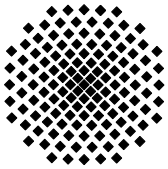


Universität Stuttgart

master thesis

Samer Karam

3D-Rekonstruktion von Gebäuden unterschiedlicher Höhenstufen aus LiDAR Befliegungsdaten

Tutor: Prof. Dr.-Ing. Peter Reinartz (DLR, IMF-PBA)
Examiner: Apl. Prof. Dr.-Ing. Norbert Haala
(Universität Stuttgart, IFP)

University of Stuttgart
Faculty of Aerospace Engineering and Geodesy
Department of Geomatics Engineering
Institute for Photogrammetry

University of Stuttgart
Germany

Master Of Science Thesis:

3D-Building Reconstruction with Different Height Levels from Airborne LiDAR Data

Samer Karam

(MSc. GEOENGINE Student, Stuttgart University)

In Cooperation with:

DLR

German Aerospace Center (DLR)
Earth Observation Center (EOC)
Photogrammetry and Image Analysis (IMF-PBA)

Supervisors: Dr.-Ing. Tahmineh Partovi (DLR, IMF-PBA)
Prof. Dr.-Ing. Peter Reinartz (DLR, IMF-PBA)

Examiner: Apl. Prof. Dr.-Ing. Norbert Haala (Universität Stuttgart, IFP)

2016

Abstract

More than 50% of the world population inhabit in urban or suburban areas, thus detailed and up-to-date building information is of great importance to every resident, government agencies, and private companies (utilities, real estates and etc.). 3D city models become increasingly important and necessary for supporting numerous applications such as crisis management, urban planning and so on (Verma u. a. [2006]). As LiDAR (light detection and ranging) considers a superior technology for 3D data acquisition from Earth's surface and delivers point clouds with high quality. This research is conducted by merely utilizing LiDAR data of an interested area without another data sources such as a cadastral map or photogrammetric aerial images. In this thesis, a developed automated data-driven approach is proposed 3D building models from airborne LiDAR data starting from segmentation of the point cloud, extracting roof patches and ending with regularization of the patches boundary to extract 3D model. Unlike photogrammetry, it is well known that LiDAR data does not record exact position of the edges of buildings. So the first challenge in this research is to extract a relatively accurate building boundaries from 3D point clouds LiDAR data and obtained results are evaluated by comparing with footprints in the cadastral map. The second challenge is to deal with step edges among the adjacent flat roof patches even with the small height differences and solve overlapping problem in 3D space.

For evaluation, the methodology is applied on another point cloud with different resolution. Also the extracted building polygons are compared with reference building polygons from cadastral map using a PoLiS metric based on polygon comparison (Avbelj u. a. [2015]). In addition, the area of individual roof patches and the area of whole footprints are computed and compared with their correspondences from cadastral map. Furthermore, RMSE distances (root mean square estimate) from raw LiDAR point clouds to the estimated 3D roof planes are computed.

Acknowledgements

I am grateful to the God for the good health and wellbeing that were necessary to complete my research.

I would like to express my very great appreciation to Dr. Tahmineh Partovi , my research supervisor from DLR, for her valuable and constructive suggestions during the planning and development of this research work.

I would like to express my deep gratitude to Prof. Peter Reinartz, my research supervisor from DLR, for his enthusiastic encouragement and useful critiques of this research work.

My sincere thanks goes to Prof. Norbert Haala, my research supervisor from Stuttgart University, for his insightful comments, guidance to finalize this research and writing this thesis.

I would also like to thank the staff and lecturers of the following organizations for the unceasing encouragement, academic support and experience which enable me to reach master degree:
Aleppo University
Stuttgart University
German Aerospace Center(DLR)

I am extremely thankful and indebted to my close friends for their appreciation of the pressure of work through my research and mitigate me by their cheerful spirit. And my special thanks are extended to new persons in my life.

Last but not the least, I would like to thank my family: my parents, brothers and sisters for supporting me spiritually throughout.

Contents

Chapter 1

Introduction

1.1 Motivation

The way of representing the earth surface has changed in the last years. Classical 2D maps have turned into 3D realistic representations of rural and urban environment. Since cities are made up of dense mixtures of artificial and natural urban features with immense varieties in size, shape, compositions and arrangements (Mesev [2003]), reconstruction of the buildings, are the most important objects in 3D city models and repeatedly appearing topic in photogrammetric research activities.

The relevance and importance of the corresponding research stems from the great benefits of 3D city models applications which have been exploited in the last decades. 3D city models have been increasingly applied in the form of data source, 3D visualizations or communication method in several fields such as archaeological reconstruction, urban planning and city management, architecture, tourism, civil engineering, mobile telecommunication, energy supply, navigation, environmental simulation, disaster management, game industry and many of other important fields. In addition there is a wide range of applications such as sustainable urban development, damage assessment, solar radiation potential assessment, water flow modeling, city climate studies, map updating, , land monitoring, pollutant diffusion, virtual tour, navigation, gaming, environmental evaluations, surface analysis, architectural design and so on. Although aforementioned applications share the common demand for 3D information, their special requirements regarding accuracy, details, quality, actuality and interoperability differ considerably. Specially, this work is of utmost interest for me due to the significant role of 3D model in rebuilding the cities and archaeological sites during crisis. The situation of urban areas in my home country, Syria, provides an important example of the potential of this approach. Most of cities, such as Aleppo, one of the oldest cities in the world, are damaged. Yet there are no 3D models available to provide support in future rebuilding process or even in damage assessment operations.

Generally, there are two main methods in the automatic building reconstruction (Maas u. Vosselman [1999]). The first one is the model-driven method that deals with parametric building and

searches for the most appropriate model among group of basic model samples contained in a given library. The second method, which is the focus of this research, is data-driven method or so-called non-parametric modelling method. This method deals with generic models which are comprised of a series of building surfaces. Since the generic models are very complex and abstract, they can be classified to prismatic and polyhedral models (Förstner [1999]). The polyhedral model is the most important one. It assumes a building is bounded by planar surfaces. This assumption is true for the majority of actual buildings. (Huang u. a. [2013]) developed model-driven combined with data-driven approaches to reconstruct 3D building models from LiDAR points cloud. Based on a pre-defined primitive library, a generative statistical modelling is conducted to reconstruct roof models. Compared with systems using parametric models, a system using generic models is more difficult to implement.
Regardless of the aforementioned approaches, building reconstruction is composed of two steps (Tarsha-Kurdi u. a. [2007]): the first step is building roofs modelling and facades modelling.
Concerning the building roof modelling, different techniques have been evoked previously to detect the roof planes, like RANSAC, Hough-transform and region growing techniques. These techniques sometimes use complementary data from images or maps in addition to the building point clouds, either to improve the plane roof segmentation or detection, or to improve the quality of result 3D building models.
For the purpose of building facades modelling, two possibilities exist. The building contour polygon has to be detected either before segmenting the roof in planes, or after the building roof segmentation. In the first case, it is necessary to use line generalization algorithms which allow simplifying or segmenting the building contour polygon according to its facades such as Douglas-Peucker technique. In the second case, the building contour polygons are extracted automatically following to the roof segmentation. The difference between these two cases is that in the first one, one facade is presented by several vertical planes according to the number of their adjacent roof planes. Whereas in the second case, one facade is presented by only one plane, under the assumption that the facade was previously well filtered (noise attenuation).
The next step is the determination of the neighborhood relationship between the building roof planes using Voronoi diagram (Ameri u. Fritsch [2000]). According to (Rottensteiner u. Briese [2003]), the mutual relations between every two neighbors roof planes have to be determined (intersection, step edge or intersection and step edge together).

Building reconstruction can be distinguished to semi-automatic and automatic procedure depending on levels of user interactions in producing the model (Weidner u. Förstner [1995]). In spite of many researches in this field, fully automatic procedure is still challenging issue and many problems remain unsolved. Detailed building models are requested for video and computer games (Wang [2013]). Different methodologies of building models generation exist and are classified according to their level of detail (Haala u. Brenner [1999]) and (Kolbe u. a. [2005]). For increasing the reliability and range of applications of the building models, additional knowledge of buildings have to be involved into the reconstruction process (Dorninger u. Pfeifer [2008]). In (Alharthy[1] u. Bethel [2004]) planar roof facets are separated and consequently building models are extracted. LiDAR data are well fitted for automatic building reconstruction. In contrast to optical stereo imagery,

where stereo matching is required in order to produce 3D geometry, LiDAR data contains accurate 3D information (Haala u. Kada [2010]) and (Meng u. a. [2009]). (Arefi u. a. [2008]) proposed a projection based method for 3D model of the building from LiDAR data.With increasing the capacity of LiDAR sensor and the density of point clouds, efforts on 3D building reconstruction has set a focus on LiDAR data (Haala u. Kada [2010]) and (Geibel u. Stilla [2000]).

Kim u. Muller [2002] proposed a method for 3D reconstruction from IKONOS GEO stereo imagery and ITS application for object identification. They developed a new technique to define the building and tree extents using 3D height points where IKONOS provide potentially useful information for the identification of individual surface objects compared with previous satellite image sources (e.g. SPOT, Landsat).

Syed u. a. [2005] presented a method on semiautomatic 3D building model generation from LiDAR. LiDAR data are utilized to estimate the orientation and height of building roof faces. Outline of roof facets are extracted by utilizing building footprints from object-oriented classification of coincident high resolution imagery.

Tebourbi u. BELHADJ proposed a new automatic technique for 3D building extraction. They used object oriented approach which started by automatically extracting different building objects from IKONOS stereo images. They combined a morphological operator and texture- based segmentation and applied the combination on the original images.

Lee u. a. [2011] integrated high resolution satellite images and digital vector maps in order to reconstruct buildings automatically. A cross correlation matching method along the vertical line focusing on the IKONOS images deployed to recover building heights.

Kada u. Wichmann [2012] proposed a 3D building reconstruction methodology that is based on the notion of sub-surface growing as a means for point cloud segmentation of planar surfaces. In addition, boundary regularization was implemented using series of techniques starting by generation of alpha shapes of the segments and ending with a quadratic least squares adjustment.

Moreira u. a. [2013] presented a digital reconstruction of LoD1 and LoD2 building models obtained with commercial packages and different input data in order to assess the achieved geometric accuracy of the 3D building models. DSM data derived from a GeoEye-1 stereo-pair, an aerial blocks and LiDAR.

Partovi u. a. [2015] proposed a methodology for 3D building reconstruction from Digital Surface Models (DSM) of stereo image matching of the space borne satellite data which cover larger areas than LiDAR datasets in one step data acquisition and also can be used for remote regions. In their approach, both top-down and bottom-up methods are utilized to find building roof models which exhibit the optimum fit to the point clouds of the DSM. In the bottom up step of this hybrid method, the building mask and roof components such as ridge lines are extracted. In addition, in order to reduce the computational complexity and search space, roofs are classified to pitched and flat roofs as well. Ridge lines are utilized to estimate the roof primitives from a building library including width, length, positions and orientation. Thereafter, a top-down approach based on Markov Chain Monte Carlo and simulated annealing is applied to optimize roof parameters in an iterative manner by

stochastic sampling and minimizing the average of euclidean distance between point clouds data and model surface as fitness function.

It should be noted that buildings that have only planar roofs are a quite common assumption. Since the parallel planar roofs with small height differences are difficult to segment, in this research a data of different types of flat roof buildings is used such as (L-shape, T-shape and even more complex shape of flat roof buildings). Moreover, other types of buildings are also considered such as (Gable roof and Hipped roof). Since the traditional manually building extraction from close range imagery is highly labor-intensive, time consuming and very expensive. This research looks into the development of a methodology for 3D building reconstruction using source of data that can cover a large area in short time, such as LiDAR. LiDAR technology is able to provide 3D point clouds data directly which is preliminary step for the derivation of 3D building models. Furthermore, LIDAR research community is becoming very active in building reconstruction (Maas u. Vosselman [1999], Maas [1999a], Maas [1999b], Maas [1999c], Stamos u. Allen [2000], Alharthy u. Bethel [2002], Haala u. Hahn [1995] and Haala u. Kada [2010]). Many research works on building reconstruction begin with the process of a DSM, which is either obtained from LIDAR points or imagery data. One of the challenges in this research is that building reconstruction process is applied directly on the LiDAR point clouds without the need to produce DSM or use a complementary data like ground plans.

1.2 Objectives:

1) The aim is to develop a methodology to automatically reconstruct accurate 3D model of building with different height levels from sparse point clouds, such as LiDAR data, taking into account resolution, cost and speed of the process.

2) Improving the reliability and geometric accuracy of 3D modeling based on LiDAR as alone source of data without using any complementary data.

3) Improving 3D building reconstruction, as buildings are the key features in 3D city models.

4) Testing applicability of the developed methodology for 3D model generation of different types of building using LiDAR data of Munich.

5) Developing the connectivity of workflow stages to overcome some common problems such as gaps in the LiDAR point clouds and the fact that LiDAR does not record the edges properly.

1.3 Thesis Outline:

Thesis consists of five chapters:

Chapter 1 contains an introduction to the thesis including motivation and objectives of the research.

Chapter 2 contains a literature review of point cloud and reconstruction methods. It begins with a definition, history and principle of LiDAR technology. Moreover a comparison between LiDAR and RADAR technologies is included. Then the data-driven as reconstruction methodology in the research, is explained including definition of the common segmentation techniques such as RANSAC, region growing. Then brief definition of model-driven as another reconstruction method is presented.

Chapter 3 contains the proposed methodology in the research starting from pre-processing of the LiDAR data and ending with 3D modelling.

Chapter 4 contains test and evaluation of the proposed methodology using different methods.

Chapter 5 contains conclusion of the thesis including some remarks and recommendations for the future work.

Chapter 2

Literature Review

2.1 Point Cloud

A point cloud is a set of data points in some coordinate system. In a 3D coordinate system, these points are usually defined by X, Y, and Z coordinates, and often are intended to represent the external surface of an object. Point clouds may be captured directly by 3D scanners or generated from images matching figure 2.1.

Scanners are devices that measure a large number of points on an object's surface, and often output a point cloud as a data file. The point cloud represents the set of points that the device has measured.

Automatic generation of high-quality, dense point clouds from digital images by image matching is a recent, cutting-edge step forward in digital photogrammetric technology. (Rosnell u. Honkavaara [2012]) generate point clouds by image matching using aerial image data collected by quadrocopter type micro unmanned aerial vehicle (UAV) imaging systems. One of the most exciting new software technologies is ENVI. It is able to generate 3D point-clouds from spaceborne EO/IR imaging platforms via multi-ray photogrammetry techniques that involve feature detector pixel correlation and dense image matching.

Point clouds are used for many purposes, including to create 3D CAD models for manufactured parts, metrology/quality inspection, and a multitude of visualization, animation, rendering and mass customization applications. However, raw point clouds resulting from the measurement process are not clean. Outliers and noise caused by errors in the measurement or false matching. So point cloud has to be processed firstly because a well-prepared point cloud leads to strong time saving in the further surface editing/modeling processes. Point clouds are usually converted to polygon mesh or triangle mesh models, NURBS surface models, or CAD models through a process commonly referred to as surface reconstruction.

There are many techniques for converting a point cloud to a 3D surface. Some approaches, like Delaunay triangulation, alpha shapes, and ball pivoting, build a network of triangles over the existing vertices of the point cloud, while other approaches convert the point cloud into a volumetric distance field and reconstruct the implicit surface so defined through a marching cubes algorithm.

In geographic information systems, point clouds are one of the sources used to make digital elevation model of the terrain (Agarwal u. a. [2006]). They are also used to generate 3D models of urban environments (Hammoudi u. a. [2010]).

Meshing Point Clouds is One of the most requested tasks when managing 3D scanning data is the conversion of point clouds into more practical triangular meshes. Meshes can be created from point cloud by grouping the cloud points. Those meshes can use as stand-alone 3D objects, for example, in 3D modeling. You can also use these meshes to quickly project the 2D sequence onto the mesh using the Project3D node.

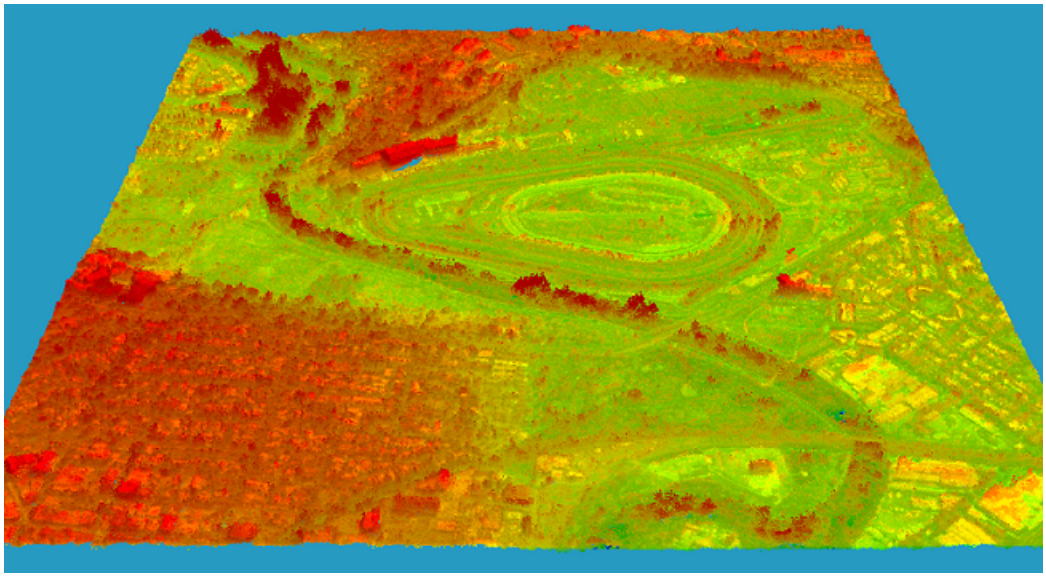


Figure 2.1 – 3D point-cloud generated by ENVI Photogrammetry Module

In this research, the used point clouds are captured from Airborne LiDAR. LiDAR (Light Detection and Ranging) is a superior technology of 3D data acquisition from Earth's surface with high precision and density. It is an active sensor mounted on an aircraft helicopter or plane. LiDAR has been used extensively for atmospheric research and meteorology due to its excellent resolution. It gathers laser measure distance, computer, high-precision clock, data storage and management systems, GPS (Global Positioning System) and INS(Inertial Navigation System) in the integral whole. Today's airborne laser scanning technology can produce dense 3D data with high accuracy called point cloud, which is an eligible data source to reconstruct 2D building outlines or even 3D building models (Albers u. a. [2016]) Figure 2.2.

One of the most recent initiatives in the areas of point cloud perception is PCL (Point Cloud Library). It is fully templated modern C++ library for 3D point cloud processing uses SSE optimiza-

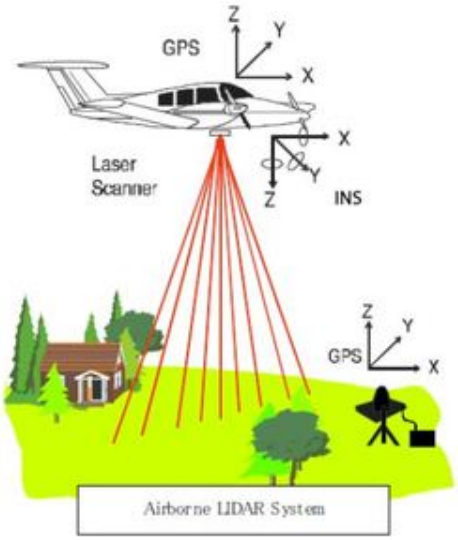


Figure 2.2 – Airborne LiDAR (USACE, 2002)

tions (Eigen backend) for fast computations on modern CPUs. PCL presents an advanced and extensive approach to the subject of 3D perception, and it's meant to provide support for all the common 3D building blocks that applications need. The library contains state-of the art algorithms for: filtering, feature estimation, surface reconstruction, registration, model fitting and segmentation. PCL is supported by an international community of robotics and perception researchers. In this research, PCL is used to help in processing 3D point clouds.

2.2 Surface Reconstruction

Reconstructing 3D surfaces from point samples is a well studied problem in computer graphics. It allows fitting of scanned data, filling of surface holes, and remeshing of existing models. The reconstruction of surfaces from oriented points has a number of difficulties in practice. The point sampling is often nonuniform. The positions and normals are generally noisy due to sampling inaccuracy and scan misregistration. And, accessibility constraints during scanning may leave some surface regions devoid of data. Given these challenges, reconstruction methods attempt to infer the topology of the unknown surface, accurately fit(but not over fit) the noisy data, and fill holes reasonably.

Several approaches are based on combinatorial structures, such as Delaunay triangulations (e.g. (Boissonnat [1984] and Kolluri u. a. [2004]), alpha shapes (Edelsbrunner u. Mücke [1994] and Bernardini u. a. [1999]), or Voronoi diagrams (Amenta u. a. [1998]). These schemes typically create a triangle mesh that interpolates all or a most of the points.

In the presence of noisy data, the resulting surface is often jagged, and is therefore smoothed (Kolluri u. a. [2004]) or refit to the points (Bajaj u. a. [1995]) in subsequent processing.

Other schemes directly reconstruct an approximating surface, typically represented in implicit form. A different approach is to form point neighborhoods by adaptively subdividing space, for example with an adaptive octree. Blending is possible over an octree structure using a multilevel partition of unity, and the type of local implicit patch within each octree node can be selected heuristically (Ohtake u. a. [2005]).

2.3 Reconstruction Methods

2.3.1 Model-driven Method

The model-driven approach searches for the most appropriate model among basic building models included in a library. Those simple buildings are so called primitive buildings which can be described by a set of parameters (Tarsha-Kurdi u. a. [2007]). In other words, model-driven approach, unlike data-driven approach, employs predefined parametric building models to reconstruct the blocks of point clouds or the DSM (Zheng u. Weng [2015]). The library contains the most probable roof shapes in a parametric form, such as shed, gabled, hipped and flat roofs (Haala u. Brenner [1999], Kada u. McKinley [2009], Zhang u. a. [2012] and Vosselman u. a. [2001]). This approach is so called model-driven approach because it starts from a model as a hypothesis, and it uses data to verify the model. This reconstruction plan is easy to understand and to implement, but it can only handle simple building models such as flat-roof and gable buildings. While in reality buildings appear in a variety of forms consequently model-driven approach cannot model very complex buildings (Ma [2004]).

In model-driven approach generally there are two types of parameters that are used to describe the building: parameters describe the building footprint (position, orientation and dimensions) and parameters describe the building roof plane equations. Based on these two parameter sets, the 3D building model can be constructed (Tarsha-Kurdi u. a. [2007]).

Model-driven methods have many advantages (Zheng u. Weng [2015]). For instance, the combination of a few primitives is much simpler than the organization of a bunch of facets (Kada u. McKinley [2009]). A classification accuracy of 95% was achieved by (Henn u. a. [2013]) who approached model selection using supervised classification. However, many types of roofs were not included such as intersecting roofs and half-hipped roofs. (Lafarge u. a. [2007] and Huang u. a. [2013]) both presented a method to reconstruct buildings from a DSM. They firstly decomposed the 2D building footprints, and then placed the 3D blocks on the 2D supports using a Gibbs model that controlled both the block assemblage and the fitting to the data. A Bayesian decision found the best fit roof primitives in the predefined library to represent the given blocks of point clouds using a Markov Chain Monte Carlo sampler associated with original proposition kernels (Zheng u. Weng [2015]). (Arefi u. a. [2010]) reconstructed building roofs based on the directions of each extracted ridge line fitted from the local maxima of the pixels using RANSAC.

2.3.2 Data-driven Method

2.3.2.1 Definition

The data-driven method is a comprehensive approach for automated determination of 3D building. It models a primitive or a complex building regardless of its predefined form using LiDAR points as initial data (Tarsha-Kurdi u. a. [2007]). It analyzes the building point cloud as a unity, without relating it to a set of parameters. This method uses series of varyingly complex operations allows to generate an unspecified 3D building models. It is starting mainly from the laser data such as (3D/2D Hough-transform, Random Sample Consensus (RANSAC), Region Growing (RG), Douglas-Peucker technique, etc) to reconstruct a complex building.

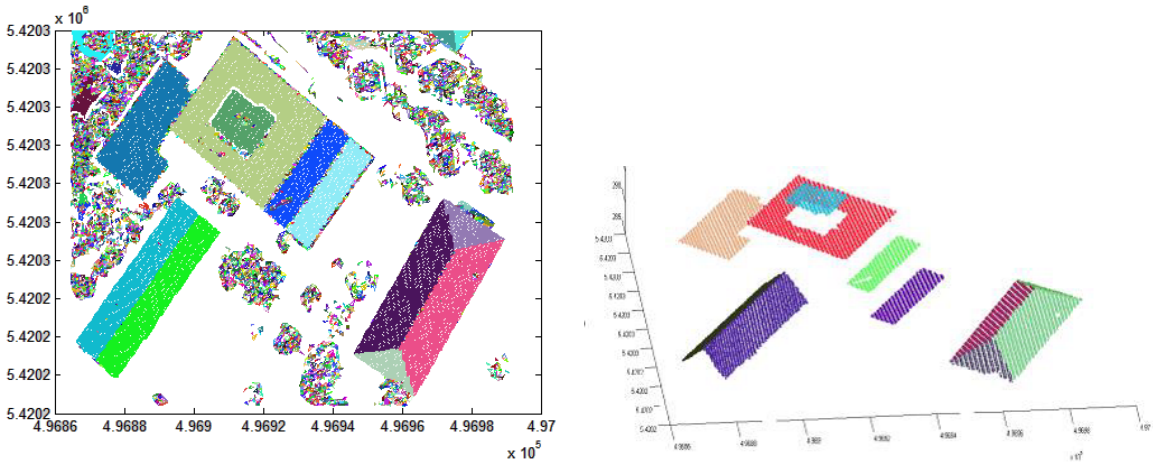
Sampath u. Shan [2010] used data-driven approach to segment the LiDAR points to planar and non-planar planes using eigenvalues of the covariance matrix in a small neighborhood. Then, the normal vectors of planar points are clustered by fuzzy k-means clustering. Afterwards, an adjacency matrix is considered to obtain the breaklines and roof vertices of corresponding planes. This method is used for reconstruction of moderately complex buildings.

For the data-driven strategies, Buildings in the real world can be represented as a set of primitives, regardless of whether they are planar facets or curved facets. Those facets are represented by the blocks of point clouds or the digital surface model (DSM). The roof planes are determined by segmenting the completed point clouds into different parts using segmentation algorithms.

2.3.2.2 Definition of Segmentation

Segmentation is the most fundamental and broader concept in building reconstruction process. It leads to divide a given data sets (point clouds) into non-overlap homogeneous regions. It collects points or pixels that belong to the same feature depending on similarity properties such as proximity, planarity and curvature.
During the last years, variation of methods and algorithms proposed to perform segmentation of the point clouds. For instance, (Sajadian u. Arefi [2014]) extracted and removed the ground and the vegetation candidate points from the original data. Subsequently, the segmentation process is conducted by triangulation of the remaining points. The objectives of aforementioned segmentation procedure is to: firstly detecting remained non-building points by removing some triangles which do not connect the edges between different objects or multi-layer structures. Secondly, labelling the points which belong to the same plane, if they are contained in triangles having the same normal vector direction. Thirdly extraction of building planes points. Segmentation results are shown in Figure 2.3.

In segmentation algorithms, a maximum distance threshold and minimum segments size are set to decide whether points belong to the same segment or not (Elberink [2008]) described segmentation



(a) Segmented triangles (b) Building segments points

Figure 2.3 – Segmentation of Point Cloud (Sajadian u. Arefi [2014]).

parameters based on the lower point density parts. (Milde u. a. [2008]) extracted the planes using RANSAC and subsequently performed region growing technique to obtain connected regions and constructing region adjacency graph(RAG) for the adjacency relations of each segmented region. While most region growing segmentation techniques assume roof segments to be planar, a robust TIN-based RG technique is designed to allow any continuous shape (Orthuber u. Avbelj [2015]) proposed method assigned one segment label to each triangle of the TIN in contrast to most RG techniques and (Dorninger u. Pfeifer [2008], Elberink u. Vosselman [2009], Sampath u. Shan [2010] and Sun u. Salvaggio [2013]) assigned one segment label to each LiDAR point.

Segmentation can be initialized by hierarchical clustering (Peter Dorninger and Norbert Pfeifer, 2008). Point clouds representing a single building can be decomposed into segments which describe planar patches where it is assumed that points belonging to the same planar region have similar local regression planes. See Figure 2.4 which demonstrates the segmentation process applied for point clouds of a single building.

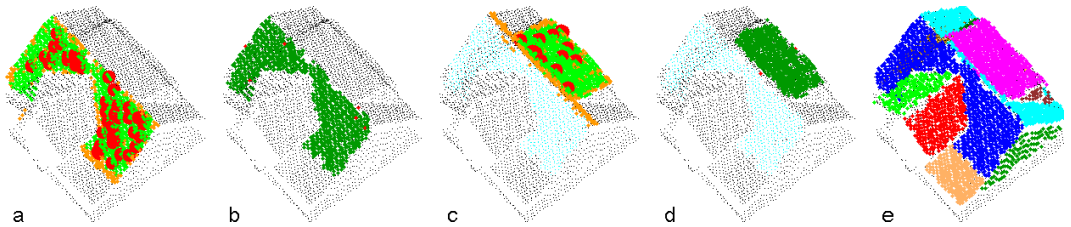


Figure 2.4 – a) and d) the determination of the first two segments is shown. a) and c) the seed clusters are emphasized (large red dots), as well as the points accepted by the object space criterion (orange) and the feature space criterion (green). b) and d) Points, finally assigned to the planar segments are shown in dark green. e) final segmentation results. (Dorninger u. Pfeifer [2008])

Segmentation algorithm can also be applied to an image to partition it into a number of homogeneous regions that correspond to surfaces in object space (Khoshelham [2005]). In this approach, the image is resampled with a smoothing kernel into different resolution layers and the segmenta-

tion algorithm is applied to each layer. In an earlier work, the watershed algorithm based on the color gradient was used to segment aerial images (Drauschke u. a. [2006]). This approach was also applied to façade images (Drauschke u. a. [2009]).

Due to the fact that the original laser scanner point cloud does not provide explicit information about the distribution of points; It is necessary to structure the irregular laser point clouds by defining the relationship between points and then search for the nearest neighboring points before processing them or to mesh the point cloud. As an adaptive approach for segmentation of 3D laser point cloud is kdtree data structure which was recommended by (Friedman u. a. [1977]) and it is a strategy for sorting and organizing a set of points. (Lari u. a. [2011]) used kdtree for the organization of 3D laser data and computation of the local point density and then adaptive cylinder neighborhood for each point is established based on the local point density and physical properties of the object surface .

The Hough transform is also employed to group independent straight line segments and calculate the parameters of every structure line (Lo u. Chenb [2012]), where they used topological elevation analysis (TEA) to detect structure lines instead of threshold-dependent concepts and predefined constraints. This study also used octree-based segmentation (Wang u. Tseng [2010]) to compare and evaluate the capabilities of TEA.

The performance of the segmentation process can be increased by using 2.5D grid representations instead of original points (Dorninger u. Pfeifer [2008]). Thus, only one height value can be assigned to an arbitrary pair of xy-coordinates. The advantages of 2.5D approaches are possible reduction of the amount of input data and the implicitly defined neighborhood by means of the grid representation.

Drauschke u. a. [2009] presented an approach to improve image segmentation for aerial imagery using multi-view analysis. Their approach relies on assumption that object surface in each region can be represented by a low order polynomial, and estimate their parameters.

2.3.2.3 Segmentation Techniques

Several segmentation techniques can be usually used in data-driven reconstruction approaches and during the segmentation process. Sometimes combination of two or more algorithms are used (Dorninger u. Pfeifer [2008]):

2.3.2.3.1 RANSAC (RANdom SAmpling Consensus) is an iterative method to estimate parameters of a mathematical model from a set of observed data which contains outliers. Therefore, it can also be interpreted as an outlier detection method. It is a non-deterministic algorithm in the sense that it produces a reasonable result only with a certain probability, with this probability increasing as more iterations are allowed. The algorithm was first published by Fischler and Bolles

at SRI International in 1981(Fischler u. Bolles [1981]). They used RANSAC to solve the Location Determination Problem (LDP), where the goal is to determine the points in the space that project onto an image into a set of landmarks with known locations.

The paradigm of RANSAC algorithm is devised in an iterative way (Zhang [2011]). In each iteration:

1. Randomly select a subset S (seed group) that contains n samples to compute the model M. The value of n is the minimum number of the selected samples required to estimate a model.
2. Compute the errors between the model and the rest of the data . The data whose errors to the model are smaller than a tolerance t are added in the Consensus Set of the current round, which is called S*.
3. Set a threshold d for the number of the data in S* to determine whether the current S* is qualified for deriving model. Since the outliers are assumed much fewer than the inliers, any S* containing outliers should be sufficiently smaller than that does not. Hence, the S* is viewed as a reliable inliers set as long as it is sufficiently large.
4. When S* is accepted, the model parameters are computed by solving the overdetermined linear system composed by S*. This is in fact an ordinary least square problem that can be solved using Singular Value Decomposition (SVD) technique. The models derived from reliable consensus sets are evaluated by the sum of the errors from all the inliers. When the last iteration terminates, the model with the smallest error wins.

The 3 parameters need to be specified for the paradigm:

1) The error tolerance t could be set as one or two standard deviations plus the average errors from the subset S to the hypothesis model generated from S.
2)The maximum number of iterations depends on the a priori probability of sampling outliers over the data. Assume that $w \times 100\%$ of the data are outliers. The probability of sampling n inlier samples is $(1-w)^n$. Then we will have a probability of $(1-(1-w)^n)^k$ for the case that we haven't found any inliers after k times' iteration. We can find a sufficiently large k that can limits the probability of failure to be under 5%.

$$(1-(1-w)^n)^k \leqslant 1-95\% \tag{2.1}$$

$$\Longrightarrow k \geqslant \frac{\log(1-95\%)}{\log(1-(1-w)^n)} \tag{2.2}$$

3) The threshold d is specified by an a priori probability y that any given data is within the error tolerance of an incorrect hypothesis model. For each evaluating process given the hypothesis model generated by n samples, is expected very small. Hence the d can be obtained by solving the inequality (Fischler u. Bolles [1981]).

$$y^(d-n) \leqslant 1-95\% \tag{2.3}$$

Ameri u. Fritsch [2000] and Brenner [2000] used this technique for detecting the roof planes. So

planes are accepted or rejected based on a list of rules which present the possible relationships between planes and ground plan edges. (Tarsha-Kurdi u. a. [2008]) proposed extended RANSAC for harmonizing the mathematical aspect of the algorithm with the geometry of roof. It gives good results even for the low point density and complex buildings.

