Automatic space subdivision for multi-story pathfinding on a 3D point cloud using an octree

MSc Geomatics Thesis Proposal

Florian W. Fichtner
f.w.fichtner@student.tudelft.nl

January 7, 2016

Under supervision of CGI Nederland

1 Introduction

Indoor navigation consists out of two parts: The localization inside a building and the route planning, which brings the user to a desired location. To successfully achieve those tasks an appropriate representation or model of the building is required (Liu and Zlatanova, 2012). In case of emergency or temporary exhibition, there is no time to create a semantically rich vector model of the interior of a building. Point clouds, however, can be acquired rapidly and relatively cheaply, but they lack a structure and semantics which are necessary for pathfinding. This thesis aims to create a workflow to subdivide the 3D space of an indoor point cloud of a building and to create a semantically enriched model for pathfinding between different floor levels.

The following chapter gives an introduction to the topics of this research. Then the related work (chapter 2) will be introduced, before the objectives of this graduation project will be addressed in more detail (chapter 3). Chapter 4 will give an overview of the methods which are going to be used to fulfill those objectives. Lastly a more detailed planning (chapter 5) of this work will be provided.

1.1 Background

A study of the National Human Activity Pattern Survey (NHAPS) showed that Americans spend 87% of their life inside buildings. 11% of the time, these buildings are neither residential, nor a workplace or restaurant which makes it likely for the person to be in relatively unknown or complex buildings, like shopping malls or airports. Only 5.5% of the time is spent in vehicles (Klepeis et al., 2001). Wayfinding and navigation for outdoors have been studied for a long time already. However, making use of indoor navigation is still not very common. This is mainly due to the difficulties in positioning indoors as there is no Global Positioning System (GPS) or other signal globally available, which allows to navigate in all buildings. Furthermore it is a complex challenge to make indoor models suitable for navigation as they need to be in 3D to optimally connect multiple stories.

The terms *wayfinding* and *navigation*, even though used interchangeable many times, differentiate because navigation offers continuous real-time guidance during the movement, while wayfinding finds a route connecting two locations (Karimi, 2015). Wayfinding or pathfinding is thus necessary for navigation.

The research of this thesis is part of the SIMs3D project, which aims to solve the problem of missing and timely 3D indoor models for many large public buildings. The safety management of public buildings, this includes mainly the fire brigade, are in need of such models. The project ranges from 3D indoor reconstruction from point clouds over 3D indoor models (geometry, semantics and topology) to 3D indoor navigation suitable for all kind of tasks and users. Finally, a rapid and low-cost 3D modeling approach which allows to identify spaces and networks, needed for a navigation applications, shall be created (SIMs3D, 2015).

During the 2015 Geomatics Synthesis Project at TU Delft the author's group developed a fast and universally reusable workflow which efficiently identifies the empty space in a point cloud and structures both the empty space and the points in an octree. The identification of empty space, in other publications also called free space, instead of using boundaries (walls, obstacles, etc.) allows the focus on wayfinding instead of only collision avoidance. This makes sense because the space where the object or person actually moves in is structured (Broersen et al., 2015). An octree is a three dimensional extension of a quadtree data structure. It consist out of a 3D volume and is recursively subdivided into eight compliant disjoint cubes (called octants) until blocks of a uniform color (either filled with points or empty) are obtained, or a predetermined level of decomposition is reached (Samet, 1989).

Regarding localization in indoor environments, van der Ham (2015) analyzed the 2D performance of the Quuppa technology system (Quuppa Oy, 2015) in the Spark Center of CGI Nederland. He found that it can be used up to a sub-meter accuracy level for asset tracking. Quuppa provides a locating system which uses the Angle-of-Arrival processing of a Bluetooth Low Energy signal for its location data.

1.2 Problem Statement

Zlatanova et al. (2014) identify four different steps necessary for a successful application for navigation through indoor spaces. The digital acquisition of available spaces (1), the structuring of acquired data (2), formalization of the data to establish relationships between different subspaces (3) and lastly applying the user requirements on the formalized and structured data (4). Subspaces are formed by subdivision of indoor space into smaller parts. They can have semantic meaning. Moreover, they can be navigable, so be used to perform activities, or non navigable like e.g. walls. Acquiring point clouds of the interior spaces in buildings became increasingly easy and cheap in the recent years. Technology like the ZEB1 laser scanner (3D Laser Mapping, 2015) or the Project Tango tablet (Google ATAP, 2015) allow to collect precise 3D models in an efficient and mobile way.

Nevertheless, this data is unstructured and does not yet contain enough information to be useful for complex tasks like navigation or localization. In many models the geometry is preserved, however the semantics as well as the topological information are lost (Diakité et al., 2014). 3D Indoor navigation modeling requires an accurate 3D topographic space of the interior of the building, which should be found in an automated and fast way (Jamali et al., 2015). The representation of a whole room as one single object is not enough and too abstract, especially if the room is big or the navigation system is used for autonomous objects (Krūminaitė and Zlatanova, 2014). A further subdivision is therefore recommended. The octree structure subdivides the space, but only classifies in occupied and non-occupied (empty) space (Broersen et al., 2015). Any semantic information is lost so there is no way to distinguish whether the space is open for navigation (e.g. stairs) or not (e.g. table). The identification of staircases, however, is essential to reach the ultimate goal of pathfinding and navigating on multi-story indoor environments (Sinha et al., 2014) to create a logical model connecting all floors.

Different logical models can be used for pathfinding, and the subdivision is dependent on the type of application (Liu and Zlatanova, 2013a). There are two different ways to subdivide, first semantically, which means to divide into meaningful subspaces (with information about the kind of room, etc.) and secondly geometrical, which divides based on geometry only (grid, Voronoi diagram etc.). Both methods have advantages and disadvantages, as the geometric approach can be fully automated, but the semantics remain unclear and for the semantic approach the reverse is the case.

In emergency cases or for temporary buildings, like for exhibitions or fairs, navigation models should be created as fast, cheap and autonomous as possible. A standard way is needed to obtain a navigation model from a point cloud connecting multiple floors. This will be addressed in this thesis using an octree structure. Semantics are added to establish relationships between different subspaces and allow to connect multiple floors in stairways. Therefore the model requires to be in 3D (or at least 2.5D) (Karimi, 2015). The technique can also help to support user requirements, such as if a person or object cannot use stairs (e.g. with a wheelchair) and is not able to use a specific path because of that.

