Automatic Model Reconstruction of Indoor Manhattan-World Scenes from Dense Laser Range Data

Von der Fakultät Luft- und Raumfahrttechnik und Geodäsie der Universität Stuttgart
zur Erlangung der Würde eines Doktors der
Ingenieurwissenschaften (Dr.-Ing.) genehmigte Abhandlung

Vorgelegt von

M.Sc. Angela Budroni

aus Sardinien - Italien

Hauptberichter: Prof. Dr.-Ing. habil. Dieter Fritsch
Mitberichter: Prof. Dr.-Ing. habil. Thomas Wunderlich

Tag der mündlichen Prüfung: 18. Juli 2013

Institut für Photogrammetrie
der Universität Stuttgart

2013

Automatic Model Reconstruction of Indoor Manhattan-World Scenes from Dense Laser Range Data

A thesis accepted by the Faculty of Aerospace Engineering and Geodesy of the Universität Stuttgart in partial fulfilment of the requirements for the degree of Doctor of Engineering Sciences (Dr.-Ing.)

by

M.Sc. Angela Budroni

born in Sardinia - Italy

Main referee: Prof. Dr.-Ing. habil. Dieter Fritsch

Co-referee: Prof. Dr.-Ing. habil. Thomas Wunderlich

Date of defence: 18 July 2013

Institute for Photogrammetry
University of Stuttgart

2013

Angela Budroni

Automatic Model Reconstruction
of Indoor Manhattan-World Scenes
from Dense Laser Range Data

Automatic Model Reconstruction
of Indoor Manhattan-World Scenes
from Dense Laser Range Data

Von der Fakultät Luft- und Raumfahrttechnik und Geodäsie

der Universität Stuttgart

zur Erlangung der Würde eines

Doktors der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Abhandlung

Vorgelegt von

M.Sc. Angela Budroni

aus Sardinien – Italien

München 2013

Hauptberichter: Prof. Dr.-Ing. habil. Dieter Fritsch

Mitberichter: Prof. Dr.-Ing. habil. Thomas Wunderlich

Tag der mündlichen Prüfung: 18.07.2013

Diese Dissertation ist auch auf dem Dokumentenserver der Universität Stuttgart veröffentlicht

<http://elib.uni-stuttgart.de/opus/doku/e-diss.php>

Contents

List of abbreviations

2D	Two-dimensional
3D	Three-dimensional
AEC	Architecture Engineering Construction
AM	Amplitude Modulation
BIM	Building Information Modeling
B-rep	Boundary Representation
CAAD	Computer-aided Architectural Design
CAD	Computer Aided Design
CADD	Computer-aided Design and Drafting
CAFM	Computer Aided Facility Management
CityGML	City Geography Markup Language
CMM	Coordinate-measuring Machines
COBIE	Construction Operation Building Information Exchange
COLLADA	COLLAborative Design Activity
CSG	Constructive Solid Geometry
CW	Continuous Wave
DTM	Digital Terrain Model
DWF	Design Web Format
DWG	Drawing
DXF	Data Interchange Format
FM	Frequency Modulation
FOV	Field Of View
GIS	Geographic Information System
GPS	Global Positioning System
GSA	General Services Administration
HDS	High-definition Surveying
ICP	Iterative Closest Point
IFC	Industry Foundation Classes
ISO	International Organization for Standardization

IMU	Inertial Measurement Unit
LiDAR	Light Detection And Ranging
LOD	Level Of Detail
OGC	Open Geospatial Consortium
PRF	Pulse Repetition Frequency
RADAR	RAdar Detection And Ranging
RGB	Red Green Blue
TLS	Terrestrial Laser Scanning
TOF	Time-Of-Flight
VRML	Virtual Reality Modeling Language
XML	Extensible Markup Language

Zusammenfassung

Die dreidimensionale Modellierung erhielt seit jeher große Aufmerksamkeit von Computergrafikdesignern und wurde zudem mit der Verbreitung urbaner Szenarien zu einem wichtigen Gegenstand für die photogrammetrische Gemeinde und für Architekten. Zur Erzeugung dreidimensionaler Modelle realer Objekte werden sowohl effiziente Verfahren zur Erhebung visueller Informationen benötigt als auch robuste Methoden zur Berechnung der mathematischen Modelle, in denen diese Informationen gespeichert werden können. Photogrammetische Methoden für die Erfassung von Objektmerkmalen erzeugen dreidimensionale Objektprofile aus herkömmlichen Bildern, die sich aus Intensitätswerten zusammensetzen. Aktive Sensoren, in Form von Laserscannern, ermöglichen es direkt dreidimensionale Punktkoordinaten eines Objekts zu erheben und damit eine Beschreibung seiner geometrischen Eigenschaften schnell und zuverlässig bereitzustellen. Die Überführung der Laserdaten in konsistente Objektmodelle führt mit bestehenden CAD-Softwareprodukten zu manuellem Aufwand. Jedoch erfordert die zunehmende Verwendung dreidimensionaler Modelle in unterschiedlichen Anwendungsdomänen automatisierte Methoden zur Erzeugung dieser Modelle. Das Ziel dieser Arbeit besteht darin, ein neues Konzept zur automatischen Erzeugung dreidimensionaler Modelle aus Laserdaten zu entwickeln.

Die automatische Modellierungsmethode zielt auf die Rekonstruktion von Innenräumen mit orthogonalem Grundriss ab. Dazu werden zwei Aspekte betrachtet: die Extraktion aller Oberflächen, welche den Innenraum umgeben, und die Berechnung des Grundrisses. Als Ergebnis umfasst das dreidimensionale Modell die Geometrie und Topologie des Innenraums in Form eines B-rep. Der zugrundeliegende Ansatz der automatischen Modellierung basiert auf einem Plane-Sweep, der vergleichbar mit der in der Computergrafik verwendeten Sweep-Darstellung von Festkörpermodellen ist. Eine Datensegmentierung, die mittels des Sweeps und eines Hypothese-und-Test-Ansatzes durchgeführt wird, ermöglicht es jeden Laserpunkt einer Oberfläche des Innenraums zuzuweisen. Danach wird der Grundriss mittels einer Zellzerlegung basierend auf einem Split-and-Merge-Ansatz ermittelt.

Zur erfolgreichen Erzeugung von Modellen ist jede Aktivität des Rekonstruktionsprozesses zu berücksichtigen. Dieser umfasst die Erhebung der Laserdaten, die Identifikation der Punktwolken, die Berechnung des Models und die Visualisierung des Ergebnisses. Diese Arbeit stellt die komplette Umsetzung aller Aktivitäten des automatisierten Modellierungsprozesses vor. Darüber hinaus wird durch die Erhöhung des Automatisierungsgrades die weitere Verbreitung dreidimensionaler Modelle in verschiedene Domänen insbesondere in BIM-Prozessen für Architekturanwendungen angestrebt.

Abstract

Three-dimensional modeling has always received a great deal of attention from computer graphics designers and with emphasis on existing urban scenarios it became an important topic for the photogrammetric community and architects as well. The generation of three-dimensional models of real objects requires both efficient techniques to acquire visual information about the object characteristics and robust methods to compute the mathematical models in which this information can be stored. Photogrammetric techniques for measuring object features recover three-dimensional object profiles from conventional intensity images. Active sensors based on laser measurements are able to directly deliver three-dimensional point coordinates of an object providing a fast and reliable description of its geometric characteristics. In order to transform laser range data into consistent object models, existing CAD software products establish a valid support to manual based approaches. However, the growing use of three-dimensional models in different field of applications brings into focus the need for automated methods for the generation of models. The goal of this thesis is the development of a new concept for the automatic computation of three-dimensional building models from laser data.

The automatic modeling method aims at a reconstruction targeted on building interiors with an orthogonal layout. For this purpose, two aspects are considered: the extraction of all surfaces that enclose the interior volume and the computation of the floor plan. As a final result, the three-dimensional model integrates geometry and topology of the interior in terms of its boundary representation. The main idea underlying the automatic modeling is based on plane sweeping, a technique referable to the concept of sweep representation used in computer graphics to generate solid models. A data segmentation driven by the sweep and controlled by a hypothesis-and-test approach allows to assign each laser point to a surface of the building interior. At the next step of the algorithm, the floor plan is recovered by cell decomposition based on split and merge.

For a successful generation of the model every activity of the reconstruction workflow should be taken into consideration. This includes the acquisition of the laser data, the registration of the point clouds, the computation of the model and the visualization of the results. The dissertation provides a full implementation of all activities of the automatic modeling pipeline. Besides, due to the high degree of automation, it aims at contributing to the dissemination of three-dimensional models in different areas and in particular in BIM processes for architecture applications.

Chapter 1

Introduction

Three-dimensional (3D) model reconstruction addresses the problem of creating digital representations of physical objects after measuring their outer surfaces. The process of measurement and evaluation of the object surfaces can take place by means of either passive methods (camera based) or active techniques (laser scanners). In the present work an active sensor is preferred since the output point cloud already contains depth information and does not require additional processing steps like point matching to generate 3D data from stereo images. Besides the advantage of providing 3D data, a laser scanner can collect a huge amount of point measurements – a dense point cloud – in a relatively short interval of time. After the object surface measurement, the pure reconstruction consisting in the model generation is usually supported by dedicated computer graphics software. These applications contain specialized tools that support the users in the conversion of unstructured point clouds into Computer Aided Design (CAD) models based on the combination of primitive shapes. In order to make the reconstruction process more efficient by eliminating the dependency on manual modeling processing, this thesis aims at contributing in improving the degree of automation of 3D point cloud modeling.

In the introduction the motivations for developing an automatic modeling method are highlighted together with general background information about potential application fields. The objectives of the thesis are set and an overview of its structure is given in the last section.

1.1 Motivation

A growing attention concentrates upon the development of new automatic techniques for building representation to facilitate the widespread availability of interactive visual environments. For a long time within the geodetic community automatic building modeling has been restricted to the simple computation of outer building shapes. However, for a more exhaustive representation the indoor environment has to be considered as well. 3D models, which include interior details, can broaden the application area of virtual architecture to other domains beyond those covered by exterior representations. In particular, indoor modeling applies to various scenarios with different purposes ranging from close-range documentation, visualization or simulation to navigation support, augmented reality and location-based services for indoor object search. Although the actual state of the art is not especially provided with research about automatic indoor reconstruction, most efforts to compute analytic topological models of interior environments come from the field of computer vision.

Whereas the collection of consistent data for 3D exterior building modeling is performed with air- and space-borne sensors besides terrestrial acquisitions for higher detailed surfaces, the reconstruction of build-

ing interiors exclusively uses close-range photogrammetric data or 3D information directly collected by a terrestrial laser scanner. Apparently, the interest towards interior reconstruction is less obvious than the attention for traditional city models. Data for achieving 3D city models might be collected and managed by public organizations like the national mapping agencies or private companies specialized in remote sensing operations. In contrast, no centralized agency is responsible neither for recording building interior measurements nor for computing 3D interior models.

Nevertheless, indoor models are useful in a wide variety of domains in the area of civil engineering, architecture and cultural heritage preservation, real estate, tourism, city management and security. Simple model visualization and data navigation can support virtual tours into indoor contexts like museums or expositions. In the same way, the implementation of intelligent buildings – smart homes – might require detailed virtual 3D maps to locate people and smart objects in order to provide interaction between users and their home environments. Methods for automatic reconstruction of interiors could also be embedded in existing applications like Google Earth and Microsoft Bing Maps for enabling the virtual representation of the indoor scenes and the simultaneous documentation and localization of the related facilities. This would allow for virtual stores and provide additional marketing potential in the oncoming scenario where users of digital infrastructures are connected anytime through the Internet. The importance of 3D interior modeling also pertains to emergency management and rescue operations. Disaster management strategies and risk assessment evaluation could take advantage of indoor models to plan evacuation strategies. If the models provide enough details and if they are geometrically accurate, they can be used for simulations by enabling realistic training scenarios for the localization of safety-relevant features [Kolbe et al., 2005]. Less critical scenarios also involve city infrastructure management which might exploit indoor models to monitor damaged structures.

For the widespread dissemination of indoor models, automation in the reconstruction process is essential in order to ensure a better trade-off between costs and efficiency of the model generation. A high degree of automation is required in all processing phases, from the data acquisition to the computation that leads to the final model. While data acquisition and low-level processing are already performed at a high automation level, model reconstruction of indoor scenarios is currently performed predominantly using manual approaches. This is in strong contrast to the successful application of automated approaches for the reconstruction of exterior building models [Brenner, 2004a]. Therefore a new method is needed that aims at filling the existing gap between the automation levels of exterior and interior building reconstruction. Computer aided facility management (CAFM) might also benefit from the automation of indoor reconstruction. CAD tools and building information modeling (BIM) might increase their representation power by integrating an automatic modeling method to their functionalities.

To guarantee high metric accuracy, spatial properties of interior scenes are typically captured by terrestrial laser scanners. Depending on the instrument settings and the network geometry, acquisitions with laser scanners can extract object point coordinates of wide surfaces yet preserving fine details. An essential part of the reconstruction is to interpret such a huge amount of points and bring it into a configuration which can be stored and visualized. Due to the fact that commercial laser scanners are expensive and not easily portable, they can be substituted by laser scanner prototypes as well as by other kind of sensors, such as consumer-grade range cameras or intensity cameras delivering image pairs.

The computation of 3D building models can take place at different detail scales. If a reconstructed building needs to be visualized in real time for example from a web application, it is convenient to create several model representations, each one providing a varying amount of details to the application. This allows the visualization of the model from progressively closer distances with a number of details that increases ac-

cordingly. The levels of detail (LODs) used to represent 3D city models are entirely defined in the City Geography Markup Language (CityGML) encoding standard [OGC, 2008], which supports multi-scale modeling ranging from regional models (LOD0) to interior models (LOD4). The latter are so-called "walkable" architecture models that contain the highest number of details.

1.2 Objectives

The work presented in this thesis aims at the reconstruction of indoor building models at the highest level of detail (LOD4). The innovative aspect of the developed method consists in the automation of the reconstruction process, which provides as outputs CAD models of existing interior scenes. As inputs to the method, 3D point clouds acquired by a laser scanner (or other range sensors) are considered. Although the point clouds delivered by the scanner just contain a list of point coordinates but no topological information about the indoor environment, semantic knowledge about the interior is not required a priori. This is directly recovered during the reconstruction by applying neighborhood constraints to adjacent areas of the indoor ground plan.

The initial accomplishment towards the 3D model generation is the automatic detection of all planar structures in the room. For this purpose a segmentation technique based on plane sweep is developed that ensures a robust localization of all data points. Then the ground plan is computed from the segmentation output. Principles of solid modeling are customized to be applied to the point cloud data. The calculation of the ground plan is an essential step for achieving the correct modeling of the interior volume. The results are investigated in order to establish how much the robustness of the modeling approach can influence the accuracy of the final reconstruction.

Compared to existing approaches to building modeling, the method described in this dissertation contains two aspects that have not been considered extensively in the context of 3D reconstruction yet. The first aspect is the level of details involved, which by definition distinguishes between outer building architecture with varying complexity and high detailed indoor models. While models of outer building shapes were already topic of research especially among the geodetic and photogrammetry communities, the reconstruction of building interiors is the target of the research in this work. The utility of detailed indoor models could be extended to applications in the context of computer vision and architecture as well.

The second feature that distinguishes the automatic model reconstruction from other approaches is the loss of manual intervention during the model computation. Overall, the combination of automation and high level of details makes the method unique. An overview of the existing approaches is given as comparison to the developed method in Chapter 4.

In conclusion, the final objectives this dissertation wants to achieve can be assigned on two different levels. One level is about the general objectives which refer to global results. These are set at the really beginning of the work and are summarized as follows:

- A robust modeling method should be defined that overcomes typical issues of least square fitting approaches such as the dependency of the output from outliers;

- The method should automatically calculate 3D models from unorganized point clouds;

- The input points are directly obtained by indoor acquisitions, therefore the models aim at preserving interior architectural details.

At a more specific level, additional objectives are set as intermediate steps to the general development of the automatic model reconstruction. These are:

- The segmentation of planar structures from the point cloud;

- The calculation of the (reciprocal) positions of such structures;

- The definition of the interior topology;

- The ground plan computation;

- The extraction of the doors as deformation of the wall profile.

In the next chapters the fine objectives will be described. By their accomplishment, the global requirements are automatically satisfied as well.

1.3 Structure of the thesis

The thesis is organized as follows. In Chapter 2 techniques and measurement systems for model reconstruction are described. A brief classification of active and passive sensors is given with particular focus on stereo vision and photogrammetric systems on the one hand, active triangulation sensors and sensors based on time delay measurements on the other hand. The characteristics and physical principles of pulsed laser scanners applied to terrestrial acquisitions are explained in detail. Finally, the pipeline for reconstructing real scenes from range data is described including the point acquisition, data registration and general modeling principles.

Building model reconstruction, representation and management are the main topics of Chapter 3. First a classification of building models according to the standard CityGML is given. Then the building extraction process is described as a sequence of three steps: detection, reconstruction and management. Reconstruction techniques are explained in detail in accordance with three major categories including decomposition models, constructive models and boundary models. The importance of building information modeling processes is pointed out in the last section of the chapter.

Chapter 4 is fully dedicated to the automatic method for indoor scene reconstruction. After defining the problem domain and illustrating the characteristics of Manhattan-world scenes, an overview of the automatic modeling algorithm is given. It follows an exhaustive description of all steps towards the indoor volume sweep reconstruction. The reconstruction is determined by both a linear sweep and a rotational sweep to compute the position and orientation of geometric primitives. Afterwards, the floor plan cell decomposition is described including the techniques for localizing the cut-lines and specifying the full cells. The approach to door extraction is also discussed in this chapter.

In Chapter 5 a selection of case studies is presented. The chapter starts with an introductory section that is dedicated to the laser scanner pipeline with reference to the sequence of measurement operations performed in the specific case studies. The next section describes the automatic modeling results in detail. An overview of all control variables, which are set to control the output of the algorithm, is provided. Then, an account is given of concrete examples used for testing the automatic reconstruction. Finally, the evaluation of the results is reported with respect to a least square fitting. Chapter 6 gives the conclusion of the described work and points out potential improvements for the algorithm and future developments. In Appendix A the moment method for plane fitting is reviewed, for which the least squares model fitting is used for the evaluation of the results.

Chapter 2

Three-dimensional sensing for model reconstruction

This chapter describes the state of the art of the measurement and processing techniques used to recover three-dimensional models from existing objects.

2.1 3D sensor overview

The primary purpose of a 3D reconstruction operation, as intended in this thesis, is the geometrical representation of real architectural objects. This approach differs conceptually from the techniques employed in the field of computer graphics for the design of artificial environments dedicated to virtual reality simulations. Modeling objects from reality [Ikeuchi and Sato, 2001] requires accurate measurements of the object points followed by post processing computations to assign the points their correct geometric configuration. This kind of reconstruction problem implies the determination of a mathematical model adequate for storing and visualizing 3D object coordinates. The investigation of such a model has been a constant research topic in the vision and photogrammetric community [Remondino and El-Hakim, 2006]. As pointed out by Budroni and Boehm [2010], the availability of 3D models has recently generated great interest among architects as well, with regards to the interoperability requirements of BIM strategies.

In the following paragraphs an overview of sensors, whose acquisitions are used as inputs for the generation of 3D models, is given. At a very general level, close range sensors for 3D measurements are distinguished into contact and non-contact sensors. The first class refers to devices that use an appropriate touch probe to map object distances (depths). This may be inconvenient if the object under measurements is delicate or valuable. However, an advantage of contact sensors is that they can reach a precision in the order of microns, which makes these devices particularly suitable for manufacturing processes. Examples of such devices are rulers or coordinate-measuring machines (CMM). On the other hand, non-contact sensors are supposed to measure object coordinates on the basis of optical principles, by hitting the object surface with an electromagnetic signal or using a camera. An integration of touch probes and optical sensors is also feasible, according to the requirements for the design of new CMMs [Sansoni et al., 2009]. Non-contact sensors are very convenient since they do not interfere mechanically with the target object. Their versatility allows for a cross-use that supports diverse practices, from the preservation of cultural heritage [Beraldin et al., 2003] and building restoration to various industrial applications, indoor navigation, disaster management or simple scene understanding.

In the last fifty years vision systems based on non-contact active sensors have found different fields of applications. Their spread is due to the advances that have been made concerning the electro-optical technologies for distance measurements [Vosselman, 2010]. In addition to optical sensors, dedicated computer softwares

constitute a powerful framework for data acquisition and on-line verification, which open the possibility of visualizing the data while collecting them, thus improving efficiency and reliability of the whole system. The development of imaging sensors, primarily challenging for scientists, became extensive also beyond the academic environment, meeting a very positive market response [Blais, 2004].