2.3.2.3.2 Hough Transform (HT) is another common method of detecting the shapes primitive. It was introduced in 1962 (VC [1962]) and firstly used to find lines in images a decade later (Duda u. Hart [1972]).(Albers u. a. [2016]) used HT to determine the main directions of the building and to extract line segments which are oriented accordingly.
The Hough transform is a feature extraction technique used in image analysis, computer vision, and digital image processing (Shapiro u. Stockman [2001]). It estimates the parameters of a shape from its points. The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure. This voting procedure is carried out in a parameter space, from which object candidates are obtained as local maxima in a so-called accumulator space that is explicitly constructed by the algorithm for computing the Hough transform. It can be used to detect lines, circles and other primitive shapes if their parametric equation is known. In principle, it works by mapping every point in the data to a manifold in the parameter space. This manifold describes all possible variants of the parametrized primitive. Making the parametrizing simpler or limit the parameter space speed up the algorithm. This is especially true for 3D shape detection, where for example to detect a plane using the plane equation ax+by+cz+d=0 requires 3D Hough space, which will quickly occupy large space of memory and performance since all possible planes in every transformed point clouds need to be examined. A plane can also be fitted based on normalized normal vectors using only two of the Euler angles and distance from origin, α, β and d. There is no need to the third Euler angle since the information when transforming around the axis in redundant (Hulik u. a. [2014]).
The advantages of Hough transform are that the concept is simple and the implementation is easy, but the computational cost of this technique is very high so the use of this technique is limited. (Rau u. Lin [2011]) use 2D Hough-transform to detect planes directly after projecting the 3D point cloud onto planes parallel and orthogonal to the principle building direction. With the 3D point cloud, the demand is increased for detecting 3D planes. In this context, the 2D Hough-transform has been extended to 3D (Vosselman u. a. [2001], DOIHARA u. SHIBASAKI and Overby u. a. [2004]). The principle of the 2D Hough-transform is the representation of a points set, defined initially in the Euclidian space, in another space. This transform allows detecting the points composing specific geometric primitives. Later, its principle has been extended to the extraction of other 3D geometric forms like cylinders (Rabbani u. Van Den Heuvel [2005]). The 3D Hough-transform uses a pure mathematical principle in order to detect the best planes from a 3D point cloud. That means that it looks for point sets which represent statistically the best planes without taking into account their signification in the building point cloud. In this context, the best plane does not mean the most probable plane calculated according to the least squares theory. But it means the plane containing the maximum number of points. Therefore, it detects perhaps a set of points which represents several roof planes or which belongs to several planes. Moreover, the 3D Hough-transform spends

a long time for calculating the matrix H and for detecting the peaks in it (Tarsha-Kurdi u. a. [2007]). An analytic comparison of both RANSAC and Hough algorithms, in terms of processing time and sensitivity to cloud characteristics, shows that despite the limitation encountered in both methods, RANSAC algorithm is still more efficient than Hough (Tarsha-Kurdi u. a. [2007]).

2.3.2.3.3 Region Growing (RG) is a simple region-based segmentation method, where the segment's form is not determined in advance such as surface fitting methods (RANSAC and Hough transform), but it depends on sequence modelling procedures. The process is iterated on, in the same manner as general data clustering algorithms (Kamdi u. Krishna [2012]). It classifies point clouds in their neighborhood contexts starting from a seed points and adding each unclassified seed point to the region's points, depending on several criteria:

- The local neighborhood of a point can be determined according to different distance measures in Euclidean or feature space like:
  - Distance-fixed threshold (Verma u. a. [2006]).
  - Grid-based window neighborhood (Alharthy u. a. [2004]).
  - Voronoi neighborhood (Awrangjeb u. Fraser [2014]and Rau u. Lin [2011]).
- RG criteria are thresholding based on similarity measures to decide whether the candidate point is added to a region or not. There are several region growing criteria based on local planes that are:
  - Angular difference of local normals of neighboring points (Lafarge u. Mallet [2012]).
  - Angular difference of the candidate point's local normal to the average normal of the region (Kada u. Wichmann [2012]).
  - Distance between a point and the region's plane (Kada u. Wichmann [2012] and Nurunnabi u. a. [2012].
  - Similarity of local plane parameters of neighboring points (Verma u. a. [2006]).
- RG seed points: can be determined randomly from the point clouds or by choosing 2Dspatial extreme points (Poullis u. You [2009]).

The advantages of RG are that the concept is similar HT and only need a small number of seed points to represent the property we want before grow the region. In addition, we can choose multiple criteria at the same time.

Alharthy[1] u. Bethel [2004] and Elaksher u. a. [2002] developed an algorithm that gathers together all pixels fitting a plane in raster data; (Rottensteiner u. Briese [2003]) extracts roof planes using seed regions and applies a region growing algorithm in a regularized DSM. Then, the homogeneity relationships between the neighboring points are evaluated by calculating the surface normals. (Orthuber u. Avbelj [2015]) propose robust TIN-based RG for roof segmentation. It is designed to allow any continuous shape unlike most of region growing techniques which assume roof segments

to be planar. Furthermore, one segment label is assigned to each triangle of the triangulation network (TIN) in contrast to most RG techniques (e.g. (Sampath u. Shan [2010]). The RG procedure iteratively starts at seed triangles defined by minimum Local Unevenness Factor.

2.3.2.3.4 Edge-Based Technique Edge-detection technique is a technique to detect edge pixels using Gradient, Laplacian or Canny filtering and then link those pixels to form contours at the end. Linking of edges, in a predefined neighborhood, depends on two criteria. The first one is the magnitude of the gradient and the second is the direction of the gradient vector. Since edges are important features in an image to separate regions, a large variety of edge detection algorithms have been developed for image segmentation in computer vision area (Shapiro u. Stockman [2001]). (Heath u. a. [1998]) demonstrate a proposed experimental strategy by comparing four well-known edge detectors: Canny, Nalwa–Binford, Sarkar–Boyer, and Sobel. (Jiang u. Bunke [1999]) presented a novel edge detection algorithm for range images based on a scan line approximation technique. LiDAR data are converted into range image, e.g. DSM (Digital Surface Model) to make it suitable to image edge-detection methods. The performance of segmentation is largely dependent on the edge detector. However, the operation of converting 3D point clouds to 2.5D range images inevitably causes information loss. For airborne LiDAR data, the overlapping surface such as multi-layer building roofs, bridges, and tree branches on top of roofs cause buildings and bridges either under segmented or wrongly classified. The point clouds obtained by terrestrial LiDAR are usually combined from the scans in several different positions, converting such kind of true 3D data into 2.5D would cause great loss of information (Wang u. Shan [2009]).

The edge based method can be preferable because it is usually less complex and the approach is similar to how humans segment images. It works well in images with good contrast between object and background. In contrast the method is sensitive to noise and robust edge linking is not trivial.

Since **data-driven** method is not restricted by a predefined catalog of buildings, all types of buildings can be reconstructed by this method. It is widely used in many applications and remains the only approach which treats the general case of unspecified building, in spite of the probable risks of obtaining disturbed models of unspecified buildings. Since the used LiDAR data is contaminated with noise and for all aforementioned reasons, data-driven method is used in this research.

Chapter 3

Proposed Methodology

3.1 Flowchart of Proposed Methodology

A methodology is proposed in this research to work directly on the LiDAR data as alone source of data without using any complementary data and reconstruct 3D model from it. It consists of sequential operations starting from filtering of the LiDAR data and going through segmentation process to collect points that belong to the same feature depending on similarity properties such as proximity,planarity and curvature. Then, extracting concave boundary that includes the outermost points of a building. Finally, the methodology is ending with extracting the footprint and 3D model of the building. The sequential operations in the methodology are clarified in the following flowchart 3.

3.2 Airborne LiDAR Data

The used data is captured by airborne LiDAR which Detect the terrain surface from the aircraft. The data is provided from " Landesamt für Digitalisierung, Breitband und Vermessung " München. The resolution of the data is approximately 5 pts/m2. The measuring points fall not only on the earth's surface, but also on the objects located there, e.g. trees or buildings. By means of suitable filter methods, the dot cloud is automatically divided into dot classes:

- Ground points
- Object points
- Points which are not attributable to the ground
- Building points

The load pulse points of the class ground points are used to calculate the digital terrain model (DGM). The digital surface model (DOM) is calculated from all points of the first pulse data set. All

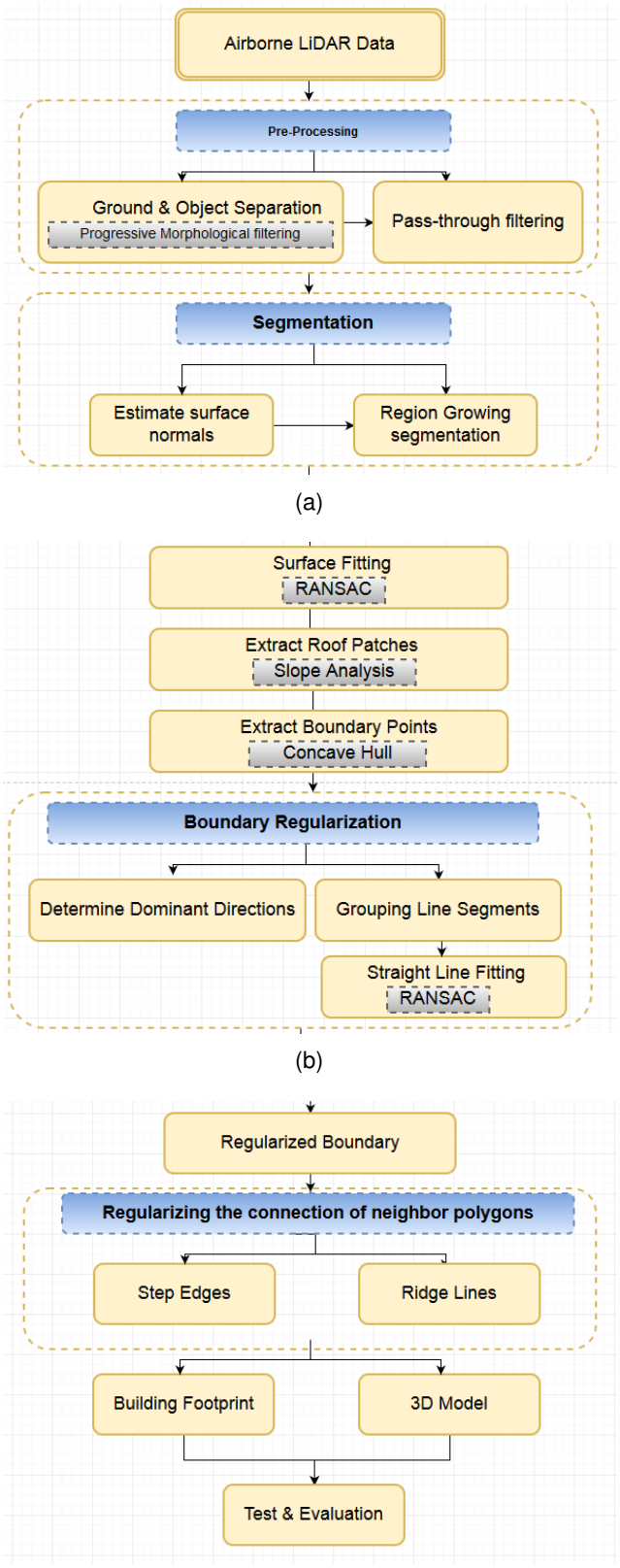


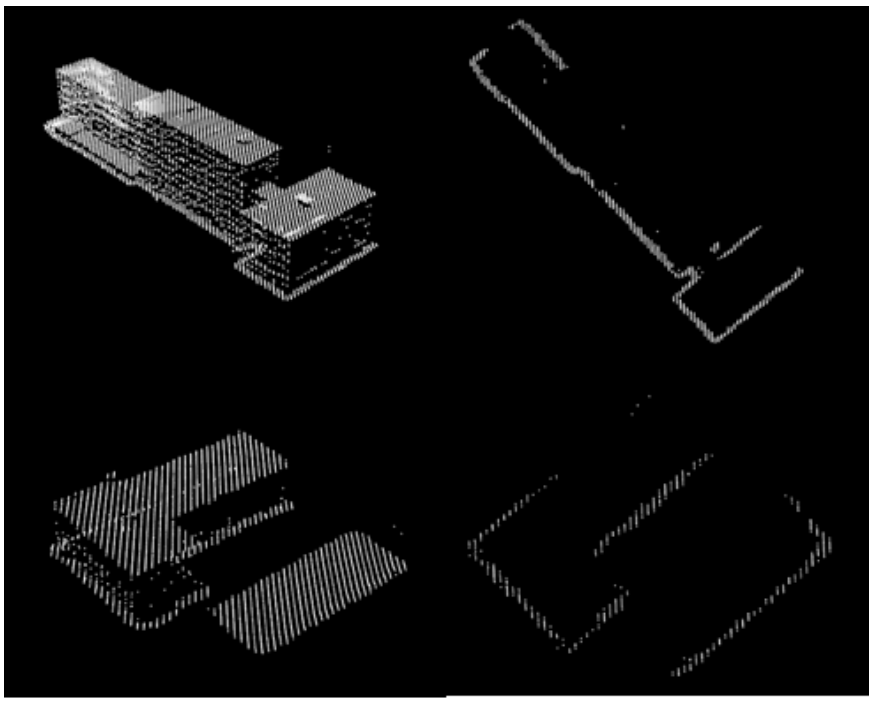
Figure 3.1 – Flowchart of Proposed Methodology.

load pulse laser points, which lie within a building outline in the floor map, are given the point class building point. They are the basis for the creation of 3D building models.

3.3 Pre-Processing

3.3.1 Pass-Through filtering

Pass-Through filter passes points in a cloud based on constraints for one particular coordinate of point coordinates. It is a simple filtering along a specified dimension. Where it Iterates through the entire input once and automatically filtering non-finite points and values that are either inside or outside a given user range. Here it is applied along z dimension of the input cloud. The resulting cloud, which will be removed, contains all points of input cloud that are finite and have z values in the given range 3.2.



(a) (b)

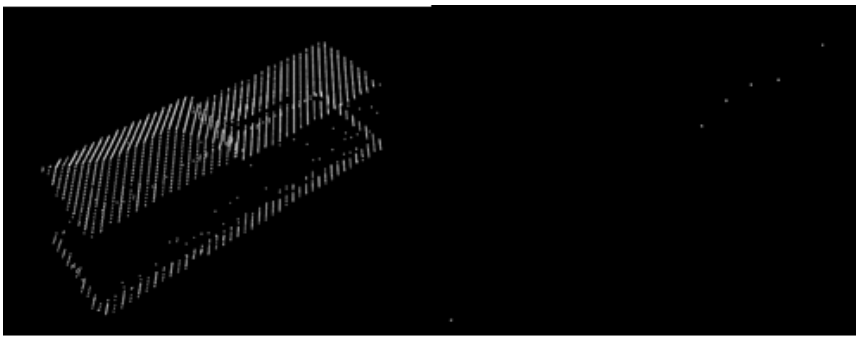
Figure 3.2 – a) Input point cloud b) filtered cloud contains the points that have been removed by Pass-Through filter

The method is not effective since it works in a given user range which need to be changed from building to another. In addition, the noise points may be not in the predefined range such as the case in 3.3 where most of ground points have height values out of the given range and consequently they did not filter out. Due to that, progressive morphological filtering has been used.

3.4 Ground & Object Separation

3.4.1 Progressive Morphological Filtering

Progressive morphological filter is used to segment the ground points. It is used here to separate object points, wall and roof points, from ground points. The commonly used filter to remove non-

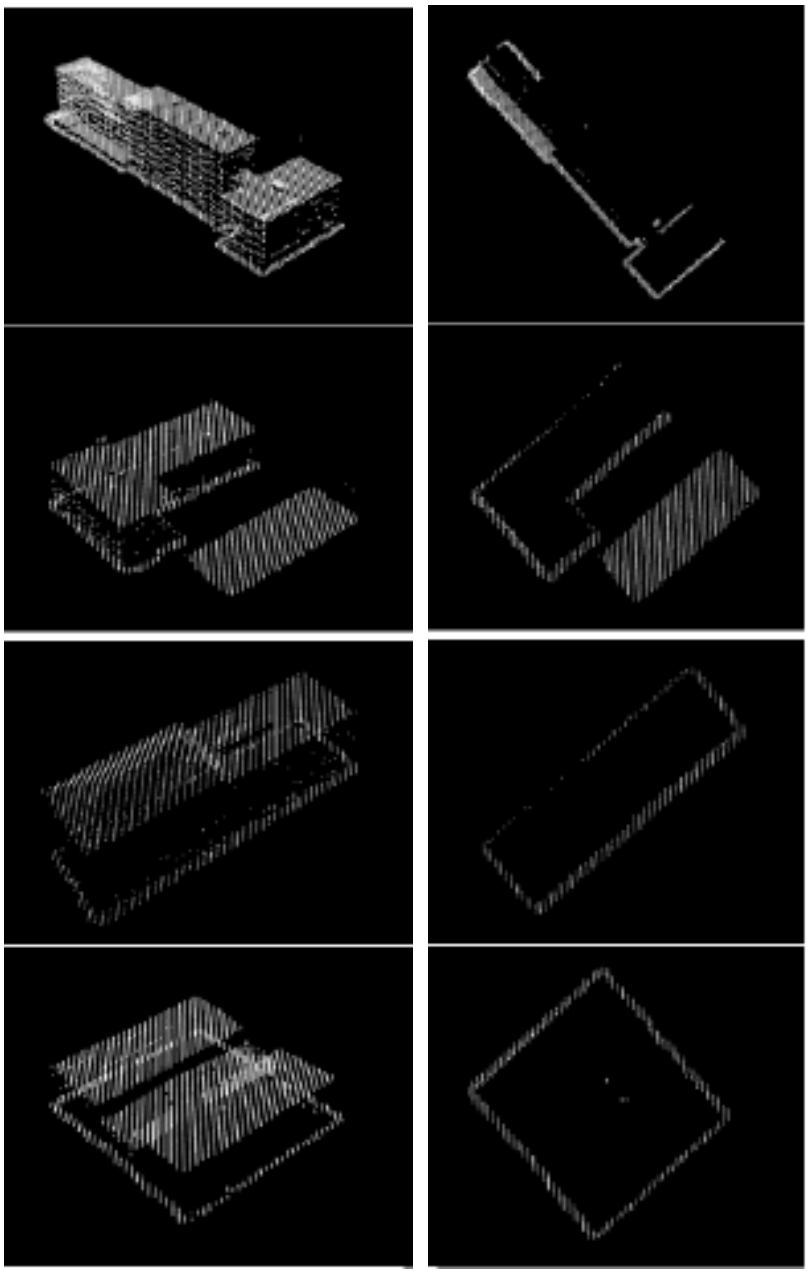


(a) (b)

Figure 3.3 – a) Input point cloud b) filtered cloud contains the points that have been removed by Pass-Through filter

ground objects is a mathematical morphological filter which is applied to agrayscale image (Kilian u. a. [1996]).
It composes of two fundamental operations, dilation and erosion which are employed to enlarge (dilate) or to reduce (erode) the size of features in binary images (Haralick u. a. [1987]). The filter was developed by (Zhang u. a. [2003]) to detect non-ground LIDAR measurements. The combination of erosion and dilation operations generates opening and closing operations that are employed to filter LIDAR data. The filter was developed to generate DTM, but in this research, I invested aforementioned filter to get rid of the ground points from the building. Thus, I have used it inversely. The results show that the filter can remove most of the non-ground points effectively figure 3.4.
The principle of progressive morphological filter 3.5 can be summarized in the following steps:

1) Generating minimum surface grid from unorganized LiDAR data by selecting the minimum elevation in each grid cell. Point coordinates xyz are stored in each grid cell.For each cell the value of nearest point measurement is assigned, if it does not contain any data.
2) The progressive morphological filter is applied to the grid surface. At the first iteration, the minimum elevation surface together with an initial filtering window size provide the inputs for the filter. In the next step, the filtered surface is obtained from the previous iteration and increased window size from Step 3) are used as input for the filter. The outputs of this step include:
a) More smoothed surface from the morphological filter.
b) The detected non-ground points based on the elevation difference threshold.
3) The elevation difference threshold is calculated with increasing the size of the filtered window. Steps 2 and 3 are repeated until becomes greater than a predefined threshold. The threshold is usually set to be slightly larger than the maximum building size which is in this research 50.
4) Finally generation of the ground points based on the dataset after nonground measurements have been removed. Since the output of the proposed approach is the indices of ground points, I used ExtractIndices filter to extract an object points from point clouds based on the resulted indices by a morphological filtering algorithm.



(a) Input point cloud (b) Filtered point clouds

Figure 3.4 – Progressive morphological filtering.

3.5 Segmentation

3.5.1 Surface Normal Estimation

Normal vector is one of the most important properties of any geometric surface. Estimating the surface normals of the given geometric surface is one of the basic steps of the following operations in this research since the used segmentation method is based on point normal vector.
It's usually insignificant to infer the direction of the normal at a specific point on the surface as the vector which is perpendicular to the surface in that point (Bogdan u. a. [2009]). However, since the

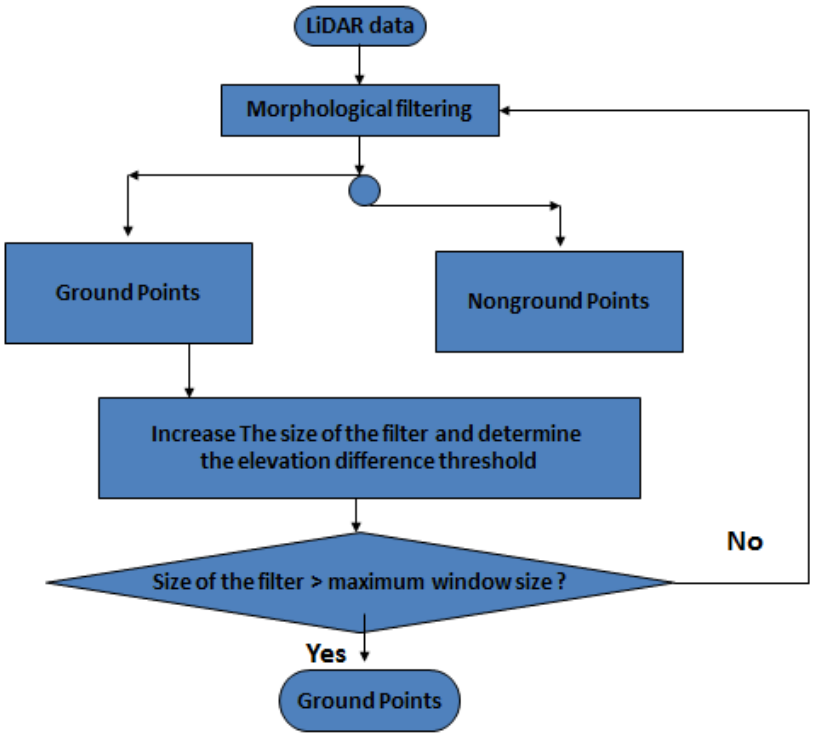


Figure 3.5 – Framework of Progressive Morphological filtering (Zhang u. a. [2003])

used point cloud represents a set of point of a real surface, there are two possibilities to estimate the normals:

- Using surface meshing techniques in order to obtain the underlying surface from the acquired point cloud dataset, and then use the result mesh to compute the surface normals.
- Directly compute the surface normals at each point in the cloud using approximations by estimating the normal of a plane tangent to the surface.

The second possibility, which depends on the approximation, is used to compute the normal vectors. The problem is reduced to an analysis of the eigenvectors and eigenvalues of a covariance matrix of point's nearest neighbors. The nearest neighbors of a query point are computed by fast kd-tree queries. A k-d tree is a space-partitioning data structure for organizing points in a k-dimensional space.
There are two types of queries:
1) Determining the neighbors of a query point within a sphere of radius r.
2) Determining the k neighbors of a query point (k-search) which is used here and it gives better results.

The applied method to compute normals can be formulated as following:

The problem of determining the point's normal on the surface is approximated by the problem of estimating the normal of a plane tangent to the surface, which in turn becomes a least-square plane fitting estimation problem. By considering the surrounding neighbors, the underlying sampled surface geometry can be captured 3.6. An analysis of the eigenvectors and eigenvalues, or

PCA – Principal Component Analysis, of a covariance matrix is needed and it is created from the surrounding point neighborhood of the query point which is so called here k-nearest neighbors. Choosing the right k is based on the level of detail that I have to capture from the input data figures 3.7 and 3.8. Moreover k factor plays a pivot role to automate the process without depending on the thresholds given by user. To better illustrate effects of k factor values, the figure 3.9 below presents the effects of selecting a small k versus a larger k. The left part of the figures depicts a reasonable well chosen scale factor, with estimated surface normals approximatively perpendicular for the two planar surfaces and small edges visible all across the table. If the scale factor however is too big (right part), and thus the set of neighbors is larger covering points from adjacent surfaces, the estimated point feature representations get distorted, with rotated surface normals at the edges of the two planar surfaces, and smeared edges and suppressed fine details.

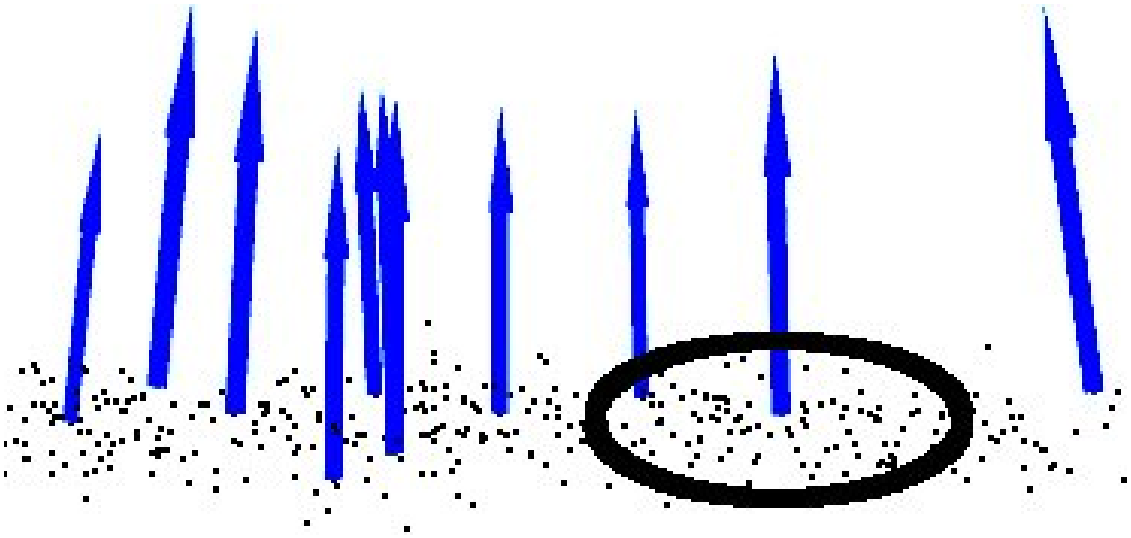


Figure 3.6 – Computing the surface normals at each point in the cloud using the nearest neighbors.

Pseudocode:

- Inputs:
  - Point cloud of 3D points: P={x,y,z}
- Surface Normal Estimation:
  - Select a set of 3D points $G \subseteq P$
  - A least-square plane fitting estimation of G points
  - Computing the normal of the plane $\vec{n}$

Empirically, I found out the right scale factor K=30 Sufficient to segment roof patches with different height level and it is tested on different types of roofs as it is clarified in figure 3.10. More specifically, for each point Pi , the covariance matrix C is formed as follows:

$$Cov_i = \frac{1}{k}.\sum (P_i - Centroid_i).(P_i - Centroid_i)\top, \quad Cov.\ \vec{v}_j = \lambda_j.\ \vec{v}_j, \quad j = \{0,1,2\} \tag{3.1}$$

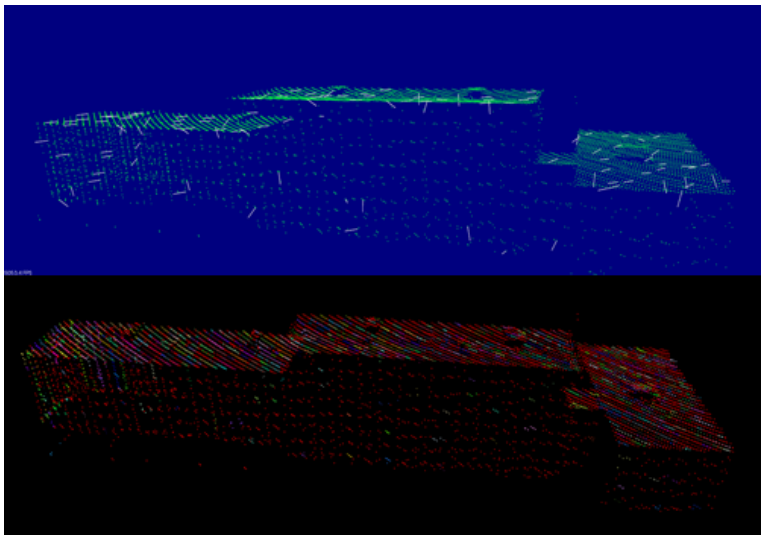


Figure 3.7 – The first photo represent normal vectors of building point cloud for K=5 and number of neighbors=6 while the second photo represent segmentation result with clear over-segmentation of the point cloud.

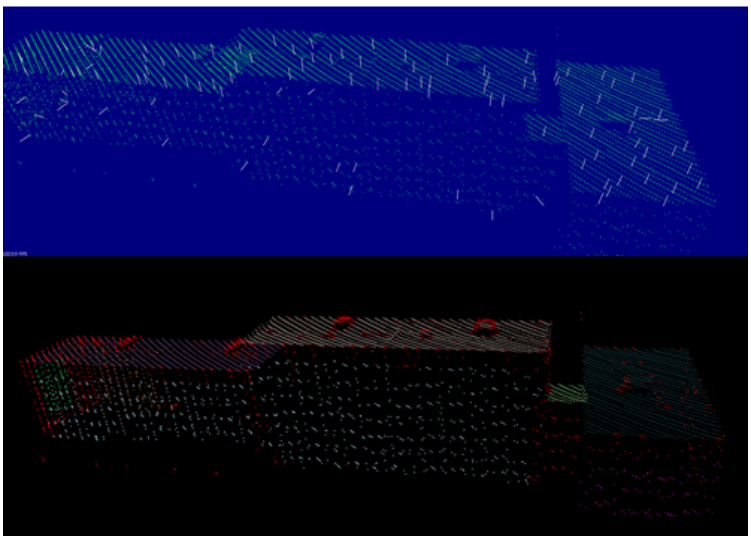


Figure 3.8 – The first photo represent normal vectors of building point cloud for K=20 and number of neighbors=30 while the second photo represent segmentation result of the point cloud.

Where *k* is the number of point neighbors considered in the neighborhood of *Pi* , *Centroid* represents the 3D centroid of the nearest neighbors, λ_j is the *jth* eigenvalue of the covariance matrix, and $\vec{v}_j$ is the *jth* eigenvector.

Firstly the point cloud is defined as the input for normal estimation, setting the Kd-tree as search tree and setting the k-search value. While the output contains the plane parameters curvature in pcd format file. The plane parameters holding the normal (a,b,c) on the first 3 coordinates xyz. Where the normal is the eigenvector corresponding to the smallest eigenvalue. The surface curvature can be described by looking at the deviation from a point p of the underlying tangent plane (Pauly u. a. [2002]). It is estimation of a point in a neighborhood as a relationship between the eigenvalues of the covariance matrix, as:

$$\sigma = \frac{\lambda_0}{\lambda_0 + \lambda_1 + \lambda_2}, \tag{3.2}$$

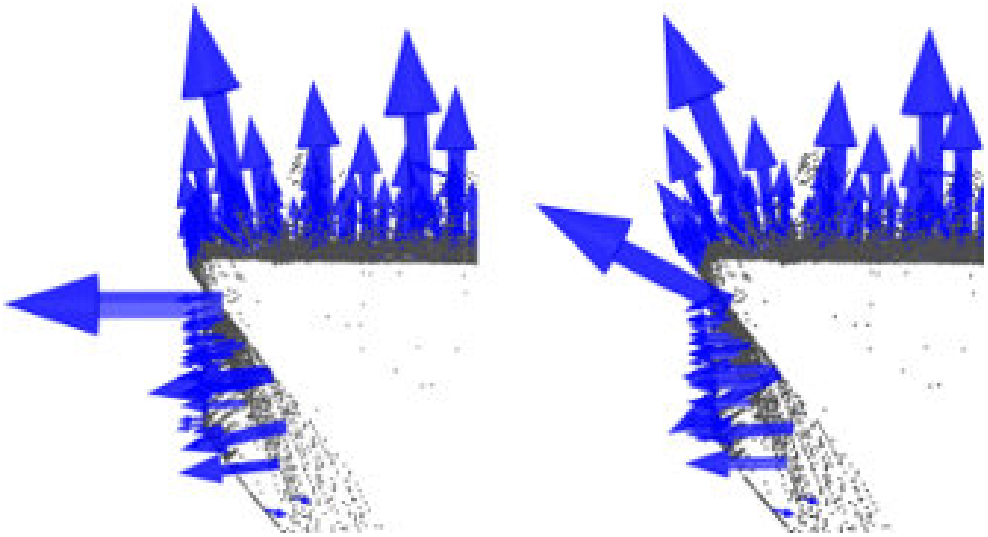


Figure 3.9 – Effects of k factor values. The left part depicts a reasonable well chosen scale factor, The left part depicts too big scale factor.

Where:
σ is the surface curvature.
$\lambda 0$, $\lambda 1$ and $\lambda 2$ are the eigenvalues of the covariance matrix.
$\lambda 0$ quantitatively describes the variation along the surface normal, i.e. estimates how much the points deviate from the tangent plane.
$\lambda 1$ and $\lambda 2$ describe the variation of the sampling distribution in the tangent plane and can thus be used to estimate local anisotropy.

3.5.2 Region Growing Segmentation

The purpose is to merge points that are close enough in terms of the smoothness constraint. The algorithm is performed by comparison of the angles between points normals. The strategy of the algorithm: is to firstly sorts the points by their curvature values. In order to reduce the total number of segments, the growth of the region begins from the flattest area. So the algorithm picks up the point that has the minimum curvature value (i.e. located in the flat area). The picked point is added to the set called seeds. The algorithm finds neighbors for each seed. Then, the neighbor point is tested for the angle between its normal and the normal of the current seed point. Comparing the angle with specific threshold and if it less then add it to the current region. After that, the neighbor is tested regarding the curvature value. If it is less than specific threshold, then add it to the seeds. Current seed is removed from the seeds.

