td>0</td><td>Errors due to limitations of the change detection method
33%</td></tr>
</table>

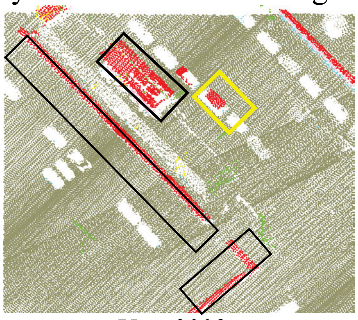
As shown in Table 6.2, 40% of the errors (false positives and negatives) are due to incorrect scene classification in Test area 1, and 67% in Test area 2. In Test area 1, which is a commercial area of Rotterdam, the main propagation errors from scene classification are large and long balconies in walls, which are incorrectly classified as roofs. In Test area 2, a residential area, the error propagation mainly comes from trees that are so dense that they are wrongly classified as building roofs. So, changes in trees were incorrectly detected as changed building objects.

Note that in Table 6.2 there are two false negatives (relevant changes that are not detected) due to our method in Test area 1. These two relevant changes are detected in the change detection method, but the 3D bounding boxes are not generated. The reason for this is that we applied a threshold value for the ratio of the length to width to exclude some thin and long objects such as fences and railings during the change quantification stage. These two relevant changes were roofs that are long and narrow, so 3D bounding boxes were not generated for them.

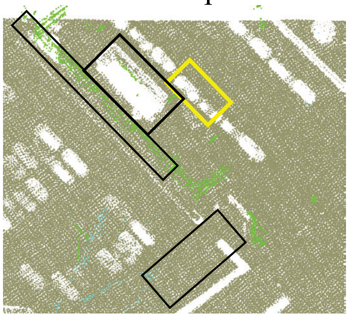
6.1.3 Change detection error due to scene classification error

Those points on the surface difference map that are labelled as part of a building are selected for the building change-detection step. When buildings are not correctly classified in the scene classification in any one of the epochs, it will influence the quality of building change detection. Errors in scene classification give rise to false positive and false negative results.

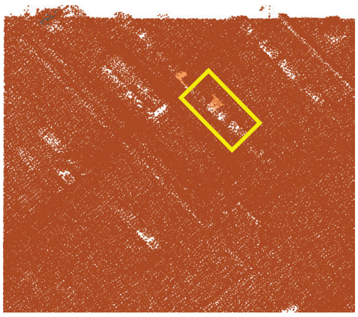
False positives occur when a change has occurred in the areas where non-building objects have been incorrectly classified as “buildings”. In the yellow rectangles shown in Figure 6.3, there is an error in the scene classification of one of the epochs such that cars are incorrectly classified (in the top row) as a building roof; the water surface is also incorrectly classified as a building roof (in the bottom row). As there are changes in these areas, and one of them is incorrectly classified as part of a building, the changes are confirmed as a “change to a building”. Although these represent real changes that that have occurred, they are not changes to buildings. In fact, most false positives occur with trees, when a tree is incorrectly classified as a building roof in one of the epochs.



Year 2008



Year 2010



Change detection in building (merged)

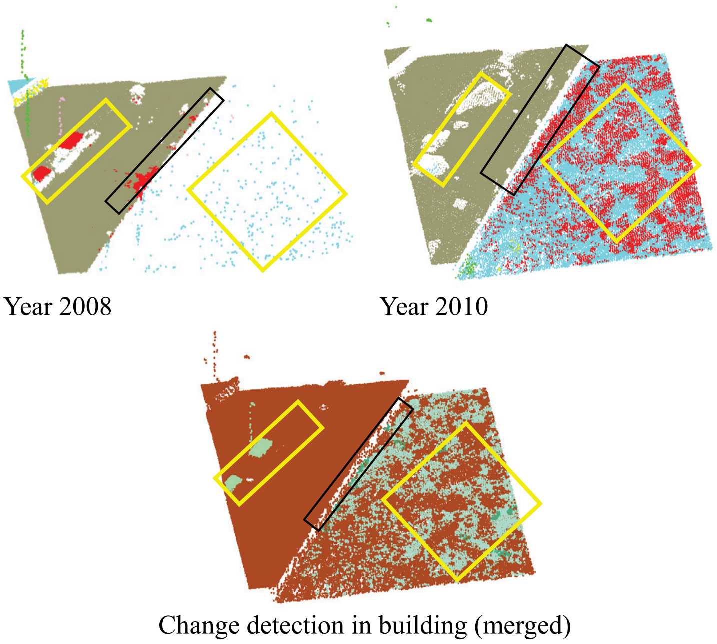


Figure 6.3. Scene classification errors (black rectangles) that have no influence on the change detection results, and ones that have a negative influence (yellow rectangles).

False negatives occur when a change is confirmed but this has not been classified as being part of a building in the data sets being compared: the change will not be signalled as a “change to a building”. Some examples of false negatives resulting from scene classification errors are shown in Figure 6.4. If one of the data sets is correctly classified, false negatives can be avoided.