The lack of a structure and semantics in geometric data of indoor environments, whose acquisition becomes increasingly accessible, can be overcome when the point cloud and its empty space are structured in a semantically enriched octree, from which a logical model can be derived. The usual approach is the reconstruction of complex vector models, but in many cases this is not necessary for pathfinding applications. It usually make more sense to show the user an intuitive set of cues of when to turn and where, than to make him look at complex models or even pictures of the same hallway he is moving through (Halsted, 2014). With the octree method only the route can be shown, the data however is still complete with a detailed point cloud in the back end.

1.3 Research Question

In the course of this MSc thesis, the author's goal is to enhance the octree structure of a point cloud and its empty space with semantic information and to automatically derive a model that allows to find a way between multiple connected floors. The research aims to answer the following question: *"To what extent can an octree data structure be used to subdivide 3D space and to create a model for multi-story pathfinding?"*

More details about the objectives and scope of this research can be found in chapter 3.

1.4 Scientific Relevance

Point clouds of buildings can be obtained quickly nowadays. But there is no way to automatically derive a subdivided model suitable for wayfinding, while at the same time allow and enhance the multi-floor connectivity via stairs. This however is essential, as there is not always enough time to create a semantically rich building model that facilitates wayfinding. On top of that, the research can form a first step for a precise logical and building model, autonomously built from a point cloud. Also Liu and Zlatanova (2013a) write that there is only very few research about the creation of navigation models derived from 3D geometry.

2 Related work

Several areas of research will be relevant for this thesis. The following chapter will provide a quick overview of related works. The first part (section 2.1) will address the semantics and subdivision of space necessary for navigation, which was studied in other works. After that the research to find an applicable automatically derived navigation model for indoor applications will be introduced (section 2.2). Finally research in indoor navigation in robotics will be explained as an example of collision avoidance, indoor navigation and as a possible future use case (section 2.3).

2.1 Semantics and subdivision of space

The success of indoor navigation is mainly dependent on the model of the building available (Krūminaitė and Zlatanova, 2014). Diakité et al. (2014) developed an framework to recover semantic information from Building Information Model (BIM) and 3D Geographic Information System (GIS) data based on the combination of geometry and topology for automated semantic labeling. The initial situation of this MSc thesis however is not based on vector models, but on point clouds and their empty space which are structured and subdivided into an octree. Therefore, to enhance subdivision and to create a logical model for indoor wayfinding across multiple-floors, features have to be detected from those point clouds. 3D modeling of the building architecture from point cloud scans is a rapidly advancing field (Turner and Zakhor, 2013) and its results can be beneficial for indoor wayfinding. The following sections will go deeper into detail about what has been researched in other works. This research will form the basis of this project.

2.1.1 Feature detection in point clouds

For extracting information, or to build an application using point clouds, the recognition of object surfaces is often one of the first steps to perform (Vosselman et al., 2004). Vo et al. (2015) were able to extract roads from point clouds using a supervised machine learning approach. Wang and Tseng (2004, 2005, 2011 and 2014) published extensive research about extraction of surface features from Light Detection and Ranging (LiDAR) data using an octree data structure. However, there is no focus on indoor environments. Yogeswaran and Payeur (2009) developed a technique to group features from a point cloud based on their proximity and similarity using an octree structure, but there was no further classification. Rusu et al. (2009) present a method to identify objects in a point cloud of a kitchen environment. They find planar areas using the point's normals and then split them into smaller parts with a region growing method. Trained conditional random fields, purely based on geometrical reasoning, are used for classification. The result is a semantic object map, which can be a help to provide context for robots. Furthermore, the Triangulated Surface Map, which is also created, can support collision avoidance and path planning.

Khoshelham and Díaz-Vilariño (2014) present a method for automated 3D indoor modeling of floors, ceilings and walls with shape grammar and histograms in point clouds. This is suitable for the detection of floors and walls as well. Also Okorn et al. (2010) use histograms to detect floor and ceiling data from 3D point clouds. Walls are identified using histograms and a hough transformation approach for line detection.

Turner et al. (2015) present a workflow to create 2D floor plans from point clouds making use of the scanner's position and voxel carving, as well as partitioning by height with histograms. Such approaches can also be useful in this work to add semantics about room separation to the model or the histograms to distinguish different levels in the building.

2.1.2 Identification of stairs

The main goal of feature detection in this research lays in the identification of stairs in a point cloud or octree. To enable indoor wayfinding, first an indoor model has to be created. The most of such a model can be shaped with walls, ceilings, floors and stairways (Sanchez and

Zakhor, 2012). The authors run a principal component analysis for every point and then they classify and segment the point cloud into the four structural elements. Their system makes use of public C++ libraries of the Point Cloud Library (PCL).

Also Eilering et al. (2014) identify those surfaces, with a specific focus on segmenting stair structures. The classification of each point is dependent on local spatial features of the point cloud, but also on the classification of close points based on probability. With a trained dataset they achieve a successrate of approximately 75%. On the other hand, there was not yet a focus on segmentation and structural relationships between the objects, but they plan to do this in the future. Python and the PCL was used for the implementation.

Schnabel et al. (2007) use random sample consensus (RANSAC) techniques to identify structure in point clouds. Oßwald et al. (2011) argue that RANSAC tends to simplify complex planar structures so that small steps merge into a sloped plane. This however, might not be detrimental for this project.

Delmerico et al. (2013) model and localize stairways on a map from depth imagery where lines represent discontinuities and depth changes abruptly. Stairs are detected by a big and regular change of depth, however the viewpoint matters and even though promising and well described method, Delmerico et al. (2013) lay the focus to detect stairs from depth images and not from point clouds directly. Another research shows an algorithm for grouping step-like obstacles from point clouds obtained by RGB-D sensor (Sinha et al., 2014). It runs in real-time, divides stairs in subsections and the authors claim that it is suitable for any real-world application. The paper contains a well described method and pseudo-code, but only the point cloud and environment within the viewpoint around the robot is considered.

Bansal et al. (2011) stream a point cloud and store the points in a voxel grid and use image processing methods to detect doors and stairs. The usage of histograms for stairs detection, even though just a small part of the research, follows a similar approach as the shape grammar method (Khoshelham and Díaz-Vilariño, 2014) and can therefore potentially be connected.

A majority of the studies are performed with the aim of improving robot navigation, more specific research about this can be found in section 2.3.

2.2 Automatically derived pathfinding model for indoor applications

There exist already different kind of digital models that can be used for indoor pathfinding purposes. They range from geometric, to semantic and topological models or to a combination of them (Krūminaitė and Zlatanova, 2014). Liu and Zlatanova (2013b) argue that there are different types of indoor paths that matter on attributes like the user size or a visibility graph. Zlatanova et al. (2014) present several 3D (and also 2D) approaches for indoor navigation models. The possibilities for a complete and partial subdivision for networks, or to use a 3D grid graph are presented.