The algorithm considered to grow the region when the seeds set becomes empty and the process is repeated again from the beginning.

Pseudocode

Inputs:

- Point cloud = {P}
- Point normals = {N}
- Point curvatures = {CV}
- Neighbour finding function NF
- Region List={L}
- Seeds List={S}
- Curvature threshold CV_{th}
- Angle threshold Q_{th}

initialization:

- Region List $L \leftarrow \phi$
- Available points list $\{A\} \leftarrow \{1, 2, ..., |P|\}$

Algorithm:

1. While {A} is not empty do following
   - Current Region $\{L_c\} \leftarrow \phi$
   - Current Seeds $\{S_c\} \leftarrow \phi$
   - Point with minimum curvature in $P_{min} \leftarrow \{A\}$
   - $\{S_c\} \leftarrow \{S_c\} \cup P_{min}$
   - $\{L_c\} \leftarrow \{L_c\} \cup P_{min}$
   - $\{A\} \leftarrow \{A\} \setminus P_{min}$
   - for i=0 to size $(\{S_c\})$ do
     (a) Find nearest neighbours of current seed point $\{B_c\} \leftarrow NF(S_c\{i\})$
     (b) for j=0 to size $(\{B_c\})$ do
     – Current neighbour point $P_j \leftarrow B_c\{j\}$
     – If $\{A\}$ contains P_j and $\cos^{-1}(|(N\{S_c\{i\}\}, N\{S_c\{j\}\})|) < Q_{th}$ then
     i. $\{L_c\} \leftarrow \{L_c\} \cup P_j$
     ii. $\{A\} \leftarrow \{A\} \setminus P_j$
     iii. If $CV\{P_j\} < CV_{th}$ then
     * $\{S_c\} \leftarrow \{S_c\} \cup P_j$

iv. end if

– end if

(c) end for

- end for
- Add current region to global segment list $\{L\} \leftarrow \{L\} \cup \{L_c\}$

2. end while

3. Return {L}

In this research, PCL library is utilized for segmentation procedure and it is modified to segment roof patches with different height levels effectively. Firstly, we need to perform initialization of RG algorithm by setting the type of the two input parameters, coordinates of the points (xyz) and their normals. So the algorithm receives filtered cloud that must be segmented and the computed normals. After that minimum and maximum cluster sizes are set. It means that after performing segmentation, all small clusters which are less than predefined minimum threshold segment size are discarded. The same thing for clusters which are more than predefined maximum threshold segment size. The used values for minimum and maximum are 45 and 'as much as possible' respectively. Then nearest neighbor algorithm is performed and the number of neighbours (k-search) to be set. The most important factors throughout segmentation procedure in this work is setting Smoothness and Curvature thresholds. The first threshold is representing angle in radian unit that will be used as permissible range of acceptance for the normals deviation. If the deviation between points normals angles is less than smoothness threshold, they would consider in the same cluster. The second one is responsible for curvature threshold. If two points have a small normals deviation then the disparity between their curvatures is tested. And if this value is less than curvature threshold then the algorithm will continue the growth of the cluster using new added point. Finally, RG segmentation algorithm is launching and the output of this algorithm is the set of clusters, where each cluster is a set of points that are considered to be a part of the same smooth surface. The figure 3.10 below illustrates the estimated surface normal and the segmentation results of buildings from different roof types with different height levels.

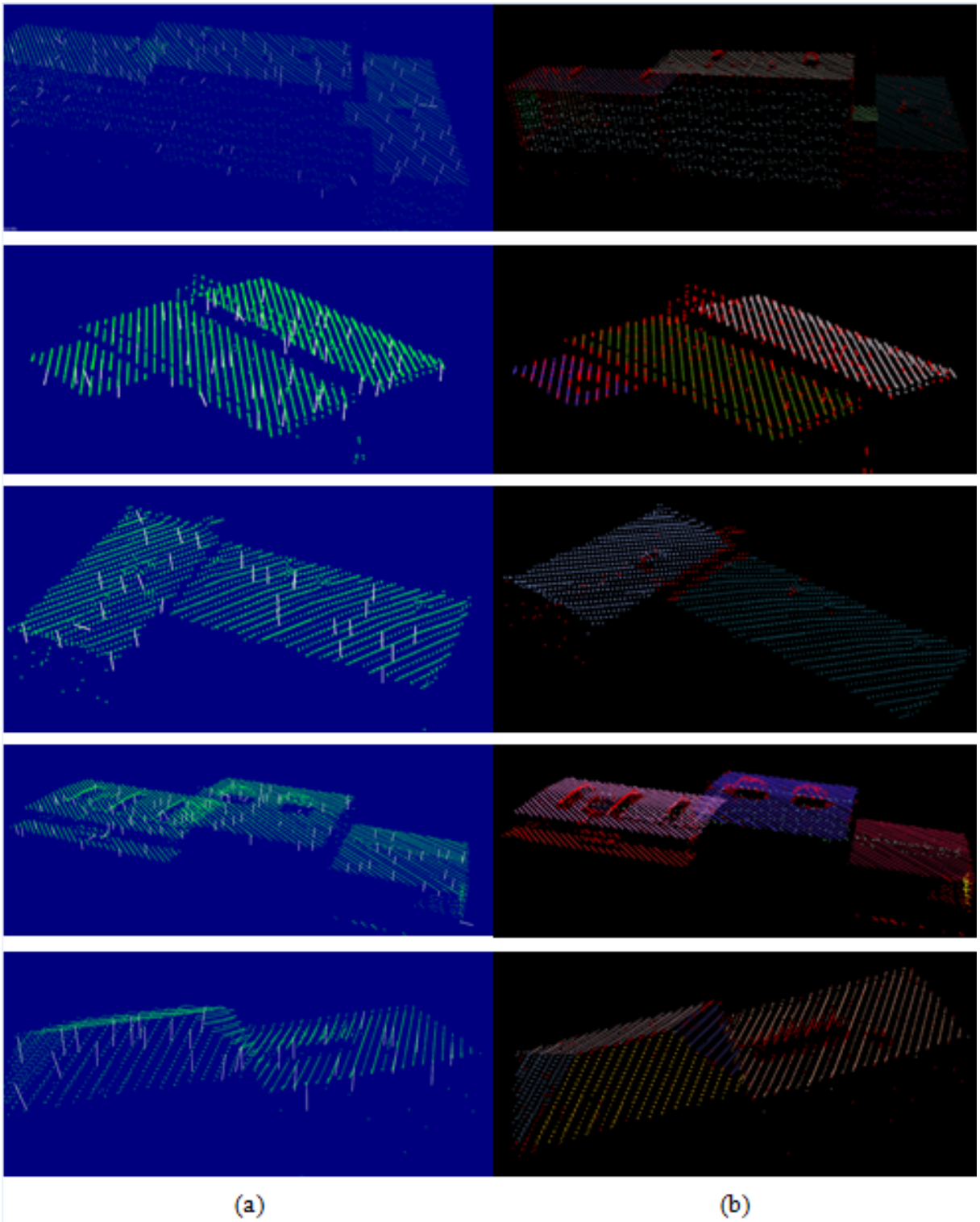


Figure 3.10 – (a) Estimated surface normals of the input point cloud. (b) Segmented building LiDAR points represented in colored cloud where each cluster has its own color. Red points represent the points that belong to the rejected clusters due to their small segments size.

The algorithm was tested also for combined building of different roof types such as flat and hipped roof as it is shown in last row of figure 3.10. Also it is applied for gable roof figure 3.11.

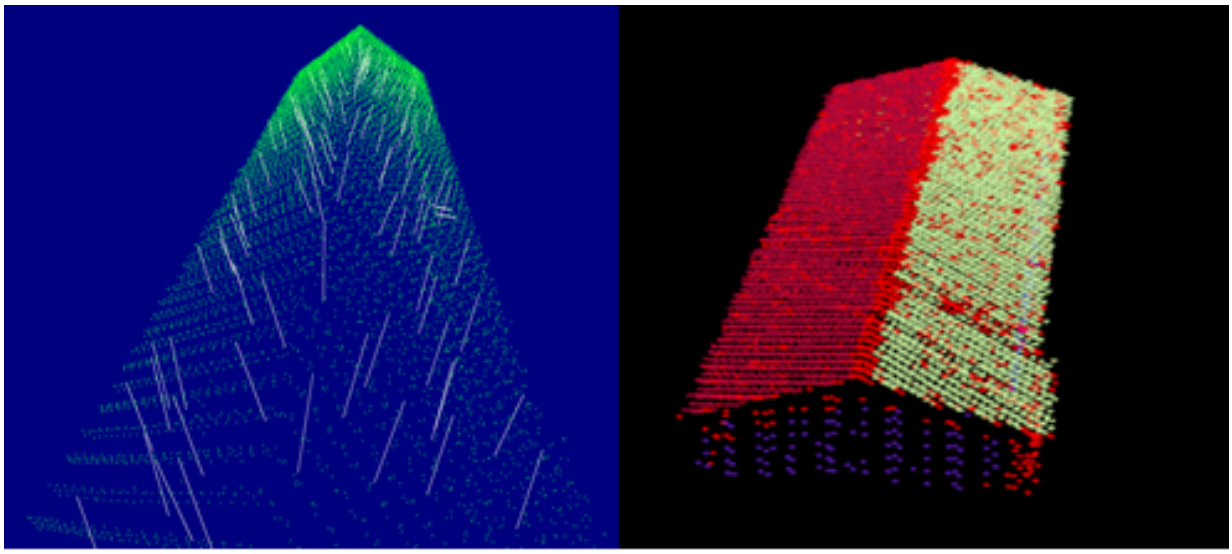


Figure 3.11 – (a) Estimated surface normals of the input point cloud of gable roof (b) Segmented building LiDAR points represented in colored cloud where each cluster has its own color. Red points represent the points that belong to the rejected clusters due to their small segments size.

3.6 Surface Fitting

Since the point cloud of building is segmented and the clusters are extracted and saved in different PCD files formats where each cluster contains points that are considered to be a part of the same smooth surface. The task is now to automatically recognize the clusters that represent roof patches. Slope analysis of the fitted surface is used to achieve that and in order to do that, the fitted surface of each cluster has to be estimated. Surfaces can be fitted to a point cloud by minimizing the number of outliers and maximizing the number of inliers. RANSAC is a non-deterministic algorithm in the sense that it produces a reasonable result only with a certain probability, with this probability increasing as more iterations are allowed.

The input to the RANSAC algorithm is a set of observed points in each cluster while the output includes the estimated parameters of the fitted plane model and its fitted point cloud as inliers. RANSAC achieves its goal by iteratively selecting a random subset of the input data. These data are hypothetical inliers and this hypothesis is then tested as it is explained in the paradigm of RANSAC section 2.3.2.3.1.

The pseudocode algorithm:

1. Input:
    - data: a set of observations.
    - model: a model that can be fitted to data.
    - n: the minimum number of data required to fit the model.

- k - the number of iterations performed by the algorithm.
- t - a threshold value for determining when a datum fits a model.
- d - the number of close data values required to assert that a model fits well to data.
- it: iterations number.

2. Output:
   - d: the number of close data values required to assert that a model fits well to data.
   - best_model: model parameters which best fit the data (or nil if no good model is found).
   - best_consensus_set: data points from which this model has been estimated.
   - best_error: the error of this model relative to the data.
3. initialization:
   - it := 0
   - best_model := nil
   - best_consensus_set := nil
   - best_error := infinity
4. Algorithm:
   - while it < k
     (a) maybe_inliers:= n randomly selected values from data
     (b) maybe_model:= model parameters fitted to maybe_inliers
     (c) consensus_set:= maybe_inliers
         – for every point in data not in maybe_inliers
           i. if point fits maybe_model with an error smaller than t
              * add point to consensus_set
         – if the number of elements in consensus_set is > d
           * (this implies that we may have found a good model, now test how good it is).
           * this_model := model parameters fitted to all points in consensus_set.
           * this_error := a measure of how well this_model fits these points
           * if this_error < best_error
             i. (we have found a model which is better than any of the previous ones, keep it until a better one is found)
             ii. best_model := this_model
             iii. best_consensus_set := consensus_set
             iv. best_error := this_error
   - increment iterations
5. return best_model, best_consensus_set, best_error

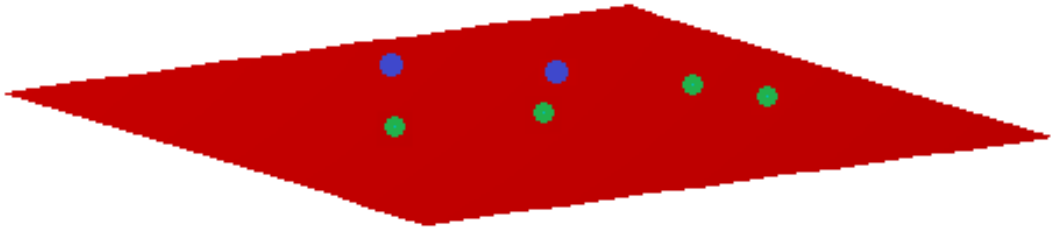


Figure 3.12 – Estimated Plane (red) by RANSAC with inliers (green) and outliers (blue) points.

The picture in figure 3.12 shows a simple application of the RANSAC algorithm on a 3-dimensional set of data. It is a visual representation of a data set containing both inliers and outliers.

3.7 Slope Angle-based Filtering

The LiDAR data is available only for some visible sides of building which means that the used LiDAR instrument in acquiring the data was not looking nadir exactly. The points on vertical surfaces(wall) are sparse because of the oblique angle between the LiDAR sensor and building walls during airborne data acquisition. It is known that in order to reconstruct the geometry of the building, the points that belong to the roof have to be identified. So once the segments (clusters) are extracted and the fitted plane models are estimated, the roof planes have to be distinguished from other segments. To achieve that, slope angle-based filtering is applied in this research. The slope angle-based filtering is performed to separate the wall points and roof points. Ground points are discarded at the beginning and the remained points are representing only the roof patches and walls where wall points are located on vertical surfaces. The algorithm starts from the idea of generating a virtual horizontal plane and then compute the slope angle θ of each cluster between the estimated plane of the cluster and this horizontal plane figure 3.13. The plane, which is not perpendicular to the horizontal plane, is considered as a plane belongs to the roof and the corresponding points are added to the roof point cloud. Because the point data may have some bias in acquiring, the plane that is estimated from wall points will not be exactly perpendicular. So a giving threshold helps in separating wall and roof points due to that the plane with slope angle is between 80 to 100 degrees, is defined as a wall and excluded from the later processing steps. And according to the regular wall slope angle of most buildings, this threshold works well for all tested buildings data figure 3.14.

Equation of slope angle:

$$\theta = \cos^{-1} \frac{|a1 \times a2 + b1 \times b2 + c1 \times c2|}{\sqrt{a1^2 + b1^2 + c1^2} \times \sqrt{a2^2 + b2^2 + c2^2}} \tag{3.3}$$

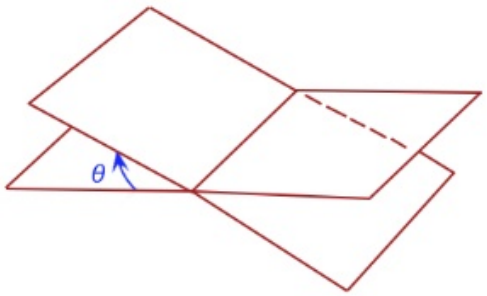


Figure 3.13 – Illustrating the slope angle between a plane and horizontal plane.

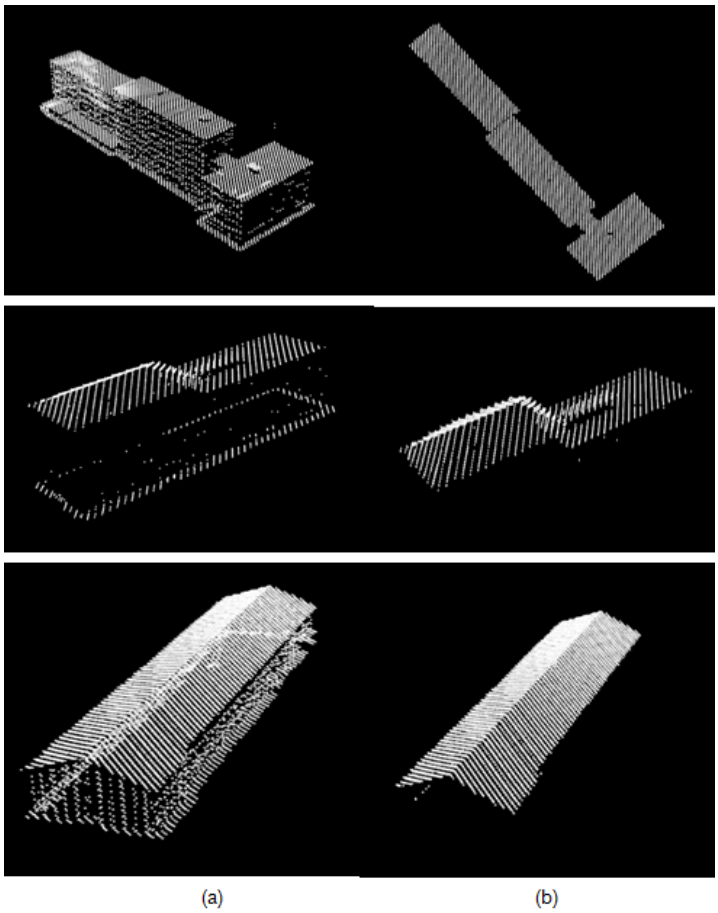


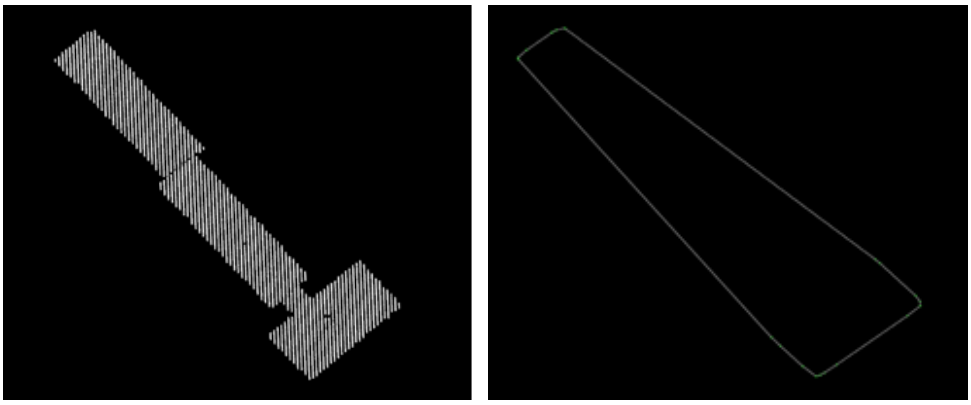
Figure 3.14 – Result of slope angle-based filtering a)The raw LiDAR data of building b) Roof points after filtering .

3.8 Extracting Bounding Hull

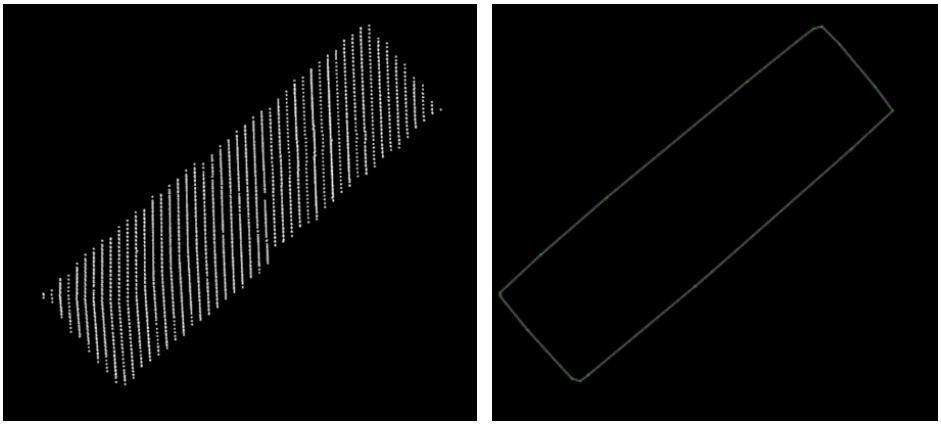
After segmenting building points, next step is to extract building boundaries which is done to find the outermost points of a building. Extracting Bounding hull, such as convex hull, concave hull and alpha shapes is very important step in this research and is implemented to reach accurate 3D model of buildings. Three methods have been tried during the work.Bounding hull contains points that form building outline.

Convex hull of a set of points is convex polygon with the minimum area that represents the region occupied by the points. Convex hull is widely used in various fields such as geographical

information systems (Liu u. a. [2003]) and pattern recognition (Zhang u. Xu [2002]). There are a lot of algorithms computing convex hull such as QuickHull (Eddy [1977], Bykat [1978]), Javis March(Jarvis [1973]), Divide-and-Conquer (Preparata u. Hong [1977]). A lot of researchers have been worked to compute the convex hull such as (Graham [1972]) and (O'Rourke u. Streinu [1998]). In PCL library, a ConvexHull algorithm is offered and it does not always specify the points region accurately. Although it gives more regularized boundary than other two methods. In addition, it gives good results for data with convex frame, but it does not work properly in extracting the boundary points of the whole roof points which does not have convex frame. Some examples of boundaries were extracted by this algorithm are presented in Figure 3.15.



(a) Roof1 point cloud (b) Roof1 Convex hull



(c) Roof2 point cloud (d) Roof2 convex hull

Figure 3.15 – Convex Hull Extraction.

Then **2D alpha-shape** algorithm is used to extract bounding hull of the segments. The alpha shape was originally introduced by (Edelsbrunner u. a. [1983]). The alpha shape of a points set is a sub graph of Delaunay triangulation (Dt) of points such that two points are connected if there is an empty ball of radius 1/alpha touching two points. When alpha=0, the ball of radius 1/alpha is replaced by half-plane. Hence the alpha shape of points will be equal to the convex hull of these points. So, the alpha shape is a generalization of convex hull such that it is applied in various fields (Asaeedi u. a. [2013]).

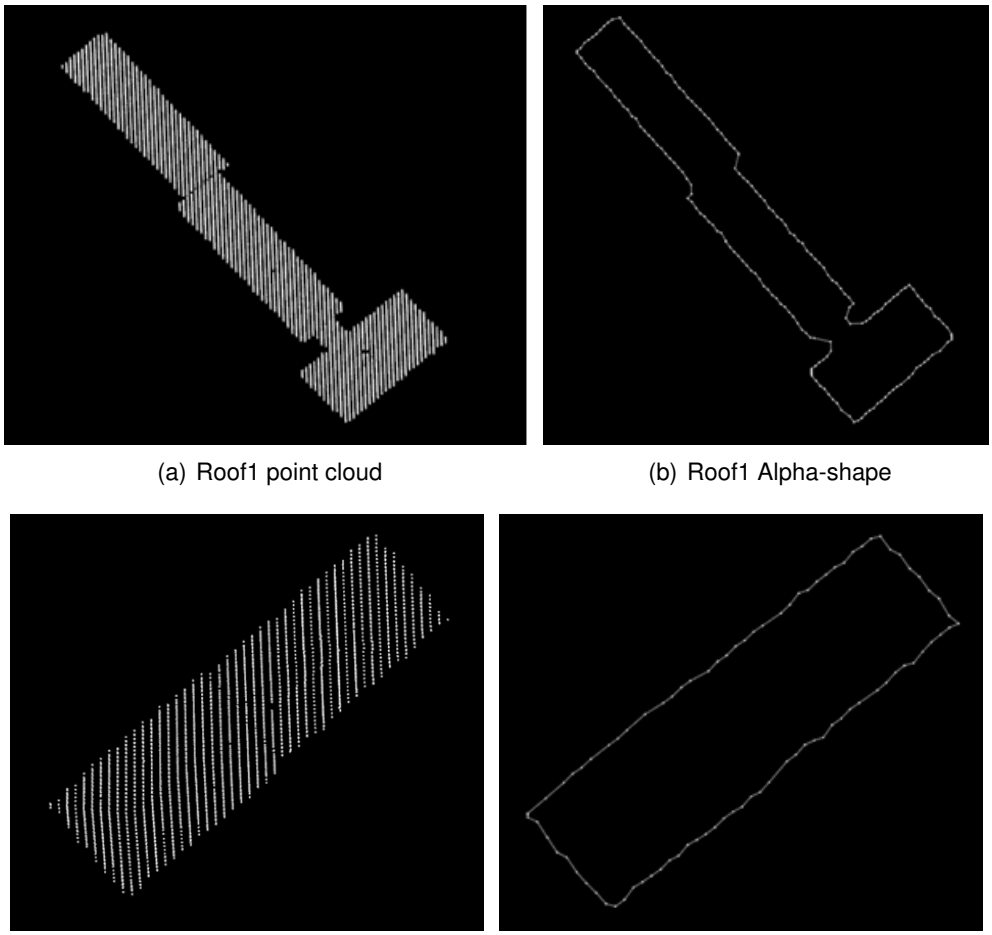
Filippovska u. a. [2016] created an initial building outline from a given set of building points with the alpha shape algorithm which will be applied here to build basic alpha shape using a Delaunay triangulation as underlying triangulation. The applied algorithm was taken from the class Alpha_shape_2<Dt> in the Computational Geometry Algorithm Library (CGAL). The class Alpha_shape_2<Dt> represents the family of alpha-shapes of points in a plane for all positive alpha values. It maintains the underlying triangulation Dt which represents connectivity and order among squared radius of its faces. Each k-dimensional face of the Dt is associated with an interval that specifies for which values of α the face belongs to the alpha-shape. There are links between the intervals and the k-dimensional faces of the triangulation (Attali u. Boissonnat [2002]).
In addition, this class provides functions to set and get the current alpha-value, as well as an iterator that enumerates the alpha-values where the alpha-shape changes. It provides iterators to enumerate the vertices and edges that are in the alpha-shape, and functions that allow to classify vertices, edges and faces with respect to the alpha-shape. They can be in the interior of a face that belongs or does not belong to the alpha-shape. They can be singular/regular, that is be on the boundary of the alpha-shape, but not incident to a triangle of the alpha-complex. Finally, it provides a function to determine the optimal alpha-value such that the alpha-shape satisfies the following two properties or at least the second one if there is no such alpha that both are satisfied:

1) The number of components equals to a number of your choice.
2) All data points are either on the boundary or in the interior of the regularized version of the alpha-shape (no singular edges).

The framework for the used 2D alpha-shape algorithm is based on the articles (Edelsbrunner u. Mücke [1994] and Edelsbrunner [1992]). In which, initialization of the family of alpha-shapes is done by the points in the range (first point,last point) and introduces an alpha_shape for a positive alpha-value.
For many examples, the used 2D alpha-shape algorithm represents bounding points more accurately than convex hull. In addition it provides the edges that connect the bounding points and form the hull polygon as depicted in Figure 3.16.

Another generalization of convex hull is the **concave hull** that represents the tighter area occupied by the points than convex hull. In another words, the concave hull, unlike convex hull and alpha shape, constructs nonconvex enclosure on a set of points by generating the non-convex polygons. (Galton u. Duckham [2006]) has introduced the concave hull and then (Moreira u. Santos [2007]) have developed it and they have suggested an algorithm based on the k-nearest neighbors approach to compute the concave hull. PCL library offers ConcaveHull (alpha shapes) algorithm. The algorithm was implemented on the extracted segments and it provides more efficient and accurate results in comparison with aforementioned algorithms Figure 3.17.

The framework of the algorithm starts from finding the fitted plane parameters of each roof patch using RANSAC. Then the inliers are projected on the plane. Thus, the resultant cloud is input of the concave hull algorithm. Setting the alpha value as the only parameter is very important since



(a) Roof1 point cloud (b) Roof1 Alpha-shape

(c) Roof2 point cloud (d) Roof2 Alpha-shape

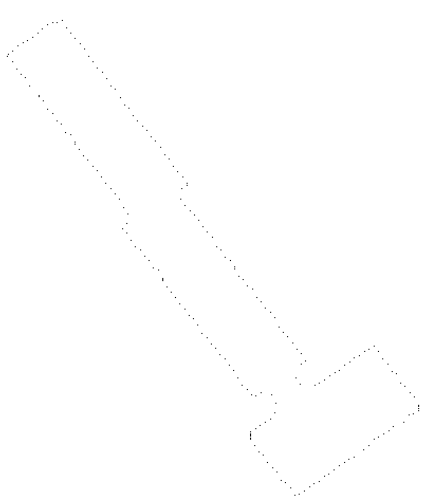
Figure 3.16 – Alpha-shape Extraction.

it limits the size of the resultant hull segments. And for smaller alpha, the more detailed hull is produced. So the alpha value plays main role in the smoothness level of produced bounding hull Figure 3.18. This alpha represents the maximum length from a vertex to the facet center (center of the voronoi cell). So the facet is accepted only if the distance from any vertex to the facet center is smaller than alpha. So alpha has to be positive and non-zero value. The resultant points are concave hull vertices.

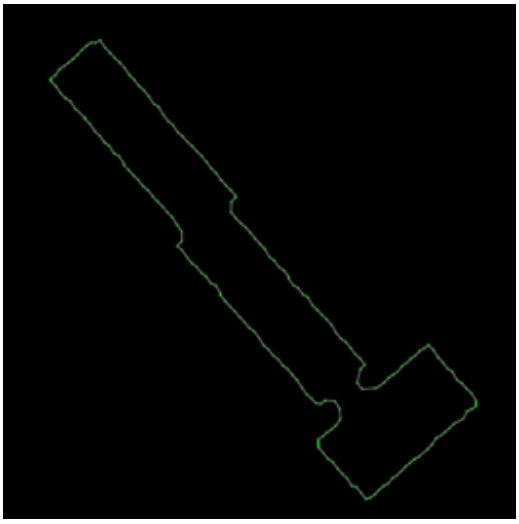
Once concave hull of roof patches are extracted and the points are sorted in clockwise mode, then area of roof patch's hull is computed using the following equation. Table 3.2 represents a comparison of the computed area of the extracted hull for both concave and convex hull algorithms.

$$area = |\frac{(x_1y_2 - y_1x_2) + (x_2y_3 - y_2x_3)....+ (x_ny_1 - y_nx_1)}{2}| \tag{3.4}$$

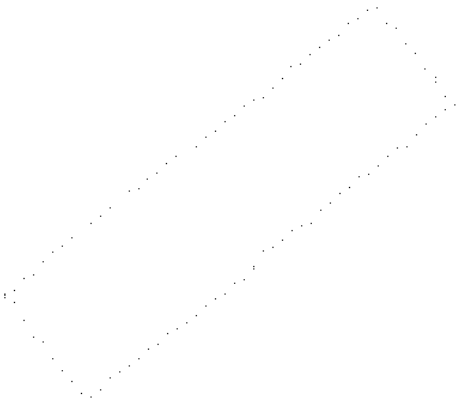
Alpha shape and concave hull are generalizations of convex hull , (Asaeedi u. a. [2013]). Experimentally the difference between concave hull and 2D alpha shape are very small Figure 3.19 (a). While the difference between convex hull and alpha shape is very small just in case the outer frame of the points forms a polygon which all interior angles between subsequence edges are less than or



(a) Concave Hull Points

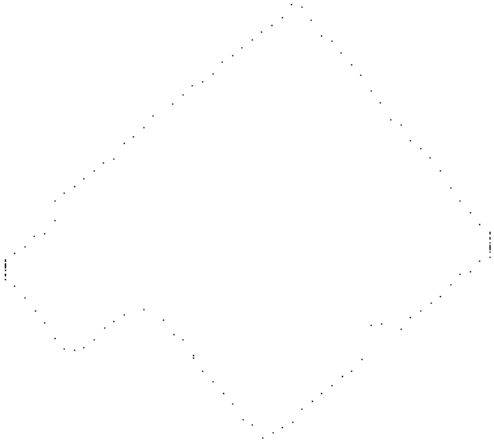


(b) connected line segments

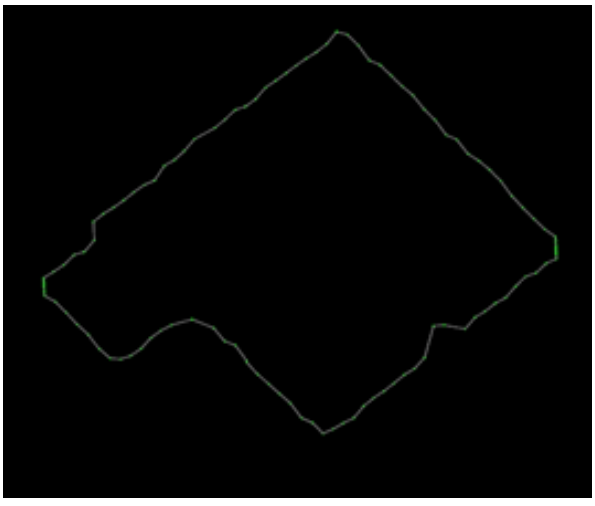


(c) Concave Hull Points

(d) connected line segments



(e) Concave Hull Points



(f) connected line segments

Figure 3.17 – Concave Hull Extraction.