The rest of the chapter focuses exclusively on non-contact sensors, describing both active and passive sensors (see classification in Figure 2.1). Such devices differ in the necessity of lighting up the object with an additional source while measuring its profile. Passive sensors require sources of external light during the acquisition, whereas active sensors emit their own light radiation. In this case, the backscattered radiation of the emitted wave determines single point positions along the wave path. As a result of the measuring process, range data is available from the acquisitions with active sensors, while passive sensors provide intensity images mainly collected using modern CCD cameras. A similar classification between techniques resulting into range data (direct techniques) and measurements that infer 3D information from monocular images (indirect measurements) can be found in [Sansoni et al., 2009]. As the inputs for this thesis have been generated by acquisitions from interiors of buildings, the distances involved are limited. This allows to restrict the following sensor classification to close range measurement systems.
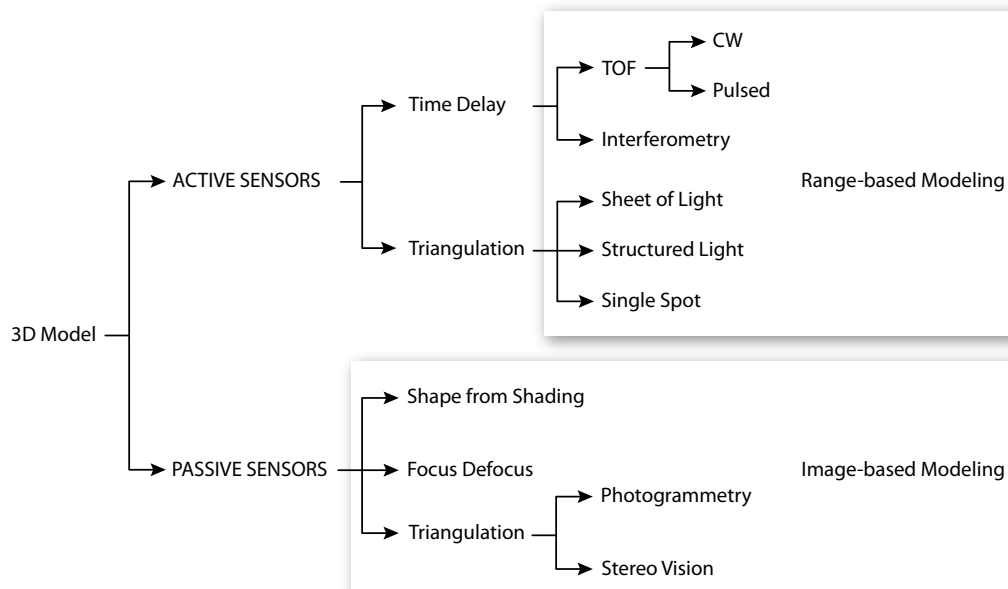
Figure 2.1: Overview of 3D sensors for model reconstruction. Depending on the type of sensor, range-based and image-based modeling techniques are distinguished.

2.1.1 Passive sensors

Passive sensors reconstruct 3D coordinates from the analysis of two-dimensional (2D) intensity images. Either several images from different views are needed to find point correspondences or techniques such as shape from shading are used to evaluate the object profile under given light conditions. Passive systems are usually compact, thus easily transportable and not particularly expensive. They provide a high accuracy on targets with well-defined textures. On the other hand, they require projective geometry computations [Hartley and Zisserman, 2003] as a post processing phase to generate 3D information in the form of point sets, better known as point clouds.

The study about interior reconstruction characterizing this thesis processes data acquired by an active device. Nevertheless, passive sensors are mentioned here for the full consistency of the 3D sensor overview and for motivating the choice of a laser scanner as a measuring system. In particular, passive sensors that recover 3D points using triangulation techniques are described. A distinction between triangulation approaches in computer vision and in photogrammetry is made according to the existing literature, which especially focuses on the different calibration methods adopted by the two communities.

Stereo vision

One fundamental issue of techniques based on passive triangulation is the computation of intrinsic (optical) and extrinsic (geometrical) camera parameters needed for the solution of the 3D reconstruction problem. The mathematical determination of internal sensor parameters – focal length, principal point and lens distortion – is carried out by the camera calibration, while pose estimation recovers camera position and orientation with respect to the global object coordinate system. Several calibration techniques are available for the calculation of the optical characteristics of the camera. Common calibration procedures for stereo vision systems, such as those of Tsai [1987] and Zhang [2000], make use of reference grids. The calibration matrix that contains the intrinsic camera parameters is determined by the knowledge of a given array of point coordinates on the calibration object. On the other hand, self-calibration does not require a calibration object but uses geometric invariants [Mundy, 1993], which exploit fix relationships among geometrical entities in the recorded images. Invariants are independent of both camera motion and camera orientation. According to Remondino and Fraser [2006], computer vision applications cannot profit from a universal calibration model due to the imaging hardware characteristics (variable zoom lenses) and the poor network geometry adopted.

Typically, stereo vision systems include a couple of slightly displaced cameras to provide perspective views of the object. The pipeline of a stereo project starts with the image rectification through epipolar geometry: given an observed point on one image, correspondence analysis constrains the position of the related point on the second image. The problem complexity is reduced and the search for image correspondences in the following step – stereo matching – is restricted to a linear domain. From the resulting disparity map, point depths are recovered by triangulation for a full 3D reconstruction of the scene. Since for many computer vision applications such as object recognition or navigation fast scene interpretation is essential, the primary requirement in implementing stereo vision systems is a high-speed response. Another challenge is the reduction of human intervention during the stereo matching phase in order to provide (full) automatic generation of 3D models. Imaging systems for stereo vision often consist of commercial off-the-shelf cameras.

Close range photogrammetric systems

Photogrammetry is another example of passive triangulation. It is traditionally associated with surveying techniques for natural and cultural heritage site documentation and industrial metrology. Referring to the categorization reported by Luhmann et al. [2006], close range photogrammetric systems operate in a configuration at a maximum distance of 300m from the object. Although photogrammetry shares many goals with stereo vision, it aims to a higher accuracy of the reconstruction process [Mundy, 1993]. Accuracy of photogrammetric measurements is determined by several factors, among which the most relevant are the camera resolution, the distance camera-object, the number of photographs available and the acquisition layout.

Differently from stereo vision, the photogrammetric calibration recovers principal point and lens distortion in addition to the camera focal length. This kind of calibration is needed for precise photogrammetric

measurements, which might require accuracies of 1:20,000, whereas a typical accuracy provided by stereo vision systems is 5% of the distance camera-object [Remondino and Fraser, 2006]. According to that, in a configuration where the camera is one meter away from the object, stereo vision systems show an accuracy that is one order of magnitude lower than the accuracy typical of photogrammetric applications. The high accuracy of photogrammetric measurements is also guaranteed in the calibration phase by a suitable network geometry that allows for the acquisition of convergent and rotated images and ensures a stable estimation of the camera parameters.

The parameters needed for the sensor calibration and orientation are usually estimated by bundle adjustment, whose mathematical model is based on collinearity constraints (the ray connecting object point, principal point and image point is a straight line). It is within a simultaneous calibration [Luhmann et al., 2006] that the bundle adjustment retrieves 3D object point coordinates besides the camera interior orientation. Examples of software packages that are nowadays available commercially for these purposes are Australis from Photometrix and Photomodeler from Eos Systems. The camera calibration performed by such software takes as an input a network of digital monoscopic images acquired in sequence by a camera that moves relatively to the scene.

A methodological classification of cameras suitable for photogrammetry is given by Kraus [2007]. Metric cameras are large format cameras specially designed for photogrammetric purposes. They have fixed intrinsic parameters yielding a stable camera orientation. Also, lenses are free from distortion. Semi-metric cameras have been developed for photogrammetry as well. However, the stability of the camera intrinsic parameters is not guaranteed. Non-metric cameras are off-the-shelf devices thought as commercial products, usually used for low accuracy applications.

In conclusion, it should be noticed that photogrammetry stands for any general process that generates three-dimensional information out of photographed objects or images in general. Photogrammetry also includes those techniques that not necessarily reconstruct the third dimension from multi-view approaches. If no information originated by a stereoscopic view is available, photogrammetric principles may be applied thanks to the support of other techniques, such as the vanishing point detection procedure adopted in [Van den Heuvel, 2001]. Therefore, not only stereo pairs like those used in the field of stereo vision are material of photogrammetry, but also monocular images are sufficient to provide a valuable three-dimensional object representation. In this regard, Luhmann et al. [2006] also specify a categorization of photogrammetry by number of measurement images. The result is a distinction between single image photogrammetry, stereophotogrammetry and multi-image photogrammetry.

2.1.2 Active sensors

Active sensors emit their own energy radiation toward the target to be investigated. They include optical measuring systems, which obtain dimensional information from the interaction between a light wave and the object under measurement. Object coordinates are retrieved from properties that are not characteristic of the object itself but refer to the entire configuration sensor-target. The reflection of the light signal is used to recover object depths from features such as interference or time-of-flight (TOF). The latter refers to the time spent by the emitted signal for traveling from the device to the target and then back to the sensor. Other techniques based on active triangulation measure the position of projected patterns on the object surface. As outlined in [El-Hakim et al., 1995], active 3D vision systems – for instance laser scanners – have a wide operational domain since their integrated illumination source makes them appropriate for measurements in most environments. However, their acquisitions are influenced by high-reflectance surfaces and sharp

discontinuities. Differently from passive sensors, they perform good results in case of missing textures on the object surface.

Referring to the overview given by Besl [1988] about active imaging sensors, the performance parameters that characterize such measuring devices are resolution, repeatability and accuracy. Besl [1988] defines the range resolution as the smallest change in range that a sensor can report. This corresponds to the ability of the sensor to distinguish two targets as separate if they are positioned at a minimum given distance from each other. Range repeatability refers to the sensor response to repeated measurements of the same distance, while range accuracy defines the shift of the measured distance from a known true value (the accuracy of a sensor can be improved by averaging multiple measurements). Besides, in his work Besl [1988] also raises a classification of active sensors. These are mainly divided into imaging radars and active triangulation sensors. The first category includes time-of-flight sensors based on pulsed measurements, amplitude modulation or phase modulation. Another category comprises triangulation methods based on structured light techniques: from point triangulation to 3D measurements of lines or coded binary patterns. Beyond these two classes of devices, Moiré techniques and holographic interferometry are also mentioned. The first use projected fringes – Moiré patterns – for mapping surface profiles. As discussed in [Nishijima and Oster, 1964], Moiré patterns might be obtained by the superposition of either two rulings of parallel lines with different spacing or two images representing concentric circles. Holographic interferometry projects laser light and computes displacement measurements that are caused by stress applied on the target object. The sensor classification given by Besl [1988] is consistent with the classification overview in Figure 2.1 and will be taken as a reference for the following theory about active devices.

Active triangulation-based sensors

3D sensing using triangulation methods applies the fundamental theorems of trigonometry to compute the optimal configuration of the measuring system and recover the object surface points. The typical setup of a triangulation-based device includes a laser projector at one end of the system and, at a given distance B from it (baseline), a camera lens used to map the projected pattern.

The simplest triangulation method addresses the problem of recovering the position of one point-like pattern at a time. This is the case of a laser source emitting a narrow beam, which spots a single point on the object surface. The light backscattered by the surface point travels back to the camera, where it is imaged on the position sensor according to the pinhole camera model simplification (Figure 2.2). In order to vary the laser spot position on the scanned surface and thus to measure a multiplicity of points on it, either deflection mirrors are used or the laser source is manually moved in the scene. By the knowledge of the baseline B and the orientations of both the emitted and the reflected radiation path, the coordinates of the target point on the surface are retrieved through the cosine law. With reference to Figure 2.2, the baseline is calculated as the sum of the X coordinate of the laser spot on the object and its orthogonal distance to the lens along X:

$$B = X + Ztan\beta = Ztan\alpha + Ztan\beta. \tag{2.1}$$

Therefore, from equation (2.1) the orthogonal distance Z between the measured point and the system yields

$$Z = \frac{B}{tan\alpha + tan\beta} = \frac{Bf}{ftan\alpha + p} \tag{2.2}$$

where $p = ftan\beta$ gives the position of the backscattered laser spot on the imaging sensor. The last term in equation (2.2) is obtained by a multiplication of both numerator and denominator by the camera focal length
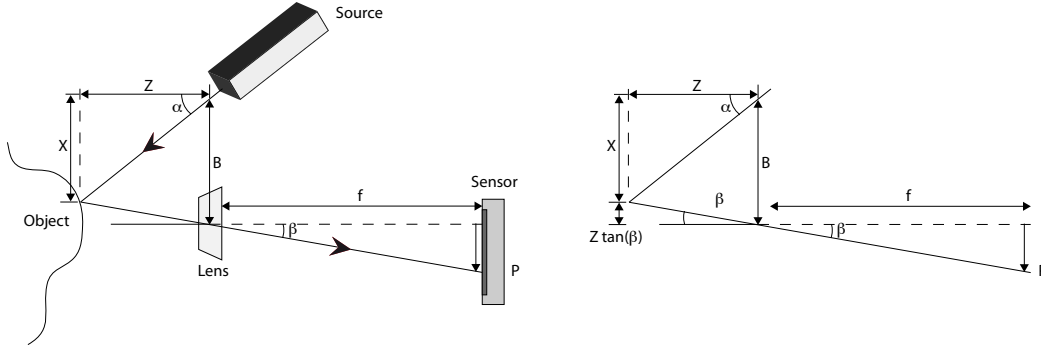
Figure 2.2: Point triangulation geometry: the laser beam is reflected by the object surface toward the position sensor (left). The position of the measured point on the sensor is computed by trigonometric laws (right).

f. According to Vosselman [2010], the error in Z is inversely proportional to the baseline B and the focal length f, but directly proportional to the square of Z:

$$\delta_z \approx \frac{Z^2}{fB}\delta_p. \tag{2.3}$$

The dependance of the error δ_z on the distance sensor-target explains why triangulation-based sensors are available commercially only for applications requiring a maximum range of 10 meters [Beraldin et al., 2003]. Besides, the uncertainty in the range measurement is directly proportional to δ_p, which is a measure of the uncertainty of the laser spot position on the sensor.

Focal length and baseline are both system-dependent critical parameters. The baseline B cannot be arbitrarily modified because it determines the field of view (FOV) of the system, which fixes the laser beam orientation and controls shadowing artifacts. In order to improve the performances of a triangulation-based system, Rioux [1984] has proposed the introduction of polygonal or pyramidal mirrors for the implementation of autosynchronized scanners. A single laser scanner is used as a projector and detector at the same time, thus eliminating problems related to the configuration of the baseline geometry.

Due to the fact that systems based on point triangulation can only measure a single point each time, they give lower performances if compared to other triangulation approaches. Faster methods are preferred, such as the projection of light sheets or structured patterns. These methods share the same triangulation principle described above but their commercial implementation provides stereo measurements with a better accuracy and a higher scan rate. The projection of light sheets is realized with the addition of a cylindrical lens in front of the laser source. This causes the emitted light beam to be reshaped and diffused along a planar path. When the planar light radiation intersects the object surface, points on a line are illuminated simultaneously, thus they can be measured by the position sensor at the same instant of time. An extension of light sheet projection systems is represented by sensors based on structured light. They further increase the number of multiple measurements at a time by projecting two-dimensional coded patterns, rather than a linear illumination.

In general, the most interesting advantage of laser triangulation systems is their high accuracy, which makes single point sensors suitable for diverse industrial applications (e.g., reverse engineering) and light sheet projectors widespread in the field of cultural heritage.

Sensors based on time delay measurements

Active sensors based on time delay measurements exploit the time-of-flight principle. TOF techniques have primarily been used with radio waves becoming known as RADAR (RAdio Detection And Ranging). The same concept extended to light waves is nowadays referred to as LiDAR (Light Detection And Ranging) and is widely used in optical vision systems to compute depth values along variable ranges. LiDAR systems measure the traveling time interval of a laser wave from the instant of its emission until the backscattered signal is received by the sensor. Each pixel maps the traveling time (depth) of one single point in the scene. This implies that the wave travels toward the target where it strikes a surface point, returning then back to the source. To compute the roundtrip travel time of the emitted signal, the property of propagation velocity of the light wave is considered. In a given medium with refractive index n, the speed of the light wave is known and given by the constant c/n, where c is the light speed in a vacuum, approximately $3x10^8$m/s. Therefore, considering that the distance sensor-target (range) is half of the complete path covered by the radiation, the value of depth is computed by the sensor as

$$d = c\frac{\Delta t}{2},\qquad(2.4)$$

assuming $n = 1$ as the approximation of the refractive index in air at standard temperature and pressure. Equation (2.4) shows how the time-of-flight Δt is directly proportional to the measured distance.

Sensors based on the TOF principle are available for applications requiring different ranges. Short and medium range sensors are capable of measurements up to 10m and 15m, respectively, while long range devices can provide depth maps of objects positioned at distances between 15m and 100m [Sansoni et al., 2009]. Long range measurements also give good results in terms of depth resolution, whereas short range acquisitions (small objects) need a high-frequency clock to detect TOF differences in the order of picoseconds with 1:1,000 accuracy. Unlike triangulation-based systems, TOF sensors are not affected by shadow effects as the path of the emitted radiation coincides with the trajectory of the reflected signal. On the other hand, the reflectivity of shiny surfaces might disturb the measurements.

TOF ranging techniques include pulse runtime and phase shift determination methods. The first technology directly measures the roundtrip time of a single pulse from equation (2.4). It is fast and mainly suited for long distance applications such as scans of whole buildings, rooms and archeological sites. In many cases, however, the high pulse energy required to map long distances becomes a critical issue in the consideration of laser safety. Both airborne and terrestrial laser scanners implement pulse runtime approaches to carry out long range measurements. Example of pulsed laser scanners available on the market for terrestrial surveying are the new Leica ScanStation C10 and the Riegl VZ-Series. The principle of operation of pulsed laser scanner is explained in detail in the following section.

The second class of ranging systems computes the phase shift generally using amplitude modulation (AM), even though frequency modulation (FM) might also be applied. The source emits a continuous wave (CW) whose intensity is modulated with a defined function, for instance a sinusoid. The TOF is determined by the computation of the phase difference $\Delta\phi$ between the incident and the reflected wave. The subtraction of the two waves yields

$$s(t) = Msin(2\pi f_m t - \Delta\phi) = Msin\left(2\pi f_m\left(t - \Delta\phi/2\pi f_m\right)\right),\qquad(2.5)$$

where M and f_m indicate the amplitude and the frequency of the modulated signal, respectively. The TOF is given by the term

$$\Delta t = \frac{\Delta\phi}{2\pi f_m} \tag{2.6}$$

with $f_m = c/\lambda_m$. Due to the periodicity of the modulated signal, the wavelength of the light source λ_m limits the unambiguous range to $\lambda_m/2$. The range resolution is proportional to the wavelength as well. In fact, the combination of equation (2.6) with (2.4) shows that $d_{min} = (\Delta\phi/2\pi)(\lambda_m/2)$. Thus, if the phase resolution is kept constant, a shorter wavelength corresponds to a more precise measurement. However, the shorter the wavelength, the less extensive the unambiguous range whose measure defines the domain where different echoes are actually detected as separate. A method to widen the unambiguous distance is to take two or more measurements, each one with a different modulation frequency. The effective unambiguous range results then increased to the least common multiple between the two initial ranges [Gokturk et al., 2004]. Laser scanners based on phase shift measurements are preferred to pulsed scanners if millimeter or sub-millimeter accuracies are required. They are cheaper than pulsed scanners and they are generally used for very fast acquisitions of room-sized scenes. Valuable examples of CW systems are the Photon laser scanner from FARO as well as their new Focus[3D] (Figure 2.4b) or the Z+F IMAGER® from Zoller+Fröhlich. A comparison between the characteristics of pulsed and CW scanners is given in Table 2.1.

Beyond laser scanners, imaging systems that exploit time delay (phase shift) measurements include also compact low resolution cameras, mostly suited for acquisitions of short range dynamic scenes. Collision prevention for parking assistance and out-of-position analysis are some examples of current applications of such TOF devices in the automotive industry [Hsu et al., 2006]. The growing potential of these devices has aroused the interest of the computer vision community as well [Kolb et al., 2009]. If compared to laser scanners, TOF cameras are small-sized and low cost, thus they allow for a higher portability. However, their low resolution – last generation TOF cameras provide an image size of 200x200 pixels – becomes inappropriate for recording high detailed large scenes. An architectural surveying for collection of data as described in this thesis could not be feasible with the employment of a 3D TOF camera alone.

TOF	Scanner	Wavelength	Range	Range Accuracy	Scan Rate (points/s)
Pulsed	Leica ScanStation C10	532nm green light	from 0.1m to: 300m @ 90% reflect.	at 1–50m: ±4mm	50000
Pulsed	Leica HDS3000	green light	from 0.1m to: 300m @ 90% reflect.	at 1–50m: ±4mm	4000
CW	Faro Photon 120	785nm red light	from 0.6m to: 153m @ 90% reflect.	at 25m: ±2mm	976000
Pulsed	Riegl VZ-400	red light	from 1.5m to: 350m @ 90% reflect.	at 100m: ±5mm	122000

Table 2.1: Specifications of laser scanners from different vendors as reported on their technical sheets.

2.2 Terrestrial laser scanning systems based on pulse runtime determination

Sensors that measure distances by computing the pulse runtime have been already mentioned in the previous section. In this section, the characteristics and the physical principles of pulsed laser scanners for terrestrial applications are described in detail. Terrestrial acquisitions refer to medium and long range measurements performed by the instrument usually operated on a tripod. Under these conditions, laser pulses are emitted to different directions according to intrinsic scanning mechanisms. The faster the pulse repetition frequency (PRF), the higher the number of object point distances that are measured in the time unit. Although pulsed laser scanners are less accurate than phase shift scanners, they are generally preferred for general purpose surface measurements [Luhmann et al., 2006]. A comparison of the technical specifications of a pulsed scanner – the ScanStation C10 – and a CW scanner – the Photon 120 – shows how both range accuracy and scan rate of the phase shift scanner deliver better values than the pulsed scanner. FARO claims 2mm accuracy at 25m and a scan rate of less than 1,000,000 points/sec, whereas the scanner from Leica provides an accuracy of 4mm and an acquisition speed of 50,000 points/sec. However, the ScanStation C10 can operate at a maximum range of 300m if the surface to be scanned is characterized by 90% albedo. With the same reflectivity conditions, the Photon 120 can measure surface points at a maximum range of 153m (Table 2.1).