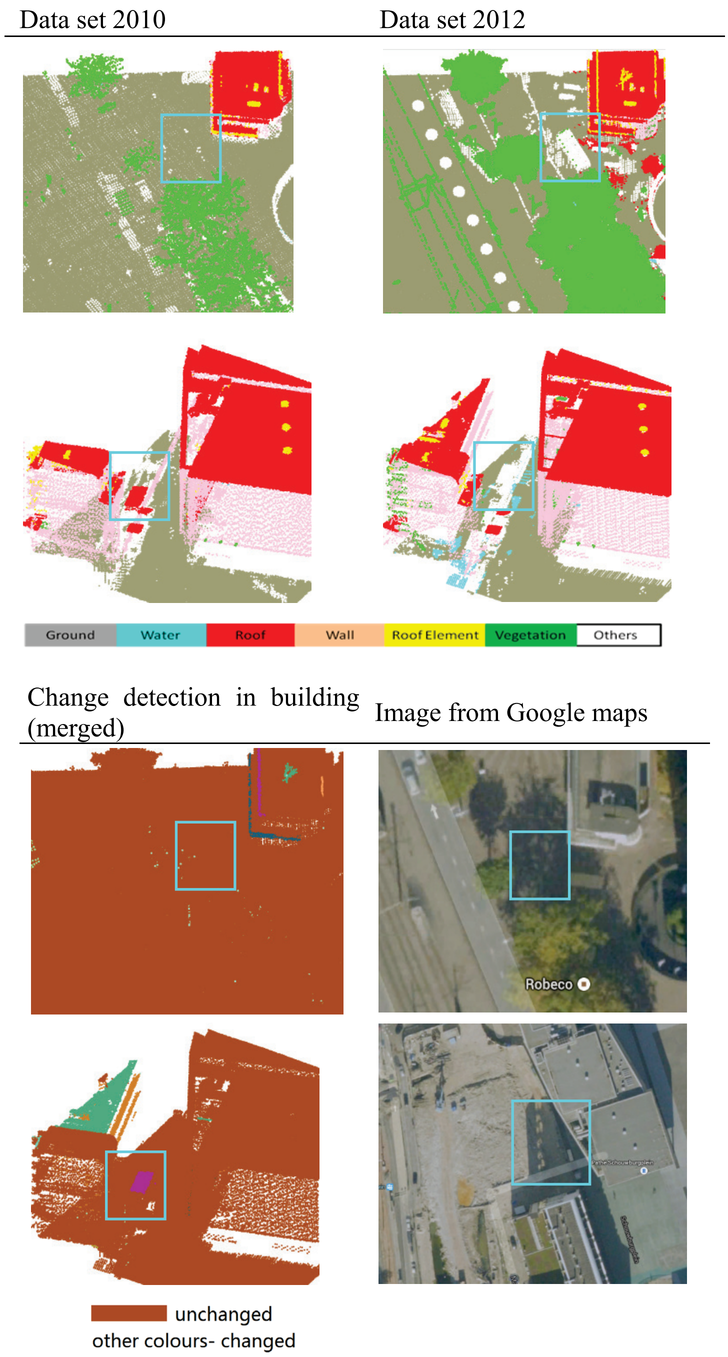


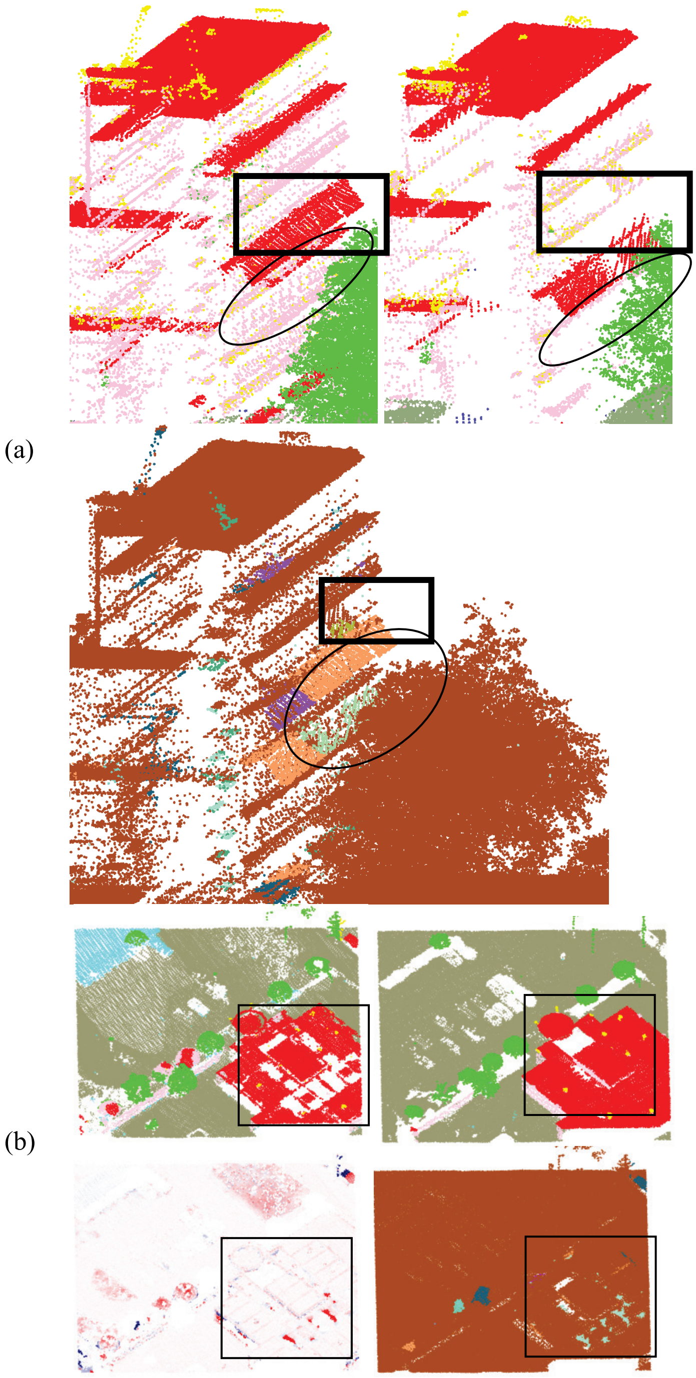
Figure 6.4. Examples of false negatives caused by scene classification errors. Because a building has been classified as an “undefined object” in the “Data set 2012” (in the rectangle), changes to this building will not be detected (Change detection(merged)).

In addition, sometimes cranes, bridges, etc., are classified as being part of a building. Nevertheless, we did not analyse changes in these objects because they did not fall within the scope of our research and we had not defined their features. Consequently, errors related to these objects have been ignored and not discussed in this thesis.

6.1.4 Errors in change detection due to our algorithm

In addition to errors propagated from scene classification results, other errors arise that are caused by limitations of the change detection algorithm used. From visual inspection, we could conclude that no changes were missed (false negatives) due to our change detection method; some unchanged walls were, however, classified as changed (false positives). Figure 6.5 (a) shows the incorrect classification of occluded walls as "changed". We assumed that if no change to the roof of a building had occurred, the attached walls would not change either, even if they had a large difference value. Under this assumption, all wall points in the left-hand image of Figure 6.5 (a) should be labelled "unchanged". However, there are some projections –balconies, sun shades, etc. – on the walls that are far away from the roof, and these projections will be identified as "changed" because they have a large difference value and there was no unchanged roof found in their 2D neighbourhood.

Figure 6.5 (b) shows an example of what should be "unknown" points being incorrectly labelled as "unchanged". The lack of data for the roofs in one epoch was caused by a water layer that absorbed the laser signals. In the other epoch, the point distribution is rather regular. We expected that the entire area for which data are missing would be classified as "unknown". However, only the central part of the area was classed as "unknown" (light blue). The reason is because only the central parts of these areas in the merged data sets have a large difference value (> 1.0 m radius). In other parts there are points from another epoch with a difference value is less than 10cm. Finally, these errors are not important because they did not influence the identification of the "changed" areas; they only resulted in a mix up between the designation "unchanged" and "unknown".



(1) Classification result for the compared epochs of data

(2) Difference map and detected change

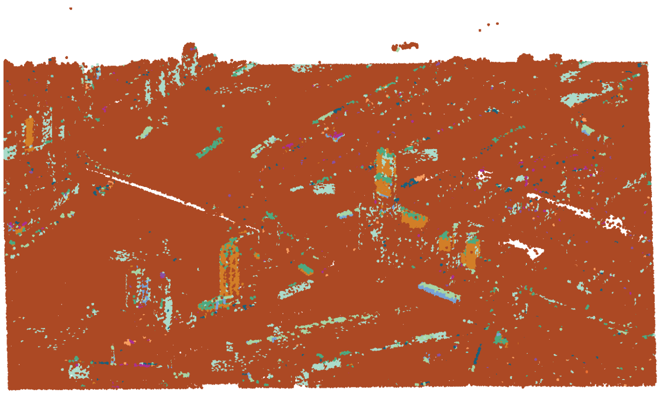
Figure 6.5. Errors due to our change detection algorithm. Colours in this figure represent the same objects as in Figure 6.4. Group (1a:) some errors in the circle are due to scene classification errors, and some in the box are due to lack of the contextual information (wall points far away from the roof). Group (1b): unknown points labelled as unchanged. As seen in the classification result of Group (1b, left-hand), there is a lack of data (some holes) due to limited reflection (red) in the left-hand image of the roof, but the roof in the right-hand image is quite well covered . We expected the entire area where data is lacking to be labelled "unknown", but in the result in (2) only the central parts of the "gaps" are "unknown"(light blue), while other parts are labelled "unchanged" (brown).