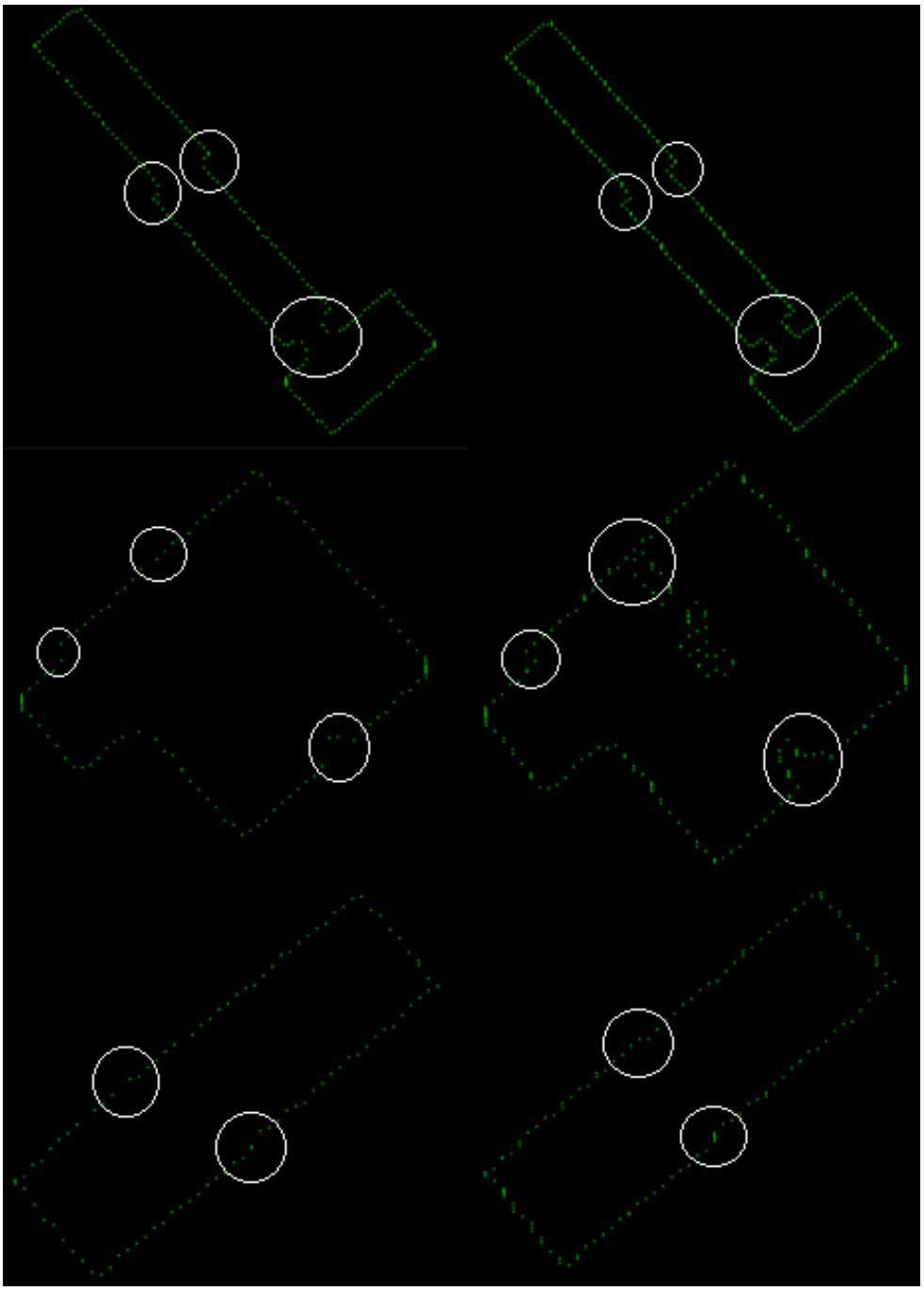


Figure 3.18 – Illustrating the effect of the parameter alpha on the concave hull (i.e. maximum distance from a vertex to the center of the voronoi cell). Photos in left column represent concave hull for large alpha value. Photos in right column represent concave hull for small alpha value. White circle to recognize the effect of alpha value on the level of details where for small value more detail are represented

equal to 180. Which is generally the case of the roof patch when it is processed individually Figure 3.19 (b). While the difference is very big in other cases like the hull of footprint (whole roof points of building) Figure 3.19 (c).

The presence of concavity in irregularly distributed points makes it difficult to trace all the boundary

Building1	Patch1	Patch2	Patch3	Patch4	Roof
ConcaveHull	**632.862**	**599.409**	**566.674**	**34.444**	**1863.468**
ConvexHull	**641.956**	**616.634**	**578.618**	**34.645**	**2837.448**

Tab. 3.1 – Area(m2) comaprison between Concave and convex hull of building1

Building2	Patch1	Patch2	Patch3	Patch4	Patch5	Roof
ConcaveHull	**102.928**	**337.506**	**43.883**	**109.637**	**44.342**	**673.580**
ConvexHull	**111.443**	**346.742**	**44.292**	**113.353**	**43.779**	**704.852**

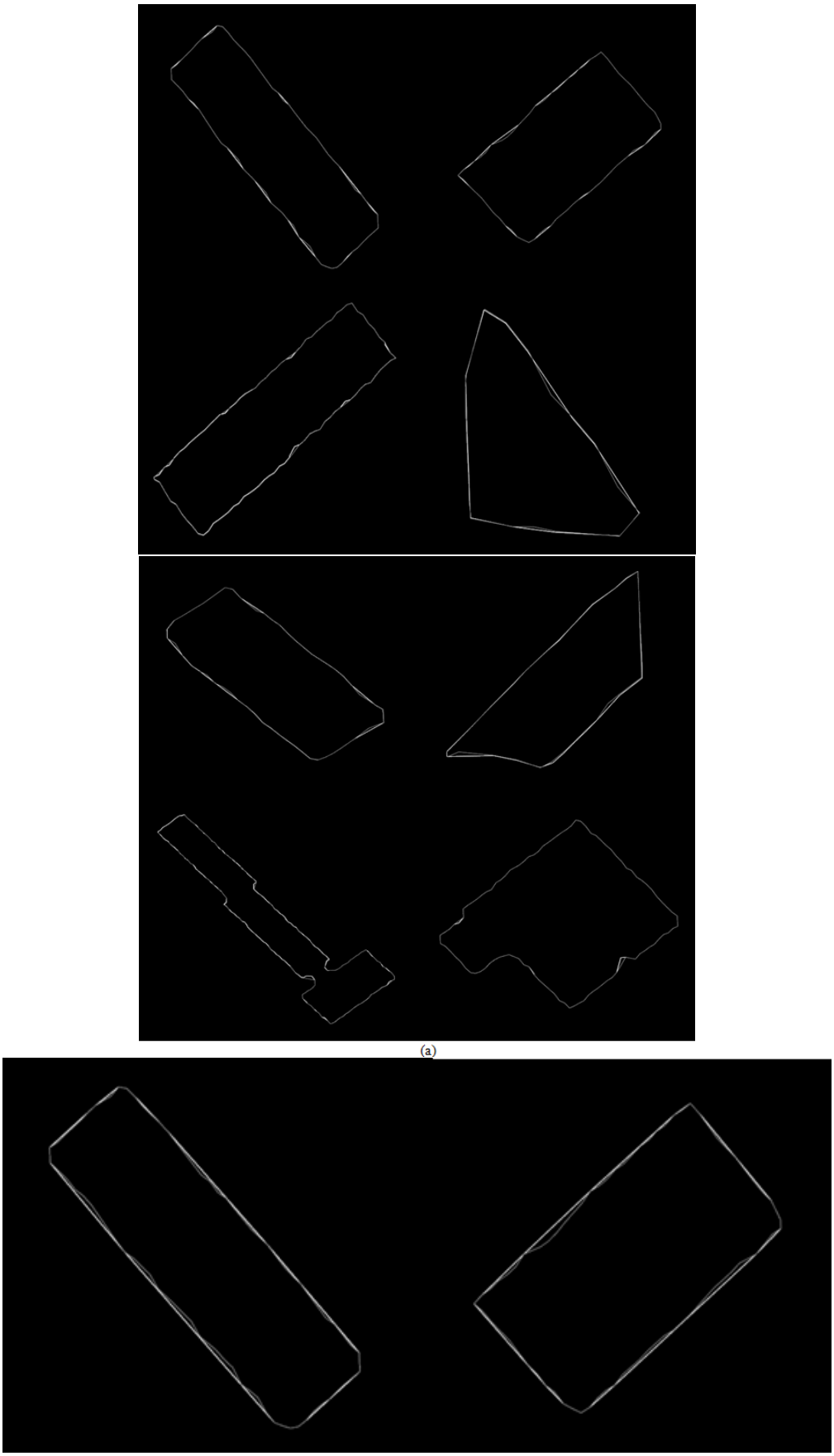
Tab. 3.2 – Area(m2) comaprison between Concave and convex hull of building2

points by convex hull algorithm without missing the concave corner. Furthermore, concave hull algorithm provides more accurate and detailed hull than convex algorithm and provides 3D hull points comparing to 2D hull points in 2D alpha shape algorithm. Therefore, its results were used for further steps of the 3D reconstruction process in this research.

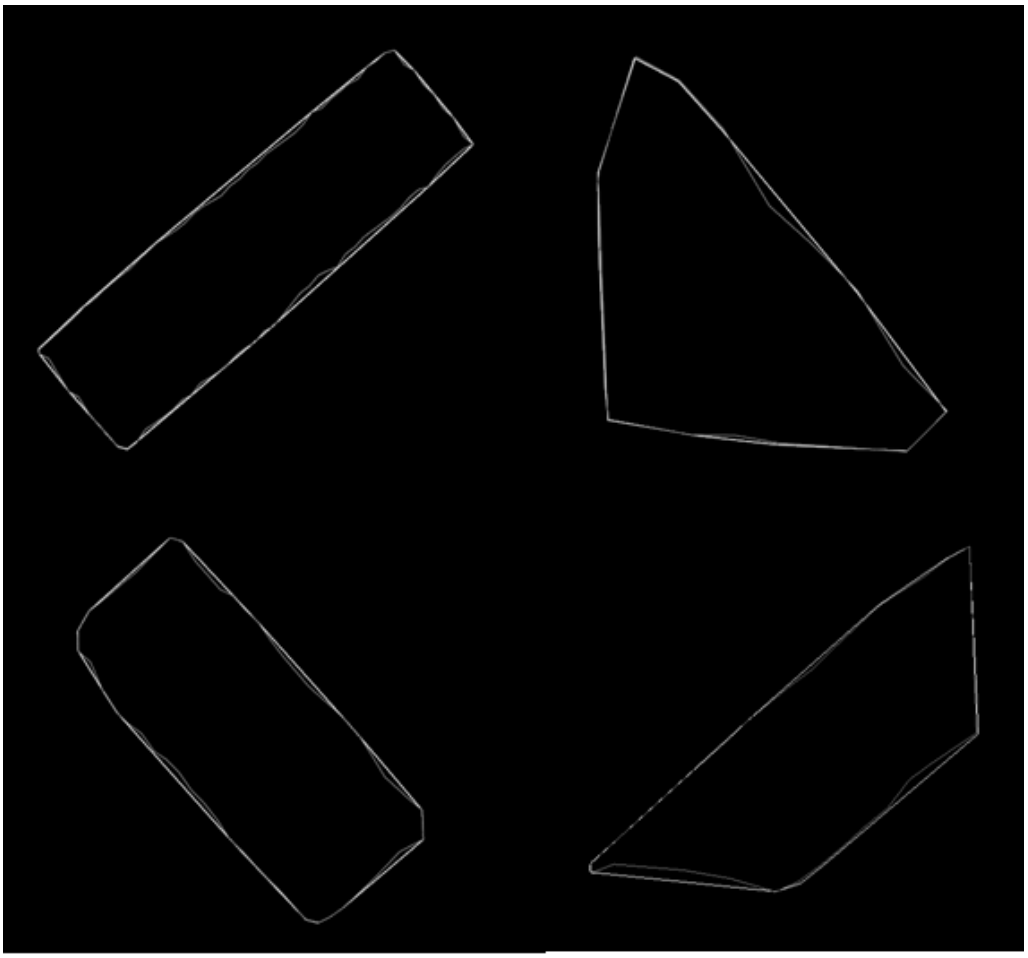
3.9 Boundary Regularization

Subsequent reconstruction steps require the individual segments to be clearly separated from each other. Due to the point cloud sparsity, the extracted boundary are jagged and has irregular shape because it contains diversity of not aligned line segments. Consequently, the adjacent segments will generate overlapping areas or gaps in between as shown in Figure 3.20 which representation of the extracted segments are shown in different colors with corresponding jagged boundaries for many different types of buildings. So regularization process is highly demand in order to obtain structural lines. (Filippovska u. a. [2016]) used Hough transform to determine the main directions of the building and to extract line segments which are oriented accordingly. The main goal of boundary regularization is to simplify the bounding polygon by identifying the critical points that show an obvious turning in shape. Those critical points will be called vertices. Two regularization approaches have presented in this research to generate the segment outlines with regularized shapes from the extracted bounding points.

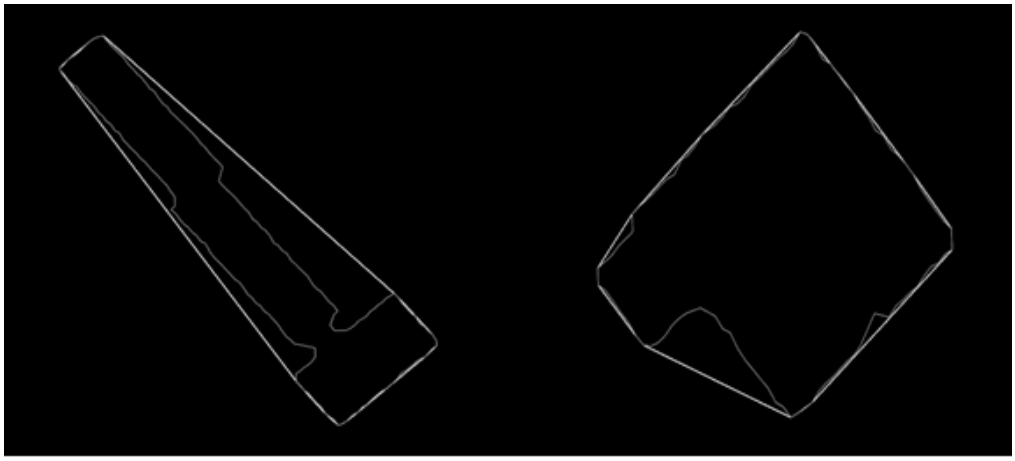
Both approaches employ RANSAC algorithm and dominant direction to regularize the boundary. Assumption is that buildings are rectilinear and adjacent edges should either be parallel or perpendicular to the dominant direction. In the both approaches, the direction of the line segments have been limited (Kada u. Wichmann [2012]) because it is related to the dominant directions of building. The first method applied on the entire bounding hull points while the second one is applied on groups of line segments individually. The workflow of the second method which is considered in this research as a regularization process is shown in Figure 3.1. It works for the buildings that have mutually perpendicular directions.



(a)



(b)



(c)

Figure 3.19 – For illustrating the difference between the concave hull and alpha shape and between the convex hull and alpha shape. (a) Bold lines represent the concave hull in xy plane while others represent the 2D alpha shape and the difference is small. (b) Bold lines represent the convex hull in xy plane while others represent the 2D alpha shape and clearly the difference is small. (c) Bold lines represent the convex hull in xy plane while others represent the 2D alpha shape and clearly the difference is very big.

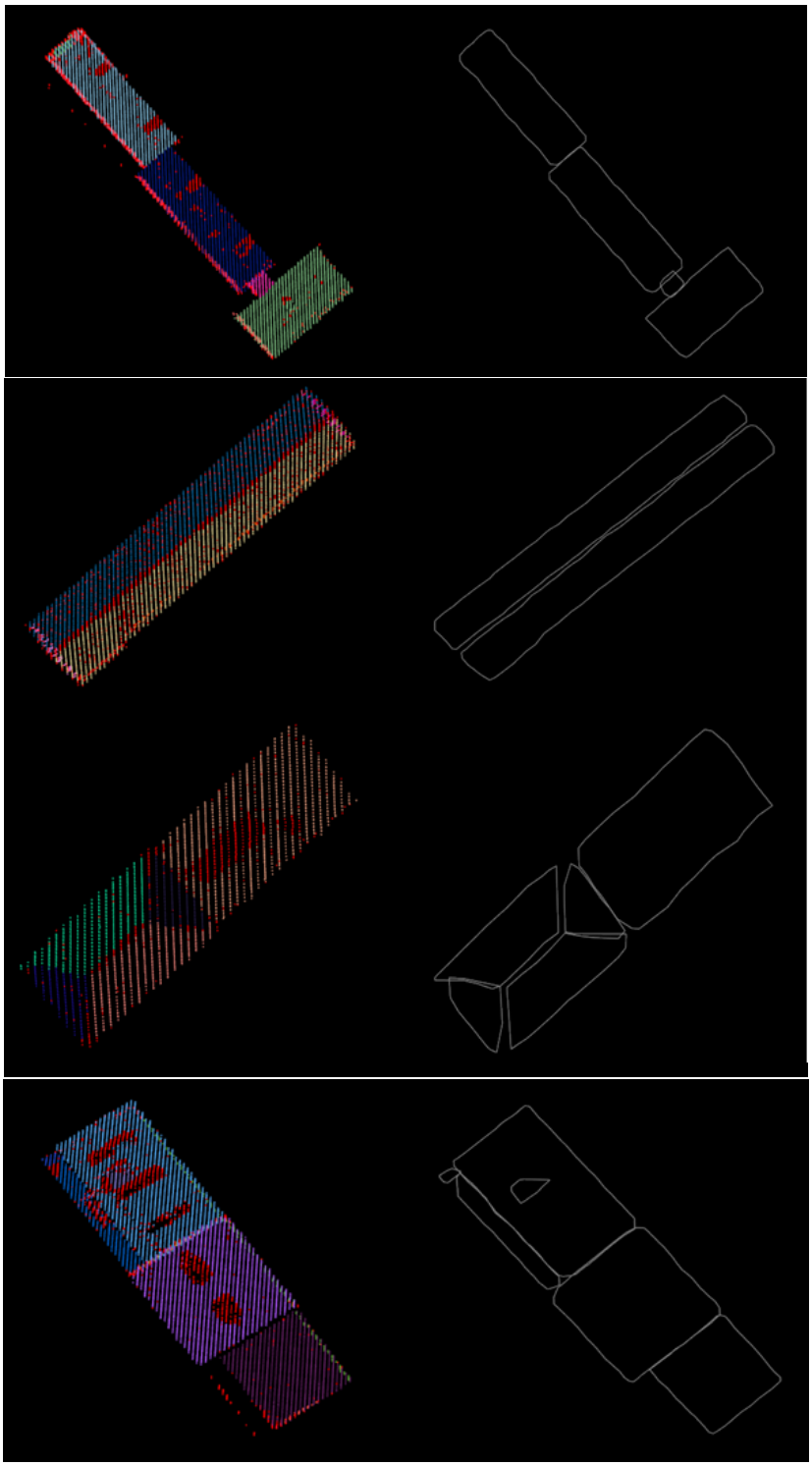


Figure 3.20 – Illustration, for many different types of buildings, the overlapping areas among the adjacent jagged boundaries as well as the gaps. (a) represents the extracted segments shown in different colors (colors arbitrary). (b) represents the corresponding jagged boundaries of the segments

3.9.1 Dominant Direction

In order to obtain more accurate boundary, the dominant direction has to be computed. It is used to limit the direction of the line segments (Kada u. Wichmann [2012]). For determining the dominant direction, the lengths of all boundary line are computed. The main dominant direction is determined from histogram analysis of all line segments' directions in the bounding hull. The idea was inspired from (Kada u. Wichmann [2012]) and the direction which has the peak in the histogram will be chosen as the main dominant direction.

Orthuber [2014] used histogram for determining the dominant direction, but they made histogram analysis of all triangle edges of the Delaunay triangulation which are inside the boundary polygon instead of the hull's line segments. The direction which has the peak in the histogram will be chosen as the main dominant direction.

Since the building corners angles are mostly 90 and 45 degrees, the angular field is divided to 8 equatorial ranges by 45 degree increments in order to compute the histogram values. Then find the bearing of each line segment using the equation 3.5:

$$Bearing = \tan^{-1} \frac{|dx|}{|dy|} \tag{3.5}$$

After that the number of lines and their lengths are added to the corresponding range. Finally the histogram is plotted where the horizontal axis represents the angular ranges with 45 degrees increment and the vertical axis represents the cumulative line segments length ın each direction. As can be noticed from the extracted hulls of different buildings that the longer sides of the building are most likely to represent its dominant direction. So the direction that has the maximum cumulative length (i.e. which has the peak in the histogram) is considered as the main dominant direction Figure 3.21.

Pseudo Code for dominant direction algorithm:

- Input:
  - P input point clouds of bounding hull.
  - Coef coeficients of the estimated line.
  - R_{k45} are angular ranges for k=1,2,. . . .,8
- Output:
  - L_{k45} cumulative line segment length of k range.
  - N_{k45} Number of line segments.
- initialization:
  - N_{k45}=0
  - L_{k45}=0

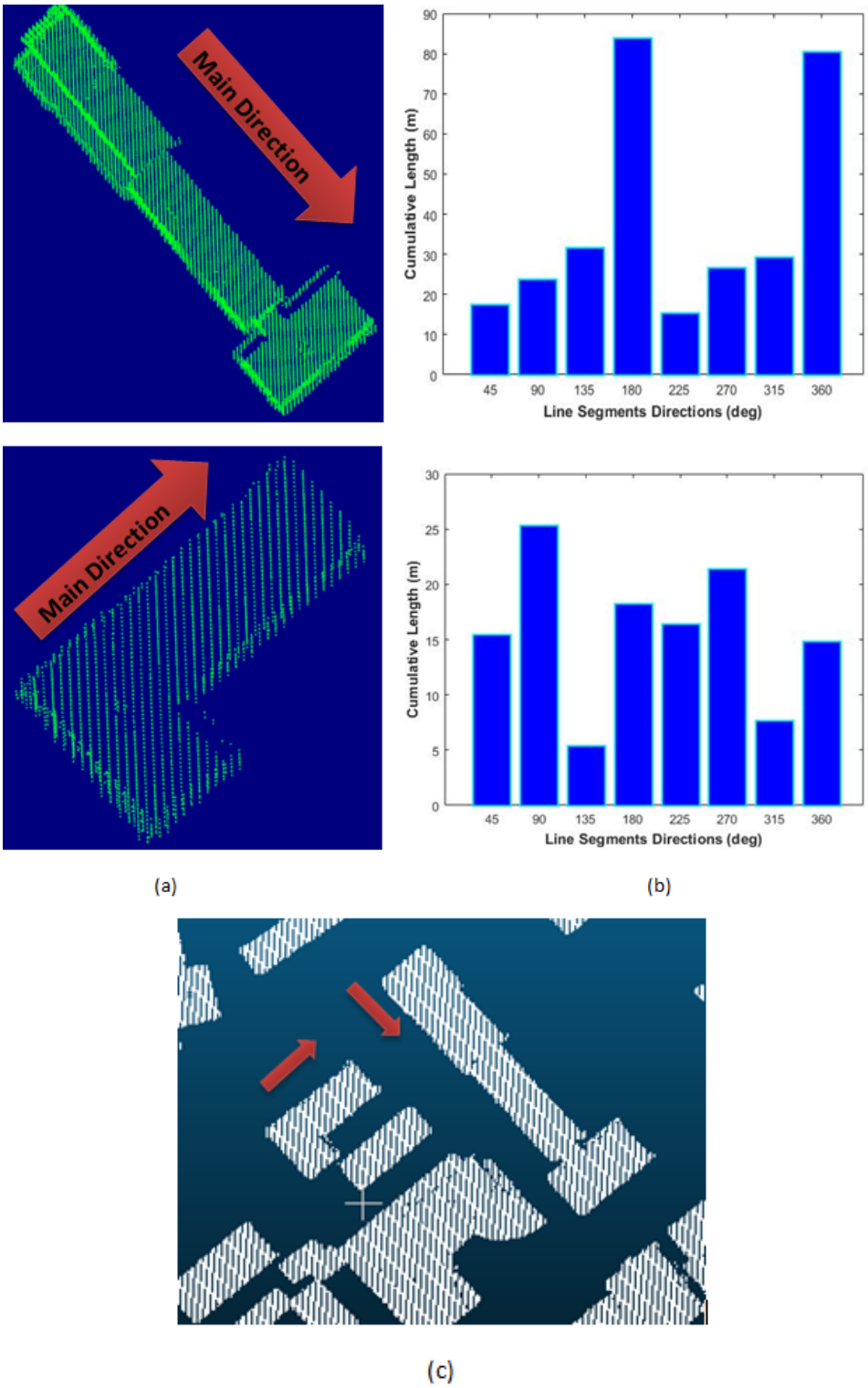


Figure 3.21 – (a) Building point cloud with arrow indicating the main direction. (b) Histogram of directions and bincenter with the highest value selected as the main direction. (c) Illustrates how both buildings are directed corresponding to each other in reality.

- Algorithm:
  - for i=0 to size(P) do
    - Coef= line coeficients of line estimated from points P by RANSAC

  - $P_i\_P_{i+1}$ = start_end line segment points
  - B_i Bearing of the line segment i: $B_i = \tan^{-1}(|dx_i/dy_i|)$
  - D_i length of the line segment i: D_i=sqrt(dx^2+dy^2)
  - If $B_i \in R_{k45}$ range then
    - L_{k45}=L_{k45}+D_i
    - N_{k45}=N_{k45}+1
  - end if
- end for

Then Direction with the maximum cumulative length is considered as the main dominant direction.

The coefficients of the main dominant direction are computed from the included line segments points using RANSAC. Thus the start and end points are computed in order to plot it on the bounding hull points.

3.9.2 Grouping Line Segments and Straight Line Fitting

Once the boundary points of concave hull are extracted and sorted in clockwise mode, boundary points can be grouped. By sequentially following the boundary points and collecting the consecutive line segments that has close slopes within a given tolerance to one group Figure 3.22.

The idea of grouping the line segments was inspired from (Sampath u. Shan [2007]). The difference here is that the solution was not determined by the least squares, but RANSAC method has been used to estimate the parameters of the line that fit sets of point in each group Figure 3.23 (b) . And thus the line will be modelled by equation 3.6.

$$Ax + By + C = 0 \tag{3.6}$$

Since the main dominant direction and the perpendicular direction have been computed, the estimated lines can be classified in accordance to parallel and perpendicular directions. If two adjacent lines have similar slope with a giving tolerance, they are combined and checked once more against the main directions. The start point of the newly derived line will be the start point of the longer line while the end point will be the intersected point of the perpendicular from the end of the shorter line to the longer line and its coordinates are computed using the following equations.

$$k = ((y2 - y1) * (x3 - x1) - (x2 - x1) * (y3 - y1))/((y2 - y1)^2 + (x2 - x1)^2) \tag{3.7}$$

$$x4 = x3 - k * (y2 - y1) \tag{3.8}$$

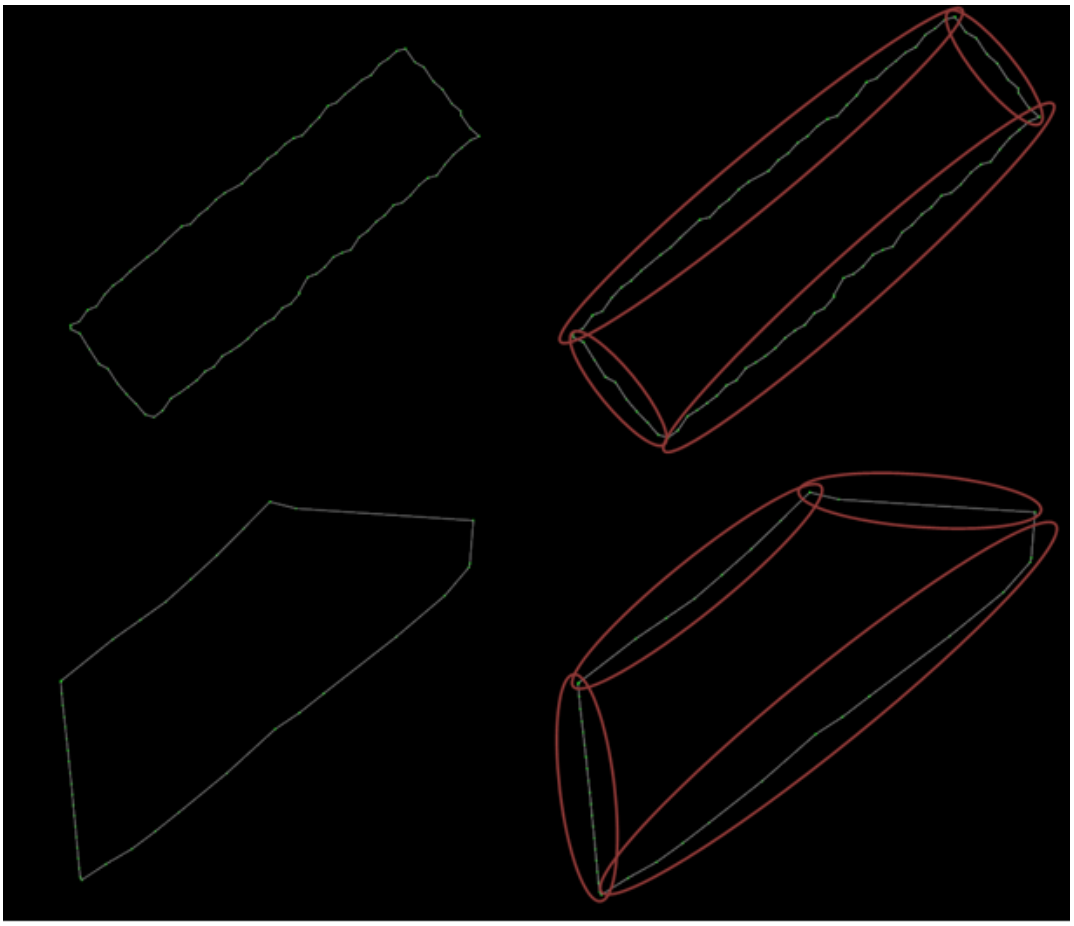


Figure 3.22 – (a) represents the connected line segments of the concave hull. (b) represents the grouping of line segments where each red circle contains one group of line segments.

$$y4 = y3 + k * (x2 - x1) \tag{3.9}$$

In order to consider the hipped and gable roof in this research several constraints are taking into consideration and applied progressively on the estimated lines to determine whether the line is a candidate line for the building boundary or not:

1) Interior angle *Ddir* between two consecutive lines from the difference between bearings of both. It is computed using the following equations:

$$DIR_i = \tan^{-1} \frac{|dx_i|}{|dy_i|} \tag{3.10}$$

$$DIR_{i+1} = \tan^{-1} \frac{|dx_{i+1}|}{|dy_{i+1}|} \tag{3.11}$$

$$Ddir = DIR_{i+1} - DIR_i \tag{3.12}$$

2) The length of the line and its percentage to the consecutive line length and to the main dominant direction length.

3) The slope angle from the main dominant direction.

In advanced steps of the algorithm, if the direction of the line is too far off and its length comparing to the main direction line is high, then the line keeps its varying direction, otherwise it is adjusted once more to the main direction for best fitting.

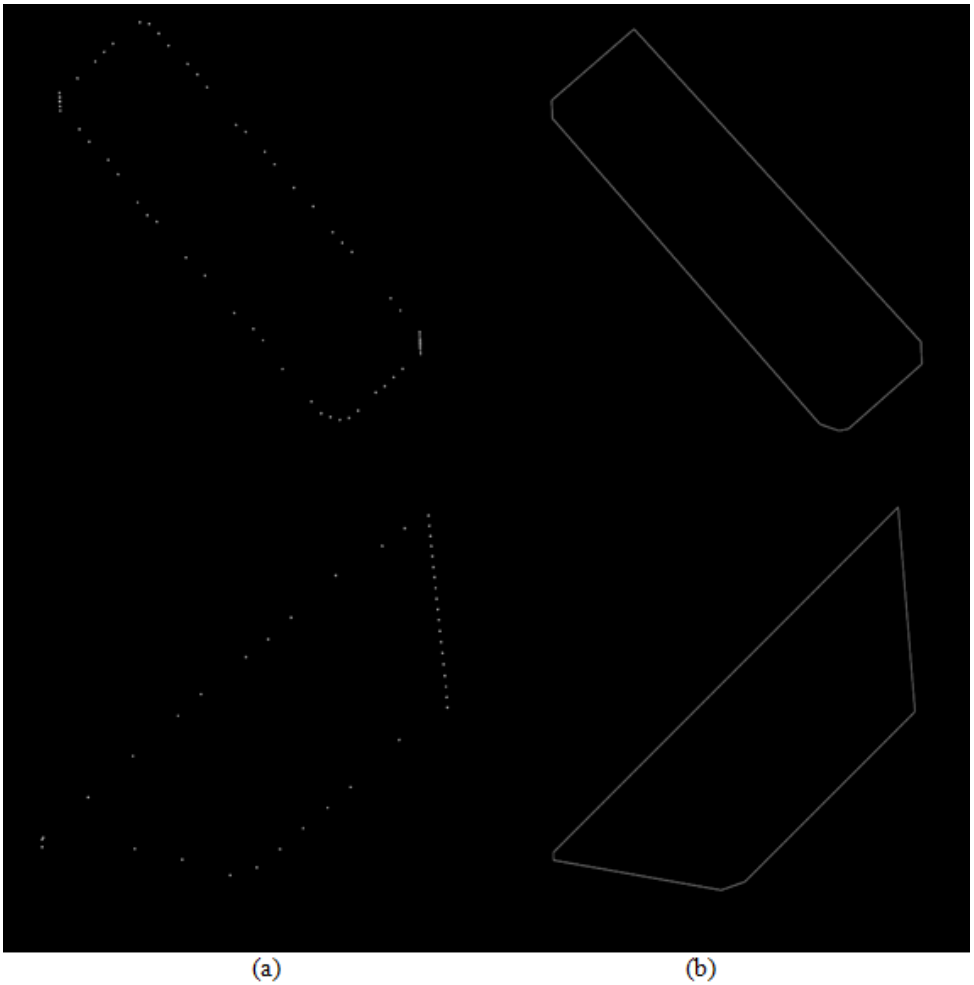
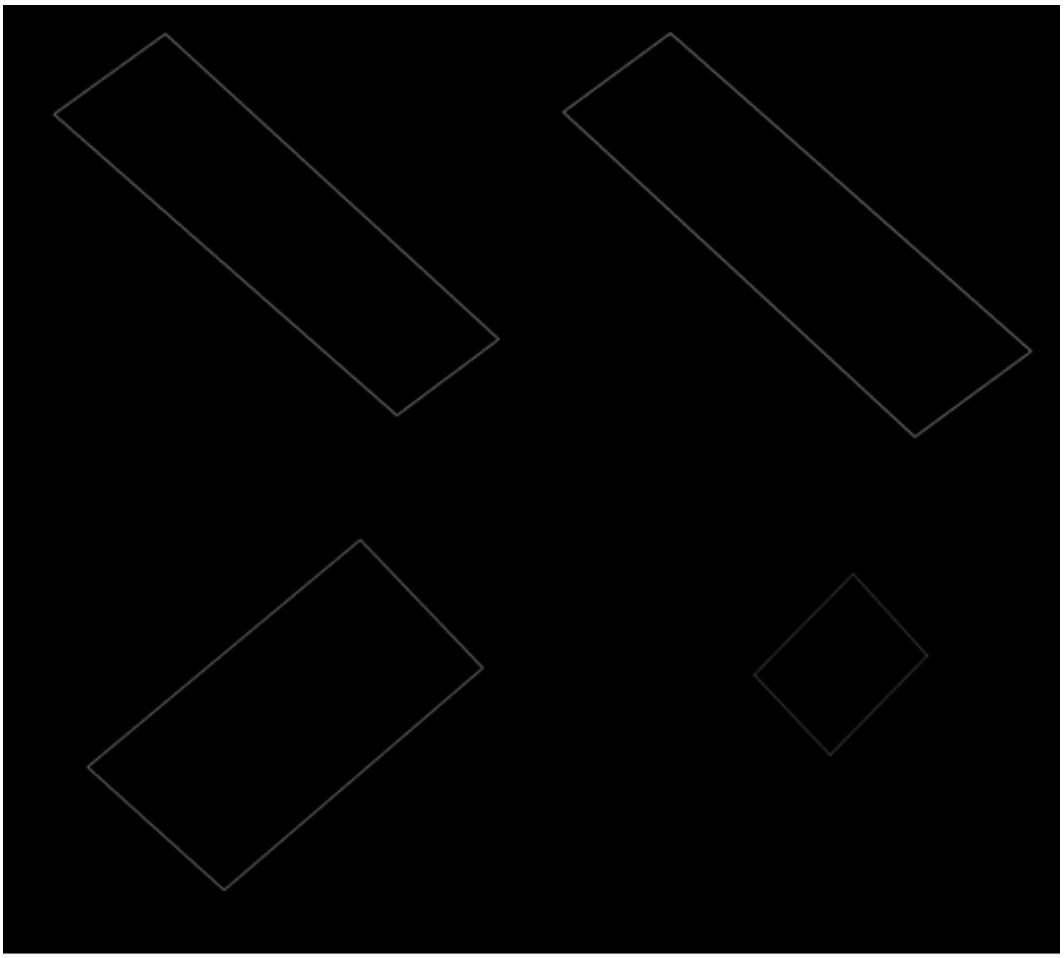
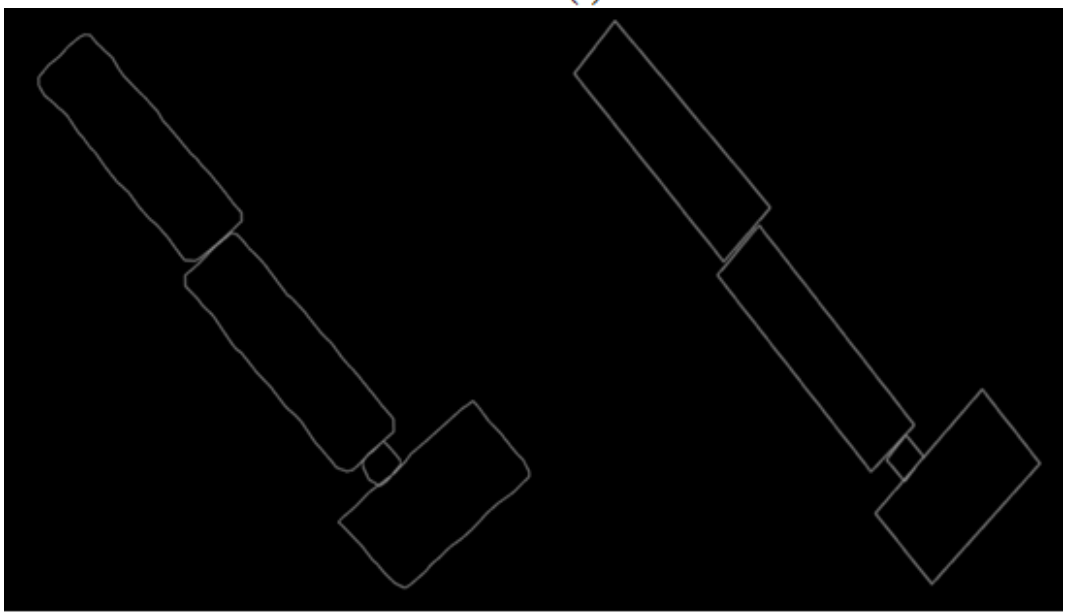


Figure 3.23 – (a) represents points of line segments' groups, (b) represents estimated lines by RANSAC from points in groups.