A pulsed laser light source produces as an output a periodic signal whose amplitude oscillates between energy peaks and zero values. Each peak has a duration – pulse width – and a rise time that determine the range resolution of the system. In fact, reflected echoes from different object points are discriminated only if their reciprocal distance is greater then half of the pulse length [Vosselman, 2010]. Typically, a pulse width of 5ns results in a length of 1.5m, from (2.4). The pulse shape can easily be approximated to a rectangle, although more realistic Gaussian-shaped pulses are often considered. In addition to the time distribution of the laser pulse, a distribution in space causes a reduction of the pulse energy along the direction perpendicular to the propagation path. Indeed, the laser beam diameter is limited by the boundary of the circular region in which the energy is higher than $1/e^n$, with $n = 1, 2$ [Pfeifer and Briese, 2007]. The energy irradiated with the pulse depends on the PRF as well: the higher the number of pulses transmitted per second (higher frequency), the lower the pulse energy (short pulse duration), thus decreasing the measurement range. This effect becomes observable in the technical specifications of the pulsed laser scanner VZ-400 from Riegl [Riegl, 2012], where the maximum range performance is classified relatively to the PRF.

Terrestrial scanning is a stationary process in the sense that the scanner is not moved while sending the pulses (that, of course, does not apply to mobile mapping). Therefore, to allow for a transition of the laser beam from the vertical to the horizontal direction, internal deflection mechanisms are needed besides a scanner rotation about its vertical axis. Such beam deflection mechanisms are performed inside the scanner head. The main scanner unit, or scanner head contains the range sensor electronics that constitutes the core component of the whole laser scanning system. Usually, additional modules are necessary during scanning to support the acquisition process. These modules are the control unit, the power supply – internal and external batteries – and, potentially, the tripod. The control is manually operated via a laptop computer that is connected to the scanner as an interface for setting the desired sampling parameters and saving the acquired data set. A software installed on the laptop also lets the 3D points be displayed as soon as they are measured by the sensors, thus facilitating a comprehensive scan interpretation. In general, modern laser scanners do not need to be controlled by an external computer device since they often provide onboard graphic interfaces, as the Leica ScanStation C10 or the new FARO Focus[3D]. Within the scanner unit, a laser rangefinder computes object distances through TOF measurements. In [Amman et al., 2001], a

block diagram of a rangefinder shows its essential components: a transmitter and a receiver unit, the time measurement electronic and the optics. Separately from the range sensor, the laser beam deflection unit is responsible for the scanning mechanisms.

2.2.1 Scanning mechanisms

The scanner emits one pulse at a time and measures the backscattered echo to recover the distance of the illuminated spot. In order to obtain densely sampled surfaces, echoes from a multitude of closely spaced object points must be measured. Three basic internal scanning configurations (Figure 2.3) are distinguished in terrestrial laser scanning (TLS) for fast sequential acquisition of multiple points. All these configurations use mirrors to deflect the laser beam. The first mechanism employs two oscillating mirrors, which are mounted orthogonally enabling a horizontal and vertical deflection of the light. The oscillations correspond to partial rotations of the mirrors about two not intersecting axes. Since the mirrors oscillate out of phase with each other, the laser beam reflected by the first mirror never spots the second mirror on the same area. This mechanism is typical for scanners with a window-like field of view, for instance 40°x40°, such as the Cyrax 2500, one of the first terrestrial laser scanner later renamed HDS2500 by Leica.

The second configuration uses only one oscillating mirror which allows a deflection of the laser beam on the vertical plane. In addition, the light deflection on the horizontal plane is performed by a rotation of the scanner head. The direction of the laser beam intersects the mirror axis orthogonally, therefore allowing the emitted light to continuously spot the mirror on a limited area. This restricts the vertical field of view of all systems that implements multiple 3D measurements with a single oscillating mirror. An example is the laser scanner Trimble/Mensi GS100, which has a vertical field of view of 60° and a complete 360° horizontal field of view due to the scanner head rotation. In order to acquire the object surfaces that are not covered by the 60° field of view, the scanner should be properly angled.

The last configuration consists of a single mirror that rotates at high speed operating the beam deflection along the vertical direction, while the horizontal deflection is again accomplished by a mechanical rotation of the scanner head. The mirror is usually positioned at a 45° angle with respect to the incident light ray. The scanning mechanism might be summarized as follows. As the pulsed laser beam is generated by the sensor electronics, it travels in the same direction of the mirror axis until it strikes the mirror surface always in the same point. However, at the interface with the mirror, the angle of reflection of the light depends on the position of the mirror. Because of the continuous rotation of the mirror, the angle of reflection is subject to infinitesimal variations, thus progressively projecting the laser light on a full profile of the object surface. Once the desired amount of point distances have been acquired along one profile, the scanner head
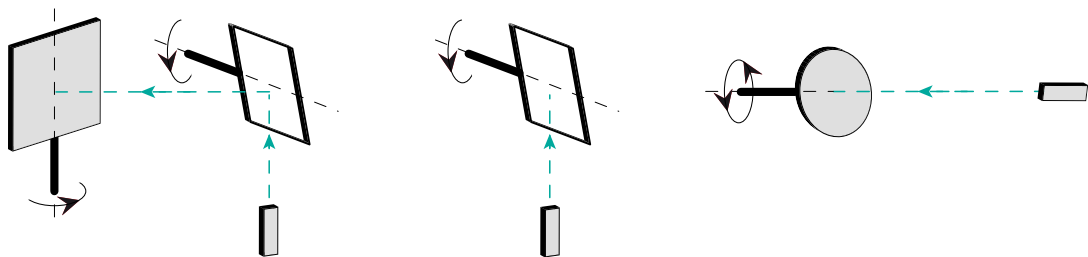

Figure 2.3: Scanning mechanism configurations: laser light deflection operated by two oscillating mirrors (left), one oscillating mirror (middle) and one rotating mirror (right).

automatically rotates around its vertical axis and the process is repeated. The configuration with a single rotating mirror is typical of scanners with a panoramic or hemispheric field of view. An example of scanner characterized by a panoramic, front and upper window design is the Leica HDS3000 (Figure 2.4a). The follow-up scanner from Leica, the ScanStation C10 uses a single mirror scanning mechanism too. Besides, it provides the innovative system X-Mirror which spins and oscillates allowing for a faster scan time [Leica, 2010].

(a) Leica HDS3000   *(b) FARO Focus³D*

Figure 2.4: Examples of two laser scanners based on the time delay measurement principle: (a) the pulsed scanner HDS3000 and (b) the CW system Focus³D.

2.2.2 Pulse detection methods

The major issue in the pulse runtime determination is the exact computation of the arrival time of the reflected signal at the receiver. In order to accomplish an accurate detection, the observation of very short incoming pulses with fast rise times is needed. However, due to the difficulty to obtain narrow width pulses, the correct calculation of the instant in which the pulse is detected may become challenging, thus compromising the accuracy of the TOF measurement. Pulse detection methods indicate all those criteria for extracting time-stamped trigger pulses from the backscattered waveforms [Wagner et al., 2004]. From the computation of the trigger pulses, the TOF of the reflected echoes is estimated, thus allowing to directly recover the distance sensor-target. Depending on the detection method used to define the trigger pulse, the resulting measurement accuracy might be subject to variations due to changes in the pulse position.

The easiest concept for pulse detection implementation is the determination of a threshold to set the minimum value of the echo rising edge. This means that a pulse is triggered when the rising edge of the reflected signal exceeds a given threshold. Unfortunately, this method is sensible to the amplitude and the width of the echo. Alternatives to threshold detection are maximum or inflection point detection. The former technique considers the first derivative of the echo waveform, while the latter computes the zero crossing of the second derivative. The activation of trigger pulses when maxima occur may become problematic if the echo contains more than one pulse. However, both maximum and inflection point detection are invariant to the echo amplitude variations.

2.2.3 Laser scanner used in the case studies

This section shortly introduces the instrument used for the experiments during the study on automatic modeling. The pulsed laser scanner that has been used for collecting 3D points of interior environments is the HDS3000 from Leica Geosystems. The acronym "HDS", which distinguishes many of the laser scanner systems of the swiss manufacturer, indicates a product family particularly suited for high-definition surveying. The HDS3000 is a medium range TLS system, originally thought for geology and civil engineering applications. It is characterized by a panoramic field of view resulting in 360°x270° (Figure 2.5) and an accuracy of 6mm at 50m range. The beam divergence at 50m is smaller than 6mm.

At the time when the HDS3000 was first released, in September 2003, it was a rather innovative tool, thus confirming a growth in the field of laser scanning technology. Since then, it has been used in a multitude of surveying and research projects, as attested by the related literature. A brief overview of the fields of operation of the HDS3000 is given below. Many projects involving the HDS3000 concern point acquisitions for deformation monitoring of diverse structures. For instance, an accuracy investigation of the HDS3000 has been carried out and tested in relation to a project aiming to inspect tunnel deformations [Lindenbergh et al., 2005]. The same type of deformation analysis has been also applied to a cooling tower [Ioannidis et al., 2006] and to a dam surface [Alba et al., 2006]. More common applications of the HDS3000 are connected to the preservation of cultural heritage. For this purpose, the data from the scanner are often used in combination with images in order to deliver realistic 3D models of architectural scenes [Rizzi et al., 2007]. An interesting contribution of the HDS3000 in archeology documentation is the survey of the ancient roman forum in Pompeii [Guidi et al., 2008]. A multi-resolution modeling methodology is used, thus requiring the HDS3000 to collect data of the entire forum for the middle resolution model, whereas both the low resolution model and the representation of detailed areas are achieved with different sensors. Also, in [Cavagnini et al., 2009] it is shown how the HDS3000 provides 3D data for applying a reverse engineering process in crime scene investigation.

The brief description of the Leica HDS3000 has been given here to provide a complete profile of the thesis research, including the instrument used to collect the data. However, the developed approach to 3D model reconstruction is completely independent on the type of device used to measure the point coordinates. In

Figure 2.5: Field of view of the laser scanner HDS3000 (left) and examples of point clouds acquired with it (right). The point clouds represent two views of Piazza del Duomo in Pisa (color scale) and the fortified gate of Bab al-Barqiyya in Cairo (from the CyArk [2011] archive).

Chapter 5, it will be shown how the modeling method is also able to perform good results in the case the input data is not provided by a laser scanner.

2.3 Real scene model reconstruction

3D models may be either created from existing scenes or from the synthesis of abstract contents supported by CAD programs and computer graphics. Among the techniques for 3D modeling from reality, image-based and range-based approaches can be distinguished. This classification is basically related to the type of sensor used to collect visual information concerning the observed object (see Figure 2.1). Image-based modeling refers to interactive reconstruction methods supported by a set of images captured by a camera, thus recovering the depth information by passive triangulation. In order to compute 3D point coordinates, 2D feature points on separate images are matched by solving the correspondence problem. The results obtained with image-based methods provide outputs that are photorealistic renderings of the scene. However, image-based techniques are affected by limitations due to the triangulation geometry: the system configuration imposes a trade-off between high resolution (large baseline) and minimal ambiguity in matching (small baseline). As pointed out by [Voltolini et al., 2006], the object coordinate accuracy is also an issue in relation to critical parameters, such as the radial lens distortion and the focal length. The reconstruction of architectural scenes is a typical context in which image-based methods are widely used. In [Debevec et al., 2006] a photogrammetric modeling for architecture is the first step of a more complex hybrid modeling strategy, which also includes a model-based algorithm to verify the correctness of the initial reconstruction. Further applications of image-based modeling are augmented reality [Van den Hengel, 2010], surface measurements of complex and detailed objects [Remondino and Zhang, 2006] and semi-automatic generation of 3D city models [Gruen, 2000].

On the other hand, range-based modeling approaches do not need to operate a transformation of the object points from 2D to 3D, since active sensors, which are used for this type of reconstruction, already provide range images as measurement results. Typically, range-based techniques rely on sensors that are expensive, not (easily) transportable and necessitate having basic knowledge about their physical mechanisms. Nevertheless, they allow to achieve high accurate 3D reconstructions of a great variety of object geometries. Range-based systems work efficiently outdoor and indoor, but interior environments are particularly well-suited to active measurements since their illumination condition can be controlled, thus reducing the interaction with the light source of the sensor. Because of the direct measurement of 3D point coordinates, range-based modeling is a relatively fast process. Only if the final model requires a textured appearance for realistic visualizations, additional acquisitions of the scene with a camera sensor are needed. In [Moussa et al., 2012], a flexible approach for the integration of laser measurements and digital images is proposed, which aims at filling gaps in the point clouds and in general at increasing the resolution of the laser data. Most of the sensors for range-based modeling are provided with an integrated color camera too, which makes the process of recording color information of image pixels simultaneous to the range measurements.

According to the specifications of each single project and to the tools available, the most appropriate modeling technique is chosen. Sometimes, a combination of different reconstruction methods can contribute to a final 3D model with improved characteristics [Remondino et al., 2009]. Especially in case of large scenes with very detailed objects, one modeling technique alone might not be sufficient for a reconstruction process that fulfills the initial requirements in terms of accuracy and photorealism, as well as efficiency and portability. Examples of reconstruction techniques that incorporate both image-based and range-based approaches are proposed by many authors, mostly in the field of cultural heritage. Beraldin et al. [2002]

combine photogrammetry (for the outer details) and (indoor) laser scanner to realize the photorealistic 3D model of a crypt. In their work, El-Hakim et al. [2004] use image-based modeling for basic shapes and structural elements, and laser scanning for fine details and sculpted surfaces. Also research in the area of computer science benefits from combined measurements, like the acquisitions of building interiors performed by Bahmutov et al. [2006] with a device that captures both color panoramas and depths in real time.

2.3.1 Reconstruction from range data

Range data consist in a special type of digital images whose pixels encode distance values instead of intensities, thus implicitly providing a full description of the surface shapes in the scene. By measuring distances of visible object points with respect to a reference position, the full configuration of the observed environment is reconstructed. The 3D representation resulting from a range data acquisition is often referred to as a range image or 2.5D image, since distances in 3D are mapped into a 2D image plane. Range images can be specified by a matrix whose elements – depth values – lie on a 2D grid mapped on the image plane. This makes the spatial relationships between objects in the scene easy to understand without any processing step. However, range measurements may have different representation forms. They may be simply stored as lists of 3D point coordinates without any organization related to the topology of the environment. Such unorganized sets of vertices define point clouds, the typical outputs of laser scanners.

Referring to range data as unorganized point clouds, the goal of the reconstruction is basically to interpret the information contained in the x-, y- and z-coordinates and rearrange it into a configuration that can be stored and visualized. Therefore, the reconstruction from point clouds is implemented by computing a 3D model that gives a visual representation of the topological organization of the recorded points. This is achieved through a numerical description of the whole scene. Current research about 3D reconstruction cannot be restricted to a unique framework because of the variety of scenarios concerning the types of targets, the desired degree of automation and the acquisition tools. In the following sections, the pipeline of generic 3D reconstruction approaches from unorganized point clouds is explained. Thus, outputs of laser scanning measurements are considered in order to reconstruct the geometric appearance of the scanned objects. The surface appearance may eventually be recovered by image texturing. It is assumed that no prior knowledge about the scanning scene is available: shape and topology information are not required for the reconstruction.

Point cloud acquisition

Most of the point clouds of the case studies in Chapter 5 are directly generated by a laser scanner, according to the operational principles described in the previous sections. The standard procedure for operating a correct scanning session will be discussed below. Other methods for generating point clouds from images involve an intermediate point matching step.

Laser scanning assumes theoretically that the point acquisition is independent on the light conditions and the viewpoint; in practice, the viewpoints should be accurately planned to avoid partial obstructions between adjacent measurements. This explains why the acquisition of the point cloud is typically preceded by a design phase to define the network geometry. The design phase establishes the configuration of multiple scan measurements, wherein for large scenes or complex objects several acquisitions from different viewpoints are needed. A multi-view configuration is necessary if the sensor has a limited field of view, the object cannot be entirely observed with a single scan or some parts of it are hidden by occlusions (caused by the object itself or by other objects in the scene). The scanning standpoints – also named stations – are previously

determined by the user with the aim of maximizing the overlap between separately illuminated areas. In addition, in order to be able to integrate point clouds from different scans in the following registration phase, homologous geometric features or artificial targets are used as transformation constraints. Thus, designing the network geometry – before starting the acquisitions – addresses also the choice of proper target locations, taking into consideration the scanning standpoint scheme as well.

Even though the scans are automated, a preliminary manual intervention is also necessary to set some imaging parameters. One of these parameters is the sampling density, directly connected with the scanning duration (the denser the point cloud, the longer the scan). For the acquisition of dense point clouds suitable for automatic generation of 3D models, an average sampling density of 15mm has been selected. That corresponds (specifically for the HDS3000) to a duration of about 40 minutes for each 360° scan if the scanner is positioned at an estimated distance of 5m from the target surface. At every adjustment of the scanning standpoint, imaging parameters might be set again.

The point acquisition is the only step of the reconstruction that is performed online. Registration and modeling can be easily completed offline once the data acquisition is finished and the point cloud has been saved in a proper data format. As a result of the scanning process, separate point clouds are available, each one containing a collection of 3D coordinates in a local reference system. The goal of the next step is to register all the clouds into a common coordinate system.

Registration

In TLS, a multi-view configuration is typically required for a full observation of the scene. The registration applies to a 3D rigid body transformation between multiple point clouds acquired from different viewpoints, thus locally referring to separate coordinate systems. It aims at computing a common coordinate frame for integrating the point clouds into a global domain that fully characterizes every surface in the scene, independently on the scans. According to Pfeifer and Boehm [2008], registration approaches may be mainly traced back to the domain of computer vision, classical surveying and photogrammetry. Consequently, the existing registration techniques are classified into data-driven, marker-based and sensor-based approaches. Data-driven and marker-based approaches have similarities in the way of establishing the global alignment of individual scans. Such similarities are based on the shared ability of computing the common reference system by overlapping artificial or geometric features captured from separate stations. The registration of two or more point clouds is usually supported by the fine acquisition of homologous tie points whose coordinates are used for the estimation of the six parameters of translation and rotation. Tie points are either natural geometric features of the scene (e.g., corner coordinates) or artificial targets specifically placed for the measurements. According to the type of tie points, feature-based or target-based registration are distinguished.

Feature-based registration methods might be defined as data-driven alignment techniques. They often implement the popular Iterative Closest Point (ICP) algorithm originally developed by Besl and McKay [1992], Chen and Medioni [1991] and Zhang [1994]. ICP works for the registration of misaligned data in different formats: besides point clouds, also line segment sets, triangle sets, implicit curves, parametric curves, implicit surfaces, parametric surfaces and meshes can be integrated. The basic idea of the algorithm is the use of least squares for estimating nearest neighborhood relationships between tie points of different scans. An initial transformation provides the parameters for a preliminary alignment: distances between homologous points are reduced. The final integration of the scans is achieved as a result of an interactive process that is considered completed when the distances between correspondent points are minimized. The convergence

of the algorithm to a global minimum is strictly connected to the initial approximation of the transformation parameters. Open source software – MeshLab – is available to automatically apply ICP to pairs of overlapping meshes [ISTI-CNR, 2010].

In its original form, ICP was restricted to the alignment of pairs of entirely overlapping data sets. Improvements and extensions of the method have been proposed since the basic algorithm was published. Currently, it results to be the most effective technique for multi-view point cloud registration. Unlike pairwise approaches, this kind of global alignment also allows for a distribution of the registration error across the several scans [Besl and McKay, 1992]. Apart from the critical dependency of the convergence of the final solution on the initialization parameters, ICP provides optimal results if some requirements on the intrinsic characteristics of the point clouds are fulfilled. Therefore, the acquisitions should guarantee areas of overlap between the data, a constant sampling density and lack of outliers. Additionally, enough geometric features are essential for the computation of the transformation parameters. In order to reduce the manual interaction in the initial approximation phase of ICP, techniques of coarse registration have been developed. These typically require the computation of surface curvatures (mainly for cultural heritage objects such as statues) or planar regions (in case of urban environments) to derive the initial pose parameters automatically.

On the other hand, target-based registration requires the employment of tie points that are placed in the scene on purpose and are measured at high density from two adjacent scans. The measurement of at least three targets from two viewpoints provides enough constraints to transform the point coordinates of one cloud to the reference system of the other one. Typical shapes for artificial targets are spheres or planar black and white patterns. These targets, differently from feature targets, are automatically detected and measured by the scanner. For a successful conclusion of the registration operations, it is important to arrange artificial tie points in a configuration that guarantees enough displacement constraints for the transformation. Also, due to redundancy (the measurement of a tie point might fail) it is generally recommended to plan the scans so that they share more than the minimum number of tie points required. This method is assumed to be more accurate than feature-based methods. Automatic algorithms to solve target-based registration are implemented in commercial software like Cyclone or SCENE.

Cyclone also implements a data-driven registration technique that is based on cloud constraints. Instead of looking for corner points in the scene or artificial targets, the entire area of overlap of two point clouds is considered and registered by ICP. Within the overlapping area, common points that are physically close, although acquired by separate scans, are reduced to a single point in the aligned coordinate system. This is done by least squares error minimization. As a result, the surfaces of the overlapping scans are globally registered without the need of placing target or searching for existing features. This solution might be a useful backup when the number of tie points is not sufficient to provide enough registration constraints.