Liu and Zlatanova (2013a) point out, that there was only a limited amount of research to derive topological models from 3D geometrical models at the time of writing. With the aim to automatically derive a logical model, first navigable spaces and doors need to be identified in order to build a connectivity network. They also argue that to generate a multi-floor navigation model from floor plans, semantics have to be added and that there is no standard way of creating a network from floor plans. Jamali et al. (2015) use a laser rangefinder for the acquisition of inaccurate geometry from buildings and propose a 3D modeling method for a topological navigation network. The model and connectivity is built up step by step, fully autonomous, but each space requires an ID, specified during the scanning process. Topology is mapped using dual edges for adjacent spaces. Yang and Worboys (2015) use combinatorial maps and their duals to derive a navigation graph, while Yuan and Schneider (2011) use their LEGO model and build a graph for navigation by representing blocks by nodes and connectors by edges. As they first merge larger blocks, they create a structure quite similar to an octree and might therefore be useful for this research as well. Generally deriving topology using Poincaré duality is a common approach (Lee and Zlatanova (2008), Becker et al. (2009) and Boguslawski et al. (2011)).

Narasimhan et al. (2006) extract a graph from a CAD model and store it in an octree. The octree enables them to determine the path to the closest destination.

2.3 Indoor navigation in robotics

Even though this research's main goal is not aimed on robotics only, experiences in this field can be useful for this study. In the area of robotics indoor navigation is well researched for quite a long time already, many approaches also use an octree data structure and Simultaneous Localization and Mapping (SLAM) methods. Herman (1986) argues that an octree representation of obstacles enables fast wayfinding algorithms. Kitamura et al. (1995) represent everything in the environment by an octree structure and generate a potential field from the black leafs in the octree by successively adjoining regions with white cells. This way they were able to create a collision free navigation method, which is not graph based. An approach that also Wu and Hori (2006) follow, who developed a collision free wayfinding method based on an octree model, which was faster than graph based solutions. However, the research was limited on the movement of a robotic arm. Jessup et al. (2014) show that octree based occupancy grids are a suitable representation for multi-robot 3D mapping and that these grids are updateable.

Robotics need context and semantics, like the room structure, estimated from the environment (Olufs and Vincze, 2011). Shen et al. (2011) describe a method for autonomous multi-floor indoor navigation of micro aerial vehicles (MAVs), while only using on-board sensors without prior knowledge of the environment. Therefore 2.5D environment models are used, which are formed by collections of vertical walls and horizontal floors or ceilings. Wang et al. (2014) were able to find the 3D coordinate of an unmanned aerial vehicle (UAV) using a 2-laser-setup and structured indoor features with preknown key corner coordinates.

However, in the field of robotics there is no direct research about shortest wayfinding in an octree representation of 3D spaces. Instead, sensors are added on the moving objects to avoid collision when in motion. On top of that, sensors are also used for SLAM, so measuring an environment, building a map and finding itself in there with the help of feature point matching like in Bergeon et al. (2015), where they use point of interests in images obtained with a Kinect to localize the robot. The octree is used often as a supporting data structure.

2.4 Discussion

There is a broad research about feature detection from point clouds, that can be re-used in this thesis (2.1.1). Especially for stairs detection (2.1.2), but also for other purposes indoor navigation and structuring is researched in the field of robotics (2.3). The focus of indoor navigation in robotics is mainly based on SLAM, while in this research the environment or at least the geometry is gathered beforehand using laser scanning methods. Also the viewpoint plays a role in many methods found. This makes sense for robotics applications, but it may not be suitable for this work, as the model has to be global and not just consider the environment around the current position.

The goal is to find and perform a workflow to automatically subdivide the space and to create a logical model for wayfinding based on geometry similar to Liu and Zlatanova (2013a) or also Jamali et al. (2015). Those works did not use a point cloud as input data though. Work focusing on point clouds on the other hand are usually limited to modeling or to specific robotic applications or tasks. Also there was no work found, which derives a pathfinding model in a way it is aimed throughout this research. The methods which are planned to be used, based on the findings in this section, will be further explained in section 4.

3 Research Objectives

This chapter provides an overview of the main research questions (section 3.1) and defines the objectives and scope (section 3.2) of this master thesis.

3.1 Research Questions

The main research question of this work is: *"To what extent can an octree data structure be used to subdivide 3D space and to create a model for multi-story pathfinding?"* To answer this, the following subquestions have to be answered as well:

- To what extent can floors and walls be distinguished in the octree structure or the point cloud?
- How to detect stairs in the octree structure or point cloud?
- How to derive a pathfinding model from the structure?
- How can the Quuppa location be found inside the model and put into relation with the environment?

3.2 Objectives and scope

The aim of this thesis is to create a workflow that takes an indoor point cloud from a laser scanner as input. It then identifies and structures empty space and the points in an octree, before it enhances the structure with automatically derived semantic information to subdivide the space and to create a model for multi-story pathfinding. Thus, while Broersen et al. (2015) only found and structured the empty space in a point cloud, this work explores how to add constraints and semantics to the resulting model. The pathfinding model should relate to the movement of humans, so be sticked to floors and stairways only as they are walkable and can be used for a path. After they have been identified all obstacles can be excluded from membership of the wayfinding model. The resulting representation should be layered to keep the connectivity (at the stairs) between different floors in the building and therefore be in 2.5D. Lastly the Quuppa coordinates can be mapped to the model.

More information about the scope can also be found in the MoSCoW table in section 5.3.

4 Methodology

The following chapter provides an overview of the methodology and tools which are going to be used throughout this thesis. First the precondition of the data (section 4.1) will be explained. Then the methodology planned to be used for the subdivision (section 4.2) and the generation of a pathfinding model (section 4.3) will be further introduced. Following the method to match the model with the localizer will be illustrated (section 4.4) and finally the data and tools to be used will be summarized in section 4.5. The workflow of the proposed methods can be found in figure 1.