The lines which fulfill all conditions will generate the final boundary at the end and the vertices will be the linking points among the adjacent lines. The results of applying the algorithm on the concave hull points of each roof patch are presented in Figure 3.24.



(a)



(b) (c)

Figure 3.24 – (a) represents regularized boundary of each roof patch of the building. (b) represents the boundary of all patches before regularization. (c) represents the regularized boundary of all patches generating together the shape of the roof

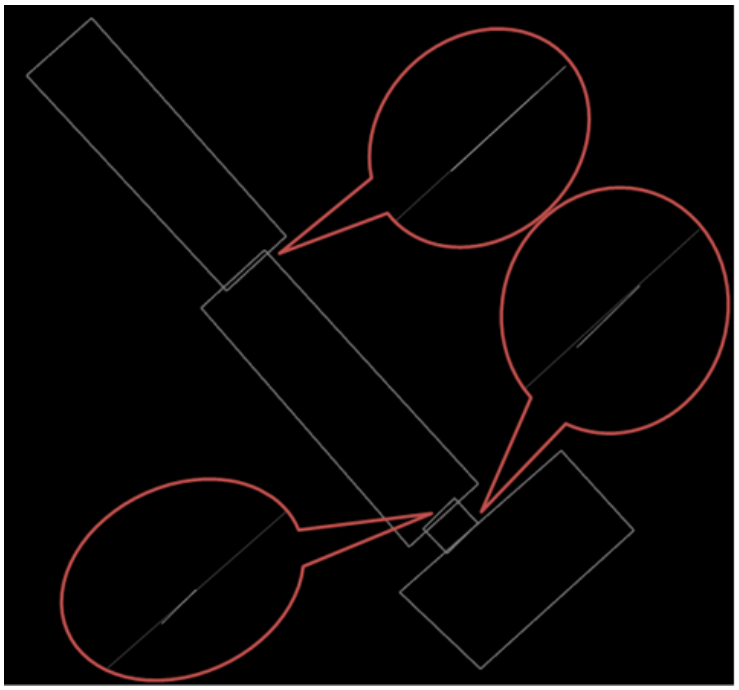
3.9.3 Model Regularization and 3D Modelling

Because of the sparsity of point cloud and the computation errors, the extracted roof faces are isolated and not connected or overlapped even though they belong to the same building and their boundaries have been regularized as shown in Figure 3.25. Consequently, further refinement need to be carried out before inputing into a geospatial database. Firstly, vertices located within a close proximity to each other are grouped together. Another step is performed to enforce the alignment between vertices which appear to be in a line. This was done by computing the distance between each vertex and the closest line and if this distance is less than the pre-defined threshold, the vertex will be shifted to that line. Interior nearby vertices are grouped together at an average location of their positions.
Faces at so called jump edges, connecting the building boundary and the eaves, but occurring in several cases within the roof of a building as well, are -by definition- vertical (Dorninger u. Pfeifer [2008]).
Step edges represent 3D lines between adjacent segments with different height levels. OR(Ideally, a closed roof face boundary can be composed of eave edges and intersection edges of neighbouring faces (Dorninger u. Pfeifer [2008]).
Also for aforementioned reason, those points have not exactly the same xy coordinates but they are located within a close proximity to each other in xy plane. Step edges appear in case of adjacent parallel planes with different height level. In the connecting area between those planes, two approximately parallel lines are located. The regularization includes computing the shift values of each line from the boundary points of the corresponding roof patch. The line with minimum shift is projected on another plane.

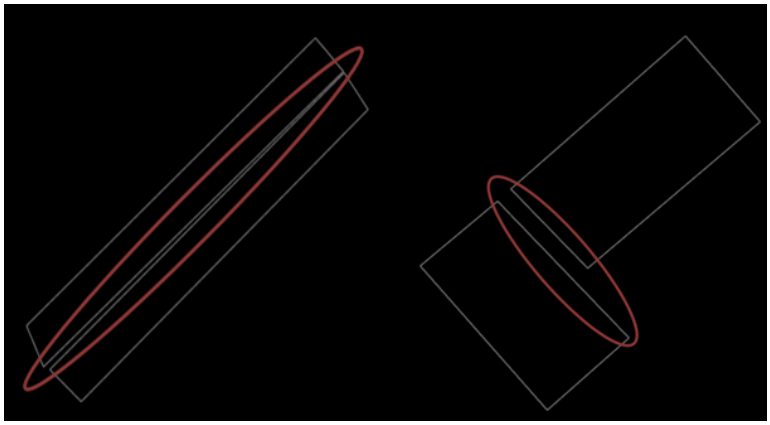
The building model is defined as a composition of a set of planar faces, which could be the best approximation of the given point cloud. Generally, these faces can be categorized as terrain intersection faces, wall faces, or roof faces. According to the definition of CityGML (Kolbe u. a. [2008]). In this research the building is modeled by composing the extracted roof faces. The goal of modeling procedures is to create closed polyhedral 3D building models consisting of vertices and faces from a segmented building point cloud. Traditional approaches of modeling analyze the segments' neighborhood relations and create and connect intersection edges and step edges between adjacent roof segments using different heuristics. After model regularization, the adjacent roof faces are connected properly and 3D roof vertices are connected with 3D ground vertices. Consequently, the final 3D models are generated for many buildings as shown in the following figures.

3.9.4 Building Footprint Extraction from LiDAR Data

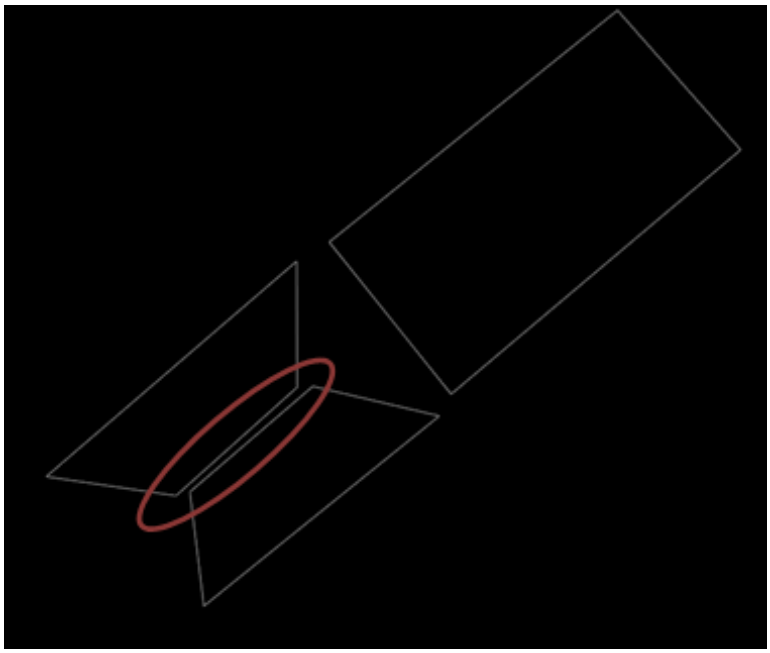
Compared to the traditional method using DSM to extract footprint, a novel technique of this research is to generate building footprint directly from LiDAR point clouds. Starting with extracting the regularized boundary of each roof patch individually and then solving the connection problems among the adjacent polygons in 3D space. Finally building footprint can be extracted simply by projecting the boundary in corresponding xy plane figure 3.29.



(a)

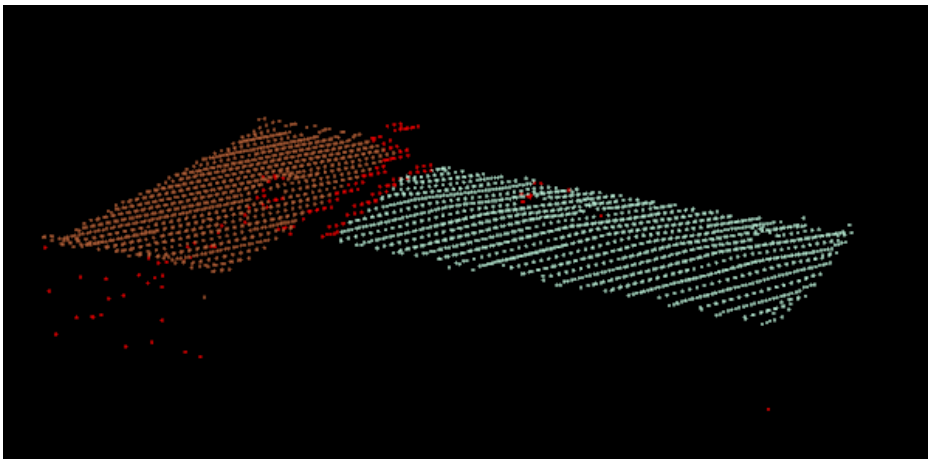


(b)

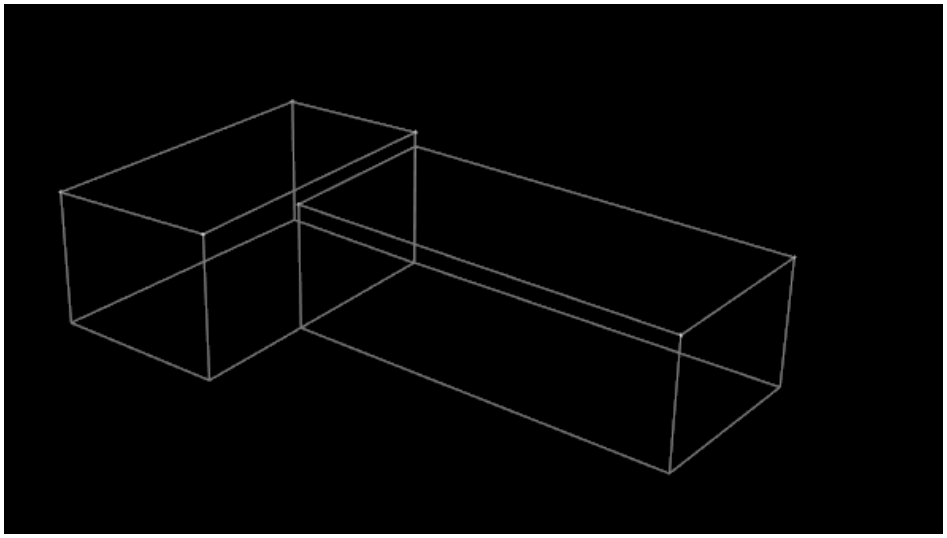


(c)

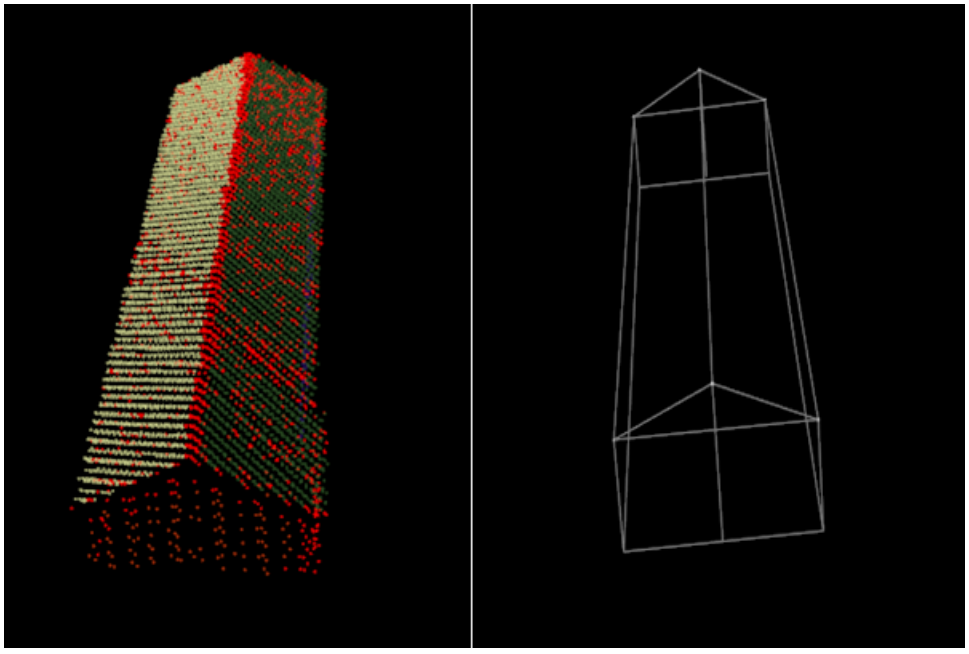
Figure 3.25 – Represents how the extracted roof faces are isolated and not connected or overlapped. The red circle is used to highlight points in the connection area that represent a ridge line or step edge. Clearly, they are located within a close proximity to each other.



(a) Segmented cloud



(b) 3D model of L-shape flat roof building with different height levels.



(c) 3D model of gable roof building

Figure 3.26 – 3D models and segmentation result of several buildings

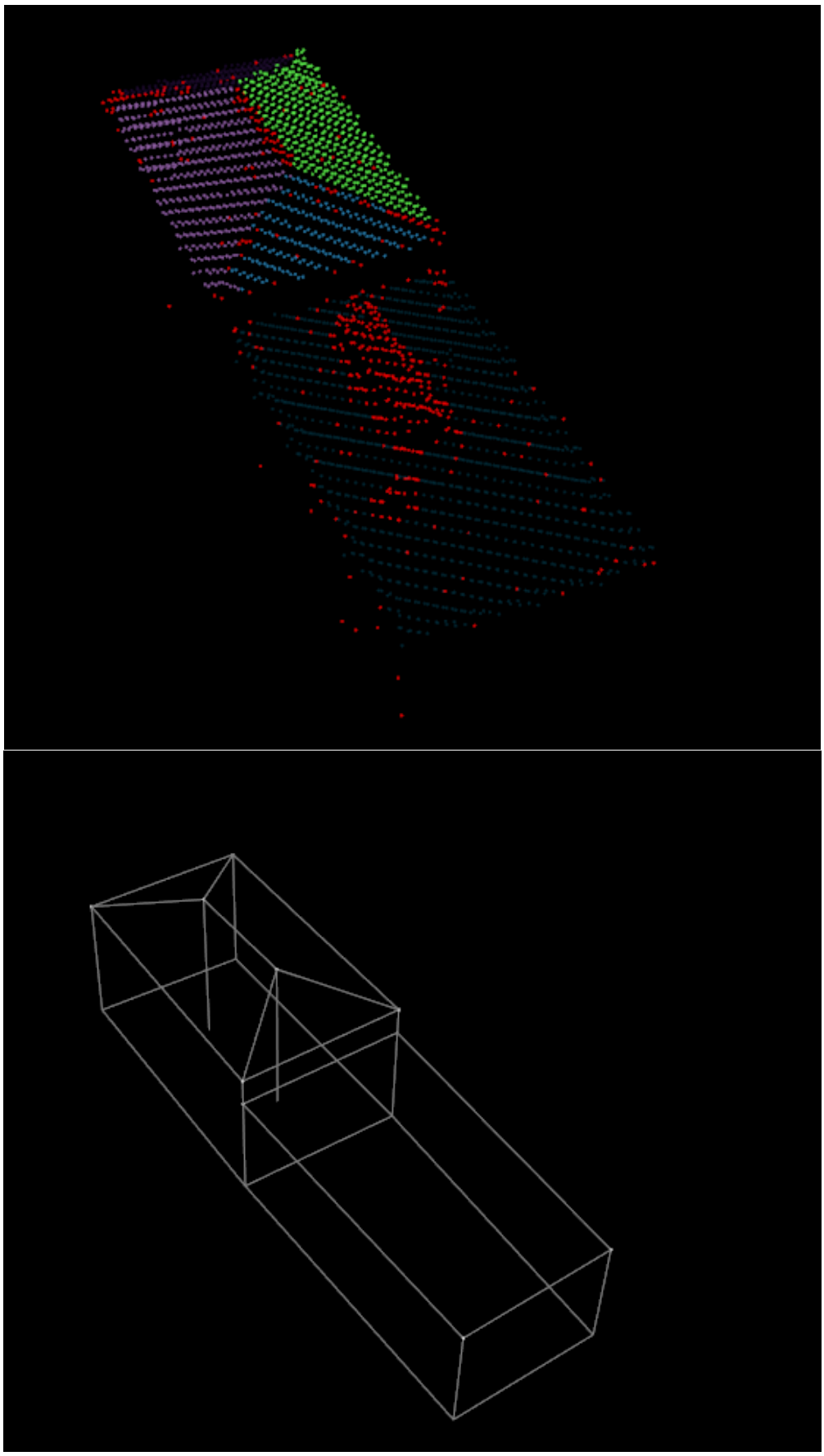


Figure 3.27 – 3D model of mixed (flat & hipped) roof building with different height levels.

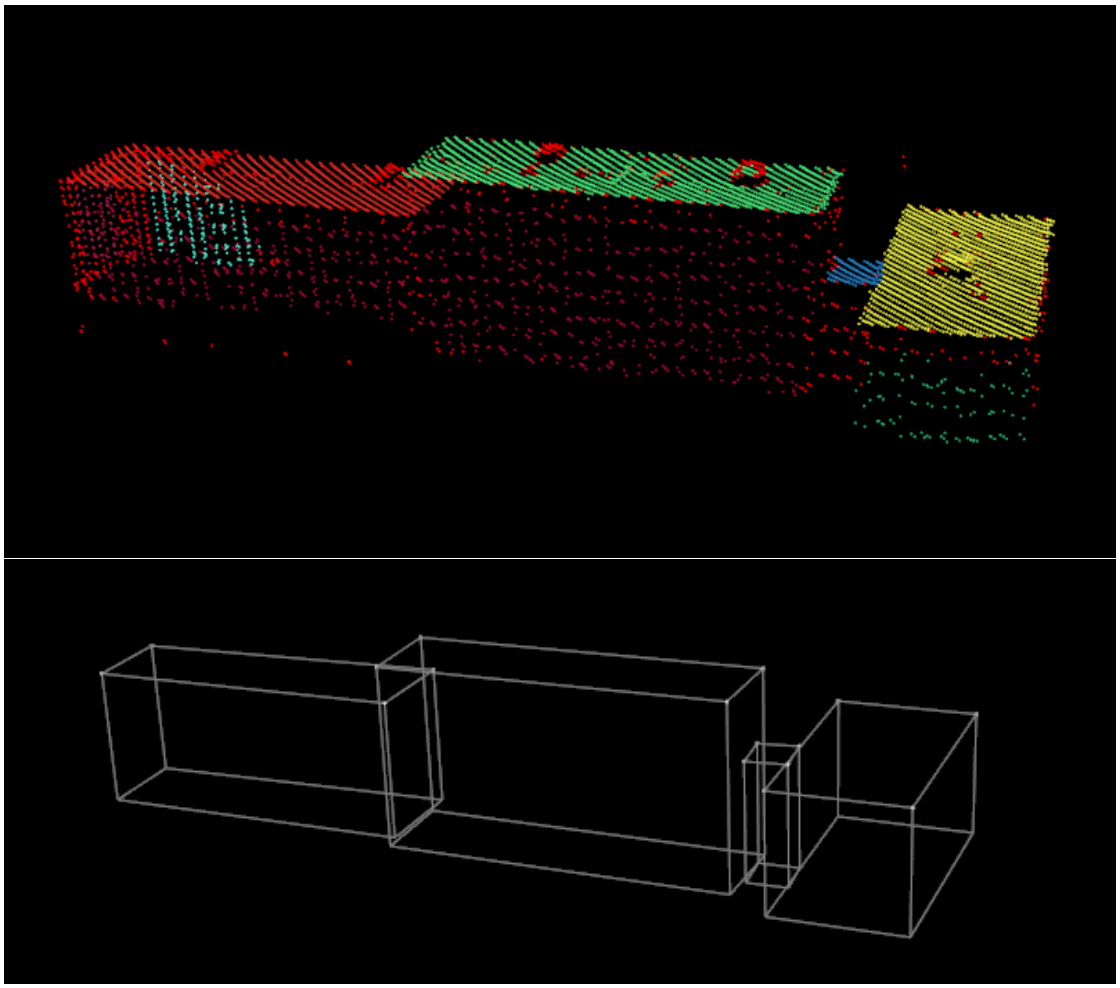


Figure 3.28 – 3D model of flat roof building with different height levels.

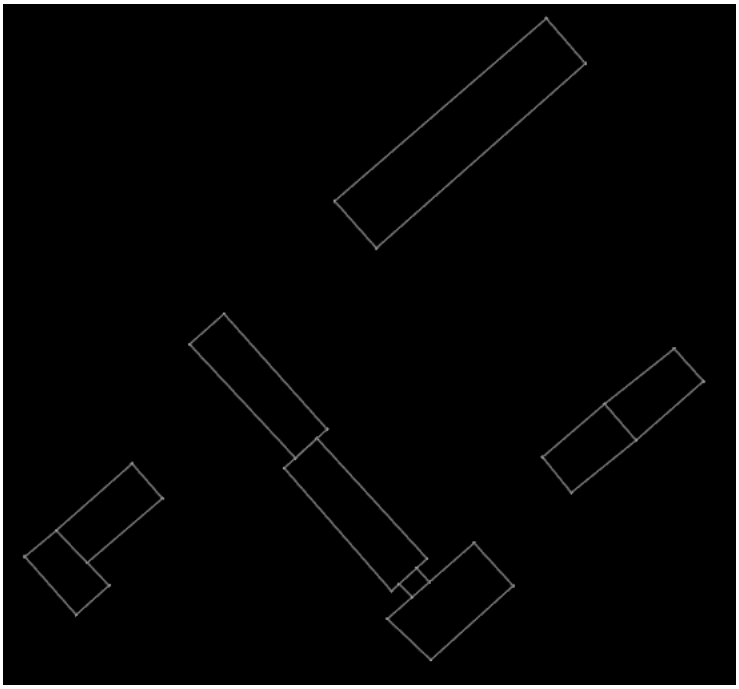


Figure 3.29 – represents LiDAR-derived buildings boundaries of different roof types of buildings (flat, gable, hipped and L-shape) collected manually together in this figure.

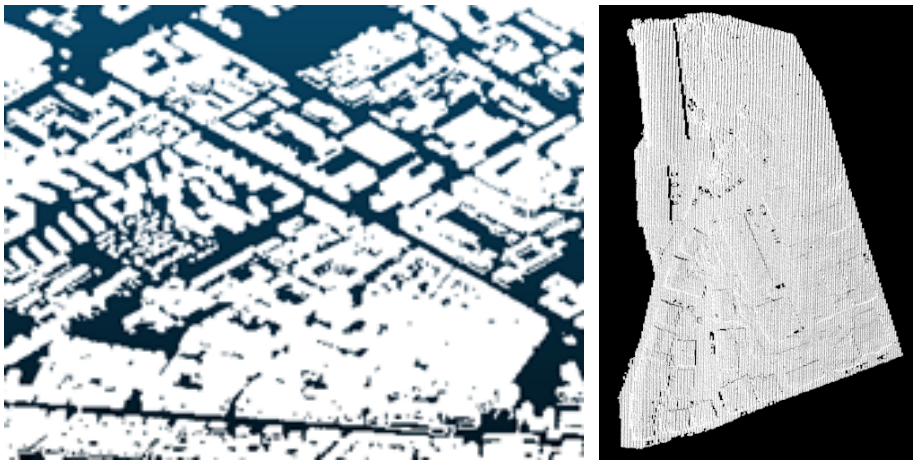
Chapter 4

Experiments and Evaluation

The proposed methodology is based on the sequential determination of individual building models from airborne LiDAR point clouds. An activity diagram of this methodology was shown in Figure 3. Input is a point cloud of the building.

4.1 Test Data

Many scenes of different data characteristics and different building complexities are chosen for testing and evaluation the proposed methodology figure 4.1. Some scenes are chosen from Munich data set which is provided in DLR. Another scenes of higher point density are chosen from the ISPRS test data set of Vaihingen. They are used to test all processing steps in the proposed methodology such as filtering, segmentation, boundary extraction and 3D reconstruction.



(a) Munich test scene (b) Vaihingen test scene

Figure 4.1 – Shows the samples of the test data point cloud.

4.2 Experiments

The test data is airborne LiDAR data of Munich with a resolution of approx. 5 pts/m2. It includes high number of buildings and high building complexity.

Tab. 4.1 – Parameters for region growing segmentation.

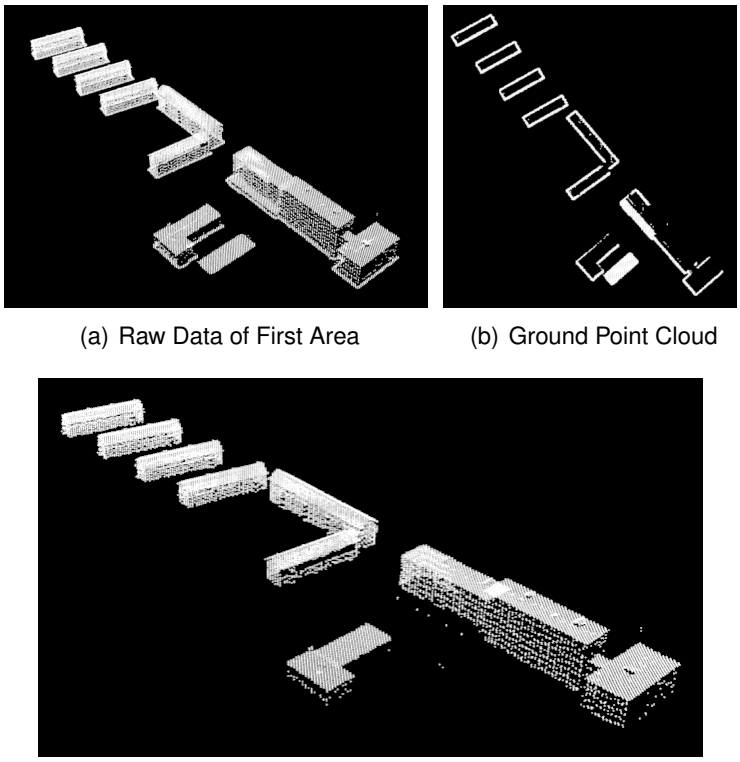
Parameter	Value
Curvature threshold	3
Angle threshold	3.7 rad
Min Cluster Size	40
Number Of Neighbours	30

Tab. 4.2 – Parameters for the progressive morphological filter.

Parameter	Value
Max window size	50
Slope value	0.19
Initial height threshold	3.5
Max height threshold	7

A **progressive morphological filter** was applied on the test data. At every step, the opening operation was applied in z direction to discard ground points. The utilized filtering parameters in this experiment are listed in table 4.2. The window size was incremented exponentially to reduce the number of iterations in aforementioned filter. It works efficiently to discard most of the ground points as shown in Figure 4.2, 4.3 and 4.4.

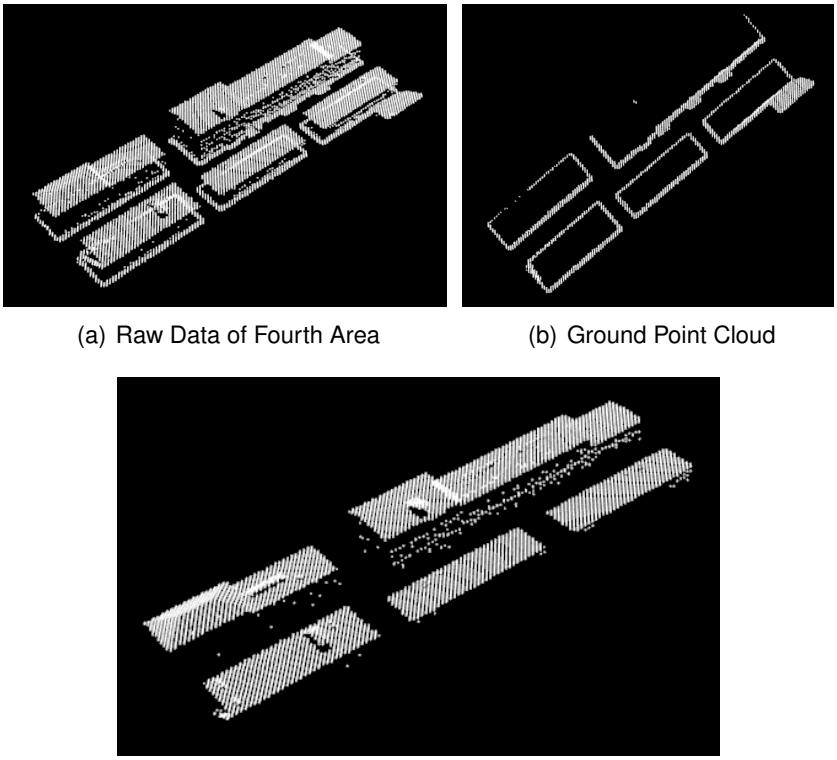
The filter is also applied on four areas of test data of Vaihingen. As it is shown in the figure 4.4.



(a) Raw Data of First Area (b) Ground Point Cloud

(c) Wall & Roof Cloud

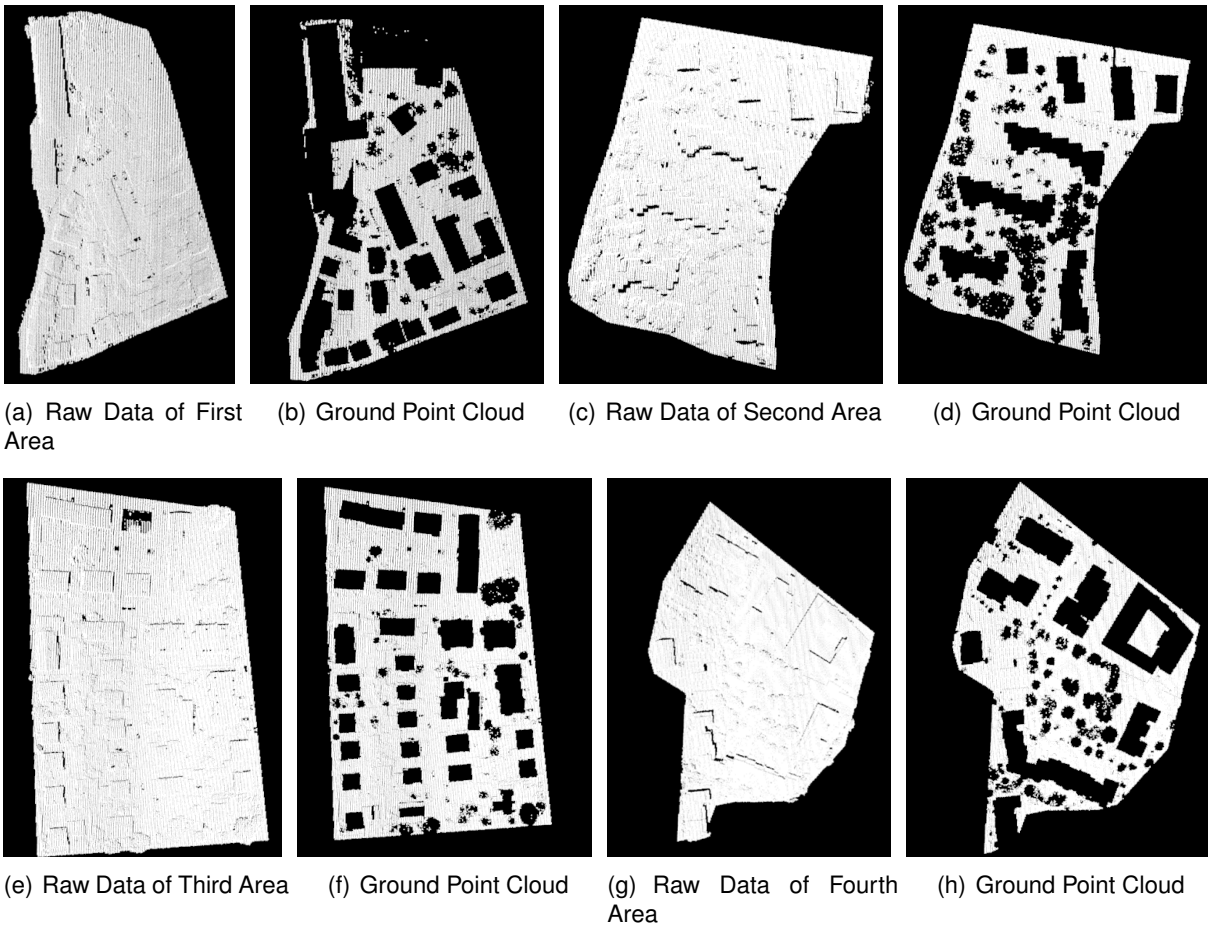
Figure 4.2 – Represents the filtering result of first of Munich data set.