The last type of approach proposed for point cloud alignment is sensor-based registration. The use of additional sensors to the LiDAR measurements is traditionally bounded to photogrammetry but it proved to be an efficient alignment technique for data acquired by mobile mapping systems as well. A successful sensor-based registration has also been proposed for static terrestrial laser scanning by Schumacher and Boehm [2005]. The idea behind it is that of computing the position and orientation of the laser scanner thanks to the support of a GPS (Global Positioning System) and an IMU (Inertial Measurement Unit).

Modeling

The term modeling indicates the process of generating closed surfaces from unorganized point clouds. Given a set of 3D points, their surface representation provides a geometrical and topological interpretation of the

scanned object. A surface model M_s is created by assuming that a certain number of points lies on – or is close to – an unknown surface S, thus allowing to approximate S with M_s. If the surface S is arbitrary, with unknown shape and topology, the reconstruction might become difficult. In the case of the reconstruction described in this thesis, it will be assumed that the surfaces to be reconstructed are planar. Another problem, which may occur while modeling a surface, is the absence of a pre-existing geometric organization of the 3D points. Also, data sets that are noisy and contain outliers increase the ambiguity of the reconstruction result. Thus, a sampling density that guarantees the allocation of enough details is a primary issue for a correct reconstruction in every scanning project.

In [Bernardini and Rushmeier, 2002], the process of modeling a surface is referred to as "scan integration" in order to underline the necessity of integrating scanned points into a unified surface representation. Bernardini and Rushmeier [2002] suggest how modeling techniques that deal with unorganized point clouds might not perform in a robust way. Although this is generally true, the method proposed in this thesis is robust – not affected by outliers – and well-suited to compute 3D models from unorganized point clouds.

Algorithms for reconstruction from range data can be classified in several ways. An overview of the different classification criteria is given in [Remondino, 2003]. Many of the reconstruction methods are based on the conversion of the scans into triangle meshes. The mesh generation produces a polygonal model with vertices, edges and faces that share common edges. The polygons of the mesh are typically triangles or quadrilaterals in 2D. Depending on the geometry of the input data, the creation of meshes can be extended to a 3D domain as well, thus partitioning the space into tetrahedra which meet at common faces. A well known technique to build a 2D mesh is the Delaunay triangulation.

Chapter 3

Building model representation

The goal of this chapter is to analyze existing building reconstruction techniques and explain how they are linked to the Manhattan-world modeling method. First, an overview of building geometry classification is given according to multiple building levels of detail. Then, well-established techniques for extracting geometric features are discussed. Finally, the concept of BIM is described with reference to a hypothetical integration with Manhattan-world models.

3.1 Classification of building models

In the following paragraph, a classification of building models is given in accordance with the CityGML standard.

3.1.1 CityGML standard

It is common sense within the geodetic community that the definition of architectural levels of detail has become a significant topic concerning the multi-scale representation of geometric and thematic building characteristics. Spatial properties and appearances of real urban environments can be currently classified through CityGML. CityGML is an extension of the international standard GML3 issued by the Open Geospatial Consortium (OGC) and the ISO TC211. It is an XML-based format that deals with the definition of common rules for the unambiguous representation of virtual cities and regional models including all their attributes and relationships. CityGML specifications were conceived and developed by Kolbe et al. [2005] to support, besides the representation, storage, exchange and maintenance of 3D city models (see [Kolbe, 2012]). In line with the variety of aspects covered by CityGML, building representation can be defined as multi-purpose as well as multi-scale.

Another characteristic of CityGML is its capacity of comprising in the same representation geometry, topology, semantics and appearance of a building. Geometry is intended as a descriptor of local shapes and profiles while topology involves the global correlation of separate geometric attributes, i.e. it indicates how geometric shapes are linked together. Further information about semantics of the 3D models can be used in the conversion process which generates computer graphics represented files such as COLLADA (collaborative design activity) or X3D [OGC, 2008]. Appearance refers to visual data and in particular to properties of the external building surface. These properties might also specify infrared radiation or structural stress. The workflow for the generation and use of 3D city models is summarized in [Kolbe et al., 2009].
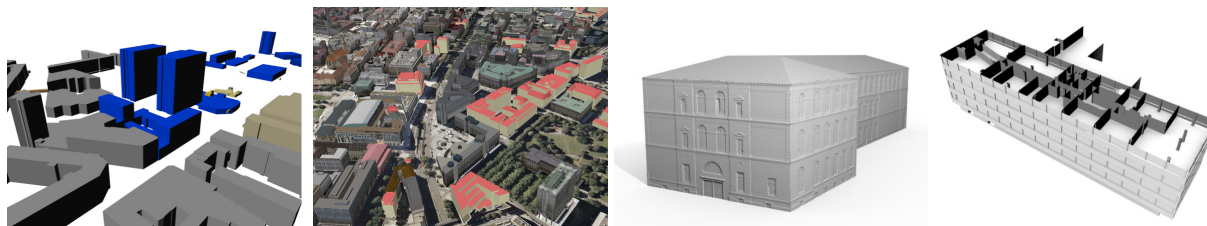
Figure 3.1: Representation of LODs from LOD1 to LOD4. From left to right: block model, site model with roof structures, detailed architectural model and interior model.

CityGML supports five different LODs that provide a hierarchical description of building entities. A higher classification level corresponds to a more detailed representation of building features. Figure 3.1 shows examples of LODs from LOD1 to LOD4. The first four levels of detail ranging from LOD0 to LOD3 describe external characteristics of buildings and natural features, either from close range or from aerial images. In particular, LOD0 is a two and a half dimensional digital terrain model (DTM). LOD1 is a block model which represents houses as prismatic volumes (without roofs). The next LOD2 adds roof structures to LOD1 and may include vegetation objects. LOD3 is a detailed model of external architectural elements such as outer walls, balconies, bays and projections. Accurate vegetation items and traffic components may be also exhibited at this level as well as high-resolution textures on the facades. The most elaborate representation is done at LOD4, which ensures the highest degree of detail for architectural models of interiors. The goal of the automatic reconstruction is to achieve an indoor model that is consistent with the degree of resolution typical for a model at LOD4.

The multi-scale building modeling described in the CityGML standard also allows for the simultaneous representation of the same object in different LODs. The aggregation of all object characteristics over different LODs can be computed by generalization. However, the scope of CityGML includes much more than just providing a simple or joint classification of objects among the LOD scale. CityGML models additionally contain other important features like taxonomies and aggregations of architectural sites (constructions in general), vegetation including water bodies, transportation facilities (roads and railways) and DTM.

In [OGC, 2008], the CityGML encoding standard is fully specified including motivations, main features and applications. The latter are identified, among others, in the areas of urban planning and management [Doellner et al., 2006], mobile telecommunication, disaster management, facility management, tourism, indoor navigation [Maes et al., 2006] and environmental simulations. Referring to disaster management, CityGML provides a framework for augmented reality with interoperable access, whose characteristics are helpful to monitor critical structures by enabling the visualization of different objects with accuracies up to 0.2m in position and height. If the model is accurate enough, it can be used during training simulations for the localization of safety-relevant components. In case of a severe destruction caused for example by an earthquake, the reference data would allow a faster rescue operation of the people involved or better performances in the reconstruction of the damaged sites afterwards. Especially for management of large urban areas, the development of city models also carries the issue of automation. This topic is discussed in [Ribarsky et al., 2002] where a semi-automatic tool is developed for the computation of 3D city models with the purpose of visualization and navigation.

Other formats for data interoperability

Beside CityGML, other standards exist for storing and sharing building models. Each standard is established on the basis of particular features that support a specific field of application. In geoinformatics, geodata is managed by geographic information system (GIS) software products which are suited for the representation and visualization of large areas including buildings as well as vegetation, road networks and topography. A software that is widely adopted for processing this type of geographic data is ArcGIS from ESRI. The resulting 2D and 3D geometry representations can be then stored in the "shapefile" format [Vosselman, 2010]. The amount of operations that ArcGIS and other GIS softwares can currently perform is huge but according to Goodchild [2011], GIS functionalities are still confusing and not provided with a rational organizational structure.

Enhanced modeling capabilities are available in CAD systems and computer graphics programs. Standard AutoCAD files are saved with the extension DWG (drawing), while special formats for CAD data exchange are DXF (data interchange format) and DWF (design web format, for sharing drawing information also over the web). A list of all AutoCAD file extensions can be found at the link in [Wiley, 2012]. Most common data formats for computer graphics applications are VRML97 or X3D. Especially in the area of architecture, the need is growing for having geometric representations with additional semantic information. In this context, BIM processes are becoming multifunctional tools which support efficient management and interoperability of building models. An approach for integrating BIM and GIS is proposed in [Van Berlo and De Laat, 2011] where the authors point out how the two methods might take reciprocal advantages in sharing the strengths of both standards.

3.2 Building extraction

In [Vosselman, 2010], building extraction is defined as the process of detecting and reconstructing architecture by determining building outlines and computing geometric 3D models. The detection of buildings takes place at different levels depending on the amount of details required for the model generation and thus in accordance with the acquisition system. The ordinary task of building detection as intended by Vosselman [2010] consists in the separation of buildings from other natural objects such as roads and vegetation, mainly using morphological operators. In its primary meaning the detection process is supposed to be referred to building outer shapes, whereas the detection of building interiors applies to the acquisition of detailed topological elements and fine indoor components. Accurate and efficient surveying methods that are currently used for building (acquisition and) detection at LOD0 up to LOD3 range from airborne laser scanning or helicopter-based systems to TLS and mobile mapping. For the detection of interior building features, TLS is widely available also in combination with range cameras or normal RGB cameras for texture generation.

The whole building extraction pipeline including detection and reconstruction is shown in Figure 3.2, where the contribution of BIM is also indicated. BIM represents the last processing step whose purpose is the management of the pre-computed 3D model. Overall, the figure illustrates how the steps are executed in sequence, starting from the building detection through reconstruction and eventually management. In general, BIM models allow for the management of an existing building from the point of view of materials, costs or other attributes relevant for facility operations. Additionally, virtual information intrinsic in BIM objects also favors a dynamic exchange of ideas among the development team members. The advantages of BIM will be described extensively in a dedicated paragraph.

In the intermediate step between building extraction and management, the 3D model is reconstructed on the basis of the raw data available (images, point clouds). The reconstruction aims at calculating a mathematical representation that comprises entire structures by binding together unorganized measurements. The most common reconstruction techniques are discussed in the following section.

3.2.1 Reconstruction techniques

Once the building profile is detected, building features are available in the form of raw data. In order to extract geometric information from such data and give the detected building the appropriate representation, techniques from geometric modeling are available. Originally, geometric modeling approaches were based on two-dimensional techniques for surface reconstruction. Surface modeling allowed to compute and store equations of object surfaces. The follow-up technique, solid modeling, extends the geometric reconstruction to three dimensions, thus enabling the more complex calculation of volumetric models. As defined by Mortenson [1997], solid modeling is a technique used to describe the shape of a concrete object by (abstract) analytical or mathematical rules. Still helpful for documentation and drafting are old 2D and 3D wireframe models which played a role in the early development of CAD programs. However, they can only provide ambiguous representation of solid objects. The model computed from the raw data, for example a dense point cloud, should have some specific characteristics. First, it should be a 3D representation suitable for visualization. Second, graphic performances as storage or transmission should be easy to be carried out. Third, also in view of the capabilities offered by BIM solutions, a 3D model should be appropriate to be stored in a database where each architectural object might be managed separately.

General classifications of solid modeling distinguish boundary-based from volume-based techniques and also object-based from spatially-based techniques. Shah and Maentylae [1995] define three main modeling categories on the basis of the point set representation. These categories separately include decomposition models, constructive models and boundary models. Decomposition models are obtained by joining together non-overlapping primitive object types. Cell decomposition and spacial occupancy enumeration belong to this category. The first computes the object as a union of adjacent cells of either regular or irregular shape. The cells are merged together according to given criteria allowing for the additional computation of topological features such as the existence of connected spaces and potential holes. Spacial occupancy enumeration is a particular case of cell decomposition in which all cells are regular cubic elements called voxels, arranged in a fixed grid. Each voxel can correspond to solid material or space where no data points
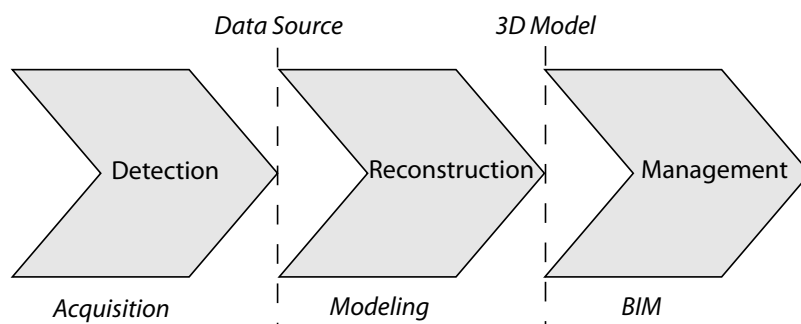

Figure 3.2: Process of building extraction in three steps: detection (acquisition) of architectural structures, reconstruction (modeling) of the building geometry and management of the 3D model through BIM tools.

occur. The voxels that denote material are represented by the coordinate of a single point, usually the cube's centroid.

Constructive models represent a point set by combining primitive sub-sets of points using Boolean operators (union, intersection, difference). This is the case of volume-based schemes as half-space modeling, constructive solid geometry (CSG) and sweeping. The latter represents objects in 3D in terms of volumes extruded by the sweep of 2D areas along a trajectory. Half-space techniques partition the space in two subsets recursively and calculate more complex objects by combining half-space primitives. Finally, boundary models deliver a hierarchical representation of a point set in terms of its boundary; for example, the 3D model of a volume is enclosed by planar faces whose boundaries are in turn one-dimensional curves. Among all geometric modeling approaches, the most widespread in CAD processes are CSG and boundary representation (B-rep). A combination of CSG and B-rep also leads to hybrid methods.

B-rep defines solid objects by calculating their "skin" which is the boundary between the model and the non-model [Stroud, 2006]. Once an object is computed in the form of a boundary representation, its geometry and topology are also available. The geometry of a reconstructed object describes its vertices, edges and faces by their positions in the recorded scene. That is, in a Cartesian reference system a vertex is localized by the coordinates (x, y, z), an edge by the coordinates of its end points and a face by its 2D boundary (given as a union of edges). Every geometric primitive has an analytical representation and is connected to other primitives by hierarchical relationships (an edge contains two end points and is part of a 2D boundary as well). Additional topology constraints indicate the way in which geometric primitives are reciprocally connected. This knowledge allows for the reconstruction of complex objects. Hence, it can be observed that B-rep is convenient because both geometry and interconnectivity information coexist in the same representation. A disadvantage is the inability of reconstructing spherical objects. A B-rep can be obtained automatically and unambiguously from CSG objects.

CSG approaches combine volume primitives using Boolean operations in order to find out the geometry of objects with higher complexity. Typical primitives are blocks, cylinders, spheres and cones with given dimensions. When combining two primitives, the resulting object varies upon the Boolean operation. The union produces an object whose volume is the sum of the primitive volumes. Intersection results in an object given by the overlapping volume. Boolean difference is defined in [Blackwell, 2012] as an order-dependent operator that produces the volume difference between the first given primitive and the common portion (intersection) to both primitives. The difference is useful to compute potential holes in the resulting shape. Topology is not an issue in CSG since it is assumed that primitives themselves are topologically correct and Boolean operations are supposed to preserve correctness [Brenner, 2004b]. However, the information about the object boundary is completely missing. On the other hand, among the advantages of CSG are the intuitive representation of 3D primitives and their parametrization, which allow reuse and collection in libraries [Vosselman, 2010].

In [Vosselman, 2010], the approach to building reconstruction is classified according to different criteria. Depending on the control in managing the data towards the model, reconstruction approaches are divided into bottom-up (data-driven) and top-down (model-driven). The first approaches build a complex object by exploring relationships between lower-level elements. Top-down approaches instead make assumptions about the global shape of an object and verify the initialization in a second step. Another classification criterium of reconstruction methods is the data source. As described in Chapter 2, either image sources or point clouds are available for building modeling. Further criteria are the type of model required (CSG or B-rep) and the degree of automation of the process. With reference to the above description, the Manhattan-

world reconstruction method developed in this thesis can be classified as a bottom-up, automatic approach which delivers a B-rep model from point clouds.

When reconstructing a building from measured data, an important issue is to ensure a correct topology. That is, all structures must be modeled correctly (with a good fit to the original data) and consistently (in an accurate reciprocal position). For example, if a plane is fitted to the corresponding data points with high accuracy but it is not correctly oriented with respect to another feature, the reconstruction will result in a wrong topology. To avoid this, the geometry of the plane should be estimated under the constraint that the fitted patch forms the correct angle with the reference feature. In Chapter 5, a case study will be described, in which the Manhattan-world assumption enforces the wall planes to be orthogonal to each other even though every single plane alone should actually have a slightly different orientation. Constraints that enforce a particular condition might be either logical (perpendicularity, parallelism, incidence) or metric (distance, angle, radius). In the specific case study, the Manhattan-world topology guarantees a relation of reciprocal perpendicularity between walls.

3.3 Beyond simple model reconstruction

This section examines the current development in the conception of modern 3D models. Nowadays, digital representations of buildings are not just the result of a pure geometric reconstruction but they might also integrate a detailed description of functional characteristics to enable facility management throughout the entire building life-cycle. Moreover, this section shows how the architectural process has evolved over time, from the introduction of digital architecture until the achievements leading up to the current status. Eventually, it will be explained why the automatic modeling can increase the efficiency of CAD-generated models towards the automatic extraction of BIM objects within a multi-functional context.

Since the power of computers has joined the world of architecture in the early eighties, the way of designing and developing architectural products has drastically turned into a more sophisticated process. In particular, the appearance of computer-aided architectural design (CAAD) software packages as commercial tools has given new perspectives to the art of designing, constructing and managing a building [Schmitt, 1999]. Engineering applications are currently responsible for the fast accomplishment of operations like space partitioning and simulations that were once carried out manually. In general, the benefits of a computerized approach to architectural methods are diverse. The most attractive is the possibility of creating virtual environments and digital models, which supports the design phase by allowing the end user to experience how buildings would look like or how they would suit in the surrounding external environment.

Each phase of the architectural design and construction process may take advantage of a modeling medium [Saleh Uddin, 1999] for different reasons. CAD systems are particularly useful during the design of a building, since they offer the developers the chance of visualizing building primitives and modifying them with low effort. Differently from traditional drafting documents, CAD projects can be easily changed without erasing and redrawing. Models generated by CAD can be also analyzed in detail thanks to integrated features that allow to magnify, move and rotate the model view. The outer profile (overall perspective) and the internal shape (cutaway drawings) of the building can be shown simultaneously within the same project. The introduction of CAD techniques has been an absolute turning point in the way of thinking through an architectural design project. However, even though CAD approaches are essential for the creation of detailed models, they are still limited by the lack of automation. Besides, they just provide static representations of the architectural scenes without any additional information about the functionality of the buildings. Conventional CAD models do not support the preliminary design in terms of development and demonstration

of design trade-offs. Instead, an interactive analysis of the trade-offs – that is a selection of initial parameter values – could help the designers in their decision making by comparing different design concepts. The interactive scenario offered by BIM, which makes available on-line simulations in every phase of the architectural process, results in better services and constitutes the main distinction between simple CAD models and parametric objects defined by BIM tools. In relation to the differences between BIM and CAD approaches, Professor Charles Eastman, co-author of [Eastman et al., 2008], observed that "BIM tools are as different from CADD tools, in the same way that a slide rule is different from a computer, or as a set of toy soldiers is different from a battle-oriented computer game" [Eastman, 2012].

This dissertation focuses on the automatic computation of a three dimensional model of a building interior from scans of an existing structure. This definition implies that the construction phase of the interactive architectural process has already been completed. Therefore, the approach developed is considered as a tool to digitize real structures of building interiors and store building characteristics for later spatial analysis. BIM processes could theoretically take advantage of the automation modeling method to further improve the efficiency in the pipeline of an architectural project.

3.3.1 Building Information Modeling

BIM can be defined as a process that aims at creating a collaborative environment among its major participants using structured information and software interoperability. The parties involved in a BIM project are professionals (architects and designers, engineers, consultants, contractors) and non-professionals (clients). BIM implements digital building models like CAD does but, while CAD representations are totally missing the aspect of shared-information, with BIM the collaboration across the team is essential.

BIM became recently a huge topic in the field of architecture, engineering and construction (AEC) due to its multiple functionalities that aggregate tools for both building construction and facility management (FM, see Figure 3.3). The origin of BIM is dated back in the mid 1980s, when the company Graphisoft brought the software ArchiCAD to the market [Morse, 2012]. ArchiCAD was the first BIM-like resource specifically implemented for providing "virtual building" solutions in the AEC industry. Shortly afterwards, the term "building information modeling" started to be used and in the early 2000s it became the common term to indicate information that is structured and shared by a model [Nisbet and Dinesen, 2010]. With the coming of BIM, another important feature is added to digital building design as intended by CAD. That is the possibility of extending the geometry of standard 3D CAD models beyond the third dimension. While CAD drawings can only rely on three primary dimensions, BIM models also support time as the fourth dimension and cost as the fifth. Time-based representations provided by 4D BIM software have great significance in a dynamic construction process; by applying the visualization of the project development along its timeline, all project partners will be able any time to participate actively in making modification as well as solving problems or simply appreciating the advances in design and construction.