6.2 Change classification

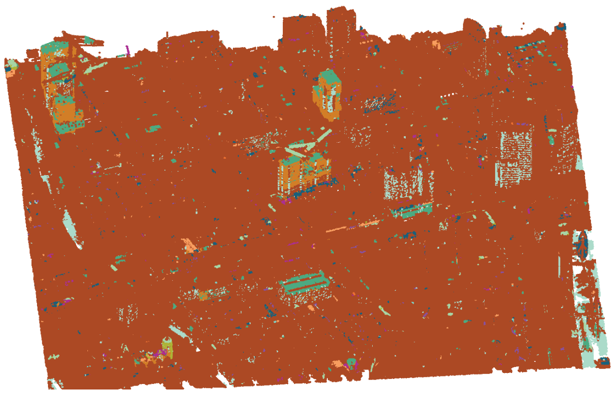
6.2.1 Results

The change classification results for the two test areas are visualised in Figure 6.6; for examples with higher level of detail see Figure 6.7. We chose several sites where changes were successfully detected and properly classified. These sites included examples of: (a) newly built dormers on roofs, (b) lack of data for roofs in one epoch (unknown) and lack of data for walls and ground because of occlusion, (c) undefined changes on roofs, (d) newly built constructions above roofs, (e) cars parked on top of buildings, (f) newly built and demolished buildings (two examples in one image), (g) add-on building constructions, and (h) insulation layers added to roofs.

Some change detection errors are also caused by errors in scene classification – in addition to those arising from the limitations of our algorithm. These are discussed in Subsection 6.1.2. More false positives are observed, lowering the accuracy, which is analysed in Section 6.3.

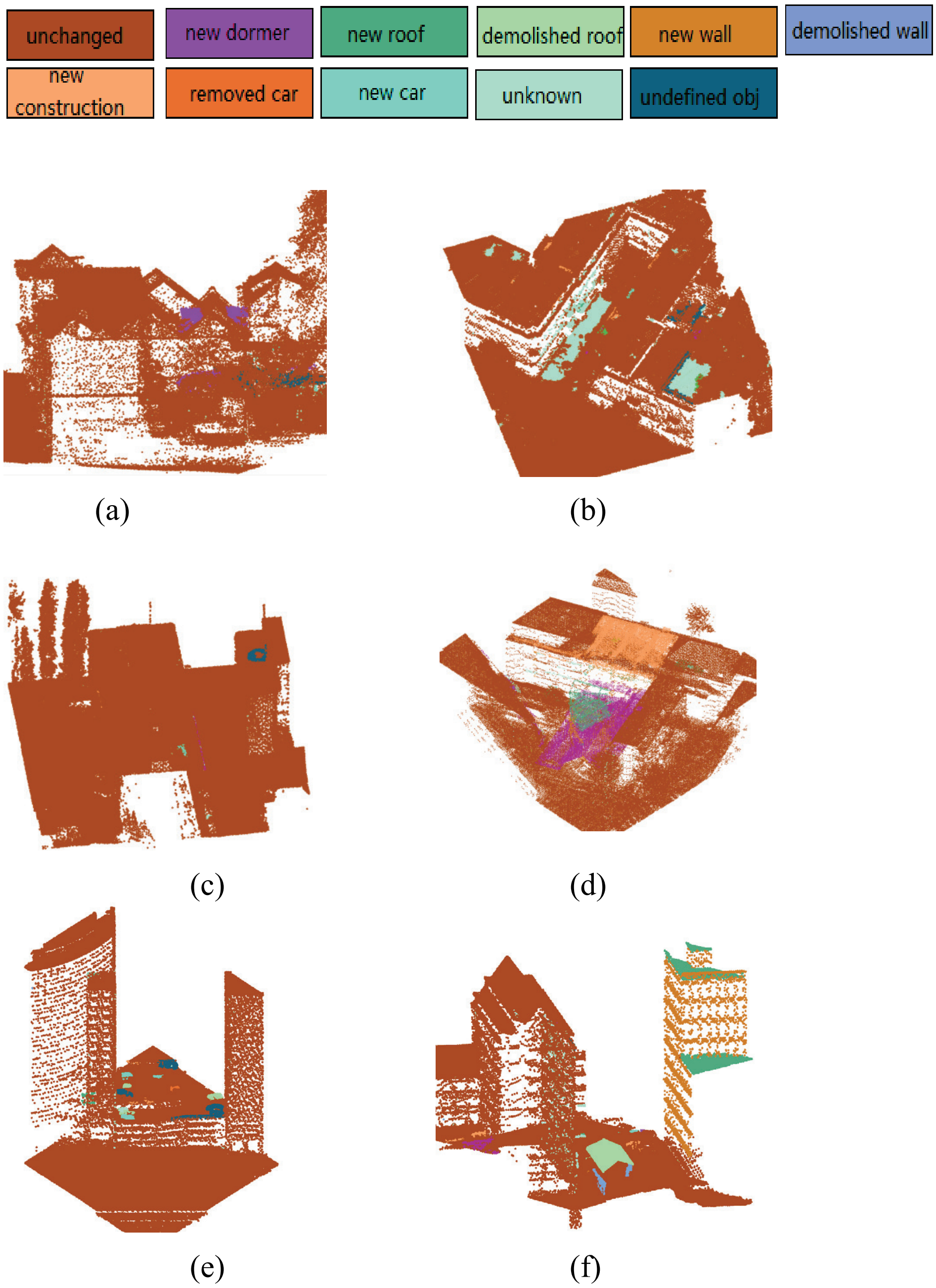


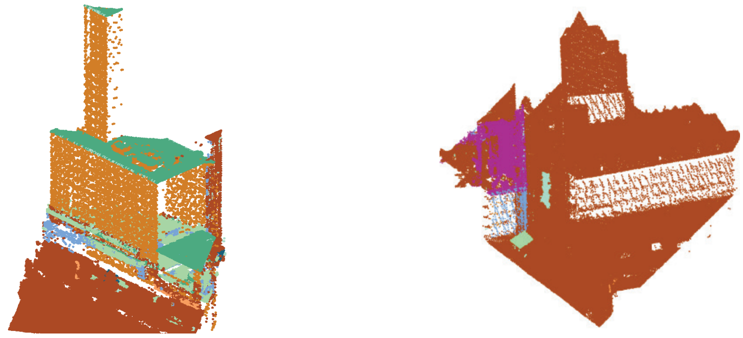
2008 vs. 2010



2010 vs. 2012

Figure 6.6. Classification of change in the test areas


unchanged
new dormer
new roof
demolished roof
new wall
demolished wall
new construction
removed car
new car
unknown
undefined obj
(a)
(b)
(c)
(d)
(e)
(f)



(g) (h)

Figure 6.7. Successfully detected and classified examples of changes.

6.2.2 Error analysis

We randomly chose another 20 tiles from each study area to cover all types of changes on roofs. Error analysis of the change classification was done by manually assessing the correctness and completeness of changes detected (see Table 6.3) .

Table 6.3. Completeness and correctness of the changed objects.