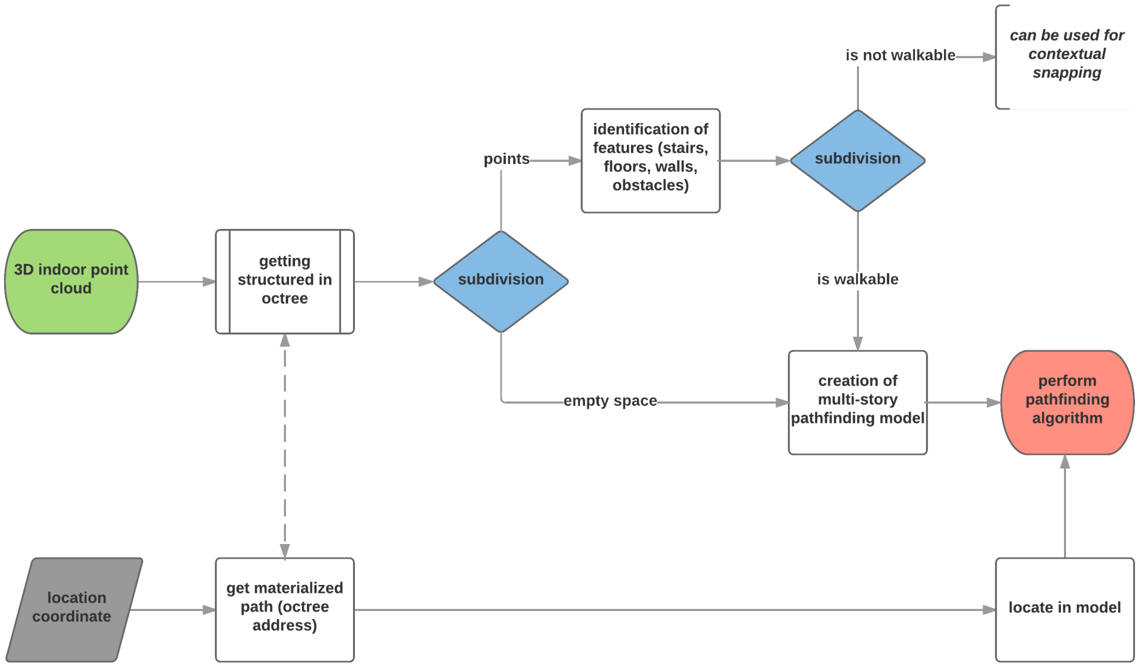

Figure 1: Proposed workflow

4.1 Precondition of the input data

To be able to create a stable framework and to reach the goals, some preconditions of the input data are necessary to be defined. Below, the criteria considered in this project is listed:

- The input data is a 3D point cloud of the indoors of a multi-story building
- The point cloud should already be processed and as clean (free of noise) as possible
- The rooms of the acquired point cloud can be furnished
- Color information is not necessary, but can be a plus

The input point cloud should be of a building with multiple stories (at least two) which are connected through a staircase. Point clouds can be noisy depending on the scanner used and the environment where they were acquired. However, cleaning of the data is out of the scope of this research as several tools to perform such tasks are already available. Thus, it is assumed that the point clouds are clean from the beginning. Color information is not needed for the workflow to be able to work, however having such information can increase the possibilities for the analysis of the data. Section 4.5 elaborates on other aspects of the data and on tools used throughout this thesis.

4.2 Subdivision and semantic enrichment of the structure

The methodology to be used in this thesis will be built on top of a point cloud structure, which was achieved during the Geomatics Synthesis Project 2015. The outcomes structured the points and the empty space in an octree (Broersen et al., 2015). Octrees for point clouds are based on the recursive and non-uniform subdivision of space and are a common approach to structure and segment 3D point clouds and for example used by Zhou et al. (2011) or Wang and Tseng (2011) for the partitioning of space, which results in a hierarchical structure. The

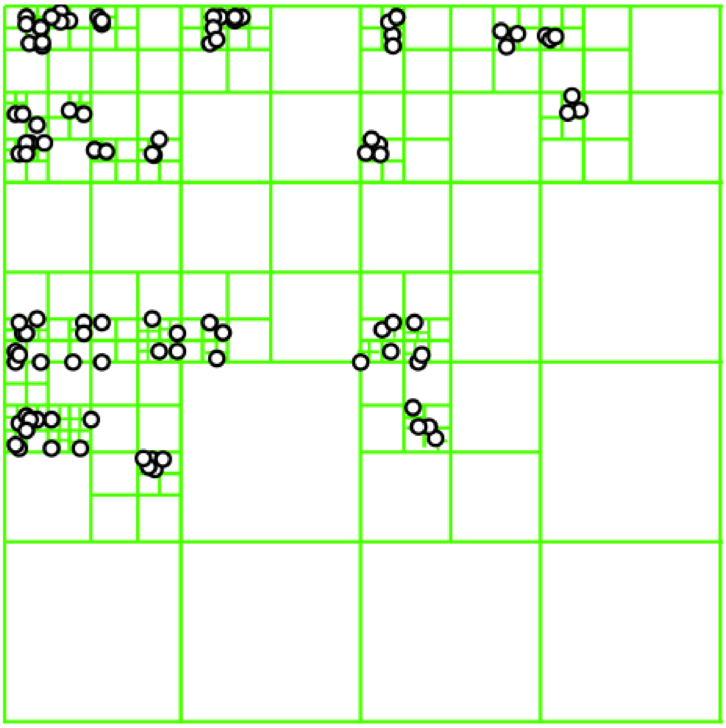

Figure 2: Current octree implementation simplified to quadtree structure

empty space, so all the free space which is not occupied by any points and therefore available for navigation is structured in an octree as well. A similar approach was used for localization by Payeur (2006).

In the current implementation of the octree structure (Broersen et al., 2015), all black leafs (with points, also more than just one) are split until the deepest and smallest level in the tree, the octree of empty space (the reverse with all white leafs) are as big and up in the tree as possible. On top of that, both octrees are merged as they supplement each other. Figure 2 show the current octree implementation simplified to 2D quadtree. Moreover a method (Vörös, 2000) has been implemented which finds all the leaf's neighbors in the octree on the fly during a basic pathfinding calculation.

So after the Synthesis Project the empty space is subdivided into an octree structure, but to find which parts of the model can be used for wayfinding the floors and the stairs (together they make the *walkable space*) need to be identified. In section 2.1.2 a number of methods for stairs detection in a point cloud were introduced. However, first floors and walls should be identified, as they can also help to find the stairs' position because stairway typically connect two floors. The method planned to be used to find them is based on the 3D modeling approach of Khoshelham and Díaz-Vilariño (2014). They identify floors, ceilings and walls with shape grammar and histograms in a point cloud. First, the point cloud needs to be rotated using the points' normals, so that the main walls are parallel to the x- and y-axes. Histogram peaks of the z-coordinate provide information about floors and ceilings in the point cloud and subclouds can be created. Also Turner et al. (2015) use such kind of histogram approach to separate the point cloud by floor levels and to create a 2D floor plan for each of them. Khoshelham and Díaz-Vilariño (2014) then divide the space into subspaces on each level using the distance of histogram peaks. Close adjacent peaks correspond to walls and non-navigable spaces and when the distance is higher than the average thickness of walls, the subspace is navigable. This method can also be used for this graduation project, but the rules, later introduced in the paper (Khoshelham and Díaz-Vilariño, 2014) have to be adapted to the specific needs. Additionally, to decrease computation time the black leaf nodes of the octree could be used instead of the points. The data would be more regular, filtered because one leaf combines multiple points.