Every step of the architectural project benefits from modern BIM tools. At a general level, the great amount of documents that had to be printed and stored in the past is eliminated by the digitization of the processes and by the comprehensive data stored within BIM objects, which include geometry, spatial relations and attributes. The concept of object recurring in BIM environments follows the terminology used in computer science to indicate an abstract item which collects properties, attributes and relationships. As BIM objects are defined parametrically, they easily support changes and can be dynamically connected to each other. That means, if an object attribute is physically modified in the design phase, all views of that object will be automatically updated consistently. The basic idea is to initialize instances of parametric objects and to
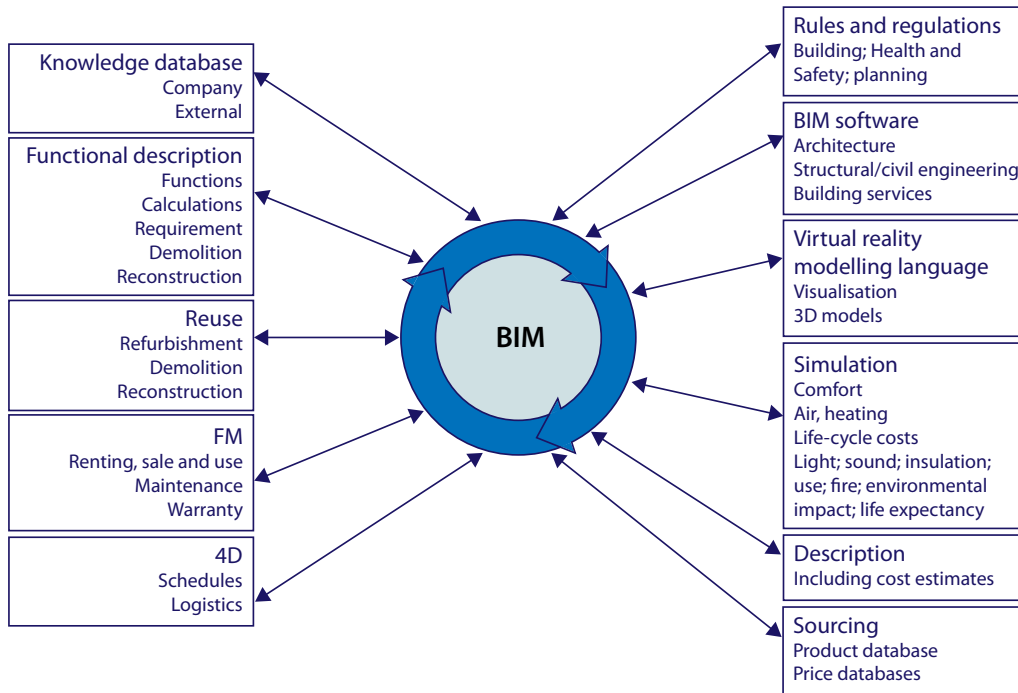
Knowledge database
Company
External

Functional description
Functions
Calculations
Requirement
Demolition
Reconstruction

Reuse
Refurbishment
Demolition
Reconstruction

FM
Renting, sale and use
Maintenance
Warranty

4D
Schedules
Logistics

BIM

Rules and regulations
Building; Health and
Safety; planning

BIM software
Architecture
Structural/civil engineering
Building services

Virtual reality
modelling language
Visualisation
3D models

Simulation
Comfort
Air, heating
Life-cycle costs
Light; sound; insulation;
use; fire; environmental
impact; life expectancy

Description
Including cost estimates

Sourcing
Product database
Price databases

Figure 3.3: Representation of BIM and all its functionalities from [Nisbet and Dinesen, 2010]. All listed functions bring shared information among the project partners.

input any change directly to the instances in order to modify all different object views at once. This leads to a drawing consistency that eliminates many errors [Eastman, 2012] and reduces the costs consistently. Also, to describe objects by parameters allows to convey modifications between related objects. Object relationships are expressed either as abstract rules (for instance geometry constraints) or physical properties (material buildings). According to Nisbet and Dinesen [2010], the concrete benefits of BIM include savings in the pre-design and design coordination as well as in the negotiation phase and finally the construction. After construction, the building might be managed through a fully functional model as it is explained in the following paragraph.

3.3.2 Model management

In the above section, it has been described how BIM adds functionalities to traditional 3D models. BIM actually creates augmented models that can be shared, analyzed and managed. In order to share 3D models among different software users without loss of accuracy, an ISO standard known as Industry Foundation Classes (IFC) defines AEC objects accordingly. Another standard for data interoperability and exchange is Construction Operation Building Information Exchange (COBIE).

Sabol [2008] describes BIM as a sophisticated technology that simultaneously provides an accurate building model and a database containing critical information for facility management. The database collects all building components and services (elements inside the building) as well as the building environment (outside). If the Manhattan-world reconstruction method would be hypothetically used for supporting BIM management, the information calculated automatically within the interior model could be stored among the building components.

3.3.3 Manhattan-world reconstruction and BIM

One of the fields of application of the method developed in this thesis might just concern the analysis and the documentation of a built construction, thus making the modeling approach strictly connected to the architectural process. The total scope of a standard architectural project includes a sequence of activities among which are the architectural program, the design phase, the working drawings (CAD and BIM detailed drawings), the negotiation phase and the realization. Besides, the maintenance of the construction over time is also supported by BIM. The automatic modeling would find a proper placing among the maintenance activities too. The software developed might be used as a tool to inspect a building interior in terms of forms and proportions, which is currently done manually by architects, or it might be also the indoor setting for multimedia implementations. In the future, virtual experiences and multimedia adventures will be the usual environments for everyday life activities that are now purely driven by human action and interaction. About its digital vision of the incoming world, Bill Gates declared: "You'll be walking in downtown London and see the shops, the stores... Walk in a shop and navigate the merchandise... in a virtual reality walkthrough" [Gibson, 2012].

In light of what was pointed out by Gates, current digital globes such as Google Earth and Microsoft Bing Maps can be extended to contain building interiors in addition to the already available external shapes. In order to support the proliferation of indoor models, a high degree of automation, which ensures the cost-efficiency optimization in the reconstruction process, should be guaranteed. Additionally, such digital scenes might be supported by BIM tools offering a wide range of management functions. Through BIM models, for instance, heating, lighting or space comfort might be controlled, safety information might be checked and maintenance operations might become easier and more efficient. Models enhanced by BIM features are meant to be "intelligent" due to the relationships and information they carry. In [Autodesk, 2011] it is described how BIM components interact with one another. An example is given of a model of a room – the basic portion of space considered in the automatic reconstruction – that is not just an abstract concept but a space contained by other building components such as walls, floors and ceiling. [Autodesk, 2011] states that a "room" is a database element that contains both geometric information and non-graphic data.

With reference to Figure 3.2, an analogy can be done between the first two steps of the building extraction pipeline and the reconstruction from range data described in Chapter 2. The latter reports how to compute a 3D model from laser data specifically for the case studies developed in this thesis. There is a close connection between 3D information collected by laser scanning during the building detection phase and BIM development. Laser scanner point clouds are the inputs for generating the models that might be later stored as BIM objects. Thus, they constitute the necessary condition to initialize the process illustrated in Figure 3.2. The relation between 3D laser scanning and BIM is also investigated by the U.S. General Services Administration (GSA), whose Office of the Chief Architect is actually carrying on projects in this field. In particular, the GSA's National 3D-4D-BIM Program [Matta and Kam, 2012] deals with the adoption of BIM technologies to manage information collected by 3D laser scanning and imaging systems.

In conclusion, a dynamic reconstruction through the automatic computation of 3D models together with an upgrade from CAD to BIM models could provide an optimal solution to building refurbishment and management. It should be finally observed that in recent times building development has globally slowed down, thus improving the need for using existing architecture. This of course encourages old building restoring and preservation, which might be regulated by BIM functions such as the "functional description" and the "FM" listed in Figure 3.3.

Chapter 4

Automatic modeling of indoor scenes

In this chapter the concept developed for the automatic reconstruction of interior scenes is described. The derived method focuses on the reduction of human intervention in the modeling process, thus aiming at a full automation of the modeling phase with the consequent optimization of the efficiency of the entire laser scanning project workflow. The reconstruction complexity is typical of Manhattan-world indoor scenes.

4.1 Existing approaches

The main goal of this section is to give an overview of the existing approaches concerning different methods for indoor reconstruction, that means for the reconstruction of building interiors at LOD4 according to [Kolbe et al., 2005]. By the mutual comparison of these approaches and by the analysis of their characteristics (and objectives) compared to the automatic modeling of indoor scene, the originality of the latter is pointed out.

More in general the approaches which have influenced the work for this dissertation are all those for the automatic and semi-automatic generation of 3D models independently from the type of target scene (interior or exterior) and from the type of input data (point clouds or images). In all these different scenarios, the main motivation for achieving 3D models is the need for a mathematical description that stores geometrical features in a digital format for better record-keeping and remote visualization. For the above reasons, current research continuously points on the development of new methods for 3D model generation. The primary contribution of the thesis is the development of an algorithm that reconstructs and visualizes indoor scenes from point clouds automatically.

In order to increase the level of automation of the reconstruction process, significant advances are observable in the current state of the art as well. Relevant work towards automatic building modeling from laser point clouds is related to feature extraction from terrestrial and aerial data. Aerial LiDAR point clouds are used by Verma et al. [2006] for the automatic recognition and estimation of building roof shapes. On the other hand, in [Pu and Vosselman, 2006] terrestrial laser scanning provides high density point clouds suitable for region segmentation to support facade extraction at closer range. In this approach, the constraints for segmentation are given by human knowledge about building features. Additionally, a window extraction tool is also developed by Pu and Vosselman [2007], whereas highly detailed polyhedron models of buildings are reconstructed automatically in [Pu and Vosselman, 2009]. This method, as the previous work of the same authors, employs a semantic analysis of the geometric features to extract walls, windows and doors. An overview of the current methods for recognizing structures in point cloud can be read in [Vosselman et al., 2004] with a special focus on the extraction of smooth and planar surfaces.

Beside the point of automation, the modeling method described in this thesis also carries the issue of computing 3D models from range data of interiors. The difficulties in modeling indoor scenes are not the same as those ones typical of building exterior reconstruction. This is the reason why model reconstruction and visualization of generic indoor scenarios is still a difficult task [Furukawa et al., 2009b]. The existing approaches that have been specifically developed for interior modeling establish the real interesting background for the modeling concept developed in this work. A distinction is done between approaches to interior reconstruction from images and from point cloud. According to Furukawa et al. [2009b], three main factors complicate the reconstruction of interiors from photographs with respect to building exteriors. The first is the absence of textures on the observed surfaces since walls and furniture present in most cases plain surfaces. Second, the visibility conditions are definitely worst due to the limited room available. Third, the scalability of the scene is challenging if objects of different sizes are present in the scene. Among these limitations, the reduced visibility conditions are an issue shared with laser scanning acquisitions.

Indoor reconstruction methods that use images as input data are particularly widespread in the computer vision community. In [Furukawa et al., 2009a] and [Furukawa et al., 2009b] the authors address the special problem of reconstructing building interiors with a multi-view stereo approach which is also designed for Manhattan-world environments. Attempts to indoor reconstruction from single images have been carried out by Olivieri et al. [1992] and Delage et al. [2006]. Coughlan and Yuille [1999] collect monocular views to design a computer vision system that calculates the orientation of a blind user with respect to the scene. This is achieved by an estimation of the most important lines in the scene – corridor boundaries and doors – assuming that they lie along the direction of a Manhattan-world grid.

The task of interior reconstruction has been already associated with range data acquisitions too. In particular, laser scanners and range finders are used to assist the navigation of autonomous robotic systems respectively in [Biber et al., 2005] and [Haehnel et al., 2002]. On the other hand, solutions to military applications are shown in [Johnston and Zakhor, 2008]. This approach exploits a technique similar to the method explained here although it collects point clouds of the interior of the building using a laser range measurement system situated outside the building. The main constraint for the success of the acquisitions is to have unobstructed and large windows that allow the laser light to hit the interior surfaces. By acquiring the point clouds from all sides of the building, a sufficient number of points is available from all building floors. Point measurements of floor and ceiling are missing.

The full reconstruction of indoor environments implies to model the surfaces that separately constitute the volume boundary. Accordingly, the central research topic in [Adan and Huber, 2011] is the modeling of surfaces from 3D laser data of complex scenes with occlusions. The paper mainly focuses on the reconstruction of occluded wall surfaces but the technique described can be also applied to floors and ceilings. Nuechter et al. [2003] present a method for the semantic interpretation of interior building features based on generic architectural knowledge. First, the data acquired by a mobile robot equipped with a laser range finder is segmented in order to extract 3D surfaces. In a second step a map of the interior is created, which contains semantic descriptions and connections between features (walls, floor, ceiling and doors). Among the techniques for surface recognition, the work of Rusu et al. [2008] investigates the problem of building 3D object maps of indoor household environments from point clouds. Once the relevant surfaces are segmented, the reconstruction part is carried out by a model fitting technique that replace the regions identified with polygons.

As highlighted in Chapter 3, the automatic reconstruction of interiors is an interesting topic among the architecture community, too. The convenience of indoor BIM models, which are easy to compute, might simplify the architectural design process by supporting a dynamic interaction between existing spaces and

new structures. A challenging study about how to generate 3D models from architectural drawings is carried out in [Yin et al., 2009]. The inputs of the method described in [Yin et al., 2009] are paper ground plans (images) that need to be converted into CAD files so as to automatically generate the model.

The general overview given about the state of the art on 3D reconstruction shows how a concept is missing that aims at the pure automatic modeling of interiors of buildings. Automatic modeling methods already exist for reconstructing outer building profiles. However, no approaches are known for modeling laser scanner point clouds into digital representations with a high level of details. The method developed in this dissertation computes the topological layout of Manhattan-world interiors from unorganized point clouds.

4.2 Modeling interiors from laser range data

As already mentioned in Chapter 2, the process of reconstructing real scenes from range data is achieved as the final result of the laser-scanning workflow. A practical evaluation of the degree of automation in each processing phase shows how the first two steps – acquisition and registration – are already sufficiently automated, whereas the modeling part still needs higher automation. By developing a 3D modeling technique that is fully automated in all its steps, the entire reconstruction process becomes faster, thus more efficient. That is due to the typical effort distribution of a laser-scanning project, which indicates the modeling phase as the most demanding task in terms of time and manual support. Therefore, a speed improvement of the modeling operations would result in a global refinement of the complete reconstruction pipeline. Also, if a high degree of automation is maintained throughout the whole processing pipeline (from the acquisition of the points until the modeling phase), cost-efficiency of the model generation is improved too.

The goal of this dissertation is to perform the automation of the modeling phase in order to achieve a 3D reconstruction of buildings at LOD4 according to the multi-scale representation CityGML (see Chapter 3). This suggests that the complexity of the obtained 3D model is compatible with the highest level of details required by the standard developed by Kolbe et al. [2005]. In particular, the main subject of the thesis is the reconstruction of a model which incorporates interior architectural details. That is, structural information – represented by walls – as well as local architectural components – such as doors or windows – will be automatically modeled. The domain that defines the field of application is given by the Manhattan-world assumption specified in the next section.

The model generation is supported by a robust method which takes the coordinates of 3D points acquired by a laser scanner as inputs. Prior information about the topology of the interior is not necessary, only basic assumptions are made. This makes the algorithm suitable in many situations where the original design model is no longer available. Few requirements on the type of input scene to be reconstructed define the problem domain.

4.2.1 Problem domain definition

The problem of determining an indoor model may apply to very general environments with distinct architectural features. Ranging from cultural heritage sites to office workspaces and industrial interiors, the whole geometry of the scene under analysis and its architectural elements vary in shape and topology. The interior of a Romanesque church for instance exhibits typical characteristics such as columns, arches and vaults whose special features cannot be found in modern construction styles. That means, the (automatic) indoor reconstruction of historical buildings from laser data must face very specific issues that do not necessary pertain to other structural contexts. Very complex and detailed interior scenes of archeological interest are

mainly modeled by generating a polygonal mesh from TLS data and manually editing the result in a separate step [Campana et al., 2009]. In these cases the automation of the reconstruction may become highly demanding because of the absence of repetitive or regular patterns.

Due to the wide diversity of the interior appearances, the domain of application of the automatic modeling of indoor scenes is restricted. This allows to develop a robust method that delivers optimal results in response to one specific type of architecture. The reason for this choice lies in the automation of the developed algorithm. An automatic tool in fact is supposed to work independently of external control, thus it requires to be specified by rules that uniquely define the algorithm within an operational domain. The rules make hypotheses on the architectural characteristics of the building interior to be modeled. In particular, the rules in this work set the following requirements:

Manhattan-world grid structure

The main rule which distinguishes the type of building interiors suitable for automatic modeling focuses on the overall distribution of the spaces enclosed by the external perimeter of the construction and by its interior walls.

Conventional indoor spaces exhibit a topology that is determined by the wall layout. With reference to a Cartesian system, all walls in the interior are defined as a set of vertical structures expressed by $V = \{w : w \parallel \mathbf{z}\}$, where $\mathbf{z}$ indicates the Z axis. The mutual configuration of the structures contained in V fixes the dominant directions of the scene. In general, two random walls of a generic pair $W_2 = \{w_a, w_b\}$ with $W_2 \subset V$ might have several reciprocal orientations. If the analysis of the indoor topology is referred to the Manhattan-world domain, w_a and w_b can be exclusively positioned in two ways. They can be either orthogonal (the angle between them is $90°$) or parallel. In the latter case, they might have the same normal vector (they form an angle of zero degree) or opposite normal vectors (they form an angle of $180°$). In view of these considerations, the vertical structures present in the Manhattan-world interior are uniquely aligned with two dominant directions orthogonal to each other (see Figure 4.1). In the rest of the chapter the dominant directions are referred to by $\mathbf{n}_\beta$ and $\mathbf{n}_{\beta+90°}$, namely by their normals.

By the observation of the horizontal section of the interior, it is noticeable that the wall projections on the floor show a 2D orthogonal grid-like structure, thus the spaces within the walls are rectangular. It is possible to conclude that the positions of the walls are fixed in place by the Manhattan-world grid layout. As pointed out in [Coughlan and Yuille, 1999], most indoor and outdoor city scenes are based on such a Manhattan-world grid. The term "Manhattan-world" recalls explicitly the typical urban layout of orthogonal streets and avenues of New York City centre.

Horizontal floor and ceiling

Another important structural characteristic that contributes to the definition of the operational domain is the orientation of the floor and the ceiling with respect to the vertical walls. According to the Manhattan-world design specified above, the set that describes the walls can be expressed as $V_m = \{w : w \parallel \mathbf{z}$ and $w \parallel \mathbf{n}_\beta$ or $w \perp \mathbf{n}_\beta\}$, where $\mathbf{n}_\beta$ refers to one of the two dominant directions. In addition, if the floor and the ceiling are both horizontal with respect to the Cartesian reference system, it yields $f, c \in H = \{h : h \perp \mathbf{z}\}$. Therefore, floor and ceiling are both orthogonal to the walls. That might sound obvious, but some examples of architectural creations exhibit shapes that do not necessary ensure the horizontal alignment of floor and ceiling. Such examples, which claim a particular originality of the shapes, are not interiors intended to be
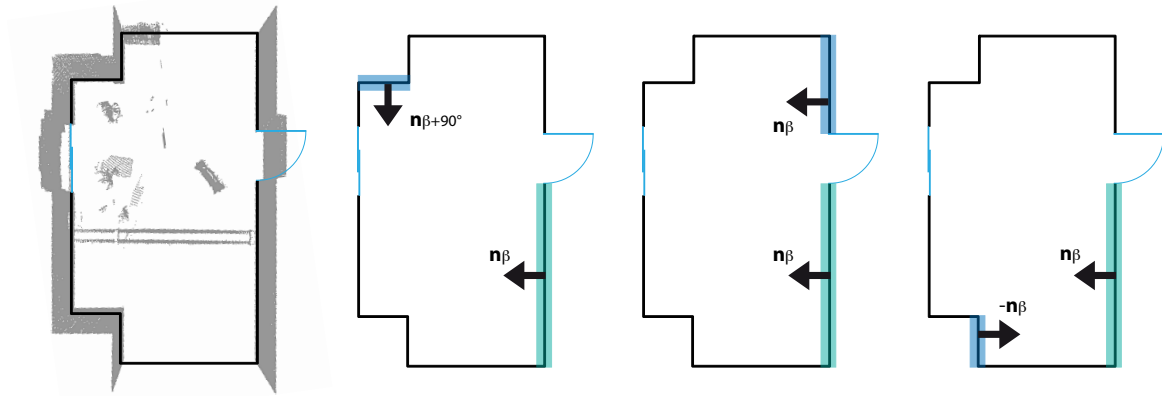
Figure 4.1: Orientation of the walls in the Manhattan-world domain. From left to right: top view of a point cloud with the related floor plan showing orthogonal walls, walls with the same orientation and walls with opposite orientations.

reconstructed with the method developed in this thesis. As there are no more requirements on the horizontal structures, the set H fully defines floor and ceiling of the Manhattan-world interior and can be named H_m accordingly.

The specifications about the orientation of floor and ceiling in combination with the 2D Manhattan-world grid rule highlight how architectural interiors often present highly structured planar modules. With regard to the principal modules of an indoor scene – floor, ceiling and walls – the following considerations are made. The floor and the ceiling are parallel to the XY plane, thus they can be described by planar surfaces, each one characterized by an unknown position along $\mathbf{z}$. Additionally, the walls are aligned with $\mathbf{z}$ but they have unknown orientations with respect to the planes XZ and YZ (and also unknown positions along the normals to their actual planes).