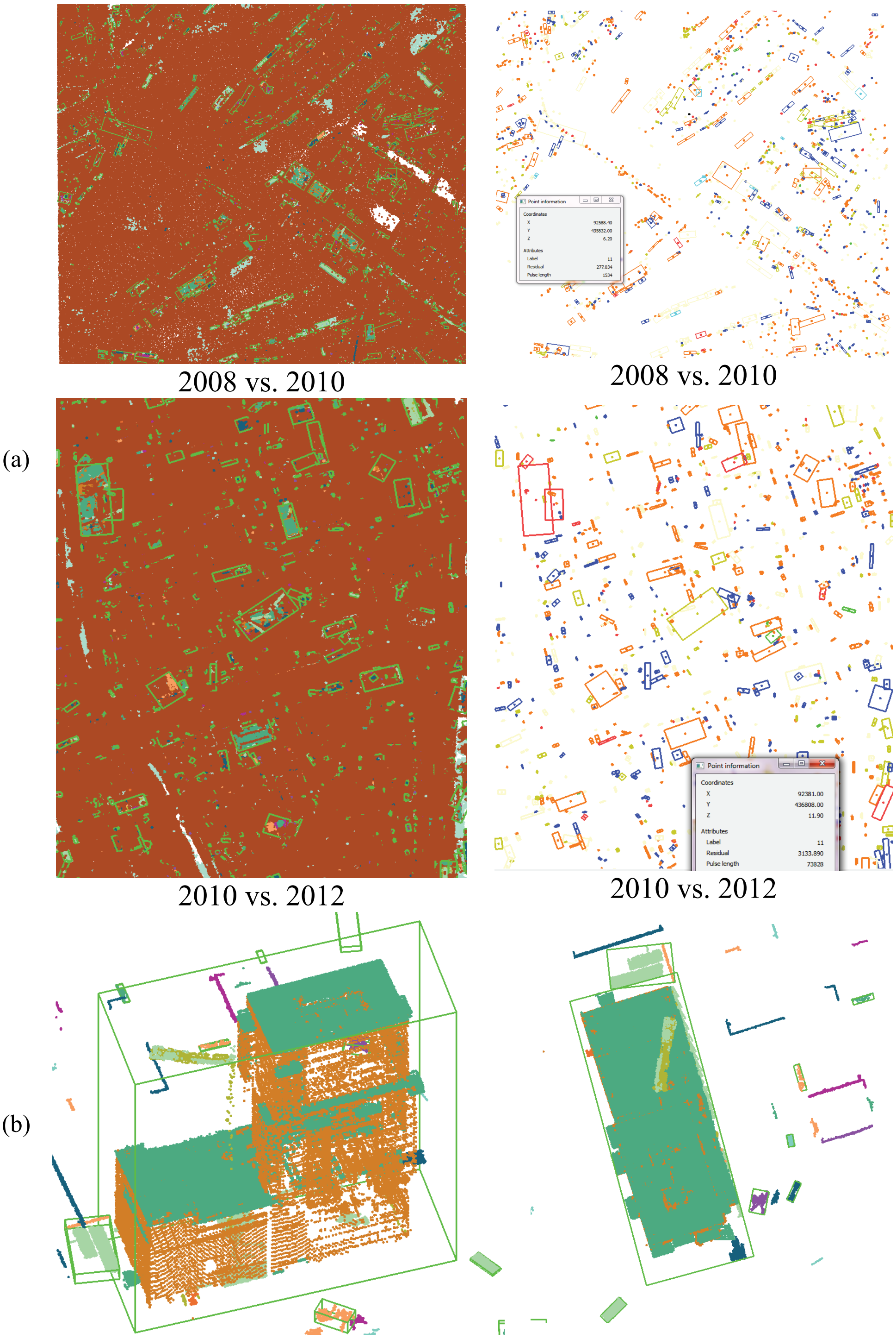
Label	*2008 vs. 2010*			*2010 vs. 2012*		
	Correct ness	*Complete ness*	*Number of objects*	*Correctness*	*Complete ness*	*Number of objects*
Undefined object	91%	35%	29	100%	26%	23
New dormer	6%	100%	1	80%	24%	17
Demolished dormer	--	--	0	--	--	0
New car on roof top	33%	11%	9	50%	7%	14
Car no longer on rooftop	100%	100%	1	--	--	0
New construction on rooftop	13%	25%	8	42%	56%	9
Demolished construction on rooftop	100%	40%	5	--	0%	2
New roof	78%	81%	26	59%	100%	17
Demolished roof	91%	100%	20	90%	100%	18
New wall	100%	100%	8	100%	100%	7
Demolished wall	100%	100%	8	100%	100%	4

The accuracy of change classification for roofs and walls is high compared to other objects because changes to roofs and walls occur most often in buildings undergoing entire change. In addition to that, scene classification is more reliable for larger objects: changes to buildings are generally large and can easily be correctly classified. Small changes are not so easily separated into their correct change classifications. Most constructions and undefined objects on rooftops are incorrectly classified as new or demolished dormers, especially if they are near a pitched roof. We have assumed that dormers are normally located near the pitched roof, but often roofs are in fact a combination of flat and pitched roofs, and there are as many changes on these roofs, which are not changed dormers but incorrectly classified as such. It is also difficult, based only on their size, to distinguish large cars on a rooftop parking lot from building constructions. As a result, some of these large cars are classified as constructions. We conclude that, even if the changes are detected correctly, in highly complex urban areas it is hard to completely and correctly classify the detected changes using geometrical and relational rules.

6.3 Object-based analysis

6.3.1 Results

Minimum 3D bounding boxes are generated around the connected components to enable selection of relevant changes in the buildings; for some results see Figure 6.8. The location of the centre point, the area and the volume of the minimum 3D bounding box have been calculated. Changes are shown in Figure 6.8 (a). Figures 6.8 (b) and 6.8 (c) give two examples of 3D bounding boxes: 6.8 (b) for a building object that has undergone complete change; and 6.8 (c) a changed building element (newly built dormer). The bounding box in Figure 6.8 (c) is larger than the dormer because of the influence of some outliers occurring in the same plane as the dormer.


2008 vs. 2010
2008 vs. 2010
(a)
2010 vs. 2012
2010 vs. 2012
(b)

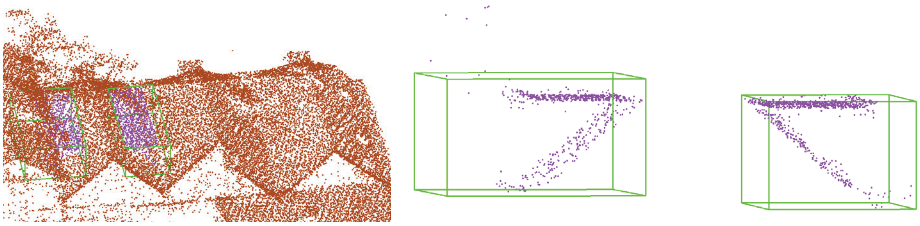
(c) 

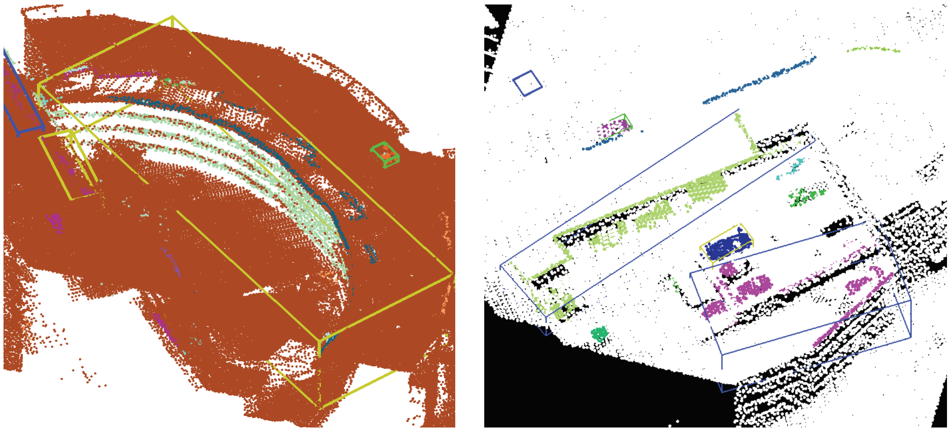
Figure 6.8. 3D bounding boxes and the centre points of the relevant changes, together with the calculated area and volume; the label, area and the volume are shown in the point number window. In this figure, points without bounding boxes are irrelevant changes.