When the walls and the floors are found the stairs have to be recognized. It might be helpful to know that stairs usually run in a slope between different floors, therefore a close connection to the previous method is preferable. Also there are several methods to identify stairs introduced in chapter 2.1.2, which can be re-used. Four methods (Sinha et al. (2014), Delmerico et al. (2013), Bansal et al. (2011) & Eilering et al. (2014)) come into closer consideration, but Bansal et al. (2011) seems to be the most feasible. The relevant parts will be summed up in the following list:

1. They first voxelize the point cloud, fit local planes and identify vertical and horizontal surfaces. The voxelization part of the approach is already fulfilled due to the octree structure (the black leaf nodes) described earlier in this section.

2. Then the voxels containing surfaces are projected to a 2D grid giving a vertical or a horizontal histogram representation respectively where the number of cells measure how many voxels above this cell contain vertical or horizontal surfaces.

3. Now some filters need to be set to detect the stairs, so that the matched filter for the vertical histogram is a rectangular block with alternating rows of positive and negative weights. The horizontal histogram only shows a continuous horizontal structure. Moreover the number of steps should be at least 3 to gain robustness for the stair detection.

4. Finally, the filter responses for the vertical and horizontal histograms of the previous step need to be combined to increase robustness.

The approach has the advantage that it provides the possibility to be connected to the identification of floors and walls described earlier (Khoshelham and Díaz-Vilariño, 2014) as both methods make use of histograms. However, the rotation of the point cloud and stairs might influence the success of this method.

Also feature extraction methods like usable parts of the other papers, such as the RANSAC method can still be considered, as it helps to structure the points in global shapes. Generally, most other approaches aim either at robotics, where the viewpoint matters, or at 3D modeling. In this thesis however just labeling points or leafs with their semantic information is necessary.

When the stairs are found a stairway attribute can be added to the leafs of the octree of points, similar like a color value. Right now each point is stored in a Postgres database, a stairway boolean would just be another column, or alternatively can also be stored in an extra table.

The identification of the other objects like furniture is out of the scope of this thesis. However, points, which are neither classified as floor, wall or stairs can be labeled obstacles (see also figure 5) and represent most likely furniture or similar objects.

4.3 Generation of pathfinding model

If the point cloud and the octree are classified, so it has been determined which points are floor, wall, obstacle or stairs the wayfinding model has to be created. There are many different approaches for wayfinding and navigation models (see section 2.2) so it first has to be defined what kind of model is best suitable. The method to derive the model can be inspired by the work of Liu and Zlatanova (2013a), where a connectivity network is created from the geometry or Jamali et al. (2015), where dual edges are used to connect adjacent spaces.

Generally the walkable path which can be used for wayfinding, so floor and stairs, is already structured because of the octree of the points and of the empty space. However, the creation of a 2.5D representation, working as a connectivity graph, derived from the 3D octree is still a challenging task, as the subdivision and semantic enrichment gained through the steps in the previous chapter have to be taken into consideration. Starting from an octree structure, the first goal is to create a quadtree of each level in the building which acts as kind of a navigable floor plan. The quadtree follows a usual black and white quadrant approach, however black quadrants can be further categorized into wall and obstacle, while the white quadrants can be classified into floor and stairs. Furthermore there should be a connection between the different floor levels at the stairs.

This results in many nodes per room or space based on the quadtree grid, opposite to the navigation graph derived by Yang and Worboys (2015), who use a complex system with different kind of nodes depending on their location. The quadtree approach makes route calculation computationally more expensive, but it can also be a lot more precise.

The creation of the quadtree can be performed by slicing the octree above floor level. This creates a structured floor plan, where adjacent squares should still be merged to create a real quadtree structure. If this does not happen the subdivision is still based on the octree. The big challenge, when slicing the dataset, is to find an appropriate height to do so, as this can have many dependencies.

Alternatively the Turner et al. (2015) approach of 2.5D modeling can be used. To re-use their method it needs to be adapted as the goal is not to create a building model focusing on walls like they do, but a wayfinding network which has the focus on the empty space. Furthermore furniture and other obstacles are ignored using this approach. They create a 2D floor plan for each level and extrude up to the ceiling height, by subsampling the point cloud to a set of representative 2D points. Samples of walls can be identified by projecting 3D points on a horizontal plane, where a high density of projected points corresponds to vertical surfaces. Also Okorn et al. (2010) use a similar method. The 3D points are first discretized into voxels (similar to what happens during the octree generation of Broersen et al. (2015)) projected on a x-y plane as well, and then a histogram gets generated. The floor plan gets generated by slicing the 2D histogram of the occupied voxels above each ground cell. This might also be promising, especially as histograms are created anyway for earlier steps. For the wayfinding model also the size and width of the movable object can play a role as well (in a manner that too small space corresponds to occupied space, because it is not usable for navigation). The approach described here can help in that manner as occupied voxels above each 2D ground cell are simply counted. The width of stairs and walkways can be determined with that way as well.

Subsequently the topology has to be derived. Following the Poincaré duality leafs in the quadtree can become nodes and adjacent faces turn into edges, which results in a dual graph. The graph can be the topological part of the Node-Relation-Structure (NRS) (Lee and Zlatanova, 2008) and simplifies the complex topological relationships between 3D spatial objects (Becker et al., 2009). Another possibility to use could be the dual half-edge (DHE) topological data structure (Boguslawski et al., 2011).

An alternative could be a projection of the semantics to the slice, which could work with a similar approach like the carving method of Turner and Zakhor (2013). They assume a voxel grid over the bounding box of the space and use ray-tracing to identify whether the voxel is interior or exterior (outside the building or inside the volume of an object). As soon as the laser intersects with a voxel it is considered to be interior. For this work the ray needs to be sent perpendicular from all octants labeled with floor or stairs. If the ray does not intersect a black node for a length of 2 meters it can be considered walkable (if the width is also wide enough) and the octants upper face of the black nodes at floor level can be merged to a raster grid. The same can happen with the stairs, just the connectivity at the start and at the end of the stairs with the different stories has to be taken into consideration.