In conclusion, in the 3D Manhattan-world environment the normal vectors to the planar modules are always aligned with three dominant directions: one of these coincides with the Z axis whereas the other two are set by the X and Y axes up to an unknown angular factor (see Figure 4.2).
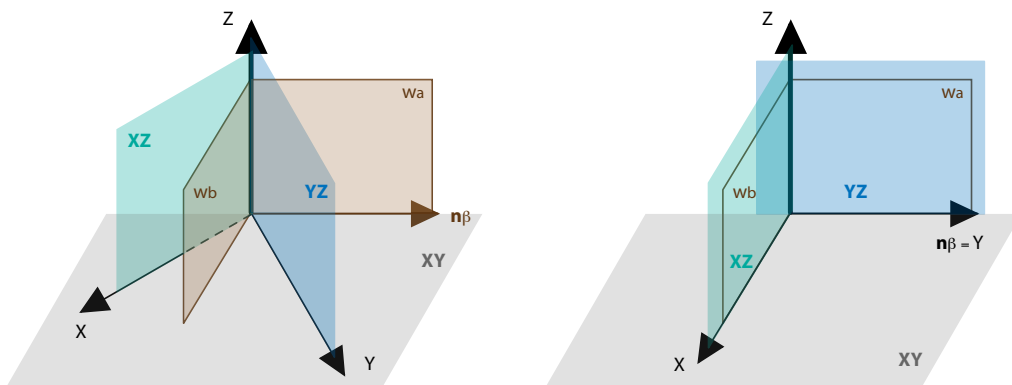
Figure 4.2: Two orthogonal walls w_a and w_b are always aligned with Z whereas their orientation with respect to the planes XZ and YZ is not known (left). A particular case occurs when all Manhattan-world structures coincide with the Cartesian planes (right).

Multiple doors

This rule together with the next one is not a necessary prerequisite to the automatic modeling but it represents a hypothesis about the efficiency of the automatic modeling in relation to the complexity of the input data. It is assumed that the reconstruction tool is also capable of recovering multiple doors which are expected to keep to standard measures. Typical sizes of interior doors correspond on average to two meters height by 80 centimeters width. In order to guarantee a successful detection, the doors are supposed to lie on a surface that is parallel to one wall surface but does not coincide with it. Open doors can be detected as well.

In [Rabausch and Volz-Graetz, 2010], an extensive functional classification of door types is reported. First, external doors are distinguished from doors that separate interior spaces. Second, such interior doors are described in detail and a deeper classification is provided. This includes, besides most common side hinged doors, sliding doors, folding doors and emergency doors as well. However, the way a door is mounted or the purpose for which it has been designed has no influence on the detection, that is every type of door can be localized while modeling the scene. This is achieved by identifying gaps in the wall measurements, and specifically by properly selecting missing-point areas with a minimum width. In most cases, the door surface does not exactly coincide with the wall plane and hence an area with no points will characterize the wall in correspondence to the door. If the gap width is set accordingly on the door size, the door is correctly localized.

Due to the algorithm strategy, the same method for door detection potentially allows for the localization of multiple windows too or any other architectural element that modifies the wall profile.

Presence of furniture

Pieces of furniture or objects with an irregular geometry are not recovered during the process of automatic indoor modeling. Nevertheless, if they belong to the interior, they do not need to be removed before the point acquisition (or later on with post-processing operations) since the robust reconstruction method classifies them as outliers, thus it discards them automatically from the final model. A method for the automatic reconstruction of indoor objects that are not part of the basic architecture may be the topic of a further research. The implementation of a furniture library might be the fastest solution to the enhancement of the rough structural model.

4.2.2 3D reconstruction in the architectural process

In consideration of the potential of the automatic modeling method concerning the support to BIM solutions (see Chapter 3), the problem domain will be set from the point of view of an architectural process too (for a better overview, read Budroni and Boehm [2010]).

If a new phase could be added to the activity list of a standard architectural project based on the overview in Chapter 3, the method presented in this thesis would find a proper placing at the top of the list, thus it would represent an additional step to be added at the end of the project. Indeed, the Manhattan-world modeling method automatically delivers a 3D model of an existing structure. Hence, the new phase could be labelled "re-construction" and could be executed among maintenance operations to analyze buildings by recreating their shapes in a digital format.

At first sight, the function of digital models generated from architectural spaces (once these are already built) may appear unnecessary since 3D models are also achieved in the previous design stage to support the

physical production. Different reasons can explain this apparent contradiction [Budroni and Boehm, 2010]. First of all, the model accuracy typical of the design phase is not ensured throughout the entire construction development. Therefore, a model created at the top of the process by reproducing real elements might be used as a reference for testing the validity of the initial design: a sort of comparison between synthetic and actual model. Secondly, most buildings must deal with structural changes during their lifetime. That means, existing architecture is object of continuous modifications for reasons such as reorganization of spaces or more generally refurbishment. Unfortunately, these changes are not documented by the original models which are static and only contain the initial design drawings. Also, prototypes in CAD format or drawings created at a very early stage of the architectural process might be entirely lost, thus it would not be possible to consult them for reconstruction purposes. A dynamic reconstruction through the automatic computation of 3D models may provide a solution to these issues.

The above definition of the rules which set the necessary characteristics for an interior to be automatically reconstructed arises from the urgency of a customized standard for indoor scenes. Since the rules apply to existing buildings, they refer to the state of construction at the top of the architectural process ("re-construction" phase). In order to formalize the modeling domain, the contribution of the well-established system of rules developed by Alexander et al. [1977] was essential. Alexander et al. [1977] fix 253 patterns organized in three categories: towns, buildings and construction. His language sets the guidelines for a correct design process, from the environmental studies (architectural program) to the practical arrangement of the structural layout (construction). As a result, Alexander's language embraces the architectural process in all its phases and provides a full description of all patterns in order "to create beautiful, functional, meaningful places".

On the basis of the reconstruction rules that are defined by the problem domain (and will be formalized below) the point cloud is interpreted using simple concepts of descriptive geometry. Hence, a robust algorithm is implemented that processes the interior data and generates a 3D structural model encompassing details such as doors.

4.2.3 Automatic modeling algorithm overview

The automatic modeling algorithm mainly consists of two subsequent phases. After their execution the topology of the interior is recovered by a robust determination of positions, orientations and sizes of all planar surfaces enclosing the indoor volume. A third phase is required to integrate the previous results and visualize the 3D model as a B-rep. Minor issues are treated in parallel to improve the results in terms of intermediate output verification (histogram visualization) and final model enhancement (door extraction). A diagram of the algorithm pipeline with all its phases is represented in Figure 4.3.
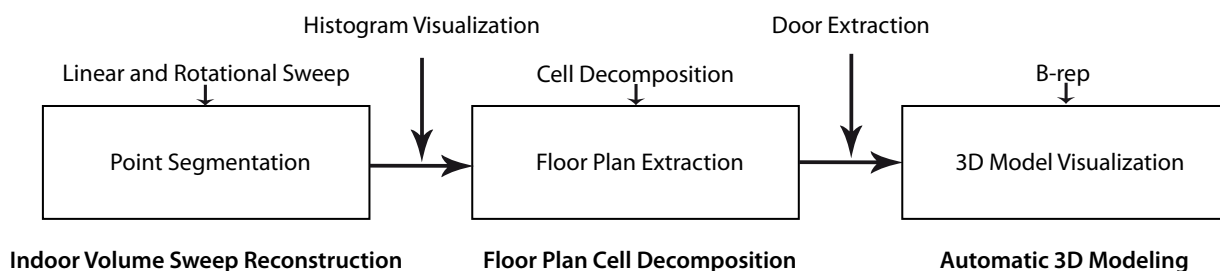
Figure 4.3: Main phases of the automatic modeling algorithm.

The first phase toward the achievement of the 3D model is the reconstruction of the indoor volume through the localization and the identification of its basic geometric primitives – rectangular shapes – from the unorganized point cloud. A sweep-based segmentation controlled by a hypothesis-and-test paradigm allows for assigning each 3D point to a well-defined primitive (floor, ceiling or one of the walls). The term hypothesis-and-test indicates a model-based approach structured in two parts: one that generates a sequence of hypotheses on the positions of the primitives and the other that verifies such positions based on the sweeping object model.

In the second phase the extraction of the floor plan is supported by a cell decomposition approach. The floor surface is partitioned into rectangular cells according to half-space modeling techniques. By detecting and merging only those cells that actually lie within the scene contours, the horizontal section cut through the building interior is well determined and the 3D model can be derived from it.

All phases of the algorithm together with the additional step for the detection of the doors are described in detail in the following paragraphs. Throughout the algorithm pipeline a point cloud is considered that refers to a Cartesian system whose Z axis is aligned to the local vector of gravity. This is feasible because the instrument is supposed to be leveled, thus its vertical axis is orthogonal to the horizontal plane. Nevertheless, the implementation of the 3D reconstruction method as well as the results of the algorithm is independent of the coordinate system.

4.3 Indoor volume sweep reconstruction

The sweep reconstruction of Manhattan-world indoor volumes is easily referable to the concept of sweep representation in computer graphics. It is in this context that a translational sweep is defined by sweeping a 2D area along a linear path normal to the plane of the area [Foley et al., 1996]. The sweep computation generates a volume whose measure is given by the sweeping area times the length of the sweep. Depending on the contour of the 2D geometric form that characterizes the sweeping area, the resulting volume assumes different shapes. The sweeping area's geometric form that optimally satisfies the criteria set by the modeling rules for Manhattan-world interiors is a union of adjacent rectangles. In accordance with the orthogonal grid layout and the horizontal alignment of floor and ceiling, the indoor volume is expected to be a solid enclosed by multiple faces which form angles of either zero or 90 degrees. Prior knowledge about the extent of the sweeping area is not required to define the indoor volume measure: the boundary (thus the shape) of the volume will be automatic determined in the following computational phase. Also, the length of the sweep can be directly derived from the point cloud by defining the maximum range between point coordinates along one dominant direction.

The algorithm for automatic modeling uses the idea of the sweep as the basic technique for 3D point segmentation. Its implementation recalls the scan-line approach for forming the intersection of geometric objects developed by Shamos and Hoey [1976]. In particular, a plane is swept across the space and is stopped at discrete positions. The stop criteria consist in predetermined (constant) distances – steps – between two subsequent sweep positions. At every stop, the number of points in the proximity of the plane is evaluated. The process is repeated for different sweep directions.

Beyond the translational plane sweep, the segmentation is supported by a rotational sweep as well. This is conceptually equivalent to the linear sweep although it computes the rotation of a plane around an axis, thus additional plane parameters should be considered. The rotation is discrete and it allows the algorithm

to count the points on the plane at each step. Several rotational sweeps around different axes are necessary to guarantee the stability of the approach. The two types of sweep are described in detail below.

4.3.1 Linear sweep

The segmentation of the geometric primitives that enclose the 3D volume is accomplished by the linear sweep. As stated above the rotational sweep is also an essential step toward the segmentation of the data. It delivers intermediate results to allow for the computation of supplementary translational sweeps. However, the translational sweeps will be considered in the following description without regard to the input they receive: the knowledge about the plane parameters might either come from predetermined assumptions or from the result of a previous rotational sweep.

In the 3D domain in which the interior is defined a unidirectional linear sweep consists in moving (sweeping) a generic plane along its normal vector. The implicit equation of the plane

$$\pi_{ik} : a_i x + b_i y + c_i z + d_k = 0 \tag{4.1}$$

is considered in its Cartesian form. The normal vector to π_{ik} is described by $\mathbf{n}_i = (a_i, b_i, c_i)$ and for ease of computation coincides with the direction of the sweep. The term d_k indicates the perpendicular distance from the origin to the plane along the direction of $\mathbf{n}_i$. Across the sweep the distance d_k varies within a given range, whereas the plane coefficients a_i, b_i and c_i stay constant. This is due to the assumption that the plane is enforced to keep its orientation unchanged while moving, therefore every π_{ik} generated by the translation is parallel to the foregoing π_{ik-1}. Separately for each sweep vector $\mathbf{n}_i$, it yields:

$$\pi_{ik} : a_i x + b_i y + c_i z + d_k = 0 \parallel \pi_{ik-1} : a_i x + b_i y + c_i z + d_{k-1} = 0$$

with $i = 1, ..., 3$. In general, planes like π_{ik} in three dimensions can have arbitrary orientations depending on the plane coefficients a_i, b_i and c_i. However, the specific π_{ik} considered in the Manhattan-world domain is restricted to three main orientations since these are sufficient alone to define the orthogonal layout. Consequently, three separate linear sweeps along three orthogonal vectors like (a_i, b_i, c_i) should be computed to fix the positions of the geometric primitives in the interiors.

Each sweep to be correctly performed requires a sweep vector $\mathbf{n}_i$ and a stop criterion set by a constant step width. For every $\mathbf{n}_i$, given the whole sequence of progressive plane positions d_k with $k = 1, ..., n$, the number of discrete steps defining an individual sweep is $n - 1$. The number of steps (and plane positions) is related to the spread of the data along the considered sweep vector, thus it varies with the sweep direction. The linear spread of the point cloud is calculated as the range between the maximum and the minimum coordinates in the direction $\mathbf{n}_i$. This determines the extremal points of the sweep, respectively d_M and d_m. Since the step width Δd is an input for the algorithm, the step number is computed by

$$n - 1 = \frac{d_M - d_m}{\Delta d} \tag{4.2}$$

where n indicates the number of stops and the parameter Δd is preferably chosen accordingly with the point sampling density. At each step the distance d_k in the sweep plane represented by (4.1) is modified by

$$d_k = d_{k-1} + \Delta d. \tag{4.3}$$

Every increment of d_k implies a new calculation of the number of points that are within a consensus distance d_c to the current sweep plane. This process is explained in detail in the following paragraph.

Hypothesis-and-test to determine the positions of the geometric primitives

The segmentation of the data by means of plane sweep is mainly structured in two alternate phases due to the approach based on a hypothesis-and-test strategy. The hypothesis phase deals with the determination of the stop positions d_k for the sweeping plane as pointed out by (4.3). At every stop the hypothesis on the plane position is tested by calculating the distances $d(P_j, \pi_{ik})$ of all 3D points P_j from the current plane at d_k. A distance equal to zero obviously indicates a point on the plane, whereas a distance between zero and d_c corresponds to a point lying in the proximity of the plane. Both these two conditions generate an increase of the number of points classified as "detected at the position d_k". The existence of a primitive-surface at d_k is then confirmed only if the number of detected points exceeds a predetermined threshold. Otherwise, no real surface is assigned at d_k and the following position d_{k+1} is tested. The signed distance of the point $P_j = (x_j, y_j, z_j)$ to the plane in equation (4.1) is:

$$d(P_j, \pi_{ik}) = \frac{a_i x_j + b_i y_j + c_i z_j + d_k}{\sqrt{a_i^2 + b_i^2 + c_i^2}} \tag{4.4}$$

and results to be positive on one side of the plane and negative on the other. Every point P_j is eligible for being considered in the proximity of the plane if

$$|d(P_j, \pi_{i0}) - d_k| < d_c \tag{4.5}$$

where π_{i0} is the plane through the origin and the distance point-plane in (4.4) can be expressed as

$$d(P_j, \pi_{ik}) = d(P_j, \pi_{i0}) - d_k.$$

The value for the consensus distance d_c is given. The motivation for a calculation in the form of (4.5) is referable to implementation advantages. That is, in a fixed coordinate system the plane through the origin with given coefficients a_i, b_i and c_i is unique ($d_k = 0$), thus each sweep step is implemented by an increment of the coefficient d_k.

The segmentation by linear sweep works separately for the detection of the structures aligned horizontally (floor and ceiling) and vertically (walls). All horizontal structures are denoted by (4.1) with the index $i = \alpha$, whereas the walls have two possible orthogonal orientations respectively labelled by $i = \beta$ and $i = \beta + 90°$. While both computations share the same mathematical properties, the detection of the walls requires additional calculations performed by the rotational sweep.

Sweep along the normal to the plane of the floor

The main issue at this stage of the reconstruction is the computation of proper plane coefficients which determine the normal vector $\mathbf{n}_i$ to the sweeping plane. The sweep advances parallel to the primitive-surfaces contained in the volume and it "matches" them at specific positions if the detection of a sufficient number of 3D points occurs. The orientation of the sweeping plane is chosen each time in dependance on the orientation of the type of surface that should be detected (see Figure 4.4). Since the floor and the ceiling (both horizontal) have the same orientation, to sweep a plane along the normal vector to the floor has the
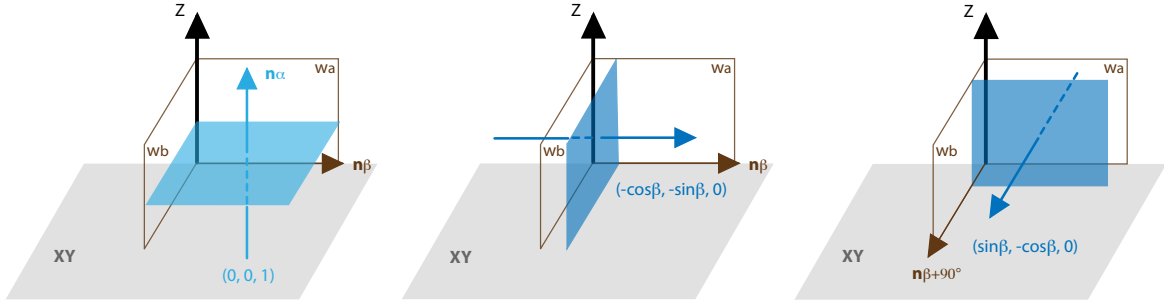
Figure 4.4: From left to right: sweep directions along the normal to the plane of the floor and separately along the normals to the planes of orthogonal walls.

same meaning of sweeping a plane along the normal vector to the ceiling. In this case the sweep is computed along the vertical direction, that means the coefficients which define the sweep direction are given by the components of the Z axis. Thus the sweep vector $\mathbf{n}_i$ for the detection of floor and ceiling is expressed by:

$$\mathbf{n}_\alpha = (0, 0, 1) \tag{4.6}$$

where it yields $a_\alpha = b_\alpha = 0$ and $c_\alpha = 1$. It follows that in this case the sweeping plane in (4.1) is reduced to the horizontal plane

$$\pi_{\alpha k} : z + d_k = 0. \tag{4.7}$$

Sweep along the normal to the plane of the walls

According to the Manhattan-world assumption, the walls of the indoor volume can be positioned uniquely along two main orthogonal directions, perpendicular respectively to the sweep vectors $\mathbf{n}_\beta$ and $\mathbf{n}_{\beta+90°}$ (Figure 4.1):

$$\mathbf{n}_\beta = (a_\beta, b_\beta, 0), \tag{4.8}$$

$$\mathbf{n}_{\beta+90°} = (a_{\beta+90°}, b_{\beta+90°}, 0). \tag{4.9}$$

The two terms c_β and $c_{\beta+90°}$ are equal to zero due to the parallelism of the wall planes to the vertical axis. Consequently, the sweep plane equations can be derived in the forms:

$$\pi_{\beta k} : a_\beta x + b_\beta y + d_k = 0, \tag{4.10}$$

$$\pi_{\beta+90°k} : a_{\beta+90°} x + b_{\beta+90°} y + d_k = 0. \tag{4.11}$$

It should be observed that, whereas the assumptions previously done for the vertical sweep have reduced the floor and ceiling detection to one degree of freedom, in the current sweep three degrees of freedom fully determine the solution (separately for each one of the above plane equations). One degree of freedom is actually the distance to the origin d_k, as before. Besides, further unknowns are added by the plane coefficients. In order to reduce the complexity of the problem, the coefficients a_β, b_β in (4.10) and $a_{\beta+90°}$, $b_{\beta+90°}$ in (4.11) are all expressed as trigonometric functions of the same angle β. Hence, the sweep vectors

defined in (4.8) and (4.9) are given by:

$$\mathbf{n}_\beta = (-cos\beta, -sin\beta, 0), \tag{4.12}$$

$$\mathbf{n}_{\beta+90°} = (+sin\beta, -cos\beta, 0). \tag{4.13}$$

Since β represents the angle between the sweep direction and the X or Y axis, the sweep vectors above practically coincide with the X and Y axes rotated about the vertical direction by β. Similarly for the plane equations (4.10) and (4.11), in order to force their dependance exclusively on the value of β, they are combined with (4.12) and (4.13). It yields:

$$\pi_{\beta k} : -cos\beta x - sin\beta y + d_k = 0, \tag{4.14}$$

$$\pi_{\beta+90°k} : sin\beta x - cos\beta y + d_k = 0. \tag{4.15}$$

In conclusion, for the complete localization of all vertical structures two independent sweeps are performed separately using the planes in (4.14) and (4.15). The value of β is automatically computed by means of a rotational sweep described in the following section, while the varying distance d_k corresponds to the sweep positions along the sweep vectors.

4.3.2 Rotational sweep

The rotational sweep aims at finding the orientation of the walls with respect to the coordinate system. The angle β introduced in the previous section is computed and its actual value is substituted in (4.14) and (4.15) to univocally determine the planes needed for the horizontal linear sweeps along (4.12) and (4.13). Since these planes are computed to localize the vertical wall surfaces, similarly to the walls they should be parallel to the Z axis, thus their Z component should be constantly equal to zero. The method of the rotational sweep for the computation of the wall angles can be applied to all interiors with vertical walls characterized by any orientation.