6.3.2 Error analysis

We used connected component labelling to form objects and their 3D bounding boxes to estimate their area and volumes. We found that connected component labelling failed to correctly form objects under two circumstances:

(1) False positives in buildings
Sometimes points on walls or roofs are false positives. These points may belong to an unchanged wall that has been incorrectly detected as changed (see discussion in Subsection 6.1.4), or they can be sparse points of plants on a balcony or rooftop (irrelevant changes). If such points are close together, connected component labelling will form an object that is as big as an entire wall or even a complete building; see Figure 6.9 (a).

(2) Small objects that are too close to each other
Small objects that are too close to each other will be connected together as one object. This often occurs with cars parked on rooftops of buildings and shelters at bus stops that are very close to each other. Figure 6.9 (b) shows an example of cars parked on a rooftop. The consequence is an incorrect shape of detected change.



(a) (b)

Figure 6.9. (a) Incorrectly indicated changes to walls and irrelevant changes on roofs; (b) Cars that are close to each other and close to some fences are represented as several large objects.

6.4 Discussion

In chapter 5 we present a method for detecting and classifying changes to buildings by using classified laser data from several epochs. The analysis of our results in this chapter leads us to put forward several discussions.

Provided distances between surfaces are greater than 10 cm and the area of change more than 4 m^2, both large and small changes be automatically detected using a surface difference map and a rule-based change detection algorithm. Areas classified as "unknown" can be correctly identified in cases of occlusions and water reflection. The surface difference map is not affected by the point density of the compared data sets.

Larger changes can be correctly assessed as belonging to a building provided that building has been correctly classified as such in the scene classification step for one of the data sets being compared. The accuracy of object recognition was evaluated by overlaying the 3D bounding boxes of the buildings for which change was detected on manually generated reference data. Our method detected 91% of actual changes in Test area 1 and 83% in Test area 2. Nearly half of the changes detected in objects were irrelevant changes.

About half the false positives occurring were caused by scene classification error. The other false positives can be mostly attributed to our change detection algorithm. Mostly they are spurious changes in a wall that is far removed from a roof or plants growing in a rooftop garden. On classifying changes, we conclude that even if changes are identified correctly, in highly complex urban areas it is difficult to completely and accurately classify the smaller detected changes using geometrical and relational rules.

The classification result for the changed object shows that large changes affecting an entire building, for example its roof or a main wall, can be detected with a higher degree of accuracy than changes made to a building element, such as dormers or construction on top of the building.

Overall, our method detected 80-90% of changes to buildings. The method is, however, not yet capable of distinguishing small irrelevant changes to objects from relevant ones. A more accurate definition of a "changed building object" and its characteristics are required in order to better interpret differences between two data sets.

Chapter 7 Conclusions and recommendations

Conclusions relating to the scene classification and change detection methodologies used in this research are presented, and limitations discussed, in Section 7.1. Finally, I make some recommendations in Section 7.2.

7.1 Conclusions

Conclusions relating to the advantages and shortcomings of the methodology used for scene classification are given in Subsection 7.1.1. Those for the change detection methodology as applied to buildings are given in Subsection 7.1.2. Finally, in Subsection 7.1.3 I present some general conclusions drawn from both the scene classification and change detection methodologies.

7.1.1 Scene classification

The methodology and features I have described in this thesis have proven to be suitable for the classification of airborne laser scanning (ALS) data: a scene classification accuracy reaching 97% could be achieved using the rule-based classifier. Over 95% of the building points were extracted and further classified as “wall”, “roof” or “roof element”. With the exception of adjustments to some parameters, the entire scene classification process is fully automated.

Nevertheless, the methodology still suffers from several limitations:

- the defined features are not suitable for terrestrial laser scanning data, as several features such as “segment size” and “average point spacing” were used, which requires that point density is not influenced by the distance between the scanned objects and the laser scanner. There are large differences in point density in the terrestrial laser scanning data.

- the strategy of using multiple entities can solve problems arising from vegetation occurring above building roofs, but it cannot solve problems resulting from vegetation near or even connected to building walls. Parts of such vegetation will be classified as “building wall” during the contextual reasoning step.

- - low vegetation, such as bushes and grass, are not classified as vegetation due to the definition of vegetation used. Only tall trees (height above DTM > 2.0 m) are classified as vegetation; lower vegetation is classified as “undefined object”.

- the methodology requires the data to be filtered before the classification.

In addition to the above, some objects cannot be clearly distinguished from each other due to limitations of the lidar data, which does not include colour and texture information. The following objects are easily mixed up:

- roofs of small buildings and large trucks (undefined object)
- building roofs and wall elements, such long balconies, sun shades or opened windows set into walls.
- building roofs and dense, flat vegetation.

In lidar data, these objects have similar geometries, so it is difficult to separate them using only x,y, z or other information from lidar data. If vegetation is too dense to allow pulses to penetrate, that vegetation may appear as a smooth surface with no features suitable for distinguishing it from building roofs in the lidar data.

7.1.2 Change detection in buildings

Change detection based on a surface difference map is not influenced by the point density of the compared data sets, so this method of change detection can avoid signalling spurious changes due to occlusion or different reflection characteristics of object surfaces.

The accuracy of the change detection is evaluated per object and the detection rate is around 90%. However, nearly half the detected changes are spurious or irrelevant changes. The spurious changes are detected because of errors in scene classification (incorrect labelling of walls and vegetation as “building roof”. The irrelevant changes are real changes, but they are not distinguished from relevant changes because the method does not have enough rules to separate one from the other.

The connected component method can also be used to compose changed building objects for either changes to an entire building or to a roof element. However, although this method is suitable for larger changes, such as those to an entire building, it is not very suitable for detecting changes to roof elements when several different types of roof elements are very near to each other (these different roof elements are then grouped to one object using the connected component).

7.1.3 General conclusions

In addition to the specific conclusions I have drawn above, there are some general conclusions to be drawn from both the scene classification and the change detection processes. The parameters, both those for the surface-growing method and those in the grouping rules, need to be adjusted for different data sets when the point density differs, since the surface-growing method, as well as some of thresholds for the features, are influenced by point density.

Furthermore, outliers in the data sets were the cause of incorrect classifications and spurious changes. Such outliers could be:

- points on floors inside the building. These points are scanned if laser beams penetrate window glass and hit the floor inside the building.