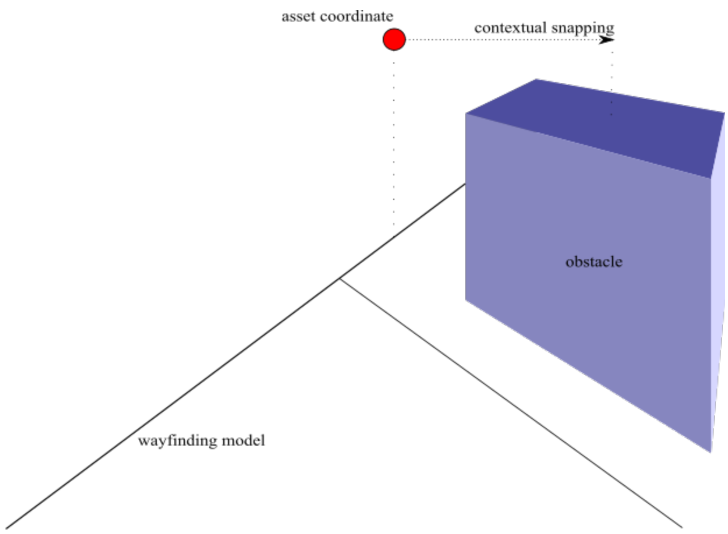

Figure 3: Contextual snapping of coordinate of asset to obstacle

4.4 Match model with localizer

The octree structures and subdivides the whole point cloud and the empty space in a bounding box. The localizer provides a 2D coordinate of a Quuppa tag in the building (van der Ham, 2015). If a height-value or Z-coordinate, which can also refer to the level in the building only, is added to position, the tag can be localized in the octree the same way as the octree of the point cloud is built. This way, if the basic octree structure is kept, the tag can be localized in the wayfinding model.

The possible use cases of this are diverse, first of all it can be the start or target of the wayfinding algorithm. Therefore a link between the coordinate and the wayfinding model has to be created. Especially if the coordinate points to the area of an obstacle and the relation cannot directly be found with an overlay of the coordinate and the wayfinding model a nearest neighbor method (or similar) to find the next node has to be applied. Furthermore, it can act as error correction for asset tracking. Depending on whether the coordinate points to a person or object, the assumption can be made whether the point's most logic location should be on the walkable space or on an obstacle (like a table or shelf) (see figure 3). A contextual aware snapping can be performed for error correction. Having a real 3D coordinate would increase the possibilities. Quuppa would be able to get a 3D coordinate, but needs to be set up for that in order to do so.

4.5 Data and Tools to be used

The method should work on most point clouds of indoor environments having multiple floors connected through a stair case. Therefore any point cloud fulfilling those requirement (section 4.1) can be used.

To be able to use the Quuppa location, using a point cloud of the Spark Center of CGI in Rotterdam, where the system is running would be recommendable. However, there is currently no stairs structure in this room.

Summing up, there is a list with the data which is going to be used throughout the project:

- Point cloud derived from mesh acquired with Project Tango tablet, or alternatively a point cloud from ZEB1 or a different (mobile) laser scanner
- Points and empty space structured in octree

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Prepare P2	1	2	1	2	100%
P2 presentation	2	1	2	1	100%
Acquire point cloud	3	1	0	0	0%
Structure in octree	3	1	0	0	0%
Rotate point cloud	3	2	0	0	0%
Create histograms & filters	3	8	0	0	0%
Identify floors & walls	4	4	0	0	0%
Identify stairs	8	8	0	0	0%
Derive pathfinding model	11	10	0	0	0%
P3 presentation	14	1	0	0	0%
Write final thesis	15	15	0	0	0%
Apply pathfinding algorithm	17	1	0	0	0%
Visualize results	17	11	0	0	0%
Results evaluation	18	2	0	0	0%
P4 presentation	19	1	0	0	0%
Map with Quuppa coordinates	20	8	0	0	0%
P5 presentation	28	2	0	0	0%

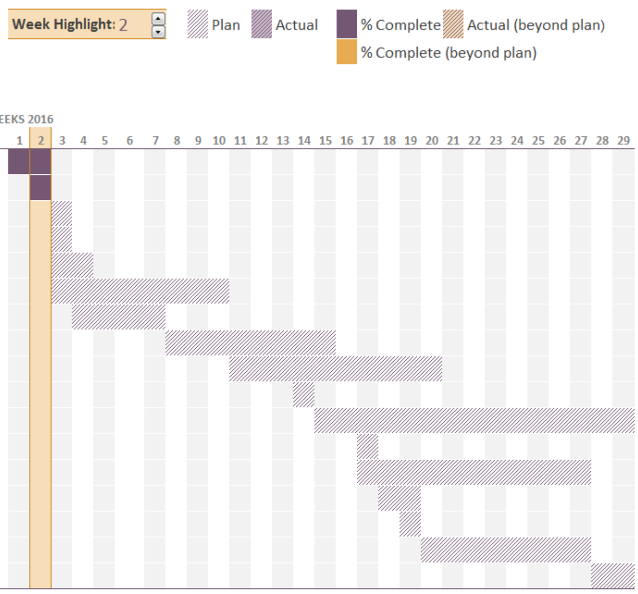

Figure 4: GANTT chart

- Coordinate of Quuppa system

The majority of this project will be build using the Python programming language and the open source database PostgreSQL.

The tools which are going to be used throughout the project:

- Python programming language
- PostgreSQL database, perhaps with PostGIS extension
- Psycopg2 to connect Python with the database, NumPy or other Python libraries
- python-pcl, the Python bindings to the PCL are an option to use

5 Planning

This chapter gives an overview over the time (section 5.1) and logical (section 5.2) planning for the following months of this thesis project. The planned scope is illustrated once again using a MoSCoW prioritization method.

5.1 GANTT

The schedule of the project - the phasing and delivery of items - is illustrated in the following GANTT chart (figure 4). The deadlines (P2, P3, P4 and P5) have fixed dates and their week number are shown in chart. A report has to be handed in at P2, P4 and P5.

5.2 Project Logic Diagram

Figure 5 shows the Project Logic Diagram of this research as Unified Modelling Language (UML). First the indoor point cloud and the empty space are structured in an octree. Then the octree or the point cloud get subdivided so that floor, stairs and other obstacles can be distinguished from one another. After that a wayfinding model, which connects different stories in the building, can be derived and implemented.