(a) Raw Data of Fourth Area (b) Ground Point Cloud

(c) Wall & Roof Cloud

Figure 4.3 – Represents the filtering result of fourth area of Munich data set.

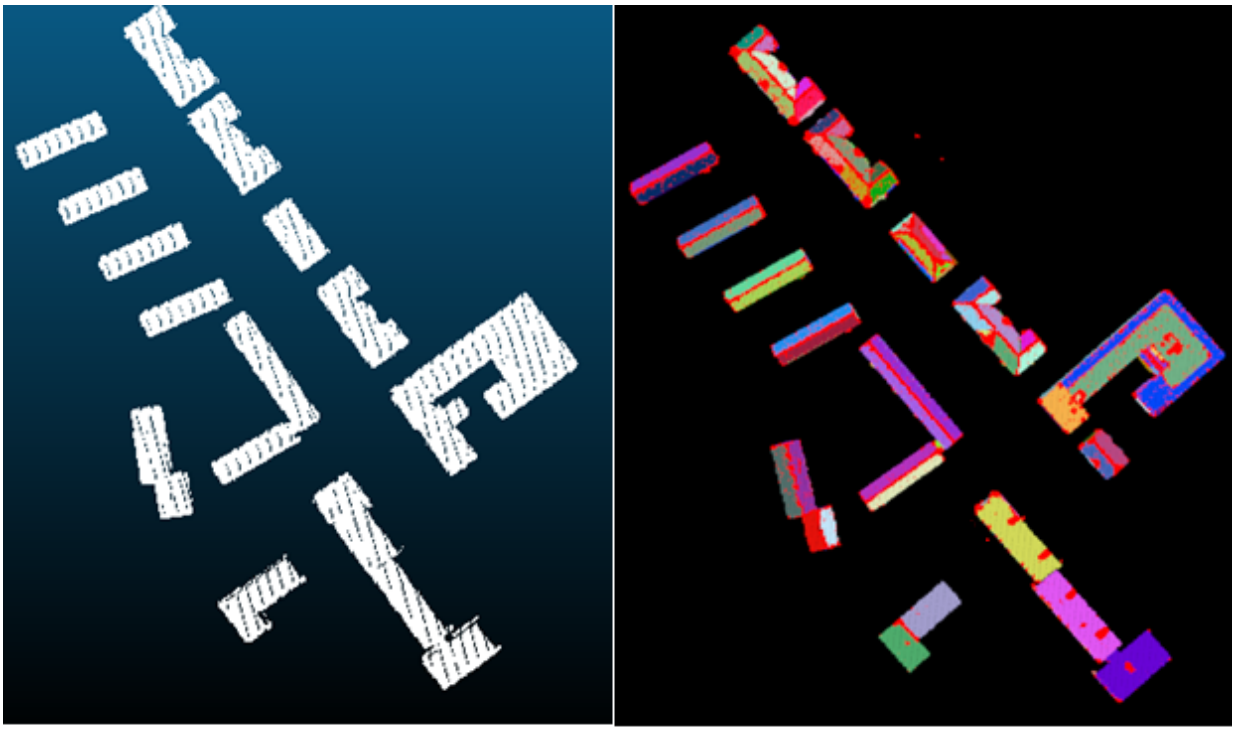


(a) Raw Data of First Area (b) Ground Point Cloud (c) Raw Data of Second Area (d) Ground Point Cloud

(e) Raw Data of Third Area (f) Ground Point Cloud (g) Raw Data of Fourth Area (h) Ground Point Cloud

Figure 4.4 – Represents the filtering result of different areas of Vaihingen data set.

Region growing segmentation algorithm was applied on the two areas of the test data. The algorithm grouped the points with similar properties in one cluster. The considered criteria are smoothness constraints which include curvature value and angle difference between normal vectors. Consequently, almost all roof segments of buildings are identified. The results of segmentation are presented in Figures 4.5 and 4.6 in which each cluster has its own color.



(a) (b)

(c)

Figure 4.5 – Depiction of the segmentation results of the first area. (a) Raw airborne LiDAR data of first area. (b) Segmented LiDAR points of the first area, different colors indicate different roof planes. (c) photo of the first area from google map.

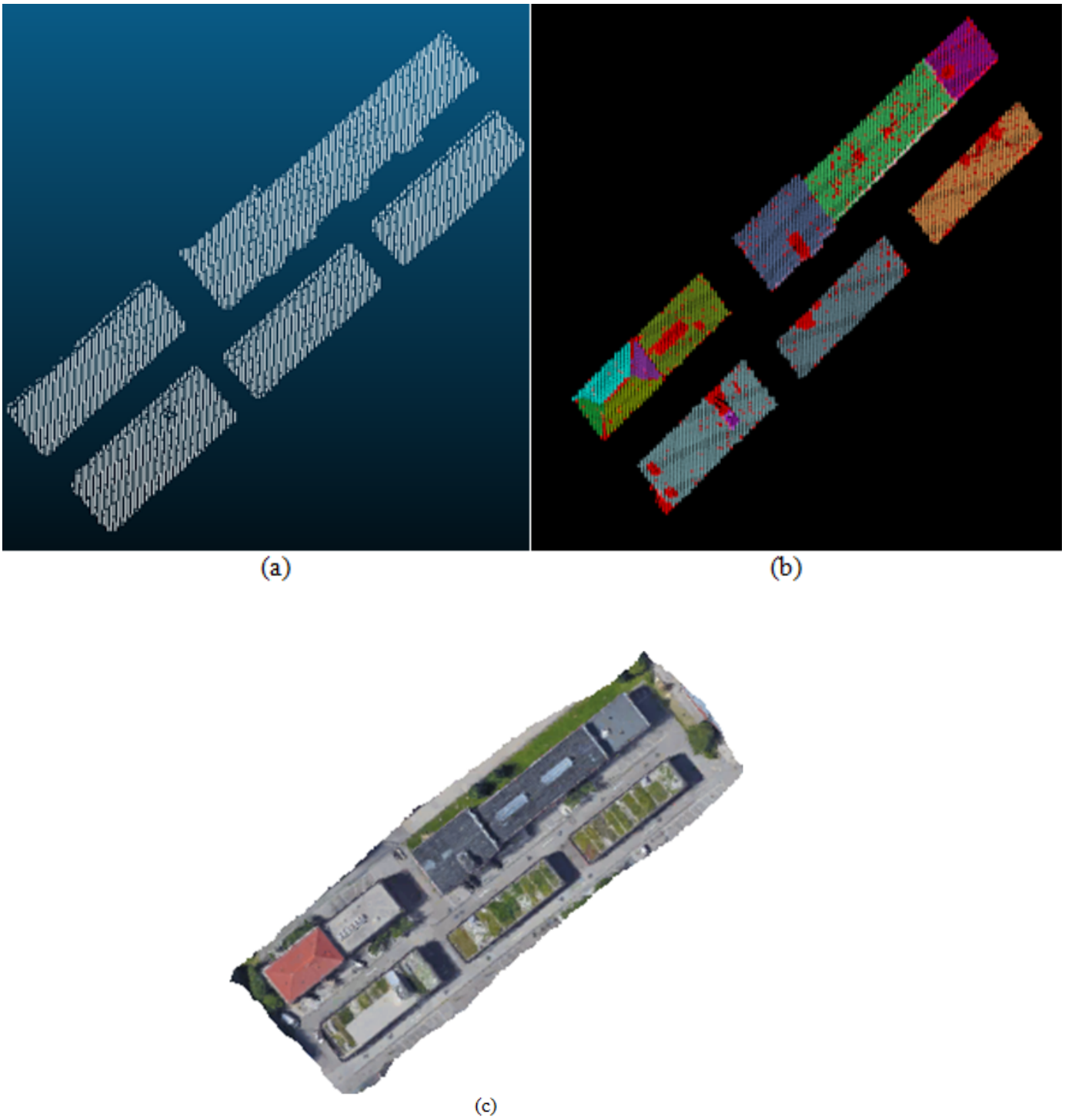
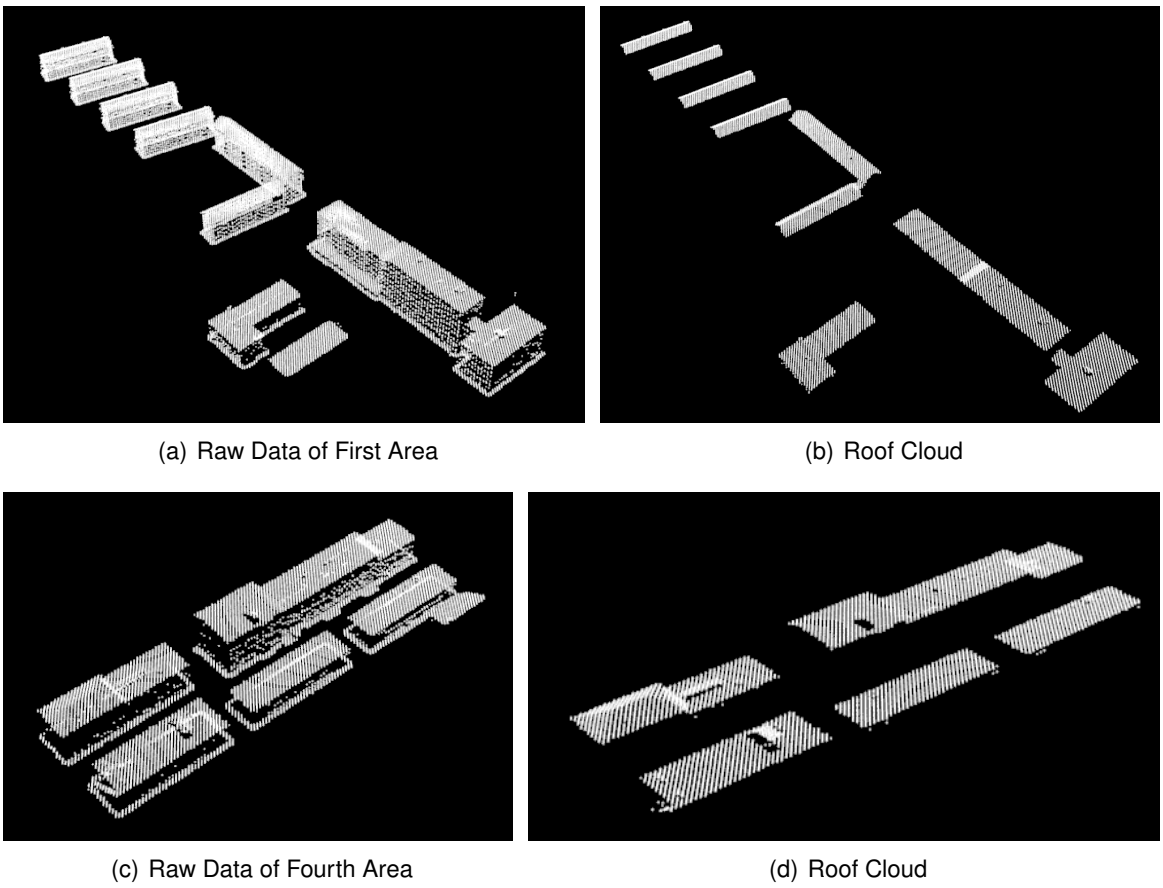


Figure 4.6 – Depiction of the segmentation results of the second area. (a) Raw airborne LiDAR data of second area. (b) Segmented LiDAR points of the second area, different colors indicate different roof planes. (c) photo of the second area from google map.

Depending on slope angle-based filtering the roof patches are extracted. Figure 4.7 shows the result of applying the algorithm on different scenes of Muich data.



(a) Raw Data of First Area (b) Roof Cloud

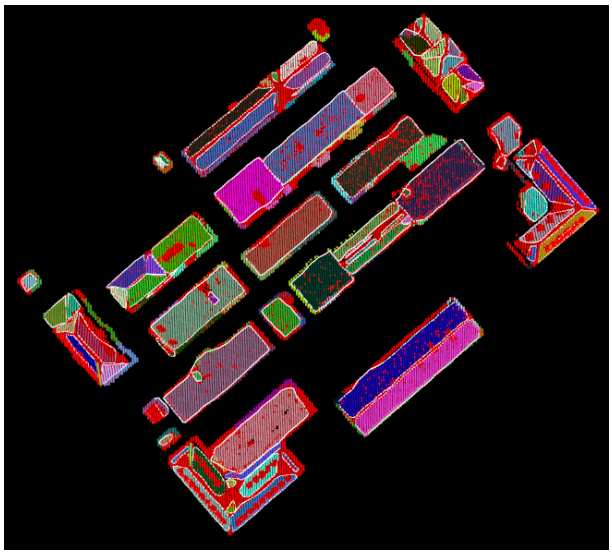
(c) Raw Data of Fourth Area (d) Roof Cloud

Figure 4.7 – Represents the roof patches as result of slope angle-based filtering of different areas of Munich data set.

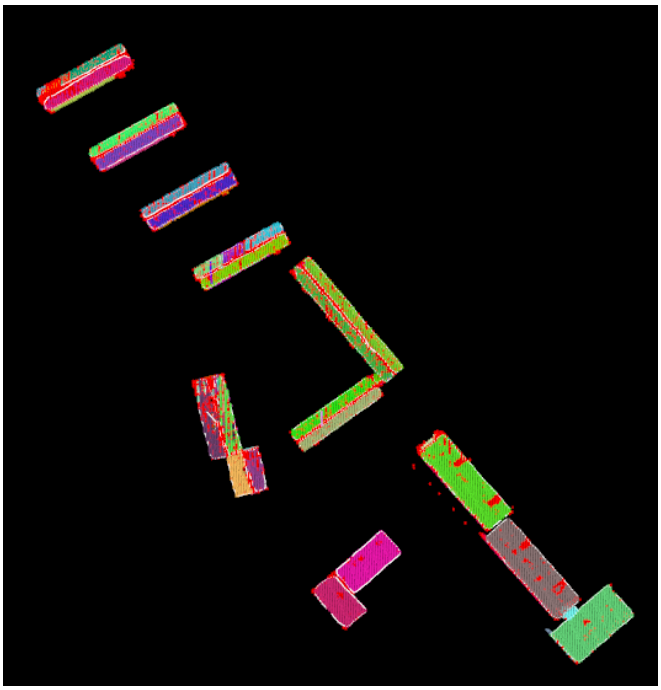
The bounding hull of each segment was extracted using **concave hull** algorithm. The following figure shows the results of concave hull algorithm, plotted over the segmentation results.

Region growing segmentation, slope analysis and concave hull algorithms in the proposed methodology are also applied respectively on the four areas of test data of Vaihingen. So the point clouds are segmented and the roof patches are determined. Then the bounding hull of each roof patch is extracted as it is shown in the figure 4.9.

3D models were extracted for different types of buildings with different height levels. The regularized outlines and the segmentation results of the roof point cloud are input to the building model generation. This approach is based on the determination of polygonal boundaries of each planar face. The following figures show the results of 3D reconstruction.



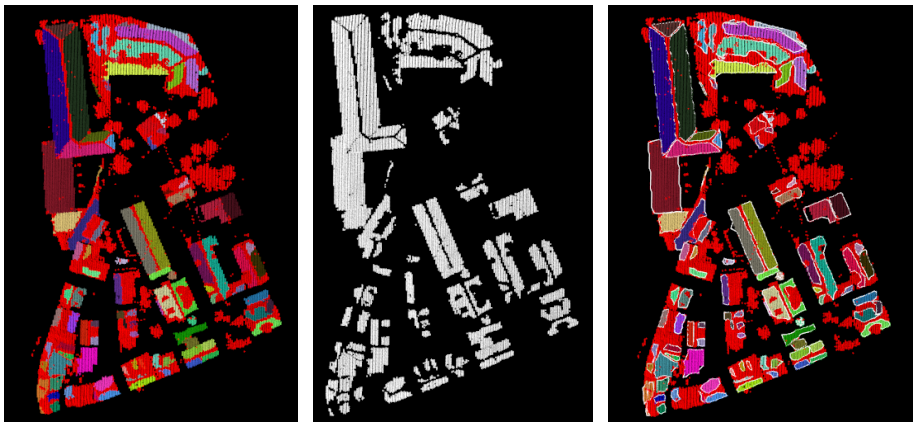
(a) Area1



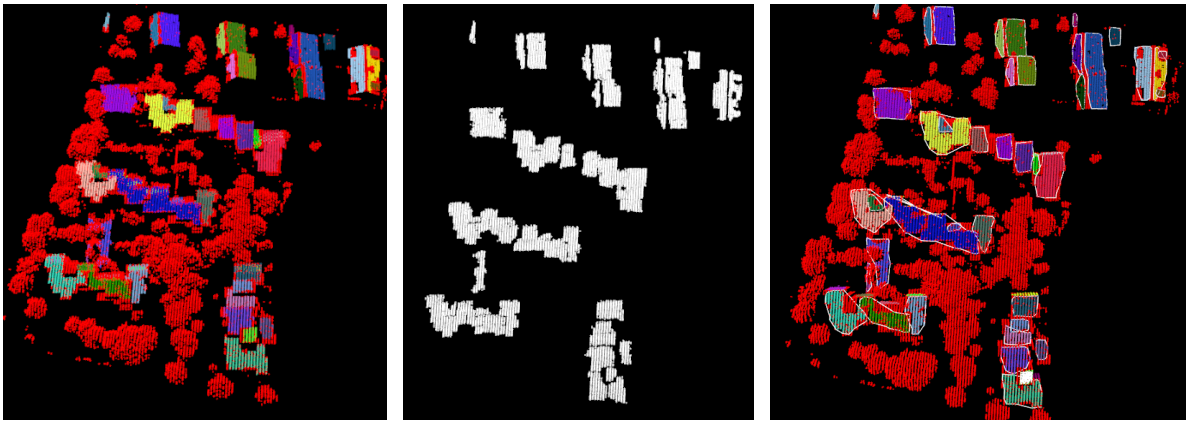
(b) Area2

Figure 4.8 – boundaries are extracted by concave hull algorithm (white lines) plotted over the segmentation results of both areas

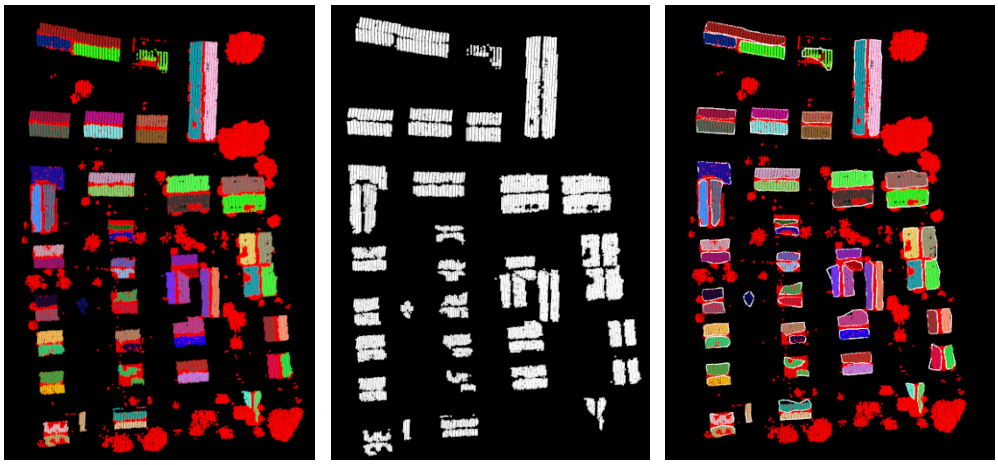
Building Footprint were extracted for different areas. The figure 4.12 shows the extracted footprint of building in different areas.



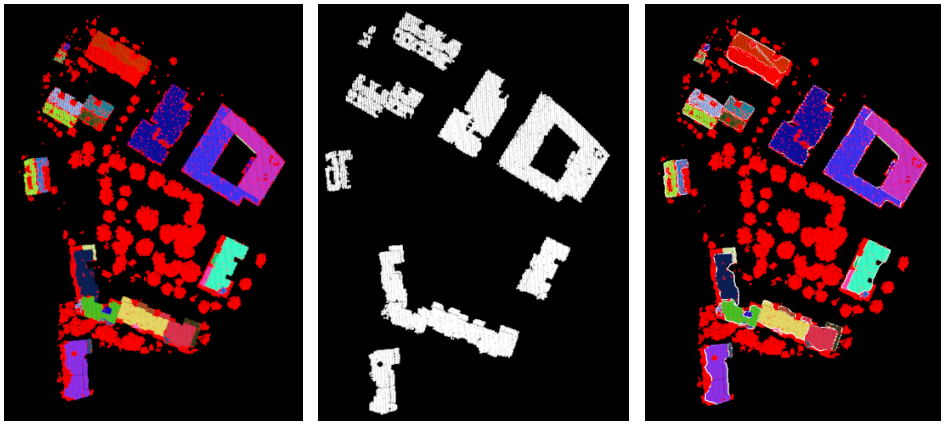
(a) Segmented First Area (b) Extracted Roof Patches (c) Extracted Bounding Hull



(d) Segmented Second Area (e) Extracted Roof Patches (f) Extracted Bounding Hull

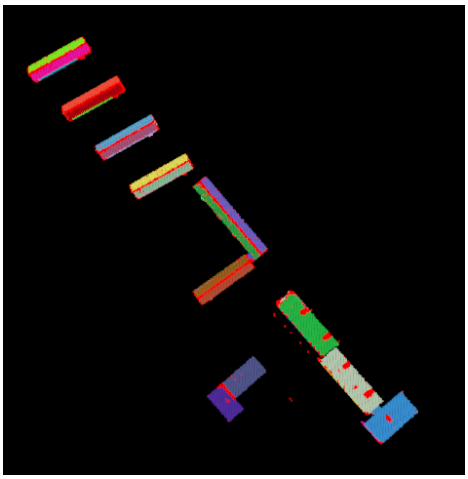


(g) Segmented Third Area (h) Extracted Roof Patches (i) Extracted Bounding Hull

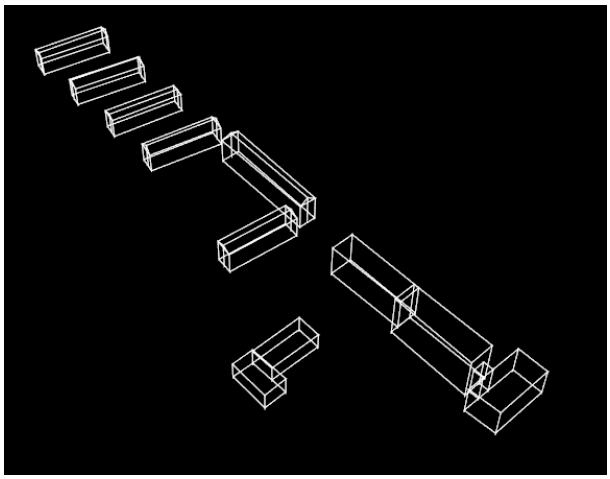


(j) Segmented Fourth Area (k) Extracted Roof Patches (l) Extracted Bounding Hull

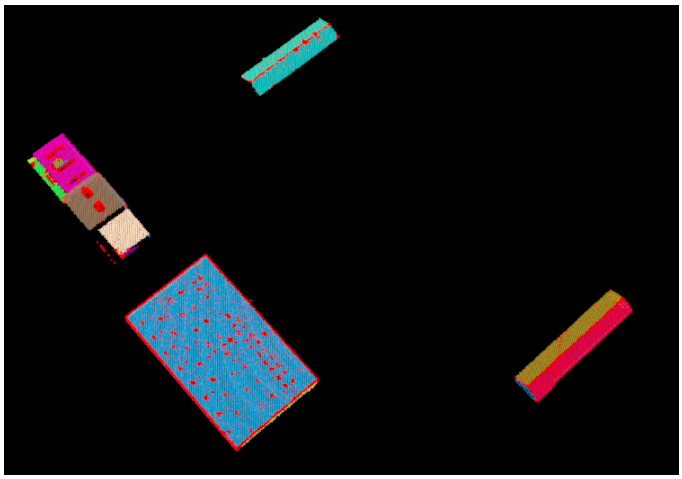
Figure 4.9 – Represents the result of segmentation, slope-angle based filtering and concave hull extraction of different areas of Vaihingen data set.



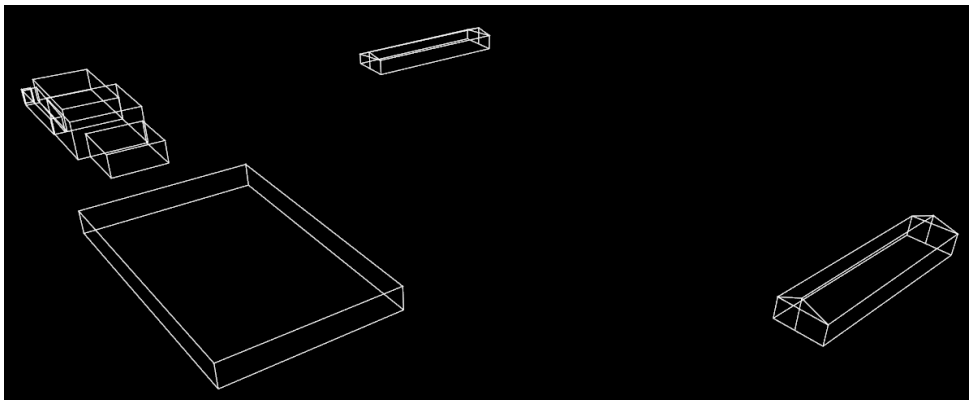
(a) First Area



(b) First Area

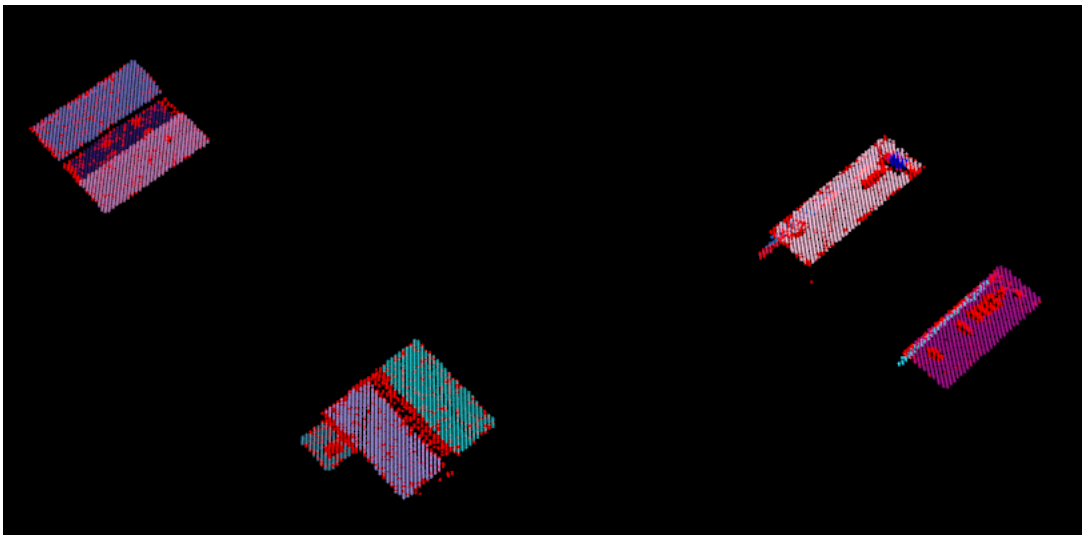


(c) Second Area

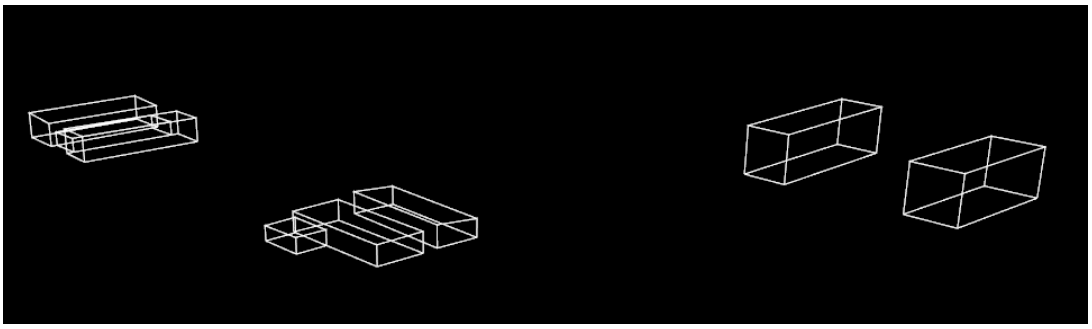


(d) Second Area

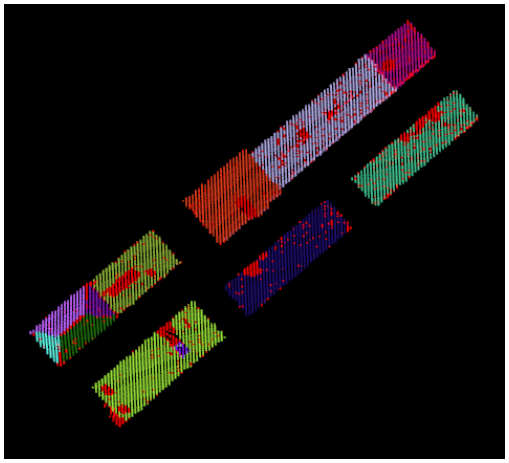
Figure 4.10 – (a) (c) represent segmentation results while (b) (d) represent 3D models.



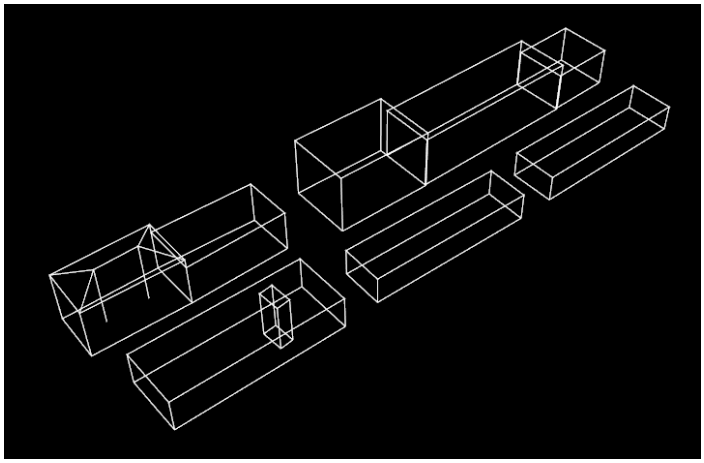
(a) Third Area



(b) Third Area



(c) Fourth Area

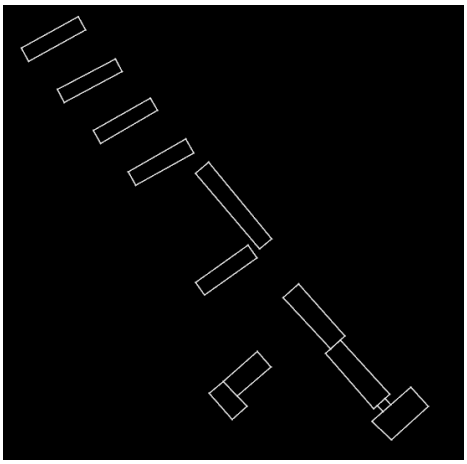


(d) Fourth Area

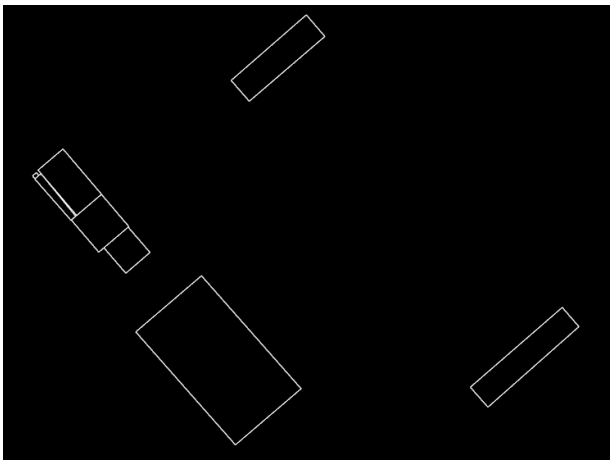
Figure 4.11 – (a) (c) represent segmentation results while (b) (d) represent 3D models.