It should be pointed out that at this stage of the calculation the initial data set is already partially segmented by the execution of the vertical sweep. This performs – independently from the rotational sweep – a preliminary detection of floor and ceiling, whose points are clustered into separate objects. All 3D points remaining after this first classification are eligible for the localization of the walls. Hence, in order to minimize the computation time the following calculations towards the full data segmentation (including the rotational sweeps) are performed using only the points that were not yet assigned neither to the floor nor to the ceiling.

To compute a rotational sweep consists in rotating a plane about a given axis that lies in the plane. Throughout the sweep, the rotation is controlled by discrete angular steps at which the hypotheses on the presence of current vertical surfaces are either validated or rejected. The axis of rotation passes through a randomly selected point $P_r = (x_r, y_r, z_r)$ on the plane and is parallel to the Z direction. In accordance with these assumptions, the rotating plane is defined as the plane through the point P_r, which has as a normal vector the generic vector $\mathbf{n}_\phi = (a_\phi, b_\phi, c_\phi)$:

$$\pi_\phi : a_\phi(x - x_r) + b_\phi(y - y_r) + c_\phi(z - z_r) = 0. \tag{4.16}$$

In (4.16) the value of c_ϕ is actually equal to zero due to the orthogonality of $\mathbf{n}_\phi$ to the Z direction. As the vector $\mathbf{n}_\phi$ changes, the equation of the plane π_ϕ in the form (4.16) corresponds to all possible planes through

P_r. By setting $\mathbf{n}_\phi = (a_\phi, b_\phi, 0)$, the previous equation becomes

$$\pi_\phi : a_\phi x + b_\phi y - (a_\phi x_r + b_\phi y_r) = 0, \tag{4.17}$$

which satisfy all initial conditions for the detection of the vertical structures. The plane π_ϕ is rotated each time by a given angle $\Delta\phi$ with respect to its previous position ϕ', thus the current plane orientation is given by

$$\phi = \phi' + \Delta\phi. \tag{4.18}$$

The angular difference $\Delta\phi$ between two adjacent planes π_ϕ and $\pi'_\phi : a'_\phi x + b'_\phi y - (a'_\phi x_r + b'_\phi y_r) = 0$ corresponds to the angle between their normals, therefore it can be computed by the following formula:

$$\Delta\phi = \arccos \frac{a_\phi a'_\phi + b_\phi b'_\phi}{\sqrt{a_\phi^2 + b_\phi^2}\sqrt{a_\phi'^2 + b_\phi'^2}}. \tag{4.19}$$

Since only the pure orientation of the walls is considered, while the topology of the interior is treated in a later step, the total sweep angle to be covered by the range of the rotation is set to 180 degrees (rather than 360 degrees) starting from $\phi = 0°$, that means $\phi \in [0°, 180°]$. Within this range the current angular position is calculated by (4.18) where $\Delta\phi = 180°/s$. Inputs are the foregoing plane orientation ϕ' and the total number of steps s required within the full range rotation.

The principal orientation of the walls β coincides with the position of the rotating plane given by an unknown rotation angle $\phi = \phi_0$. This angle defines the coefficients a_ϕ and b_ϕ in equation (4.17). In order to find the value of $\beta = \phi_0$, a_ϕ and b_ϕ must be expressed as functions of β. According to that, $a_\phi = cos\beta$ and $b_\phi = sin\beta$ are set and the following equation of the rotating plane is derived from (4.17):

$$\pi_{\phi_0} : cos\beta x + sin\beta y = (cos\beta x_r + sin\beta y_r). \tag{4.20}$$

For $\beta = \phi_0$ the above equation defines the plane whose orientation indicates the densest distribution of walls (principal orientation). That means, most walls in the interior are supposed to be parallel to the plane π_{ϕ_0}. Due to the Manhattan-world layout, the remaining walls are supposed to be perpendicular. The goal of the rotational sweep is to compute an accurate and reliable value of ϕ_0 so as to define the principal wall orientation. To achieve that, a test is carried out at each plane position determined by a certain ϕ: the number of 3D points that belong to the current plane is calculated. A threshold defines the minimal distance point-plane needed to classify the points in the close proximity of the rotation plane. To further maximize the efficiency of the algorithm only the points in a cylindrical neighborhood of the axis of rotation are checked for consensus with the plane. The process is also iterated over more rotations in order to guarantee the stability of the result. Consequently, more sweeps are performed on different axes of rotation providing a joined output that univocally fixes one of the dominant directions of the vertical surfaces. At the end of the 180 degrees rotation, the angle β is recovered by the sweep angle ϕ_0 at which the plane (4.17) has collected the highest number of points. The normal vector to the plane with the maximum number of points coincides with the sought sweep direction defined by $\mathbf{n}_\beta$ in (4.12). Thus, β is the angle between the normal vector $\mathbf{n}_\beta$ and the X (or Y) axis. The whole rotational sweep algorithm is shown in the flowchart in Figure 4.5, while the histogram computation step described in the following section is visualized in Figure 4.6.

The specific dominant direction of the Manhattan-world domain identified by β depends on the initial conditions of the rotational sweep and particularly on the localization of the randomly selected points within the point cloud. If the majority of the random points belongs to a wall whose normal vector coincides with $\mathbf{n}_\beta$,

then β is actually the angle of the X axis with the normal to those walls. On the other hand, random points localized with higher frequency on a wall with normal vector $\mathbf{n}_{\beta+90°}$ determine an output angle between the X axis and the direction $\mathbf{n}_{\beta+90°}$ (see Figure 5.6 and related example in Chapter 5). The angles computed in the two cases always differ of 90 degrees. Implicitly, the value of the angle β outputted by the rotational sweeps is strictly connected to the distribution of the points in the point cloud with respect to the reference system. That is, if the point cloud presents for instance a higher density of points which are distributed on vertical planes perpendicular to $\mathbf{n}_\beta$, there will be a higher probability of selecting a random point on these planes. Depending on the coordinates of the random points, the numerical value of β changes without anyway altering the final reconstruction.
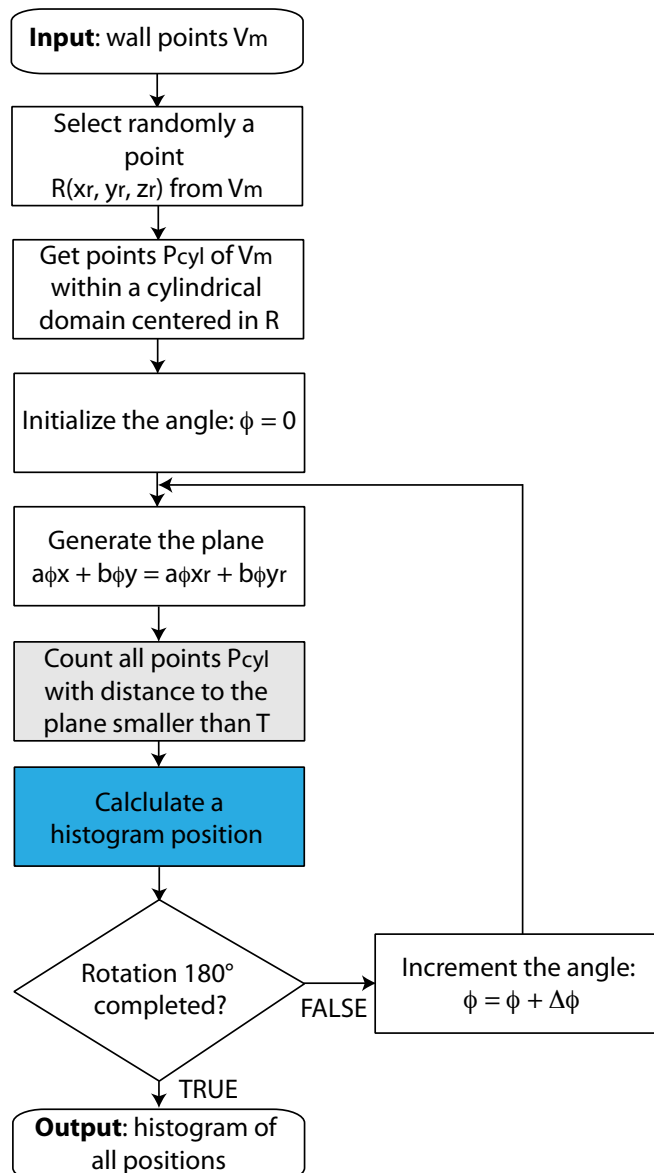
Figure 4.5: Overall flowchart for the rotational sweep computation. The output is a histogram of point occurrences. For the detailed representation of the grey box, refer to Figure 4.6.
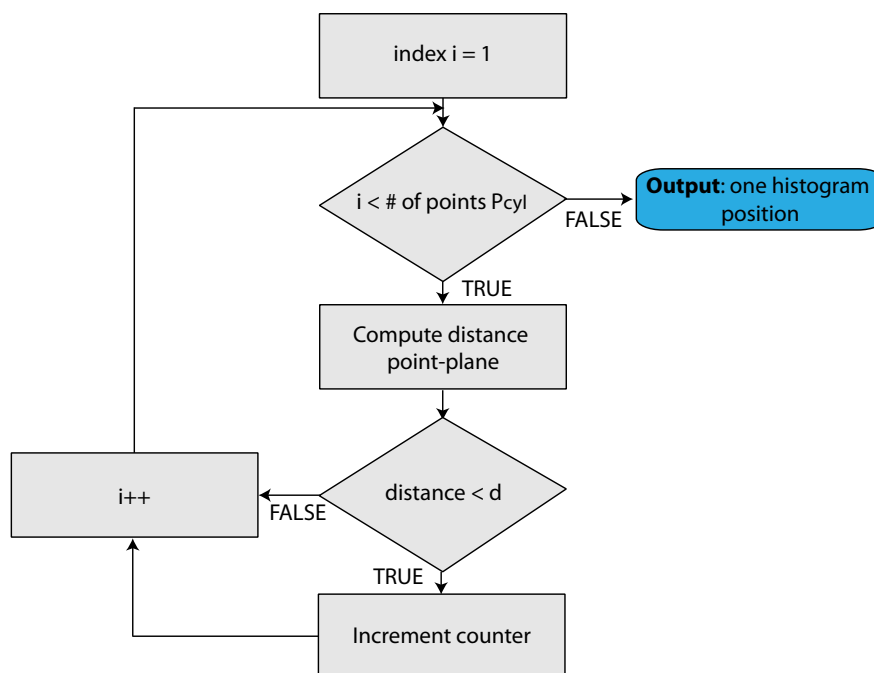
Figure 4.6: Detailed flowchart of the rotational sweep step in which the number of points corresponding to a single histogram peak is computed. For an overall representation of the rotational sweep flowchart, refer to Figure 4.5.

4.3.3 Surface segmentation

Referring to the field of computer vision segmentation is an image processing operation which partitions a digital image by grouping sets of pixels with similar characteristics into separate regions. All the pixels contained in one region are distinguished by the same label. In this way, objects of interest are isolated from the rest of the image that becomes more meaningful and easier to analyze [Shapiro and Stockman, 2001]. The 2-dimensional segmentation of images shares the same goal of point cloud segmentation. Although the latter is applied to 3D points and not to intensity pixels, it also aims at clustering the input data into different segment-objects. Such objects are the primitive-surfaces of the indoor volume localized through a histogram-based method. A histogram is calculated that gives an analytical and graphical representation of the point distribution at each stop of the linear plane sweep and at each angle of the rotational sweep.

With the results produced by the sweep reconstruction supported by the computation of histograms the 3D points can be clustered into separate surfaces allowing the point cloud to be partitioned in its geometric primitives. These are mainly the floor, the ceiling and multiple vertical surfaces.

Histogram visualization

The core part of the 3D segmentation is carried out by three linear sweeps: one along the Z axis and two separate horizontal sweeps for the localization of vertical surfaces. These sweeps perform the computations that lead to the ultimate localization of all primitive-surfaces. The rotational sweep is an intermediate step required to compute any primitive-surface orientation if this is not inferable from the problem domain. The

results of the rotational and linear sweeps are all supported by the visualization of histograms. In this section, the techniques for calculating such histograms are discussed. Eventually, the final localization of the surfaces of the indoor volume is given exclusively by the histogram of the translational plane sweep.

Once the translational sweeps along the above calculated directions are performed, every surface is associated with a plane in space, thus with a direction $- \mathbf{n}_\alpha$, $\mathbf{n}_\beta$ or $\mathbf{n}_{\beta+90°} -$ and a stop position $d_k(\mathbf{n}_i)$ along that direction. For every $\mathbf{n}_i$, the histogram $H(d_k, P_j)$ plots the frequencies of all points P_j occurring at each stop d_k for which the inequality (4.5) is satisfied. The sum of the frequencies gives the total number of points observed:

$$m = \sum_{k=1}^{n} f(d_k), \text{ with } f(d_k) = \sum_{j} P_j \ \forall \ d(P_j, \pi_{ik}) \in]d_k - d_c, d_k + d_c[.$$

For example, a plane like the one described by (4.1) is swept in order to detect primitive-surfaces perpendicular to its normal vector (a_i, b_i, c_i). The stop of the plane at the position $d_k \in [d_m, d_M]$ automatically implies the computation of the number of points that verify (4.5). In order to store a graphical visualization of the distribution of the points counted at each d_k, the results of the sweep are represented by the histogram. The peaks in the histogram occur in correspondence to the positions $d_p(\mathbf{n}_i)$ at which the number of points counted is maximum. Thus, at every position $d_k = d_p(\mathbf{n}_i)$ a planar surface perpendicular to $\mathbf{n}_i$ is expected to be detected in the interior. Each surface (peak) is fully defined since its orientation is given by the sweep direction $\mathbf{n}_i$, while its distance to the origin is determined by the position d_k.

The positions of the histogram peaks are filtered through a non-maxima suppression algorithm applied to the histogram values. Non-maxima suppression is a common technique for edge detection used in computer vision. It produces as a result the suppression of the image information that is not part of local maxima. The same basic idea is applied to the peak detection problem so as to find the primitive-surfaces (higher peaks) by eliminating the histogram values that are smaller than a given maximum. The search for non maximum values is local; it occurs each time within a given interval of distances d_k centered around the peak. After the removal of non maximum values, the remaining local maxima can be actually linked to a real surface in the volume only if their heights represent a sufficient number of 3D points. Additionally, the result is validated by a practical observation of the input data as well, which helps in defining whether distances between detected maxima are consistent with the data. Considering for instance a vertical sweep, implicit knowledge about building characteristics suggests without ambiguity that a) the distance floor-ceiling has a lower limit (e.g. one meter) and b) the floor is expected to be at a lower level with respect to the ceiling. If these conditions are verified by the observation of the two local maxima filtered in the histogram, the existence of the primitive-surfaces is confirmed and their reciprocal position is univocally established.

A similar technique for maxima localization by means of a histogram is used during the rotational sweep computation. In this case, the histogram is denoted by $H(\phi, C)$. The point distribution of the sweep is provided by the amount of points C accumulated at every rotation increment $\Delta\phi$. Thus, the orientation of the plane related to the densest observation determines the principal direction of the Manhattan-world interior (angle β), that is the direction of all parallel walls with the highest sampling density. Beyond the principal direction given by the absolute maximum in the histogram, a minor peak indicates a relative maximum that coincides with the direction of the orthogonal walls. Both peaks correspond separately to given angular positions defined by the angles ϕ_M and ϕ_m, with

$$\phi_m = \phi_M + 90°. \tag{4.21}$$

In practice, the rotational sweep algorithm carries out a single peak detection by calculating just the absolute maximum in the histogram. The second direction ϕ_m is inferred automatically by assuming a further histogram peak at 90 degrees distance from the first one. An alternative approach might consist in extracting both wall directions simultaneously from the histogram without computing additionally the perpendicular direction in a second step. Nevertheless, the choice of computing only the absolute maximum is motivated by a higher efficiency in the reconstruction. Detecting the secondary peak by adding up 90 degrees to the main peak like in (4.21) can be useful for recovering potential measurement errors. If the Manhattan-world assumption is supposed to be valid, erroneous data sets can be forced to provide a model with orthogonal walls in any case. This might happen if the data is collected with a low-cost range sensor as shown in Chapter 5. A second motivation for preferring the single peak detection consists in the ability to deal with point clouds with missing data. In the case a point cloud contains for instance only measurement points of parallel walls, just one peak would be present in the histogram, thus it would not be possible to determine all wall directions directly through the maxima extraction.

A further consideration about the rotation histogram suggests that the accumulation values outside the peak intervals are not equal to zero but they show a constant offset. This is due to the plane positions during the rotation, which allow the plane to collect at every position all points within a buffer region. Therefore, by rotating about the vertical axis the buffer regions of subsequent positions overlap in a cylindrical domain centered in the axis of rotation. This corresponds to consider the same points, those in the cylindrical domain, as multiple observation points without anyway altering the robustness of the results.

The determination of the peak positions from the analysis of the values of the translational sweeps may be affected by errors caused by the discretization of the sweep steps. Also, in order to include all points in the search for the consensus plane that best matches a primitive-surface, the consensus distance d_c in (4.5) is supposed to be bigger than the single steps. This causes a reshaping of the histogram values around the peaks. It results that the peaks exhibit a flat distribution in which several plane positions correspond to the same amount of points collected. Due to the flat peak shape, an error appears that could lead to an inaccurate detection of the peak position. However, to correct this error at a computational level an average of more positions within an interval around the selected peak is computed. The mean value is then taken as the final peak position, thus as the valid plane distance along the sweep.

4.4 Floor plan cell decomposition

Cell decomposition is one of the main representation techniques for solid modeling. In [Stroud, 2006], it is defined as the process of dividing the object space into a set of elements, each one full of the material of which the solid is composed. The same concept is applied in 2D to surface modeling to recover the floor plan. In this case the set of elements considered is represented by two-dimensional cells of variable size. Such cells may be seen as the projection on the floor of the solid cells that constitute the object volume.

As already pointed out above, the global approach to solid modeling presented in this thesis is a combination of sweep reconstruction and cell decomposition, the latter responsible alone for the floor plan extraction. Cell decomposition computes a successive region splitting similarly to split and merge algorithms for region detection. According to split and merge, an area of interest is first partitioned (split) into uniform regions that can be later unified (merged) only if they satisfy given homogeneity criteria. Half-spaces modeling is used for splitting the horizontal floor plane into polygonal cells by trimming the 2D space using straight lines.

In architecture, the floor plan specifies a horizontal cut through the building, usually at one meter height from the floor level. A view from the top of the cut shows the disposition of the spaces inside the building including walls, door openings and windows. Theoretically, this corresponds to a vertical orthographic projection of all interior features onto the horizontal floor plane. However, for the automatic modeling a generalization of the concept of floor plan is taken into account, which only concerns the permanent shape of the interior, thus just the pure floor topology and not temporary objects such as pieces of furniture.

At this phase of the modeling algorithm the knowledge about the positions of the vertical structures is employed and the floor topology is defined. The floor plan is basically extracted by partitioning the floor surface into rectangular cells derived by the multiple intersection of perpendicular walls. Among all cells obtained, only those "full of material" according to the definition of Stroud [2006] are merged together in order to compute the final floor plan contour. In a 2D view the floor plan is provided as a B-rep whose contour is given by adjacent linear elements which are defined by the edges of the single accepted (full) cells. In the following paragraphs it will be described how the cell decomposition approach is particularly well-suited for the interior reconstruction problem since the whole floor plan is not representable yet while its single cells are.

4.4.1 Localization of the cut-lines

In order to calculate the single cells into which the floor surface is decomposed, the 2D space is trimmed using straight lines. The vertices of each rectangular cell are found by the multiple intersections of four orthogonal straight lines that are referred to as cut-lines. Every cut-line divides the 2D floor surface in two parts. By recursively partitioning the space with more cut-lines the floor and the surrounding area are decomposed into a finite number of quadrilateral cells. The computation of these cells implies the localization of their vertices first, and then the definition of the sides of the cells.

The cut-lines are determined by the vertical projections of the walls on the floor plane or XY plane, thus they coincide like the walls to the density peaks computed during the two horizontal linear sweeps. Since the sweeps have been carried out along two Manhattan-world directions, the cut-lines are arranged by a perpendicular grid. With reference to the sweeps the generic notation $l_p(\mathbf{n}_i)$ for a straight line indicates the cut-lines l_p perpendicular to the vector $\mathbf{n}_i$. The cut-lines that coincide with the walls are given by discrete peak positions d_p calculated along $\mathbf{n}_i$. All cut-lines perpendicular to $\mathbf{n}_i$ can be expressed by the equation:

$$l_p(\mathbf{n}_i) : a_i x + b_i y + d_p(\mathbf{n}_i) = 0 \tag{4.22}$$

where $\mathbf{n}_i$ has components a_i, b_i and c_i along the coordinate axes and p indicates the number of cut-lines along $\mathbf{n}_i$. The intersection of the cut-lines to compute the cell vertices is performed at a global level. That is, all cell vertices are mapped simultaneously on the floor by intersecting two parallel sheaf of lines, each one dependent either on the vector $\mathbf{n}_i = \mathbf{n}_\beta$ in equation (4.12) or on $\mathbf{n}_i = \mathbf{n}_{\beta+90°}$ in equation (4.13). In a 3D visualization each cut-line in (4.22) is described by the points where the vertical planes in the forms (4.14) and (4.15) separately meet the horizontal floor plane, hence it is given by the 2D points whose coordinates satisfy the following equations

$$-cos\beta x - sin\beta y - d_p(\mathbf{n}_\beta) = 0, \tag{4.23}$$

$$+sin\beta x - cos\beta y - d_p(\mathbf{n}_{\beta+90°}) = 0. \tag{4.24}$$

The straight lines above actually correspond to the traces of the walls on the floor. Their intersection results in a grid that comprises rectangular cells of variable size.