- points that are far below ground level or up in the sky (e.g. balloons or helicopters).

Outliers related to floors are partly classified as “building roof” or “building wall” in the scene-classification step and are will be detected as changes to the building if the outliers appear in only one epoch of the data and are absent in another. Points below ground level will influence the calculation of the height to the DTM if they are misclassified as ground. Consequently, incorrect distances from the object to the DTM will be calculated.

7.2 Recommendations

Buildings either partly or fully covered by trees can be classified and changes detected. Buildings (walls, roofs or roof elements) extracted in the scene classification step can be used to detect change and for the detailed modelling of buildings. Larger trees extracted can also be used for the modelling trees, as well as change detection (Xiao et al., 2012).
The change detection methodology can be used to detect building changes that display differences in their geometry greater than 10 cm (in the data the maximum strip difference is 10 cm; this may vary with

differences in registration accuracy) and have an area greater than 4 m^2 (the area can be adjusted according to application requirements). Such changes provide insight into building construction activities and help in verifying building permits. Quantification of changes can assist in the efficient updating of maps.

As already noted in Subsection 7.1.1, the methodology studied has several limitations. These can be addressed by: (1) adding new features to indicate the size of a segment for terrestrial laser scanning data so that the method can be used for this type of data too, and; (2) using different segmentation methods to allow one segment to represent one object (something already investigated by Vosselman (2013)).

I have mentioned that some points in trees and on buildings have similar geometries and are difficult to separate during scene classification of lidar data. Here I would suggest incorporating images and exploiting the texture and colour information available to improve the accuracy of classification of high and low vegetation.

Improved scene classification can also improve the accuracy of change detection. In particular, some spurious changes are identified in trees because the trees were incorrectly classified as building roofs. Such spurious changes can be eliminated if trees and roofs are 100% separated. Irrelevant changes are difficult to distinguish from the relevant ones because the wide range of types of changes in a city complicates the detection process. This can be addressed by defining certain patterns for each type of change to assist recognition of the relevant ones. For example, for the change resulting from construction of a new dormer, surface difference values may have a characteristic distribution that distinguishes this type of change from other changes.

Bibliography

Anguelov, D., Taskar, B., Chatalbashev, V., Koller, D., Gupta, D.,Heitz, G. and Ng, A., 2005, Discriminative Learning of Markov Random Fields for segmentation of 3D Scan Data. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, pp. 169-176.

Breiman, Leo, 2001, Random Forests. Machine Learning 45 (1), 5–32.

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

Brzank A., Heipke C., 2006, Classification of lidar data into water and land points in coastal areas, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36(Part 3), on CD-ROM

Byrne, G. F., Crapper, P. F., Mayo, K. K., 1980, Monitoring land-Cover change by principal component analysis of multitemporal landsat data, Remote Sensing of Environment 10(3), 175-184.

Champion, N., 2007, 2D Building change detection from high resolution aerial images and correlation digital surface models, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36(Part 3/W49A), 197-202.

Champion, N., Rottensteiner, F., Matikainen, L., Liang, X., Hyyppä, J., Olsen, B. P., 2009, A test of automatic building change detection approaches. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 38 (Part 3/W4), 145-150.

Chehata, N., Guo L., Mallet c., 2009, Airborne lidar feature selection for urban classification using random forests.Laser scanning 2009, The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 38(Part 3/W8) , 207-212

Chen, L., Lin, L., Cheng H.,Lee, S., 2010, Change detection of building models from aerial images and Lidar data. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 38 (Part 7B), 121-126.

Chen Q., Gong P., Baldocchi D., Xie G., 2007, Filtering Airborne Laser Scanning Data with Morphological Methods. American Society for Photogrammetry and Remote Sensing 73(2), 175-185.

Choi, K., Lee, I., Kim, S., 2009 A feature based approach to automatic change detection from Lidar data in urban areas. The International

Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 38 (Part 3/W8), 259-264.

Comaniciu, D. and Meer, P., 2002, Mean shift: a robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(5), 603-619.

Darmawati, A. T., 2008, Utilization of multiple echo information for classification of airborne laser scanning data. Master's thesis, International Institute for Geo-Information Science and Earth Observation (ITC), Enschede, the Netherlands.

Di F., Li X., Zhu C., 2009, A new method in change detection of remote sensing image. IEEExplore Digital Library, Image and Signal Processing, CISP' 09. 2nd International Congress, 17-19 October, pp. 1-4.

Ferraz, A., Bretar, F., 2010, 3D segmentation of forest structure using a mean-shift based algorithm. Image Processing (ICIP), 2010 17th IEEE International Conference, 26-29 September, pp. 1413-1416.

Frontoni, E., Khoshelham, K., Nardinocchi, C., Nedkov, S., Zingaretti, P., 2008, Comparative analysis of automatic approaches to building detection from multi-source aerial data, The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 38(Part 4/C1), ON-CDROM.

Gerke, M., 2009, Dense matching in high resolution oblique airborne images. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 38 (Part 3/W4), 77-82.

Hebel, M., Arens, M.,Stilla U., 2013, Change detection in urban areas by object-based analysis and on-the-fly comparsion of multi-view ALS data.ISPRS Journal of Photogrammetry and Remote Sensing Volume 86, December 2013, pp: 52–64.

Kasetkasem, T., Varshney, P. K., 2002, An Image Change detection algorithm based on Markov Random Field models. IEEE Transactions on Geoscience and Remote Sensing 40(8), 1815-1823.

Khoshelham K., Nardinocchi C., Frontoni E., Mancini A., Zingaretti P., 2010, Performance evaluation of automated approaches to building detection in multi-source aerial data. ISPRS Journal of Photogrammetry and Remote Sensing 65 (1), 123-133.

Khoshelham, K., Oude Elberink, S.J. and Xu, S., 2013, Segment based classification of damaged building roofs in aerial laser scanning data. IEEE Geoscience and Remote Sensing Letters 10 (5), 1258-1262.

Kim, H. B. and Sohn, G., 2011, Random Forests-based multiple classifier system for power-line scene classification. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 38 (Part 5/W12) , 253-258.

Lafarge, F and Mallet, C., 2011, Modeling urban landscapes from point clouds: generic approach. Report: INRIA-00590897, Version 1-5 May, Laboratoire I3S, France.

Li S.Z., 2001, Markov Random Field Modeling in Image Analysis. Springer, 2001.

Lim, E. H. and Suter, D., 2008, Multi-scale conditional random fields for over-segmented irregular 3D point-clouds classification. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 23-28 June, 1-3, pp. 856-862.

Lim, E. H. and Suter, D., 2009, 3D terrestrial lidar classifications with super-voxels and Multi-scale Conditional Random Fields. Journal of Computer Aided Design 41(10), 701-710.

Lindenberger, J., 1993, Laser-Porfilmessungen zur topographischen Geländeaufnahme. Deutsche Geodätische Kommission, Series C, No. 400, Munich.

Lodha, S. K., Fitzpatrick, D., Helmbold, D. P., 2006, Aerial lidar data classification using support vector machines. In: Proceedings of Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT), pp. 567 – 574.

Lodha, S. K., Fizpatrick, D. M., Helmabold, D. P., 2007, Aerial lidar data classification using AdaBoost. In: 3-D Digital Imaging and Modeling, 3DIM'07. Sixth International Conference, Montreal, Quebec, Canada, pp. 435-442.