4.3 Evaluation

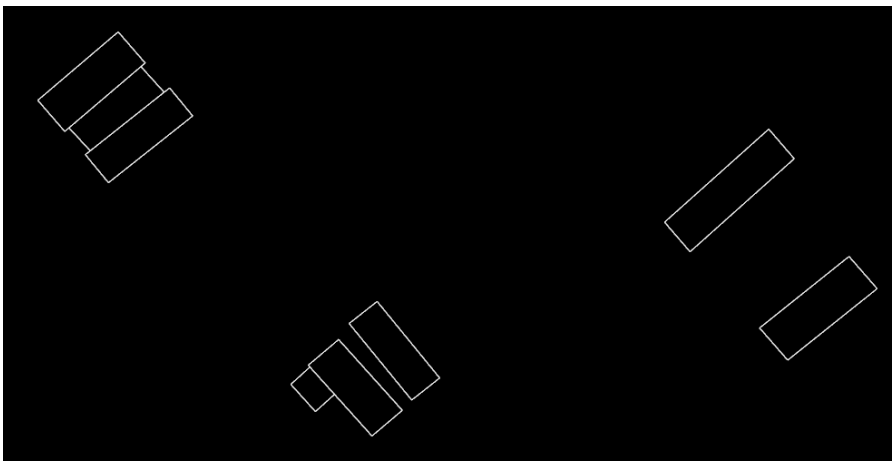
So far, all processing steps of the proposed methodology have been presented. Thereafter, evaluation is performed by determining the quality and the accuracy of the results and to decide whether a building is modeled properly or not. Both segmentation and 3D models for different types of



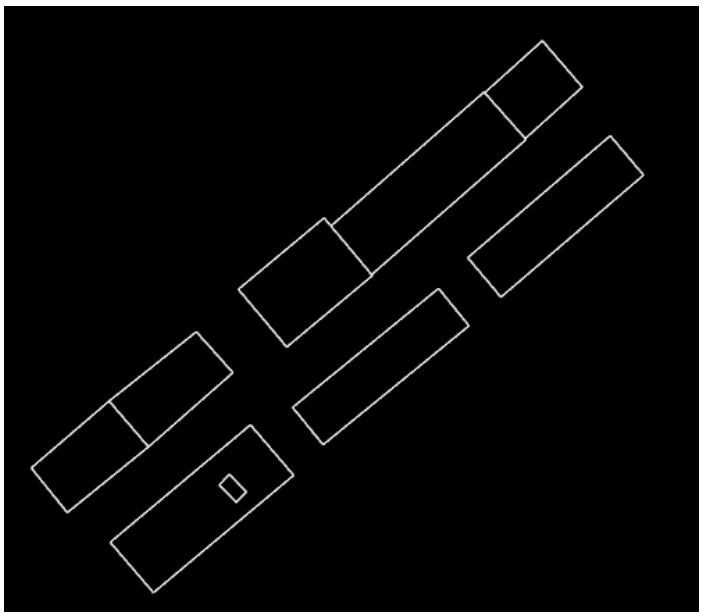
(a) First Area



(b) Second Area



(c) Third Area



(d) Fourth Area

Figure 4.12 – represent the extracted footprint of the four tested areas.

buildings are evaluated, using different methods of evaluation. The basis for the evaluation of segmentation are the 2D boundary polygons as described in sections 3.7 and 3.8. The basis for the evaluation of 3D models are the 2D-projections of 3D boundary polygons of the roof model segments and the 3D models themselves. Evaluation results can be classified to quantitative and qualitative results.

Qualitative

A check is performed if all planar segments found by the segmentation are represented by proper closed polygonal boundaries. The following figure shows the closed polygonal boundaries plotted over the points of the corresponding roof patch.

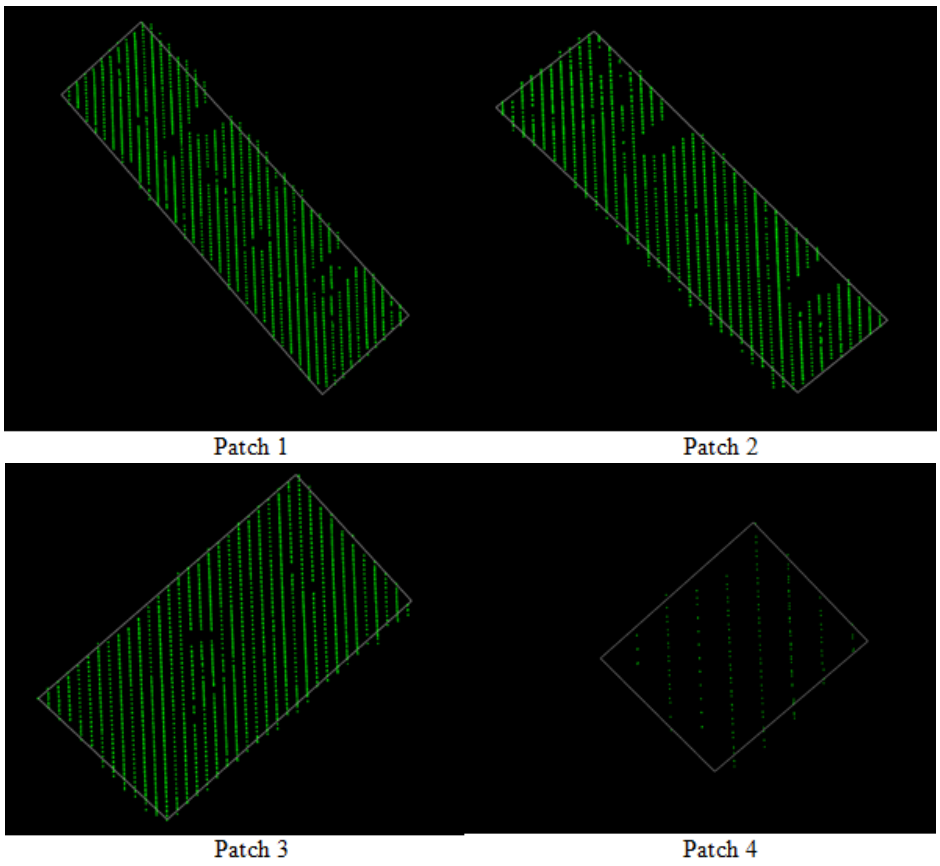


Figure 4.13 – show the closed polygonal boundaries plotted over the points of the corresponding roof patch.

2D-projections of 3D building boundary of the roof are compared with reference 2D building boundary from cadastral map using PoLiS metric method (Avbelj u. a. [2015]) Figure 4.14.

The following figure 4.15 shows visually the difference in area between the boundary that is extracted using the proposed algorithm and the reference boundary.

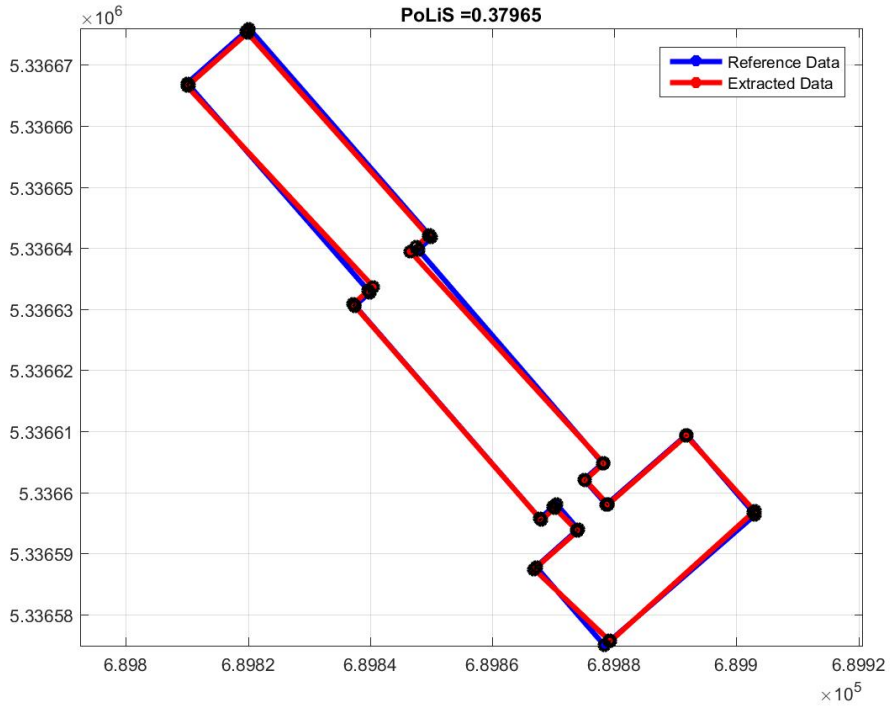


Figure 4.14 – 2D-projection of 3D building boundary of the roof is compared with reference 2D building boundary from cadastral map using PoLiS metric method.

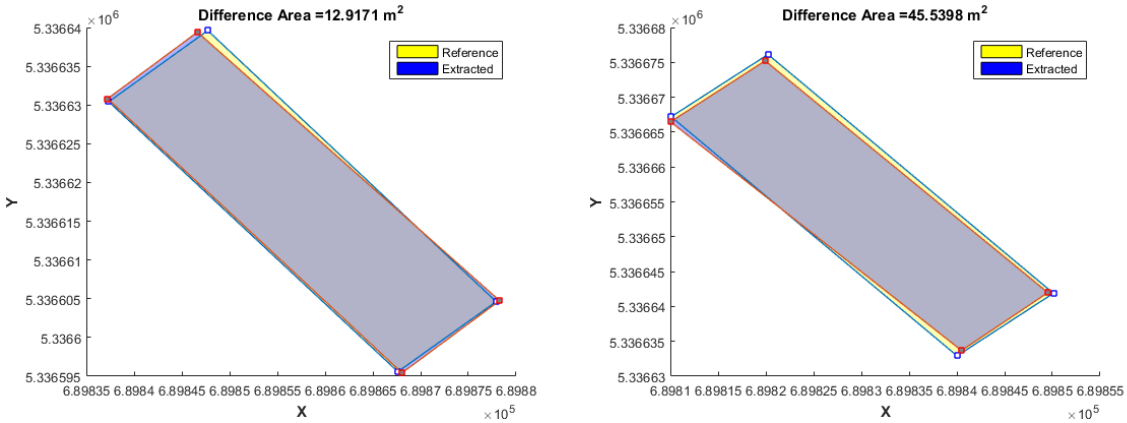
The 3D models are compared –visually- with the raw airborne LiDAR data by plotting them over the input point clouds to display how models fit to the input point cloud and the mean point density is approximately 5 pts/m2. The differences of the automatically determined model with respect to the original point cloud are shown in Figure 4.16 and 4.16 which show fits very well to the input LiDAR points.

Quantitative

Assessment of the quality of extracted 2D building polygons by comparing with reference building polygons from cadastral map using PoLiS metric method (Avbelj u. a. [2015]). The comparison of the polygons quantifies the overall average dissimilarity per polygon vetex. The PoLiS distance p estimates the similarity between polygons with different number of vertices Figure 4.18. The PoLiS distance for the roof patches and the total building boundary are listed in the table 5.

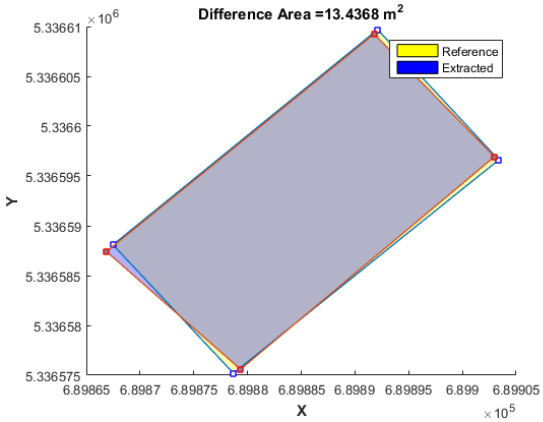
$$p(A,B) = \frac{1}{2q}\sum_{a_j \in A} min_{b \in \partial B}\|a_j - b\| + \frac{1}{2r}\sum_{b_k \in B} min_{a \in \partial A}\|b_k - a\| \tag{4.1}$$

A and B are point sets and can be considered a closed polygons. aj points of a set A represent the vertices of closed polygon A, where j=1,2,.....,q,q+1 the first and last vertices coincide, i.e., a1=aq+1. bk points of a set B represent the vertices of closed polygon B, where k=1,2,.....,r,r+1 the

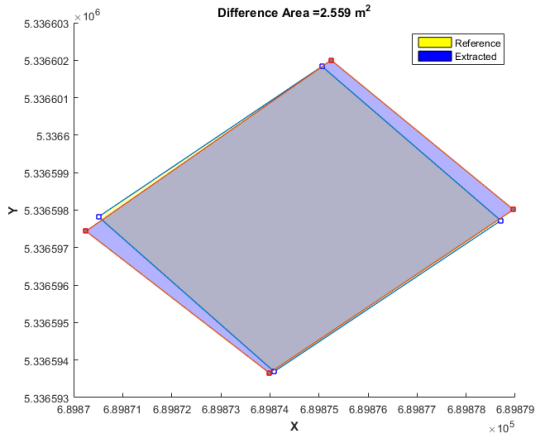


(a) Patch1

(b) Patch2



(c) Patch3



(d) Patch4

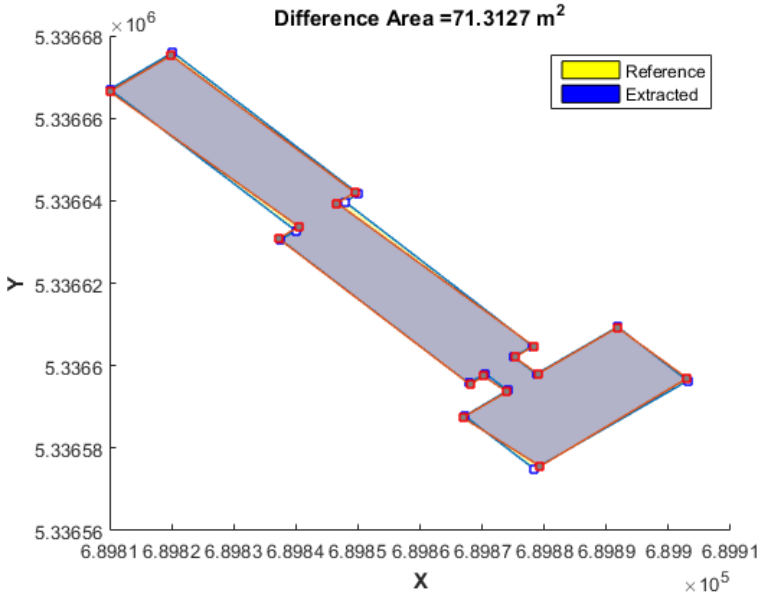
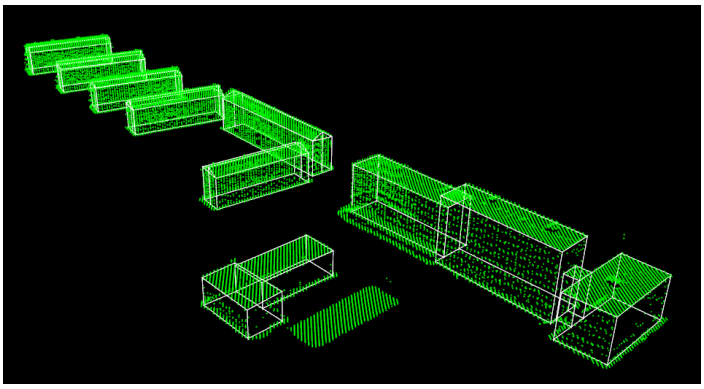
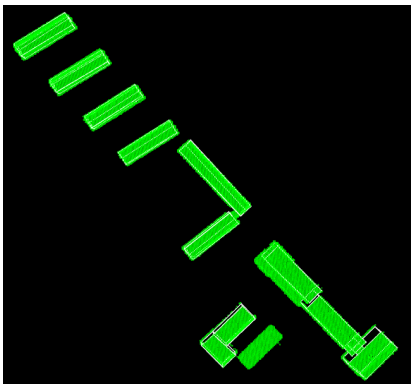


Figure 4.15 – shows visually the difference in area between and the reference boundary and the extracted boundary of patches 1,2,3,4 as well as the whole building boundary.

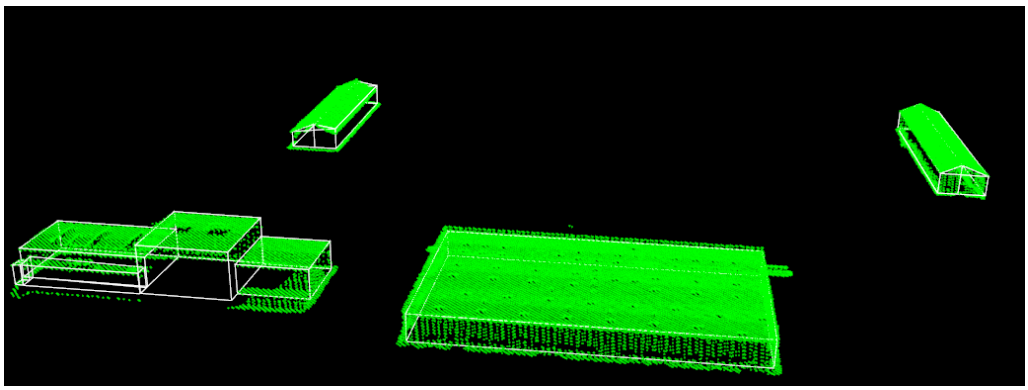
first and last vertices coincide, i.e., b1=br+1. Normalized factors (1/2q) and (1/2r) are needed to



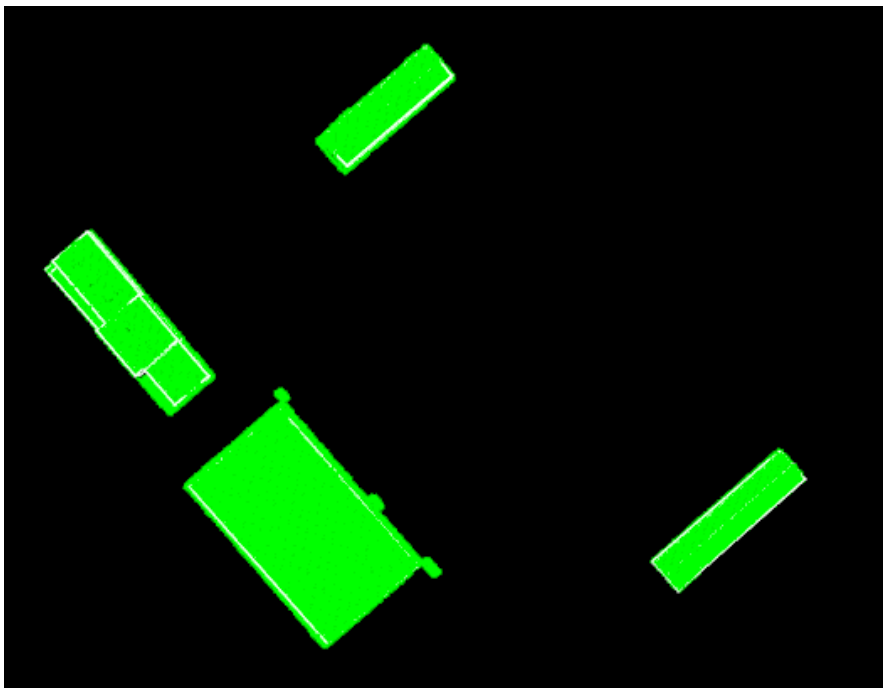
(a) Side View



(b) Top View



(c) Side View

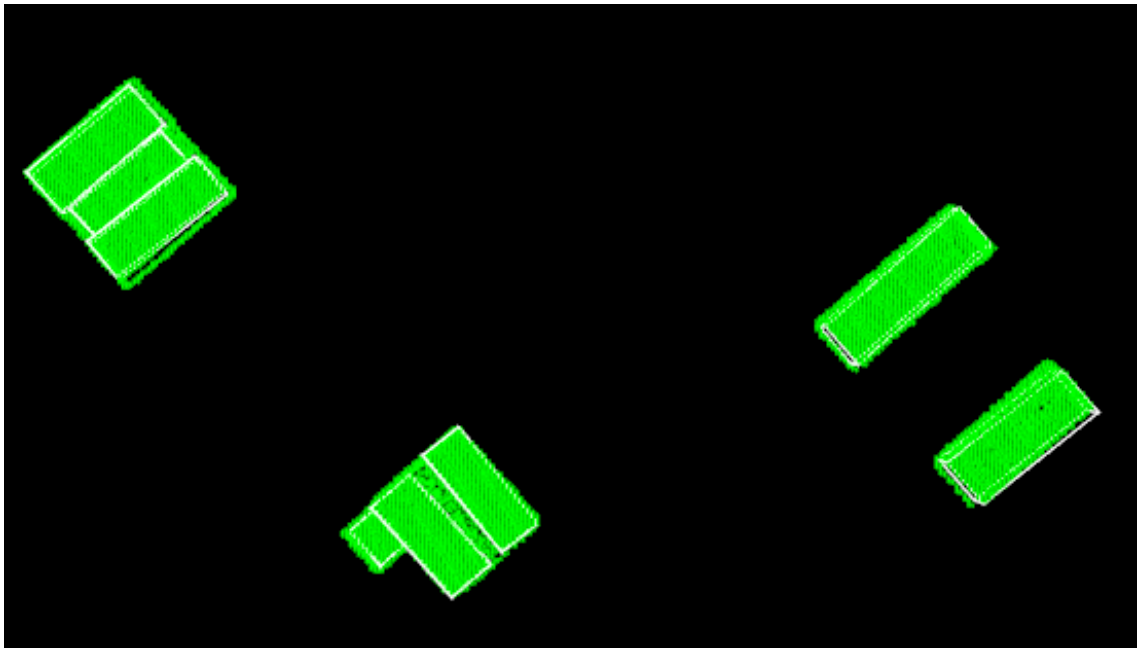


(d) Top View

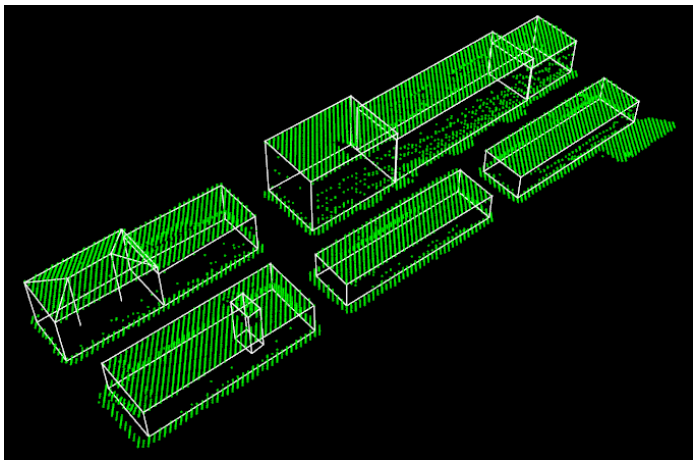
Figure 4.16 – 3D models(white) of extracted buildings over the original point clouds(green).

quantify the overall average dissimilarity per point.

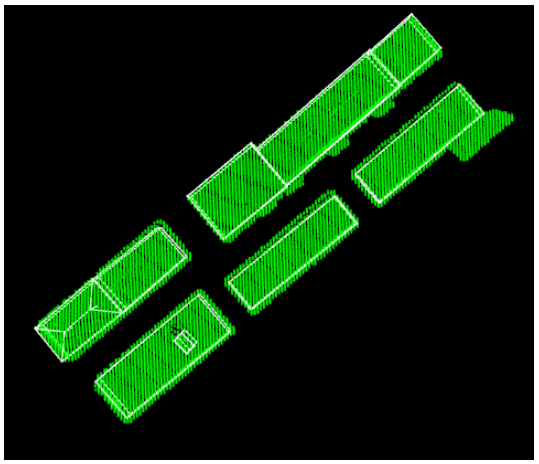
(a) Side View



(b) Top View



(c) Side View



(d) Top View

Figure 4.17 – 3D models(white) of extracted buildings over the original point clouds(green).

Reference values of the areas and perimeters of the outlines were available from cadastral map for the buildings. The area as well as the perimeter of individual roof patches boundary and the

whole footprint are computed and compared with their correspondences from cadastral map as it presents in the following table 3.2.

Table 3.2, comparison of the area and perimeter between the extracted boundary and the reference.

Patch	Area (m2)		Perimeter (m)		PoLiS
	Cadastral Map	Extracted	Cadastral Map	Extracted	
1	633.188	623.387	119.897	120.423	0.39623
2	617.390	568.880	118.363	114.764	0.57277
3	561.730	548.486	99.780	98.805	0.39112
4	32.750	36.000	22.937	24.155	0.17246
Total	1846.080	1776.753	315.309	314.604	0.53743

Root mean square estimate (RMSE) of 3D distances of the original points with respect to the 3D planes defining the roof segments are computed and listed in the following table.

Tab. 4.3 – Root mean square estimate -RMSE- of 3D distance.

Plane	RMSE(m)
Roof Patch1	0.1220
Roof Patch2	0.0812
Roof Patch3	0.2048
Roof Patch4	0.0597

Patches in the table belong to the flat roof building figure 3.28 (a). The extracted 3D planes are plotted over the original roof points. By considering a distance threshold the roof points can be classified to inliers and outliers regarding their distance to the plane Figure 4.19.

4.4 Discussion of the Regularization Process

The proposed regularization algorithm is not work efficiently for the complex boundary shape. Because of this limitation, I choose an area which is overlapped with area 1 of ISPRS test data in Vaihingen. This area has buildings which the proposed methodology is able to reconstruct 3D models of them. All sequence techniques of the proposed methodology are applied on this area. The figure 4.21 presents results of all steps respectively. During the model regularization in solving the

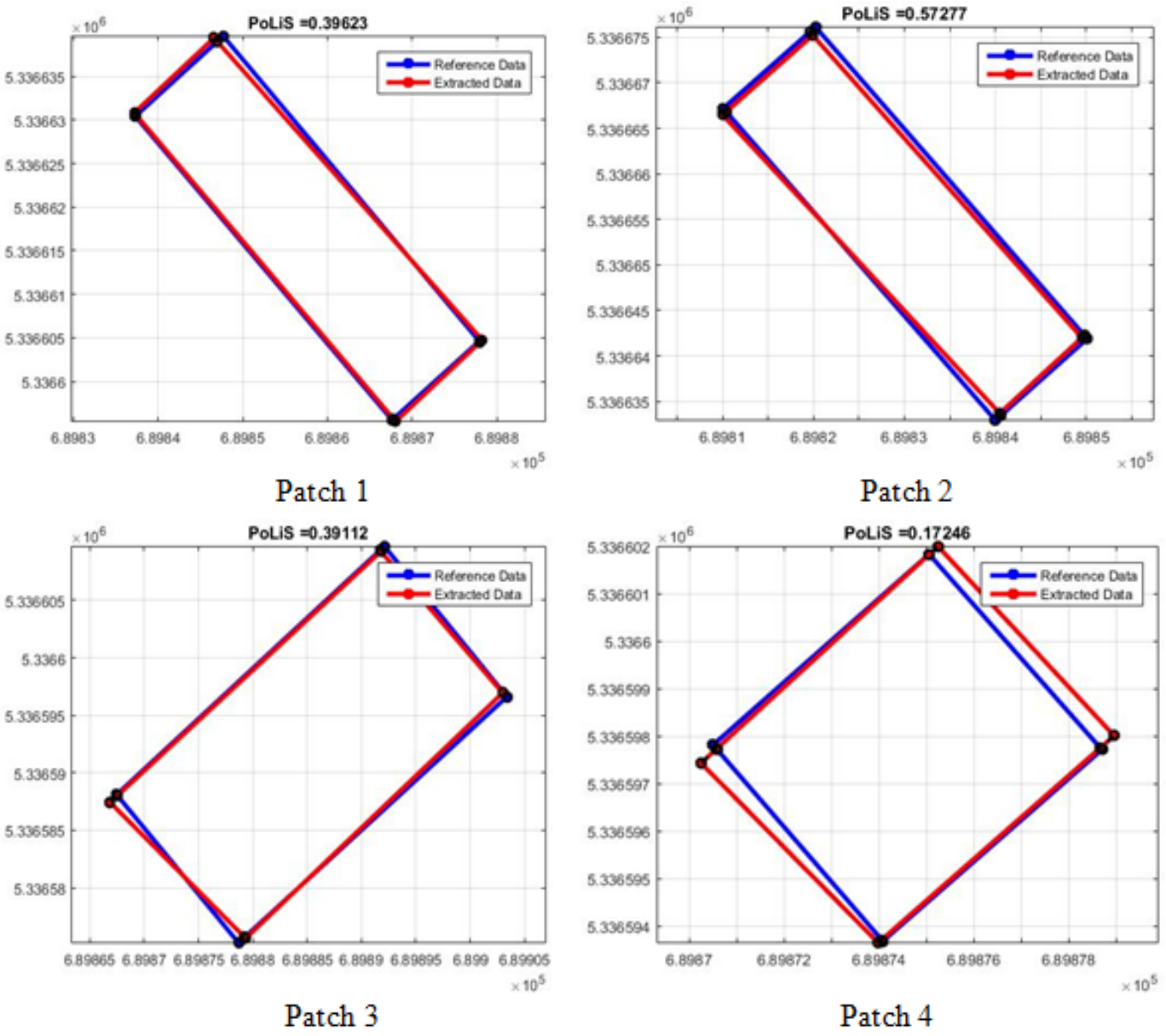
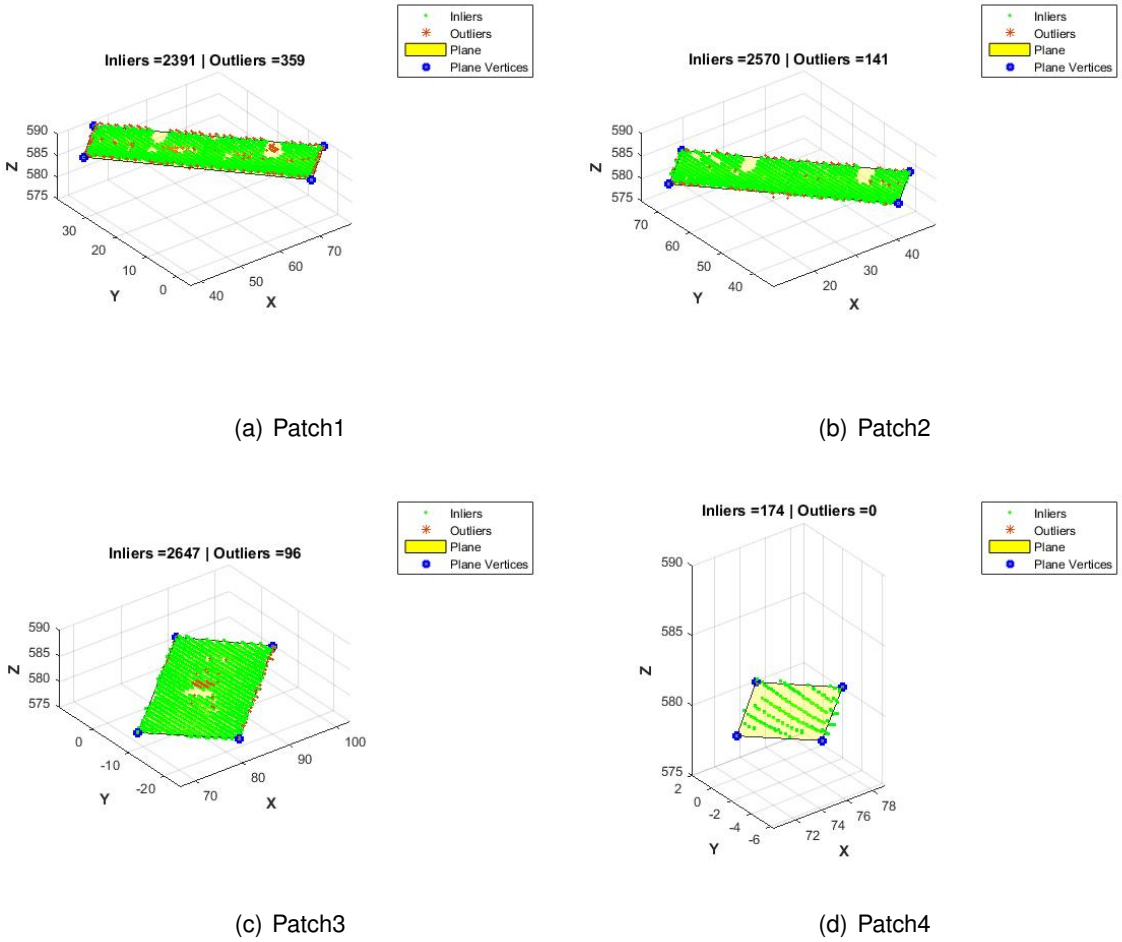


Figure 4.18 – Extracted 2D building polygons are compared with reference building polygons from cadastral map using PoLiS metric method.

connection problems among the adjacent polygons in 3D space, final 3D vertices of the model may shifted in case of having multiple adjacent polygons. As it is shown in the figure 4.20.



(a) Patch1 (b) Patch2

(c) Patch3 (d) Patch4

Figure 4.19 – Illustration how the extracted planes fit the corresponding roof patches points. (Inliers) green points while (outliers) represent points away from the corresponding plane more than 25 cm.