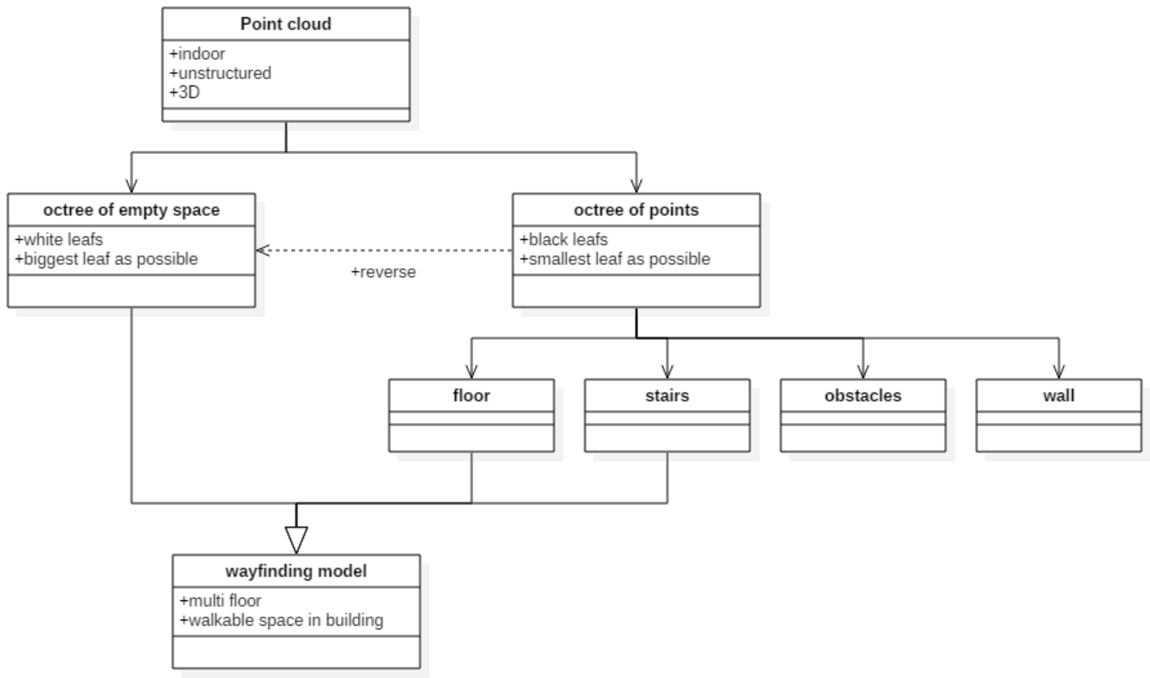

Figure 5: UML Project Logic Diagram

5.3 MoSCoW

The MoSCoW method is used for the prioritization of individual project requirements to reach a common understanding with stakeholders about each delivery. The term MoSCoW is derived from the first letter of each category seen on the left of the table 1, where the "M" stands for *must have*, or killer requirements, the "S" for the *should haves*, "C" for the *could haves* and finally the "W" for the *would like to (but most likely will not get) haves*. The *could haves* and downwards will be out of the scope (see section 3.2) of this research, but some of them might still be done if time and resources allow it.

References

3D Laser Mapping (2015). Zeb1 indoor mapping. `http://www.3dlasermapping.com/zeb1-indoor-mapping/`. Accessed 09.12.15.

Bansal, M., Matei, B., Southall, B., Eledath, J., and Sawhney, H. (2011). A lidar streaming architecture for mobile robotics with application to 3d structure characterization. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1803–1810.

Becker, T., Nagel, C., and Kolbe, T. (2009). Supporting contexts for indoor navigation using a multilayered space model. In *Mobile Data Management: Systems, Services and Middleware, 2009. MDM '09. Tenth International Conference on*, pages 680–685.

Bergeon, Y., Hadda, I., Krivanek, V., Motsch, J., and Stefek, A. (2015). Low cost 3d mapping for indoor navigation. In *Military Technologies (ICMT), 2015 International Conference on*, pages 1–5.

Boguslawski, P., Gold, C. M., and Ledoux, H. (2011). Modelling and analysing 3d buildings

Table 1: MoSCoW method

category	*task*	*further detailization*
Must	acquire point cloud	
	create octree structure of point cloud	structure empty space
	subdivide space in the octree	distinguish floors & walls
		detect stairs
		detect obstacles
Should	create network suitable for navigation	derive topology
		connect different stories
	map Quuppa coordinate to model	find 2D location in model
Could	visualize result	visualize path in point cloud
		visualize path in mesh (of Tango)
	put coordinate in relation	link asset to obstacle
		contextual snapping
	make path flexible	make path for walking person
		make path for rolling object
	create IndoorGML file	
	smooth the path	
	add real time navigation	
	add size constraint for wayfinding	connect to thesis of *O. Rodenberg*
Would	get 3D coordinate from Quuppa	
	make model updateable	apply changes real time
	add elevators to model	detect elevators automatically
	connect to Hue Lighting system	

with a primal/dual data structure. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(2):188–197.

Broersen, T., Fichtner, F., Heeres, E., de Liefde, I., and Rodenberg, O. (2015). Project pointless. identifying, visualising and pathfinding through empty space in interior point clouds using an octree approach. Geomatics Synthesis Project.

Delmerico, J., Baran, D., David, P., Ryde, J., and Corso, J. (2013). Ascending stairway modeling from dense depth imagery for traversability analysis. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2283–2290.

Diakité, A. A., Damiand, G., and Gesquière, G. (2014). Automatic semantic labelling of 3d buildings based on geometric and topological information. In *Proc. of 9th International 3DGeoInfo Conference (3DGeoInfo)*, 3DGeoInfo conference proceedings series, pages 49–63, Dubai, United Arab Emirates. Karlsruhe Institute of Technology.

Eilering, A., Yap, V., Johnson, J., and Hauser, K. (2014). Identifying support surfaces of climbable structures from 3d point clouds. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 6226–6231.

Google ATAP (2015). Project tango. `https://www.google.com/atap/project-tango/`. Accessed: 09.12.2015.

Halsted, G. (2014). Connexient blog: Google's project tango - and why a 3d model is not an indoor map. `http://www.connexient.com/blog/entry/google-s-project-tango-and-why-a-3d-model-is-not-an-indoor-map`. Accessed: 18.12.15.

Herman, M. (1986). Fast, three-dimensional, collision-free motion planning. In *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, volume 3, pages 1056–1063.

Jamali, A., Rahman, A. A., Boguslawski, P., Kumar, P., and Gold, C. M. (2015). An automated 3d modeling of topological indoor navigation network. *GeoJournal*, pages 1–14.

Jessup, J., Givigi, S., and Beaulieu, A. (2014). Robust and efficient multi-robot 3d mapping with octree based occupancy grids. In *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*, pages 3996–4001.

Karimi, H. (2015). *Indoor Wayfinding and Navigation*. CRC Press.

Khoshelham, K. and Díaz-Vilariño, L. (2014). 3D Modelling of Interior Spaces: Learning the Language of Indoor Architecture. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 321–326.

Kitamura, Y., Tanaka, T., Kishino, F., and Yachida, M. (1995). 3-d path planning in a dynamic environment using an octree and an artificial potential field. In *Intelligent Robots and Systems 95. 'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on*, volume 2, pages 474–481 vol.2.