Lu, D., Mausel, P., Brondizio, E., Moran, E., 2004, Change detection techniques, International Journal of Remote Sensing 25(12), 2365-2407.

Matikainen, L., Hyyppä, J., Kaartinen, H., 2004, Automatic detection of changes from laser scanner and aerial image data for updating building maps. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 35 (Part B2), 434-439.

Matikainen, L., Kaartinen, H., Hyyppä, J., 2007, Classification tree based building detection from laser scanner and aerial image data. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 3/W52), 280-287.

Munoz, D., Vandapel, N. and Hebert, M., 2008, Directional associative Markov Network for 3-D point cloud classification. In: International

Symposium on 3D Data Processing, Visualization and Transmission, Atlanta, GA.

Munoz, D., Vandapel, N. and Hebert, M., 2009, Onboard contextual classification of 3D point clouds with learned high-order Markov Random Fields. IEEE International Conference on Robotics and Automation, Kobe, Japan, pp. 4273–4280.

Murakami, H., Nakagawa, K., Hasegawa, H., Shibata, T., Iwanami, E., 1999, Change detection of buildings using an airborne laser scanner. ISPRS Journal of Photogrammetry and Remote Sensing 54 (2-3), 148-152.

Niemeyer, J., Rottensteiner, F., Soergel, U., 2012, Conditional random fields for lidar point cloud classification in complex urban areas. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume I-3, 2012 XXII ISPRS Congress, 25 August – 01 September, pp. 263-268.

Niemeyer, J.; Mallet, C., Rottensteiner, F., Sörgel, U., 2011, Conditional random field for the classification of lidar point clouds. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 38 (Part 4/W19) ,on CD-ROM.

Oude Elberink, S. and Vosselman, G., 2009 (a), Building reconstruction by target-based graph matching on incomplete laser data: analysis and limitations. Sensors 9(8), 6101-6118.

Oude Elberink, S. and Vosselman, G., 2009(b), 3D information extraction from laser point clouds covering complex road junctions. Photogrammetric Record 24(125), 23-36.

Priestnall, G., Jaafar, J., Duncan, A., 2000, Extracting urban features from Lidar digital surface models. Computers, Environment and Urban Systems 24 (2), 65-78

Roberts, G. and Hirst, L., 2005, Deformation Monitoring and Analysis of Structures Using Laser Scanners, from : Pharaohs to Geoinformatics FIG Working Week 2005 and GSDI-8 Cairo, Egypt April 16-21, pp.

Rottensteiner, F. and Trinder, J., 2007, Building detection by fusion of airborne laser scanner data and multi-spectral images: Performance evaluation and sensitivity analysis. ISPRS Journal of Photogrammetry and Remote Sensing 62 (2), 135-149.

Rottensteiner, F., 2007. Building change detection from digital surface models and multi-spectral images. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 3/W49B), 145-150.

Rottensteiner, F., Trinder, J., Clode, S., Kubik, K.., 2004, Fusing airborne laser scanner data and aerial imagery for the automatic extraction of building in densely built-up areas. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 35 (Part B3), 512-517.

Rutzinger, M., Höfle, B., Geist, T., Stötter, J., 2006, Object based building detection based on airborne laser scanning data within grass GIS environment. Remote Sensing for Urban Applications 7(Part 2), 37-48.

Rutzinger, M., Ruf, B., Hofle, B., Vetter, M., 2010, Change detection of building footprints from airborne laser scanning acquired in short time intervals. The International Archives of Photogrammetry and Remote Sensing 38 (Part 7B), 475-480.

Samadzadegan, F., Bigdeli, B., Ramzi, P ., 2010, A multiple classifier system for classification of LIDAR remote sensing data using multi-class SVM. Multiple classifier systems, Proceedings Book Series, Lecture Notes in Computer Science 5997, 254-263.

Samadzadegan, F., Hahn, M., Bigdeli, B., 2009, Automatic road extraction from LIDAR data based on classifier fusion. Joint Urban Remote Sensing Event 1-3, 1605-1609.

Shapovalov, R, Velizhev, A, Barinova, O., 2010, Non-associative Markov Random Networks for 3D point clouds classification. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 38 (Part3A), 103-108.

Sharma, B., Rishabh, I., Rkshit, S., 2006, Unsupervised change detection using RANSAC, IEEExplore Digital Library Visual Information Engineering, 2006. VIE 2006. IET International Conference, 26-28 September, pp. 24-28.

Singh, A. 1989, Digital Change Detection Techniques Using Remotely Sensed Data. International Journal of Remote Sensing 10(6), 989-1003.

Sithole, G and Vosselman, G, 2003, Automatic structure detection in a point-cloud of an urban landscape, 2nd GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion Over Urban Areas, pp. 67-71.

Sithole, G. and Vosselman, G., 2004, Experimental comparison of filter algorithms for bare-earth extraction from airborne laser scanning point clouds. ISPRS Journal of Photogrammetry and Remote Sensing 59 (1-2), 85–101.

Sithole, G. and Vosselman, G.,2006, Bridge detection in airborne laser scanner data, 2006. ISPRS Journal of Photogrammetry and Remote Sensing 61 (1), 33-46.

Sui, L., Wang, X., Qu J., 2008, Application of 3d laser scanner for monitoring of landslide hazards. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 37(Part B1), 277-282.

Tanathong, S., Rudahl, K. T., Goldin, S. E., 2009, Object oriented change detection of buildings after a disaster. ASPRS 2009 Annual Conference Baltimore, Maryland, March 9-13. on CD-ROM.

Vögtle, T., Steinle E., 2004, Detection and recognition of changes in building geometry derived from multi-temporal laser scanning data. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 34 (Part B2), 428-433.

Vosselman, G., 2000, Slpoe Based Filtering of laser altimetry data, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 33(Part B3/2), 935-942.

Vosselman, G., 2012, Automated planimetric quality control in high accuracy airborne laser scanning surveys. ISPRS Journal of Photogrammetry and Remote Sensing 74, 90-100.

Vosselman, G., 2013, Point cloud segmentation for urban scene classification. ISPRS Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 40 (Part 7/W2), 257-262.

Vosselman, G., Gorte, B.G.H., Sithole, G., 2004(a), Change detection for updating medium scale maps using laser altimetry. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 34 (Part B3), 207-212.

Vosselman, G., Gorte, B.G.H., Sithole, G., Rabani, T., 2004(b), Recognising structure in laser scanner point clouds. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 8/W2), 33-38.

Wei, Y., Hinz, S., Stilla, U., 2009, Object extraction based on 3D-segmentation of lidar data by combining mean shift with normalized cuts: two examples from urban areas. In: Joint Urban Remote Sensing Event, 20-22 May, pp. 1-6.

Xiao W., Vallet B., Paparoditis N., 2013, Change detection in 3D point clouds acquired by mobile mapping system. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences 2 (Part 5/W2), 331-336.

Xu, S., Oude Elberink, S., Vosselman, G., 2013, Multiple-entity based classification of airborne laser scanning data in urban areas. ISPRS Journal of Photogrammetry and Remote Sensing 88, 1–15.

Yang Z., Qin Q., Zhang Q., 2006, Change Detection in high spatial resolution images based on support vector machine. IEEE International Geoscience and Remote Sensing Symposium, 31 July - 4 August., pp. 225-228.