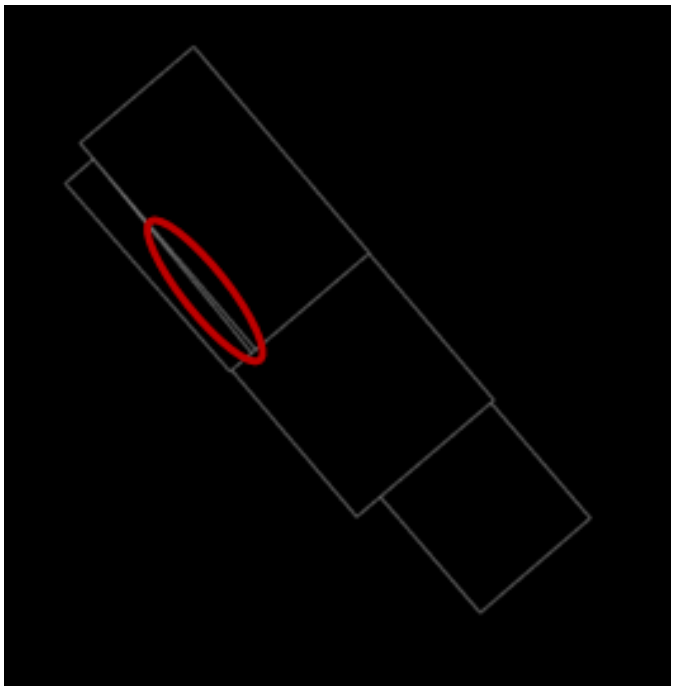
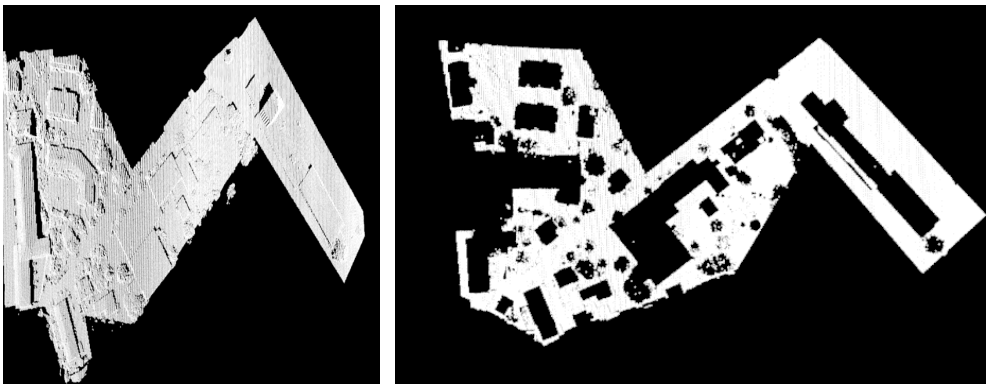
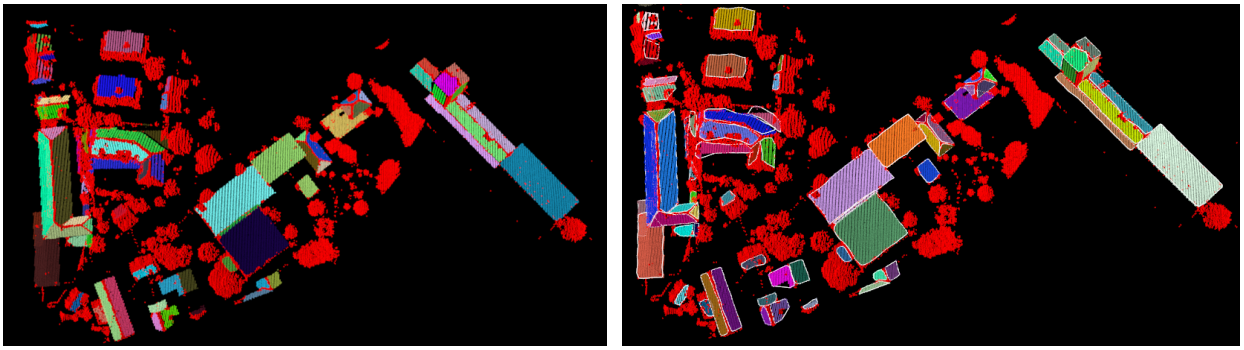


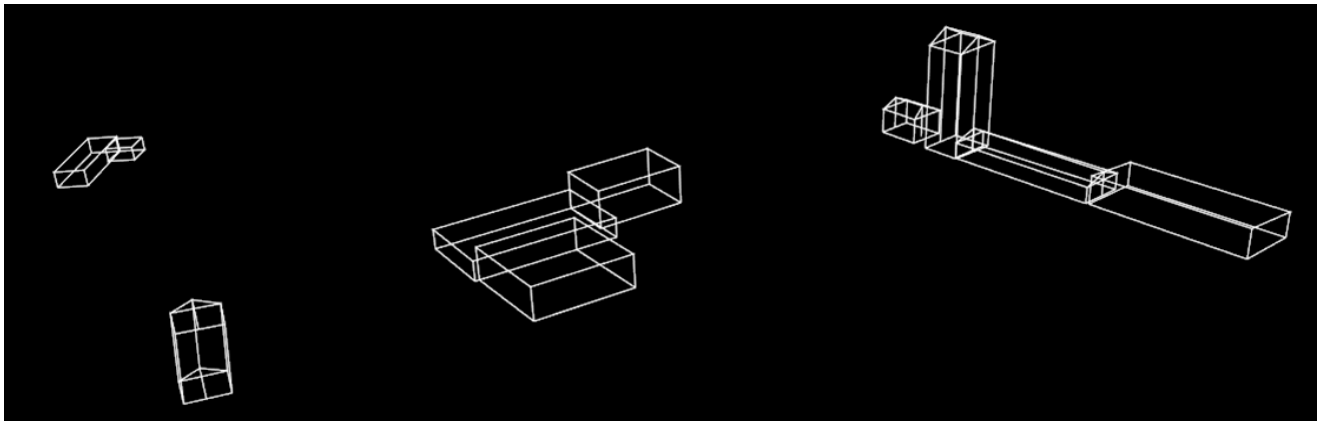
Figure 4.20 – Represents the case of shifted 3D vetices.



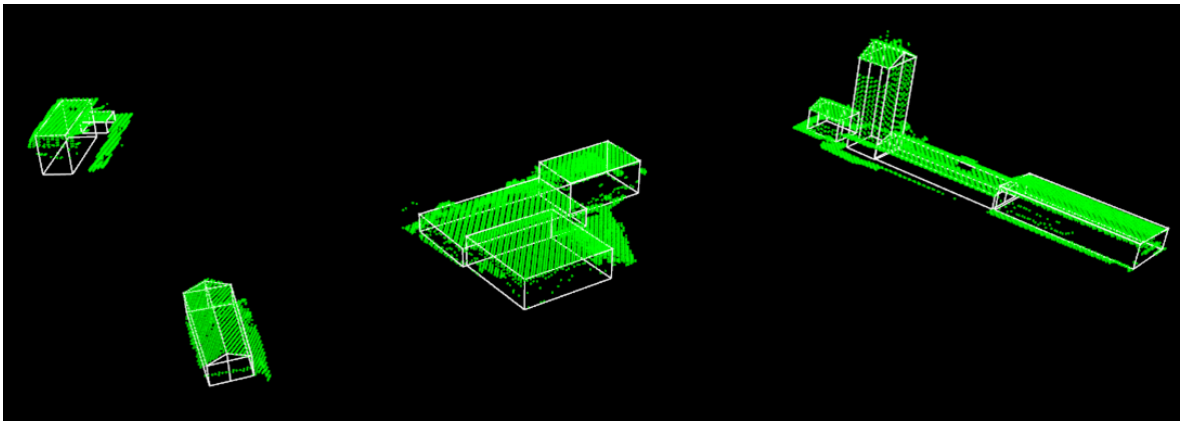
(a) Raw Data (b) Ground Point Cloud



(c) Segmented Point Cloud (d) Extracted Bounding Hull



(e) 3D Model



(f) 3D Model over original point clouds

Figure 4.21 – Represents the results of applying the proposed methodology on specific area of Vaihingen data set.

Chapter 5

Conclusion

The aim of this research is to design a methodology that reconstructs buildings with different height levels in urban areas using merely LiDAR data. The procedure is restricted to not require any other source of data rather than LiDAR. This was done intentionally to avoid the limitation of availability of other sources of information in some areas. Where sources such as ground plans, satellite imagery, airborne imagery and multispectral data are not available for every desired site.

So far, the proposed methodology, for the determination of 3D building models from Airborne LiDAR data, is presented. All subsequent steps (filtering, segmentation, extracting roof patches, outlines extraction and regularization and building model generation) are applied automatically. The applicability of all individual processing steps were demonstrated. Since the results of all evaluation techniques are plausible, the tested buildings are properly and completely modeled. The comparison of the extracted boundary polygons and reference data using PoLiS metric has shown, that a low average dissimilarity between the reference polygons and the automatically extracted boundary polygons. The comparison of the extracted models and the original LiDAR point clouds is also shown the reliability of the results.

However, some difficulties were encountered which are discussed as below. Although the segmentation procedure shows successful results, it might fail to segment roof regions in some areas. Areas where the roof segment size is not large enough to contain enough LiDAR points to estimate reliable geometrical parameters of the segment which might lead to inaccurate roof segments. In such cases, increasing the data density might alleviate this obstacle to a certain extent. In roof polygons extraction, the performance of the process was successful especially with Square-shape roof regions as shown experimental results. However, some nodes might be shifted from their true position during the joining and connecting of the roof planar segments especially with complex buildings.

The used approach is data-driven. Hence, the resulting models aim at well approximation of the given LiDAR point clouds. The sequence used techniques have demonstrated the ability to reconstruct buildings with different height levels from airborne LiDAR data as an alone data source. In other words, the proposed methodology works for 3D reconstruction of a set of buildings with different height levels including (rectangular, L)-shape buildings and building with (flat, gable or

hipped) roof. Also mixed of flat and hipped roof building is considered. Compared to model driven approaches, a data-driven approach is more flexible since it reconstructs complex buildings regardless of its predefined form. But Anywise, it is known that data-driven approach is suffering of probable risks of obtaining disturbed models for unspecific buildings. While model-driven approach is restricted by predefined library. Data-driven approach generally requires assumptions. In this research one of the assumptions is that the buildings can be reconstructed by a composition of planar faces. Another assumption is that building edges are mostly orthogonal or parallel to the main building direction. Moreover, the point clouds in general comprise certain random and systematic errors. Thus,the used thresholds in the proposed methodology may need to change for another LiDAR data for better efficiency in reconstruction process.

The 3D building generation using data-driven approach is still heavily under construction. Future work of this research, can be performed by improving the methodology to use less thresholding parameters to increase level of automation. And more robust contraints may consider in the regularization process in order to regularize more complex boundary shapes.

Bibliography

[Agarwal u. a. 2006] AGARWAL, Pankaj K. ; ARGE, Lars ; DANNER, Andrew: From point cloud to grid DEM: A scalable approach. In: *Progress in Spatial Data Handling*. Springer, 2006, S. 771–788

[Albers u. a. 2016] ALBERS, Bastian ; KADA, Martin ; WICHMANN, Andreas: Automatic Extraction and Regularization of Building Outlines from Airborne LIDAR Point Clouds. In: *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* (2016), S. 555–560

[Alharthy u. a. 2004] ALHARTHY, A ; BETHEL, J ; MIKHAIL, EM: *Analysis and accuracy assessment of airborne laserscanning system, ISPRS Congress Istanbul 2004, July 12-23, 2004, ISPRS*. 2004

[Alharthy u. Bethel 2002] ALHARTHY, Abdullatif ; BETHEL, James: Heuristic filtering and 3D feature extraction from LIDAR data. In: *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences* 34 (2002), Nr. 3/A, S. 29–34

[Alharthy[1] u. Bethel 2004] ALHARTHY[1], Abdullatif ; BETHEL, James: Detailed building reconstruction from airborne laser data using a moving surface method. (2004)

[Amenta u. a. 1998] AMENTA, Nina ; BERN, Marshall ; KAMVYSSELIS, Manolis: A new Voronoi-based surface reconstruction algorithm. In: *Proceedings of the 25th annual conference on Computer graphics and interactive techniques* ACM, 1998, S. 415–421

[Ameri u. Fritsch 2000] AMERI, Babak ; FRITSCH, Dieter: Automatic 3D building reconstruction using plane-roof structures. In: *ASPRS, Washington DC* (2000)

[Arefi u. a. 2008] AREFI, H u. a.: Levels of detail in 3d building reconstruction from lidar data. (2008)

[Arefi u. a. 2010] AREFI, Hossein ; HAHN, Michael ; REINARTZ, Peter: Ridge based decomposition of complex buildings for 3D model generation from high resolution digital surface models ISPRS, 2010

[Asaeedi u. a. 2013] ASAEEDI, Saeed ; DIDEHVAR, Farzad ; MOHADES, Ali: Alpha-Concave Hull, a Generalization of Convex Hull. In: *arXiv preprint arXiv:1309.7829* (2013)

[Attali u. Boissonnat 2002] ATTALI, Dominique ; BOISSONNAT, Jean-Daniel: A linear bound on the complexity of the Delaunay triangulation of points on polyhedral surfaces. In: *Proceedings of the seventh ACM symposium on Solid modeling and applications* ACM, 2002, S. 139–146

[Avbelj u. a. 2015] AVBELJ, Janja ; MÜLLER, Rupert ; BAMLER, Richard: A metric for polygon comparison and building extraction evaluation. In: *IEEE Geoscience and Remote Sensing Letters* 12 (2015), Nr. 1, S. 170–174

[Awrangjeb u. Fraser 2014] AWRANGJEB, Mohammad ; FRASER, Clive S.: Automatic segmentation of raw LiDAR data for extraction of building roofs. In: *Remote Sensing* 6 (2014), Nr. 5, S. 3716–3751

[Bajaj u. a. 1995] BAJAJ, Chandrajit L. ; BERNARDINI, Fausto ; XU, Guoliang: Automatic reconstruction of surfaces and scalar fields from 3D scans. In: *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques* ACM, 1995, S. 109–118

[Bernardini u. a. 1999] BERNARDINI, Fausto ; MITTLEMAN, Joshua ; RUSHMEIER, Holly ; SILVA, Cláudio ; TAUBIN, Gabriel: The ball-pivoting algorithm for surface reconstruction. In: *IEEE transactions on visualization and computer graphics* 5 (1999), Nr. 4, S. 349–359

[Bogdan u. a. 2009] BOGDAN, Radu u. a.: Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments. (2009)

[Boissonnat 1984] BOISSONNAT, Jean-Daniel: Geometric structures for three-dimensional shape representation. In: *ACM Transactions on Graphics (TOG)* 3 (1984), Nr. 4, S. 266–286

[Brenner 2000] BRENNER, Claus: Towards fully automatic generation of city models. In: *International Archives of Photogrammetry and Remote Sensing* 33 (2000), Nr. B3/1; PART 3, S. 84–92

[Bykat 1978] BYKAT, Alex: Convex hull of a finite set of points in two dimensions. In: *Information Processing Letters* 7 (1978), Nr. 6, S. 296–298

[DOIHARA u. SHIBASAKI] DOIHARA, KAZUO ODA TADASHI TAKANO T. ; SHIBASAKI, RYOSUKE: AUTOMATIC BUILDING EXTRACTION AND 3-D CITY MODELING FROM LIDAR DATA BASED ON HOUGH TRANSFORMATION.

[Dorninger u. Pfeifer 2008] DORNINGER, Peter ; PFEIFER, Norbert: A comprehensive automated 3D approach for building extraction, reconstruction, and regularization from airborne laser scanning point clouds. In: *Sensors* 8 (2008), Nr. 11, S. 7323–7343

[Drauschke u. a. 2009] DRAUSCHKE, Martin ; ROSCHER, Ribana ; LABE, T ; FORSTNER, W: Improving image segmentation using multiple view analysis. In: *Object Extraction for 3D City Models, Road Databases and Traffic Monitoring-Concepts, Algorithms and Evaluatin (CMRT09)* (2009), S. 211–216

[Drauschke u. a. 2006] DRAUSCHKE, Martin ; SCHUSTER, Hanns-Florian ; FÖRSTNER, Wolfgang: Detectability of buildings in aerial images over scale space. In: *PCV06, IAPRS* 36 (2006), Nr. 3, S. 7–12

[Duda u. Hart 1972] DUDA, Richard O. ; HART, Peter E.: Use of the Hough transformation to detect lines and curves in pictures. In: *Communications of the ACM* 15 (1972), Nr. 1, S. 11–15

[Eddy 1977] EDDY, William F.: A new convex hull algorithm for planar sets. In: *ACM Transactions on Mathematical Software (TOMS)* 3 (1977), Nr. 4, S. 398–403

[Edelsbrunner 1992] EDELSBRUNNER, Herbert: *Weighted alpha shapes*. University of Illinois at Urbana-Champaign, Department of Computer Science, 1992

[Edelsbrunner u. a. 1983] EDELSBRUNNER, Herbert ; KIRKPATRICK, David ; SEIDEL, Raimund: On the shape of a set of points in the plane. In: *IEEE Transactions on information theory* 29 (1983), Nr. 4, S. 551–559

[Edelsbrunner u. Mücke 1994] EDELSBRUNNER, Herbert ; MÜCKE, Ernst P.: Three-dimensional alpha shapes. In: *ACM Transactions on Graphics (TOG)* 13 (1994), Nr. 1, S. 43–72

[Elaksher u. a. 2002] ELAKSHER, A ; BETHEL, J ; MIKHAIL, E: Reconstructing 3d building wireframes from multiple images. In: *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences* 34 (2002), Nr. 3/A, S. 91–96

[Elberink 2008] ELBERINK, Sander O.: Problems in automated building reconstruction based on dense airborne laser scanning data. In: *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sience* 37 (2008), S. B3

[Elberink u. Vosselman 2009] ELBERINK, Sander O. ; VOSSELMAN, George: Building reconstruction by target based graph matching on incomplete laser data: Analysis and limitations. In: *Sensors* 9 (2009), Nr. 8, S. 6101–6118

[Filippovska u. a. 2016] FILIPPOVSKA, Yevgeniya ; WICHMANN, Andreas ; KADA, Martin: Space Partitioning for Privacy Enabled 3D City Models. In: *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* (2016), S. 17–22

[Fischler u. Bolles 1981] FISCHLER, Martin A. ; BOLLES, Robert C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. In: *Communications of the ACM* 24 (1981), Nr. 6, S. 381–395

[Förstner 1999] FÖRSTNER, Wolfgang: 3D-city models: automatic and semiautomatic acquisition methods. (1999)

[Friedman u. a. 1977] FRIEDMAN, Jerome H. ; BENTLEY, Jon L. ; FINKEL, Raphael A.: An algorithm for finding best matches in logarithmic expected time. In: *ACM Transactions on Mathematical Software (TOMS)* 3 (1977), Nr. 3, S. 209–226

[Galton u. Duckham 2006] GALTON, Antony ; DUCKHAM, Matt: What is the region occupied by a set of points? In: *International Conference on Geographic Information Science* Springer, 2006, S. 81–98

[Geibel u. Stilla 2000] GEIBEL, Roland ; STILLA, Uwe: Segmentation of laser altimeter data for building reconstruction: different procedures and comparison. In: *International Archives of Photogrammetry and Remote Sensing* 33 (2000), Nr. B3/1; PART 3, S. 326–334

[Graham 1972] GRAHAM, Ronald L.: An efficient algorith for determining the convex hull of a finite planar set. In: *Information processing letters* 1 (1972), Nr. 4, S. 132–133

[Haala u. Brenner 1999] HAALA, Norbert ; BRENNER, Claus: Virtual city models from laser altimeter and 2D map data. In: *Photogrammetric Engineering and Remote Sensing* 65 (1999), Nr. 7, S. 787–795

[Haala u. Hahn 1995] HAALA, Norbert ; HAHN, Michael: Data fusion for the detection and reconstruction of buildings. In: *Automatic extraction of man-made objects from aerial and space images*. Springer, 1995, S. 211–220

[Haala u. Kada 2010] HAALA, Norbert ; KADA, Martin: An update on automatic 3D building reconstruction. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 65 (2010), Nr. 6, S. 570–580

[Hammoudi u. a. 2010] HAMMOUDI, Karim ; DORNAIKA, Fadi ; SOHEILIAN, Bahman ; PAPARODITIS, Nicolas: Extracting wire-frame models of street facades from 3D point clouds and the corresponding cadastral map. In: *IAPRS* 38 (2010), Nr. Part 3A, S. 91–96

[Haralick u. a. 1987] HARALICK, Robert M. ; STERNBERG, Stanley R. ; ZHUANG, Xinhua: Image analysis using mathematical morphology. In: *IEEE transactions on pattern analysis and machine intelligence* (1987), Nr. 4, S. 532–550

[Heath u. a. 1998] HEATH, Mike ; SARKAR, Sudeep ; SANOCKI, Thomas ; BOWYER, Kevin: Comparison of edge detectors: a methodology and initial study. In: *Computer vision and image understanding* 69 (1998), Nr. 1, S. 38–54

[Henn u. a. 2013] HENN, André ; GRÖGER, Gerhard ; STROH, Viktor ; PLÜMER, Lutz: Model driven reconstruction of roofs from sparse LIDAR point clouds. In: *ISPRS Journal of photogrammetry and remote sensing* 76 (2013), S. 17–29

[Huang u. a. 2013] HUANG, Hai ; BRENNER, Claus ; SESTER, Monika: A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data. In: *ISPRS Journal of photogrammetry and remote sensing* 79 (2013), S. 29–43

[Hulik u. a. 2014] HULIK, Rostislav ; SPANEL, Michal ; SMRZ, Pavel ; MATERNA, Zdenek: Continuous plane detection in point-cloud data based on 3D Hough Transform. In: *Journal of visual communication and image representation* 25 (2014), Nr. 1, S. 86–97

[Jarvis 1973] JARVIS, Ray A.: On the identification of the convex hull of a finite set of points in the plane. In: *Information Processing Letters* 2 (1973), Nr. 1, S. 18–21

[Jiang u. Bunke 1999] JIANG, Xiaoyi ; BUNKE, Horst: Edge detection in range images based on scan line approximation. In: *Computer vision and image understanding* 73 (1999), Nr. 2, S. 183–199

[Kada u. McKinley 2009] KADA, Martin ; MCKINLEY, Laurence: 3D building reconstruction from LiDAR based on a cell decomposition approach. In: *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci* 38 (2009), S. W4

[Kada u. Wichmann 2012] KADA, Martin ; WICHMANN, Andreas: Sub-surface growing and boundary generalization for 3D building reconstruction. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences I-3* (2012), S. 233–238

[Kamdi u. Krishna 2012] KAMDI, Shilpa ; KRISHNA, RK: Image segmentation and region growing algorithm. In: *International Journal of Computer Technology and Electronics Engineering (IJCTEE) Volume* 2 (2012)

[Khoshelham 2005] KHOSHELHAM, Kourosh: Region refinement and parametric reconstruction of building roofs by integration of image and height data. In: *Int. Arch. Photogramm. Remote Sens. Spatial Inform. Sci* 36 (2005), S. 3–8

[Kilian u. a. 1996] KILIAN, Johannes ; HAALA, Norbert ; ENGLICH, Markus u. a.: Capture and evaluation of airborne laser scanner data. In: *International Archives of Photogrammetry and Remote Sensing* 31 (1996), S. 383–388

[Kim u. Muller 2002] KIM, JR ; MULLER, JP: 3D reconstruction from very high resolution satellite stereo and it's application to object identification ISPRS, 2002

[Kolbe u. a. 2005] KOLBE, Thomas H. ; GRÖGER, Gerhard ; PLÜMER, Lutz: CityGML: Interoperable access to 3D city models. In: *Geo-information for disaster management*. Springer, 2005, S. 883–899

[Kolbe u. a. 2008] KOLBE, Thomas H. ; GRÖGER, Gerhard ; PLÜMER, Lutz u. a.: CityGML–3D city models and their potential for emergency response. In: *Geospatial information technology for emergency response* 257 (2008)

[Kolluri u. a. 2004] KOLLURI, Ravikrishna ; SHEWCHUK, Jonathan R. ; O'BRIEN, James F.: Spectral surface reconstruction from noisy point clouds. In: *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing* ACM, 2004, S. 11–21

[Lafarge u. a. 2007] LAFARGE, Florent ; DESCOMBES, Xavier ; ZERUBIA, Josiane ; PIERROT-DESEILLIGNY, Marc: 3D city modeling based on hidden markov model. In: *2007 IEEE International Conference on Image Processing* Bd. 2 IEEE, 2007, S. II–521

[Lafarge u. Mallet 2012] LAFARGE, Florent ; MALLET, Clément: Creating large-scale city models from 3D-point clouds: a robust approach with hybrid representation. In: *International journal of computer vision* 99 (2012), Nr. 1, S. 69–85

[Lari u. a. 2011] LARI, Z ; HABIB, A ; KWAK, E: An adaptive approach for segmentation of 3D laser point cloud. In: *ISPRS Workshop Laser Scanning*, 2011, S. 29–31

[Lee u. a. 2011] LEE, Dong-Cheon ; YOM, Jae-Hong ; SHIN, Sung W. ; OH, Jaehong ; PARK, Kisurk: Automatic building reconstruction with satellite images and digital maps. In: *ETRI Journal* 33 (2011), Nr. 4, S. 537–546

[Liu u. a. 2003] LIU, Jiyuan ; WANG, Xinsheng ; ZHUANG, Dafang: Application of convex hull in identifying the types of urban land expansion. In: *ACTA GEOGRAPHICA SINICA-CHINESE EDITION-* 58 (2003), Nr. 6, S. 885–892

[Lo u. Chenb 2012] LO, CY ; CHENB, LC: Structure Line Detection from LIDAR Point Clouds Using Topological Elevation Analysis. In: *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 1 (2012), S. 143–147

[Ma 2004] MA, Ruijin: *Building model reconstruction from LiDAR data and aerial photographs*, The Ohio State University, Diss., 2004

[Maas 1999a] MAAS, Hans-Gerd: Closed solutions for the determination of parametric building models from invariant moments of airborne laserscanner data. In: *transformation* 2 (1999), S. 20

[Maas 1999b] MAAS, Hans-Gerd: Fast determination of parametric house models from dense airborne laserscanner data. In: *International Workshop on Mobile Mapping Technology Bangkok, Thailand*, 1999

[Maas 1999c] MAAS, Hans-Gerd: The potential of height texture measures for the segmentation of airborne laserscanner data. In: *Fourth international airborne remote sensing conference and exhibition/21st Canadian symposium on remote sensing*, 1999, S. 154–161

[Maas u. Vosselman 1999] MAAS, Hans-Gerd ; VOSSELMAN, George: Two algorithms for extracting building models from raw laser altimetry data. In: *ISPRS Journal of photogrammetry and remote sensing* 54 (1999), Nr. 2, S. 153–163

[Meng u. a. 2009] MENG, Xuelian ; WANG, Le ; SILVÁN-CÁRDENAS, José Luis ; CURRIT, Nate: A multi-directional ground filtering algorithm for airborne LIDAR. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 64 (2009), Nr. 1, S. 117–124

[Mesev 2003] MESEV, Victor: *Remotely-sensed cities*. CRC Press, 2003

[Milde u. a. 2008] MILDE, J ; ZHANG, Y ; BRENNER, C ; PLÜMER, L ; SESTER, M: Building reconstruction using a structural description based on a formal grammar. In: *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 37 (2008)

[Moreira u. Santos 2007] MOREIRA, Adriano ; SANTOS, Maribel Y.: Concave hull: A k-nearest neighbours approach for the computation of the region occupied by a set of points. (2007)

[Moreira u. a. 2013] MOREIRA, JM M. ; NEX, F ; AGUGIARO, G ; REMONDINO, F ; LIM, Nancy J.: From DSM to 3D Building Models: a quantitative evaluation. In: *Int Arch Photogramm Remote Sens Spatial Inf Sci XL1 W* 1 (2013), S. 213–219

[Nurunnabi u. a. 2012] NURUNNABI, Abdul ; BELTON, David ; WEST, Geoff: Robust segmentation in laser scanning 3D point cloud data. In: *Digital Image Computing Techniques and Applications (DICTA), 2012 International Conference on* IEEE, 2012, S. 1–8

[Ohtake u. a. 2005] OHTAKE, Yutaka ; BELYAEV, Alexander ; ALEXA, Marc ; TURK, Greg ; SEIDEL, Hans-Peter: Multi-level partition of unity implicits. In: *ACM SIGGRAPH 2005 Courses* ACM, 2005, S. 173

[O'Rourke u. Streinu 1998] O'ROURKE, Joseph ; STREINU, Ileana: The vertex-edge visibility graph of a polygon. In: *Computational Geometry* 10 (1998), Nr. 2, S. 105–120

[Orthuber 2014] ORTHUBER, Elisabeth: *3D Building Reconstruction from Airborne LiDAR Point Clouds*, TU München, Diss., 2014

[Orthuber u. Avbelj 2015] ORTHUBER, Elisabeth ; AVBELJ, Janja: 3d building reconstruction from lidar point clouds by adaptive dual contouring. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 2 (2015), Nr. 3, S. 157

[Overby u. a. 2004] OVERBY, Jens ; BODUM, Lars ; KJEMS, Erik ; IISOE, PM: Automatic 3D building reconstruction from airborne laser scanning and cadastral data using Hough transform. In: *International Archives of Photogrammetry and Remote Sensing* 35 (2004), Nr. B3, S. 296–301

[Partovi u. a. 2015] PARTOVI, Tahmineh ; HUANG, H ; KRAUSS, Thomas ; MAYER, H ; REINARTZ, Peter: Statistical Building Roof Reconstruction from WORLDVIEW-2 Stereo Imagery. In: *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 40 (2015), Nr. 3, S. 161

[Pauly u. a. 2002] PAULY, Mark ; GROSS, Markus ; KOBBELT, Leif P.: Efficient simplification of point-sampled surfaces. In: *Proceedings of the conference on Visualization'02* IEEE Computer Society, 2002, S. 163–170

[Poullis u. You 2009] POULLIS, Charalambos ; YOU, Suya: Automatic reconstruction of cities from remote sensor data. In: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on* IEEE, 2009, S. 2775–2782

[Preparata u. Hong 1977] PREPARATA, Franco P. ; HONG, Se J.: Convex hulls of finite sets of points in two and three dimensions. In: *Communications of the ACM* 20 (1977), Nr. 2, S. 87–93

[Rabbani u. Van Den Heuvel 2005] RABBANI, Tahir ; VAN DEN HEUVEL, Frank: Efficient hough transform for automatic detection of cylinders in point clouds. In: *ISPRS WG III/3, III/4* 3 (2005), S. 60–65

[Rau u. Lin 2011] RAU, Jiann-Yeou ; LIN, Bo-Cheng: Automatic roof model reconstruction from ALS data and 2D ground plans based on side projection and the TMR algorithm. In: *ISPRS journal of photogrammetry and remote sensing* 66 (2011), Nr. 6, S. S13–S27

[Rosnell u. Honkavaara 2012] ROSNELL, Tomi ; HONKAVAARA, Eija: Point cloud generation from aerial image data acquired by a quadrocopter type micro unmanned aerial vehicle and a digital still camera. In: *Sensors* 12 (2012), Nr. 1, S. 453–480

[Rottensteiner u. Briese 2003] ROTTENSTEINER, Franz ; BRIESE, Christian: *Automatic generation of building models from LIDAR data and the integration of aerial images*. Citeseer, 2003

[Sajadian u. Arefi 2014] SAJADIAN, M ; AREFI, H: a Data Driven Method for Building Reconstruction from LiDAR Point Clouds. In: *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 40 (2014), Nr. 2, S. 225

[Sampath u. Shan 2007] SAMPATH, Aparajithan ; SHAN, Jie: Building boundary tracing and regularization from airborne LiDAR point clouds. In: *Photogrammetric Engineering & Remote Sensing* 73 (2007), Nr. 7, S. 805–812

[Sampath u. Shan 2010] SAMPATH, Aparajithan ; SHAN, Jie: Segmentation and reconstruction of polyhedral building roofs from aerial lidar point clouds. In: *IEEE Trans. Geosci. Remote Sens* 48 (2010), Nr. 3, S. 1554–1567

[Shapiro u. Stockman 2001] SHAPIRO, Linda ; STOCKMAN, George C.: Computer vision. 2001. In: *ed: Prentice Hall* (2001)

[Stamos u. Allen 2000] STAMOS, Ioannis ; ALLEN, PE: 3-D model construction using range and image data. In: *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on* Bd. 1 IEEE, 2000, S. 531–536

[Sun u. Salvaggio 2013] SUN, Shaohui ; SALVAGGIO, Carl: Aerial 3D building detection and modeling from airborne LiDAR point clouds. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 6 (2013), Nr. 3, S. 1440–1449

[Syed u. a. 2005] SYED, Sohel ; DARE, Paul ; JONES, Simon: Semi-automatic 3D building model generation from lidar and high resolution imagery. In: *Proceedings of SSC Spatial Intelligence* (2005), S. 1421–1428

[Tarsha-Kurdi u. a. 2008] TARSHA-KURDI, Fayez ; LANDES, Tania ; GRUSSENMEYER, Pierre u. a.: Extended RANSAC algorithm for automatic detection of building roof planes from LiDAR data. In: *Photogramm. J. Finl* 21 (2008), S. 97–109

[Tarsha-Kurdi u. a. 2007] TARSHA-KURDI, Fayez ; LANDES, Tania ; GRUSSENMEYER, Pierre ; KOEHL, Mathieu: Model-driven and data-driven approaches using LIDAR data: Analysis and comparison. In: *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci* 36 (2007), S. 87–92

[Tebourbi u. BELHADJ] TEBOURBI, Riadh ; BELHADJ, Zied: Three-dimensional buildings reconstruction from IKONOS stereo images.

[VC 1962] VC, Hough P.: *Method and means for recognizing complex patterns*. Dezember 18 1962. – US Patent 3,069,654

[Verma u. a. 2006] VERMA, Vivek ; KUMAR, Rakesh ; HSU, Stephen: 3D building detection and modeling from aerial LIDAR data. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* Bd. 2 IEEE, 2006, S. 2213–2220

[Vosselman u. a. 2001] VOSSELMAN, George ; DIJKMAN, Sander u. a.: 3D building model reconstruction from point clouds and ground plans. In: *International archives of photogrammetry remote sensing and spatial information sciences* 34 (2001), Nr. 3/W4, S. 37–44

[Wang u. Shan 2009] WANG, Jun ; SHAN, Jie: Segmentation of LiDAR point clouds for building extraction. In: *American Society for Photogramm. Remote Sens. Annual Conference, Baltimore, MD*, 2009, S. 9–13

[Wang u. Tseng 2010] WANG, Miao ; TSENG, Yi-Hsing: Automatic segmentation of LiDAR data into coplanar point clusters using an octree-based split-and-merge algorithm. In: *Photogrammetric Engineering & Remote Sensing* 76 (2010), Nr. 4, S. 407–420

[Wang 2013] WANG, Ruisheng: 3D building modeling using images and LiDAR: A review. In: *International Journal of Image and Data Fusion* 4 (2013), Nr. 4, S. 273–292

[Weidner u. Förstner 1995] WEIDNER, Uwe ; FÖRSTNER, Wolfgang: Towards automatic building extraction from high-resolution digital elevation models. In: *ISPRS journal of Photogrammetry and Remote Sensing* 50 (1995), Nr. 4, S. 38–49

[Zhang u. a. 2003] ZHANG, Keqi ; CHEN, Shu-Ching ; WHITMAN, Dean ; SHYU, Mei-Ling ; YAN, Jianhua ; ZHANG, Chengcui: A progressive morphological filter for removing nonground measurements from airborne LIDAR data. In: *IEEE Transactions on Geoscience and Remote Sensing* 41 (2003), Nr. 4, S. 872–882

[Zhang u. Xu 2002] ZHANG, L-H ; XU, W-L: Convex Hull Based Point Pattern Matching Under Perspective Transformation. In: *Acta Automatica Sinica* 28 (2002), Nr. 2, S. 306–309

[Zhang u. a. 2012] ZHANG, Man ; ZHANG, Liqiang ; MATHIOPOULOS, P T. ; XIE, Wenqing ; DING, Yusi ; WANG, Hao: A geometry and texture coupled flexible generalization of urban building models. In: *ISPRS journal of photogrammetry and remote sensing* 70 (2012), S. 1–14

[Zhang 2011] ZHANG, Ning: Plane fitting on airborne laser scanning data using ransac. In: *Lunds Tekniska Högskola* (2011), S. 17

[Zheng u. Weng 2015] ZHENG, Yuanfan ; WENG, Qihao: Model-Driven Reconstruction of 3-D Buildings Using LiDAR Data. In: *IEEE Geoscience and Remote Sensing Letters* 12 (2015), Nr. 7, S. 1541–1545

List of Figures

List of Tables