Klepeis, N. E., Nelson, W. C., Ott, W. R., Robinson, J. P., Tsang, A. M., Switzer, P., Behar, J. V., Hern, S. C., and Engelmann, W. H. (2001). The national human activity pattern survey (nhaps): A resource for assessing exposure to environmental pollutants. *Journal of Exposure Analysis and Environmental Epidemiology*, 11(3):231–252.

Krūminaitė, M. and Zlatanova, S. (2014). Indoor space subdivision for indoor navigation. In *Proceedings of the Sixth ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness*, ISA '14, pages 25–31, New York, NY, USA. ACM.

Lee, J. and Zlatanova, S. (2008). A 3d data model and topological analyses for emergency response in urban areas. *Geospatial information technology for emergency response*, 143:C168.

Liu, L. and Zlatanova, S. (2012). A semantic data model for indoor navigation. In *Proceedings of the Fourth ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness*, ISA '12, pages 1–8, New York, NY, USA. ACM.

Liu, L. and Zlatanova, S. (2013a). Generating navigation models from existing building data. In *Acquisition and Modelling of Indoor and Enclosed Environments 2013, Cape Town, South Africa, 11-13 December 2013, ISPRS Archives Volume XL-4/W4, 2013*. ISPRS.

Liu, L. and Zlatanova, S. (2013b). A two-level path-finding strategy for indoor navigation. In *Intelligent Systems for Crisis Management*, pages 31–42. Springer.

Narasimhan, S., Mundani, R.-P., and Bungartz, H.-J. (2006). An octree-and a graph-based approach to support location aware navigation services. In *PSC*, pages 24–30.

Okorn, B., Xiong, X., Akinci, B., and Huber, D. (2010). Toward automated modeling of floor plans. In *Proceedings of the Symposium on 3D Data Processing, Visualization and Transmission*, volume 2.

Olufs, S. and Vincze, M. (2011). Towards efficient semantic real time mapping of man-made environments using microsoft's kinect. In *Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on*, pages 130–137.

Oßwald, S., Gutmann, J.-S., Hornung, A., and Bennewitz, M. (2011). From 3d point clouds to climbing stairs: A comparison of plane segmentation approaches for humanoids. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 93–98. IEEE.

Payeur, P. (2006). A computational technique for free space localization in 3-d multiresolution probabilistic environment models. *Instrumentation and Measurement, IEEE Transactions on*, 55(5):1734–1746.

Quuppa Oy (2015). Technology. `http://quuppa.com/technology/`. Accessed: 10.12.15.

Rusu, R., Marton, Z., Blodow, N., Holzbach, A., and Beetz, M. (2009). Model-based and learned semantic object labeling in 3d point cloud maps of kitchen environments. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 3601–3608.

Samet, H. (1989). Neighbor finding in images represented by octrees. *Computer Vision, Graphics, and Image Processing*, 46(3):367–386.

Sanchez, V. and Zakhor, A. (2012). Planar 3d modeling of building interiors from point cloud data. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 1777–1780.

Schnabel, R., Wahl, R., and Klein, R. (2007). Efficient ransac for point-cloud shape detection. In *Computer graphics forum*, volume 26, pages 214–226. Wiley Online Library.

Shen, S., Michael, N., and Kumar, V. (2011). Autonomous multi-floor indoor navigation with a computationally constrained mav. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 20–25.

SIMs3D (2015). About. `http://www.sims3d.net/`. Accessed 09.12.15.

Sinha, A., Papadakis, P., and Elara, M. (2014). A staircase detection method for 3d point clouds. In *Control Automation Robotics Vision (ICARCV), 2014 13th International Conference on*, pages 652–656.

Tseng, Y.-H. and Wang, M. (2005). Automatic plane extraction from lidar data based on octree splitting and merging segmentation. In *Geoscience and Remote Sensing Symposium, 2005. IGARSS '05. Proceedings. 2005 IEEE International*, volume 5, pages 3281–3284.

Turner, E., Cheng, P., and Zakhor, A. (2015). Fast, automated, scalable generation of textured 3d models of indoor environments. *Selected Topics in Signal Processing, IEEE Journal of*, 9(3):409–421.

Turner, E. and Zakhor, A. (2013). Watertight planar surface meshing of indoor point-clouds with voxel carving. In *3D Vision - 3DV 2013, 2013 International Conference on*, pages 41–48.

van der Ham, M. (2015). Real time localization of assets in hospitals using quuppa indoor positioning technology. Master's thesis, TU Delft.

Vo, A.-V., Truong-Hong, L., and Laefer, D. (2015). Aerial laser scanning and imagery data fusion for road detection in city scale. In *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*, pages 4177–4180.

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

Vörös, J. (2000). A strategy for repetitive neighbor finding in octree representations. *Image and Vision Computing*, 18(14):1085 – 1091.

Wang, F., Wang, K., Lai, S., Phang, S. K., Chen, B., and Lee, T. (2014). An efficient uav navigation solution for confined but partially known indoor environments. In *Control Automation (ICCA), 11th IEEE International Conference on*, pages 1351–1356.

Wang, M. and Tseng, Y.-H. (2004). Lidar data segmentation and classification based on octree structure. *parameters*, 1:5.

Wang, M. and Tseng, Y.-H. (2011). Incremental segmentation of lidar point clouds with an octree-structured voxel space. *The Photogrammetric Record*, 26(133):32–57.

Wang, M. and Tseng, Y.-H. (2014). Extraction of surface features from lidar point clouds using incremental segmentation strategy. *Journal of Photogrammetry and Remote Sensing*, 19(1).

Wu, L. and Hori, Y. (2006). Real-time collision-free path planning for robot manipulator based on octree model. In *Advanced Motion Control, 2006. 9th IEEE International Workshop on*, pages 284–288.

Yang, L. and Worboys, M. (2015). Generation of navigation graphs for indoor space. *International Journal of Geographical Information Science*, 26:1–20.

Yogeswaran, A. and Payeur, P. (2009). Features extraction from point clouds for automated detection of deformations on automotive body parts. In *Robotic and Sensors Environments, 2009. ROSE 2009. IEEE International Workshop on*, pages 122–127.

Yuan, W. and Schneider, M. (2011). 3d indoor route planning for arbitrary-shape objects. In *Database Systems for Adanced Applications*, pages 120–131. Springer.

Zhou, K., Gong, M., Huang, X., and Guo, B. (2011). Data-parallel octrees for surface reconstruction. *Visualization and Computer Graphics, IEEE Transactions on*, 17(5):669–681.

Zlatanova, S., Liu, L., Sithole, G., Zhao, J., and Mortari, F. (2014). Space subdivision for indoor applications. Technical report, Delft University of Technology, OTB Research Institute for the Built Environment.