Summary

Detailed change detection in buildings using airborne laser scanning data (ALS data) has become possible with the availability of multi-temporal ALS data sets. In this thesis we present a methodology for building change detection in urban scenes, which is composed of two main parts: the classification of point clouds of an urban scene and the detection of changes in buildings. A classification methodology is put forward to solve the problem of how to detect buildings in point clouds and how to distinguish the building roofs, building roof elements, and building walls. The change detection methodology is not only used to detect changes but also aims to interpret the type of change that occurred to a building. The two methodologies are suitable for application to raw ALS laser points. They do not require the ALS data to be organised in Digital Surface Models.

The thesis consists of seven chapters. Chapter 1 gives the motivation of this research and an introduction to the background of the two main topics mentioned above. Research problems and questions are raised, and goals and objectives are defined on the two topics. Furthermore, the limits of the research scope of this thesis are set. Chapter 2 introduces the study area used in this thesis, the available data, including the data quality and the data organization, and some pre-processing steps of the data.

Chapter 3 describes the methodology of the classification, explaining the entities, features, classifiers and the classification strategy. We introduce a classification procedure that combines classifications of three different entities: points, planar segments, and segments obtained by mean-shift segmentation. Seven types of objects, namely, water, ground, vegetation, roof, roof element, wall and undefined object, are distinguished based on feature values of the entities. Some features were already defined in literature. Other features are defined by us. Five commonly used classifiers (rule based classification, Random Tree, AdaBoost, SVM, and ANN) are tested. The rule-based method provides over 99% accuracy for the ground and roof classes, and a minimum accuracy of 90% for the water, vegetation, wall and undefined object classes, resulting in an overall accuracy of 97%. The accuracy of the roof element class is only 70% with the rule-based method, or even lower with other classifiers. All experimental results for the classification methodology are presented and

discussed in chapter 4. These results include the evaluation of the classification accuracy, comparisons between different classifiers and comparisons between different features derived from the different entities.

Chapter 5 explains the methodology of the change detection comprising a point-based change detection method and an object-based change analysis. The detection process starts with two data sets that are classified using the classification methodology in chapter 3. Next, a point-to-plane surface difference map is generated by merging the two data sets to be compared. By applying rules to the surface difference map the change status of points is set to “changed”, “unchanged”, or “unknown”. Rules are defined to solve the problems caused by the lack of data. “Unknown” are locations where due to lack of data in at least one of the epochs it is not possible to reliably detect changes in the structure. Points on buildings labelled as “changed” are re-classified into changes related to roofs, walls, dormers, cars, constructions above roofs and undefined objects in a second classification step. Next, all the classified changes are grouped to changed building objects. Geometric descriptions of the changed building objects, such as the location of the centre point of the change objects, the height, area and volume of the change objects, are derived from their minimum 3D bounding boxes. Performance analysis showed that 80% - 90 % of the real changes are found, of which approximately 50% are considered relevant. The results of the change detection and analysis and their accuracy are discussed in chapter 6.

Finally, chapter 7 draws the main conclusions from the test results obtained with the classification and the change detection methodology. Limitations of our methodologies are summarized and potential solutions to these limitations are suggested.

Samenvatting

Dankzij het feit dat steeds meer organisaties herhaaldelijk gedetailleerde laser scan data inwinnen is het mogelijk om nauwkeurig vast te stellen of er in de gebouwde omgeving iets veranderd is. In dit proefschrift staat beschreven hoe laser datasets van verschillende jaren gebruikt kunnen worden in stedelijk gebied om verschillen te detecteren die tussentijds zijn opgetreden. De focus in dit proefschrift ligt op veranderingsdetectie van gebouwen. Dit gebeurt aan de hand van twee belangrijke stappen: de classificatie van laser data en de feitelijke veranderingsdetectie. De classificatie stap geeft inzicht in welke objecten er aanwezig zijn in de datasets, en dus ook waar er zich gebouwen bevinden. Dankzij de hoge mate van detail kunnen daken, muren en zelfs dakkapellen onderscheiden worden. Onze veranderingsdetectie is niet alleen in staat om een verandering te herkennen, maar ook om aan geven wat voor soort verandering heeft plaatsgevonden. Voor deze twee stappen wordt gebruik gemaakt van de ruwe laser altimetrie gegevens, puntenwolken genaamd, zonder enige vorm van nabewerking zoals interpolatie naar een regelmatig raster.

Het proefschrift bestaat uit zeven hoofdstukken. Hoofdstuk 1 bevat de introductie tot classificatie en veranderingsdetectie in puntenwolken, aangevuld met de onderzoeksvragen en doelstellingen. Hoofdstuk 2 beschrijft de gebruikte data, inclusief de opbouw en geometrische kwaliteit. De classificatie methode wordt beschreven in hoofdstuk 3. De methode is gebaseerd op informatie op verschillende schaalniveaus: per laser punt, per segment met laser punten die in een plat vlak liggen, en per segment met punten die anderszins gegroepeerd kunnen worden. De laser data wordt geclassificeerd in zeven klassen: water, terrein, vegetatie, gebouwdaken, objecten op die daken, muren en een restklasse voor alle overige objecten. Op alle drie schaalniveaus worden parameters berekend die geometrische informatie bevatten over tot welke klasse dat gedeelte van de laser data behoort. De beste classificatiemethode geeft een 99% nauwkeurig resultaat voor de klassen gebouwdaken en terrein, en is voor minimaal 90% correct in het geval van vegetatie, muren, water en de restklasse. Echter, voor objecten op daken is het voor ongeveer 70% correct. Verschillende classificatiemethoden en hun resultaten worden vergeleken en beschreven in hoofdstuk 4.
Hoofdstuk 5 behandelt de veranderingsdetectie, waarbij gebruik gemaakt wordt van de geclassificeerde puntenwolken uit tenminste twee

verschillende opnameperiodes. Eerst wordt per punt de afstand tot het dichtstbijzijnde vlak in de andere dataset bepaald. Dit resulteert in een zogenaamde verschilkaart tussen twee datasets. Aan de hand van de verschilkaart en de classificatie informatie wordt bepaald of een deel van een gebouw (niet) veranderd is, of dat het onmogelijk is om een verandering te detecteren. Een gebouw is veranderd indien er een aantal laserpunten zijn die een bepaald verschil vertonen in de verschilkaart én die als gebouw geclassificeerd zijn, dus ook punten op muren, daken en dakkapellen. De gevonden veranderingen worden nader geïnspecteerd en de verder geclassificeerd om aan te geven wat voor soort verandering er heeft plaatsgevonden. De verandering worden vastgelegd als kaartobject, inclusief grootte, locatie en aard van de verandering. Onze methode kan 80-90% van alle daadwerkelijke veranderingen opsporen, waarvan de helft relevant is in praktijk. De resultaten van de veranderingsdetectie worden beschreven en geanalyseerd in hoofdstuk 6.

Tenslotte worden in hoofdstuk 7 conclusies getrokken over de classificatie en de veranderingsdetectie. Ook worden beperkingen aan de voorgestelde aanpak besproken en aanbevelingen gegeven hoe beperkingen van deze methode kunnen worden opgelost.

ITC Dissertation List

http://www.itc.nl/research/phd/phd_graduates.aspx