Besides the peak-related cut-lines, artificial cut-lines are added to represent potential walls where the input data set has been cut for processing reasons or it has not been acquired correctly. The absence of walls due to a cut in the data always occurs at the geometrical extremities of the sub-cloud, thus the artificial cut-lines are added at the maximum and minimum coordinates along a given direction. If the direction $\mathbf{n}_\beta$ is considered, the cut-lines that are needed to generate the synthetic walls are:

$$-cos\beta x - sin\beta y - d_m(\mathbf{n}_\beta) = 0, \tag{4.25}$$

$$-cos\beta x - sin\beta y - d_M(\mathbf{n}_\beta) = 0 \tag{4.26}$$

and similarly for the direction $\mathbf{n}_{\beta+90°}$ the cut-lines are obtained by substituting the distances d_m and d_M into (4.24). From the output of the (real and synthetic) cut-line intersection, a distribution of points is extracted. The four vertices of each cell are recovered from the point distribution by clustering neighboring points in accordance to a repetitive adjacency configuration. The data structure that contains the cells stores a sequence of points listed in anti-clockwise order (or clockwise, important is that all cell vertices are listed consistently). In this way the sides of the cells can be calculated by the difference between the coordinates of two consecutive points.

Rectangular cells are now available for the extraction of the floor plan of the interior. A union of cells is implemented after testing which cells are compatible with the floor. It is not guaranteed that every cell produced by the cut-line intersection lies within the floor plan contour. Therefore, only the cells containing the floor material are eligible for being merged together. A check on the number of points inside the rectangles provide a separation of valid and irrelevant cells before assigning only the valid cells to the floor plan.

4.4.2 Full cell specification

According to Stroud [2006], which refers to the octree representation, each single element that constitutes the whole solid can be classified as being full of material, partially full or empty. The cell decomposition technique used for the floor plan extraction is restricted to the case of either full or empty cells. If n is the number of cut-lines along the vector $\mathbf{n}_\beta$ while m gives the number of cut-lines along $\mathbf{n}_{\beta+90°}$, the joint number of full and empty cell is $(n-1)(m-1)$. Full cells are considered valid rectangles, thus they can be assigned to the final floor plan.

The criterion for a rectangle to be considered valid and to be accepted as a floor cell consists in the definition of a minimum number of points within the cell boundaries. If enough laser point measurements are counted in the cell, it means that the area within the cell was targeted during the acquisition, therefore it is part of the interior. The threshold value which establishes the minimum number of points is defined a priori. However, such a static definition of the threshold might produce erroneous evaluations. Due to measurement limitations and to the discrete approximation of the sweep steps, cells that are clearly outside the floor plan contour might contain a considerable number of points anyway. This happens especially for small cells adjacent to a full cell with high point density, if some points of the high density cell go beyond the boundaries of the low density cells. On the other hand, cells characterized by a low sampling density might not be able to meet the general specification about the minimum number of points per rectangle. To solve these inconsistencies, which could drive the evaluation technique to an incorrect acknowledgment of the

validity of the cells, a dependency of the threshold on the area of each specific rectangle is introduced. The definition of the dynamic threshold value is explained below.

Definition of the threshold value

The idea is to create a threshold ε that changes in dependence upon the cell size. In this way, in order to be labelled as a full cell, a small rectangle is expected to contain less points than a rectangle with a wider area. With the definition of the dynamic threshold the problem of the nonuniform point density distribution in the laser data is bypassed.

Another issue is to eliminate the point overflow into the cells which do not actually contain any material. In practice, the points causing the overflow cannot be just removed from the empty cells since they are anyway not directly responsible for the error whereas the rectangle boundaries are. These are derived from the cut-lines, thus from the peaks calculated by the discrete sweeps. Hence, they are typically affected by an approximation error which may produce a shift of the calculated cut-line with respect to the real alignment of the surface points. The solution to this problem is to resize each rectangle area by computing shrunk contours for every cell. Afterwards, the threshold ε will be related to the number of points inside the resized area that is by definition smaller than the original cell.

Relatively to the input data set, the minimum and maximum values of the sampling density, respectively ρ_m and ρ_M are defined. Therefore, for each cell with area A the minimum number of points necessary to label that cell as full is included in the interval $[A\rho_M^2, A\rho_m^2]$. Thus, the generic dynamic threshold is equal to the product between the area and the density squared:

$$\varepsilon = A\rho^2, \tag{4.27}$$

where A indicates the area of the cell before being resized while the value of ε should be compared to the number of points within the rectangle after the contour shrink.

4.4.3 Floor plan visualization

The final goal of the floor plan computation is to determine the floor plan contour in order to define the topology of the interior volume and then its 3D model. The floor plan contour can be determined once the empty cells generated by the cut-line intersections are removed. In the following steps only the full cells are needed as they actually coincide with the floor plan whereas the empty cells mostly represent the area surrounding the outer floor contour. By computing the union of the only full cells the floor plan is defined.

From the point of view of the implementation, the union of full cells is computed as the union of their sides, in particular those sides that belong to the floor contour. A side of the cell is considered as a contour element for the floor only if it does not coincide with any other side of the neighboring cells. Coincident sides are removed and the floor plan contour is given by the union of the remaining outer sides.

For the pure visualization of the floor plan, the check over the validity of the cells that establishes whether a cell is full or empty is performed dynamically within the function that plots the results. Such a function extracts the cells full of floor material and displays them highlighting their contours together with the floor plan contour. A combined visualization in which the computed floor plan overlaps the point cloud and the empty rectangles allows for a validation of the final floor plan. If the plotted point cloud is contained within the specified floor plan, the results are validated.

(a) Door gap along Y_β (b) Door gap along X_β
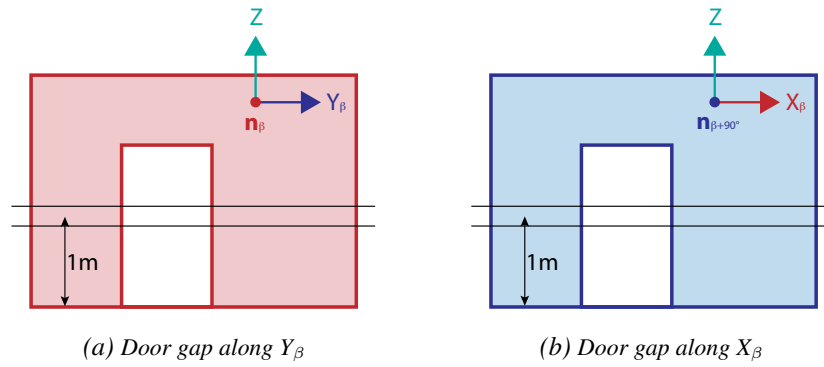
Figure 4.7: Two different configurations of the door extraction problem: door gaps on walls perpendicular to (a) $\boldsymbol{n}_\beta$ and (b) $\boldsymbol{n}_{\beta+90°}$.

4.5 Door extraction

The final 3D model may be enhanced by the addition of the doors included in the interior volume. The extraction of the doors provides completeness to the reconstruction and allows for delivering a more consistent solution to the automatic modeling issue. As a precondition for the method developed, it is necessary that the doors are open or receding since they are detected as linear gaps on the planar surfaces of the walls, similarly to [Becker and Haala, 2008]. The doors are localized on each vertical surface by processing a two-dimensional projection of the 3D point cloud on the surface itself (Figure 4.7). A comprehensive knowledge about the indoor spaces separated by the door is not needed. In particular, if the door under examination is used as a connection between two rooms, only the laser data of one of these is sufficient to recover a gap on the plane of the door.

The cut-lines derived previously are essential in the computation of the door positions. By locating the wall positions along the horizontal sweep directions, they identify the vertical planes on which the data should be projected for searching the door gaps. A new Cartesian system is defined based on the directions of the horizontal sweeps: the new coordinates refer to the orthogonal axes X_β, Y_β and Z. The 3D coordinates of a generic point in this reference system are (x_β, y_β, z). Because of the Manhattan-world assumption, two separate cases depicted in Figure 4.7 are possible for the statement of the door extraction problem depending on the alignment of the walls:

- Walls perpendicular to $\mathbf{n}_\beta$. They have a fixed coordinate $x_\beta = k$ for every cut-line $l_p(\mathbf{n}_\beta)$. The linear gap is detected along the direction Y_β (Figure 4.7a);

- Walls perpendicular to $\mathbf{n}_{\beta+90°}$. Unlike the previous walls, they have a fixed coordinate $y_\beta = h$ for every cut-line $l_p(\mathbf{n}_{\beta+90°})$. The linear gap is detected along the direction X_β (Figure 4.7b).

In order to formalize a solution to the door localization problem, the first configuration is considered. The second has the same characteristics if a rotation of 90 degrees is applied. Among the points associated with one wall plane (one specific cut-line), a horizontal stripe is selected. The stripe is centered at a height of one meter and contains points with coordinates $(k, y_\beta, 1 \pm \delta)$, where k is a constant value of x_β which depends on the wall position along the sweep vector $\mathbf{n}_\beta$. δ is the half width of the detected stripe along Z. By sorting the coordinate y_β and subsequently detecting a gap of missing points between two neighboring values, a door with the same width of the gap can be added to the model.

In the same way, for the second configuration a stripe with point coordinates $(y_\beta, h, 1 \pm \delta)$ contains all points necessary for the localization of the doors along the perpendicular walls. The coordinate h is a constant value of y_β, which localizes each time a primitive-surface along the sweep vector $\mathbf{n}_{\beta+90°}$.

In conclusion, the method for the detection of the doors is robust and the doors can be located with an accuracy that is affected only by measurement errors. The only limitation of the door extraction algorithm is the risk of false detections. Every convex or concave object deforming the wall profile might cause a gap in the data stripe on the plane of the wall. On the other hand, this could also be considered as an improvement to the flexibility of the described approach allowing for the detection of any other architectural element which does not overlap with the vertical surfaces. In the next chapter, it will be described how the receding space for an elevator's door is also classified as a gap. Windows detection could be implemented based on the same concept too although with modified specifications. The width of the gap is set to 80 centimeters referring to the standard measure of a door (80 x 200 centimeters). For the same reason, the point stripe is selected at one meter from the floor level; at this height the probability of detecting potential door gaps is maximum. Moreover, short objects on the ground will not interfere in the detection by modifying the wall profile.

4.6 Automatic 3D modeling

The algorithm for automatic 3D modeling is divided into the modules described above and summarized in Figure 4.8. Each module produces a partial result toward the whole reconstruction of the indoor volume.
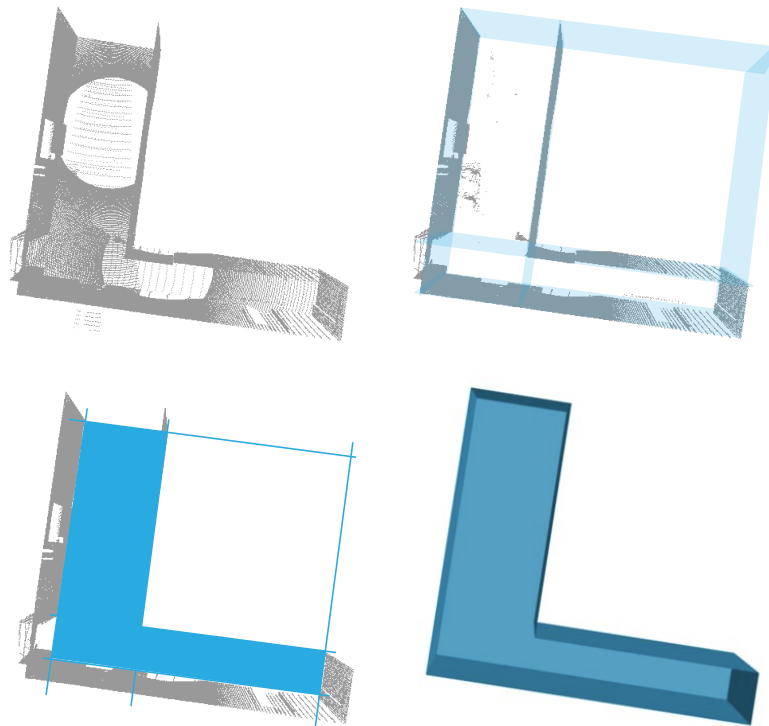
Figure 4.8: Representation of the sequential phases toward the computation of the 3D model (bottom right): the point cloud (top left) is segmented and half space modeling techniques (top right) are used to compute the floor plan topology (bottom left).

The computational phases are executed in a sequential timeline since the result of one module becomes

the input for the next phase. The last step – the assembly and the visualization of the 3D model – aims at the integration of the results as well as at the representation of the model by means of a B-rep technique. As explained by Van den Heuvel [1999], in architecture a surface representation is preferable to a solid model obtained using CSG. B-rep allows the building primitives to be handled separately whereas this is not possible if the model is computed with a volumetric approach.

In order to automatically generate and visualize the 3D interior model, the following inputs are needed. First, a reference to the point cloud is necessary to establish the correct location of the derived representation. Second, the union of the full rectangles produced by the cell decomposition is used as a shape-generator of the floor plan contour. Given the heights of floor and ceiling, the model is extruded from the floor plan by computing the walls as protrusions of the floor contour up to the ceiling level. If required, the creation of the pure model can be also enhanced by the addition of the doors. By the knowledge of position and width of the door gaps, every door can be added on the related wall surface in the model. Besides, for a verification of the results the model may be overlapped to the point cloud too.

The code for the implementation of the automatic reconstruction is written in C++. This is a standalone software for the generation of interior architectural models from point clouds. The so-computed 3D model reconstructs geometry and topology of the Manhattan-world interior in terms of its B-rep. In [Stroud, 2006], B-rep techniques are defined as solid modeling representations that describe 3D objects by the "skin" around them. The "skin" of an object is either composed by surfaces or edges. In the case of the automatic modeling illustrated in this chapter, each primitive-surface in the interior is framed by edges, at which two planes intersect. The whole indoor volume is eventually contained into the 3D space defined by the primitive-surfaces.

Chapter 5

Case studies

In this chapter, the results of the 3D modeling method developed for Manhattan-world interiors are described. Particularly the efficiency of the proposed indoor volume reconstruction is shown by giving an account of different case studies, in which the outputs of the automatic reconstruction in response to diverse point clouds are compared. The process of modeling the data is only applicable once the point clouds are first acquired with separate measurements and then registered to a common reference system.

Although the reconstruction method is independent from the instrument used to measure the 3D points, this chapter mainly refers to case studies that have as input data point clouds acquired with the pulsed laser scanner Leica HDS3000.

5.1 Laser scanning pipeline

As described in Chapter 2 for the general case, a typical laser scanning project begins with the point acquisition. However, the prerequisites for an accurate acquisition should be set beforehand in a preparation phase, in which the instrument and the scene are fixed for delivering the desired results. During the project preparation, the network geometry is defined. This applies to strategically placing scan targets in the scene in order to provide the acquisitions with shared measurements. In addition, before the acquisition starts, some scan settings should be also configured via the instrument's interface. Critical settings are mainly the measurement range and the sampling density. The range between a sample point and the scanner is estimated through a single point measurement and is used as an average reference value for the scanner distance to all points in the scene. The sampling density indicates the number of measured point per unit area (or unit distance) and is strictly connected with the time spent for the acquisition. Usually, a given number of acquisitions is performed in dependance with the extent and the geometry of the target scene. For large scenes or areas with obstructions multiple acquisitions are needed for a correct reconstruction. Moreover, if the site is complex or characterized by narrow corridors, additional measurements are necessary to guarantee a sufficient number of target constraints. As already pointed out the scanner is leveled, that means every scan provides a point cloud that is aligned to the Z axis. The acquisition phase is followed by the registration of all clouds into a common coordinate system. The pipeline ends with the modeling phase, whose process is automated by the implementation of the developed reconstruction method. Figure 5.1 depicts the whole laser scanner pipeline for the case studies described in this chapter and points out the degree of automation of each phase. Although the preparation phase is fully manual, it does not entail extensive effort in term of time and operational resources. On the other hand, the automation of the modeling process noticeably improves the efficiency of the whole indoor reconstruction.
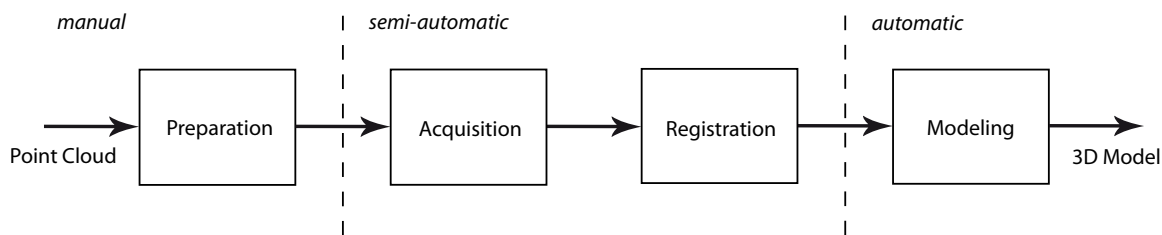
Figure 5.1: Laser scanner pipeline that applies to the case studies of this chapter for which the Leica HDS3000 has been used.

5.1.1 Data acquisition

The acquisition of all target points is achieved with progressive strategic re-placement of the instrument in order to obtain a global point cloud that is consistent with the whole indoor space. Once the final point cloud is derived by several laser scanner observations, each surface in the interior is represented by a discretization of 3D points.

The measurement process is composed by separate point cloud acquisitions. Depending on the size and the geometry of the indoor volume, the number of acquisitions varies. Each acquisition is accomplished by the scanner at a specific standpoint whose coordinates determine the origin of the local reference system (x_s, y_s, z_s) related to the cloud. From every standpoint all points observable within a $360°$ horizontal field of view are measured with a sampling density set before the start of the acquisition. The measurements are made possible by the rotating scanner head and are performed by a sequence of vertical scans detecting points along a line up to a field of view of $270°$. However, the acquisition of the ceiling points with the HDS3000 takes place in a stage that directly follows the acquisition of the points of the walls. In particular, at first the front window of the scanner allows for acquisitions of points within a vertical field of view (from $-45°$ to $32.5°$) resulting in a total angle of $77.5°$ (see Figure 2.5). Then, from the upper window the remaining points, mostly ceiling points, are captured by scans with an angular extension of $62.5°$ (from $27.5°$ to $90°$). The overlap between the fields of view of the two acquisitions – from the front and the back scanner window – is of $5°$, which becomes $10°$ within the full field of view of $270°$.

At every standpoint, the above described operations for data acquisition are automated by the scanner's software. On the other hand, the issue of planning the layout of the standpoints is a task of the user. This is a critical process since the relative positions of the standpoints should guarantee a certain overlap between neighboring point clouds. Also, in view of the following target-based registration, it is necessary to place artificial tie-points in the scene in order to provide enough transformation parameters for the cloud alignment. For a successful registration, some basic rules apply to selecting the correct locations of the tie-points. Typically, tie-points should be placed far apart from each other and should be distributed in the scene in every direction to constrain all six degrees of freedom. According to this rule, tie-points in a line should be strongly avoided. It is also a good practice to place tie-points far from the scanner and to ensure acquisitions with long baselines. Besides, a given redundancy in the number of tie-points shared by two adjacent observations is also recommended. Although three is the minimal number of tie-points that are normally necessary for the registration of two adjacent point clouds, it is always preferable to place additional tie-points in the scene, in case one or more points cannot be measured correctly. An example of a correct network geometry adopted during the surveying of an interior scene is illustrated in Figure 5.2.

A parameter which should be defined in order to perform effective acquisitions is the sampling density. That means, the point spacing is set by the user depending on the type of cloud required (and on the usage of the expected model). The lower the spacing between the laser spots onto the target, the denser the point cloud that results from the survey. Denser point clouds require a high acquisition time in contrast to sparse data acquired through an object-driven approach, as described in [Kwon et al., 2004]. The work of Kwon et al. [2004] points out how, for particular applications, it is just sufficient to measure a minimum number of range points by selecting clusters of objects instead of targeting the whole scene. However, this approach is better suited to rapid or local 3D modeling whereas the automatic modeling aims at a robust and accurate description of the interior. The differences between dense and sparse cloud acquisitions mainly consists in the level of integration of human assistance in the acquisition process and in the related degree of automation of the point measurement operations. In the case of object-driven data acquisitions, the human interaction to the sensor is crucial in order to extract significant portions of the target. On the other hand, if the survey aims at delivering a dense point cloud, a fully automated acquisition phase precedes a data processing step that still requires intensive manual effort. Unlike the previous approach, the processing step of the automatic reconstruction method eliminates the need for human support despite the dense data set available.

The acquisition of multiple scan clouds from separate standpoints requires high density measurements of all tie-points. As already observed, these have to be already positioned in the scene before the survey starts. The best way to measure the tie-points is to perform their acquisitions directly after the acquisition of each single scene from a certain standpoint. This implies that from every standpoint all "visible" tie-points are measured with a density which allows for a high accuracy estimate of the tie-point coordinates during registration. Precisely, it is the center of the tie-point that is automatically extracted by the registration software. Density values suited for fine acquisitions of tie-points correspond to a point spacing of 1.5mm while the typical accuracy reached in the calculation of the tie-point centers is about 2mm, depending on the scanner and on the target. Once the tie-point coordinates are measured, they are automatically stored with the related point clouds in their local reference systems. The scanner is also provided with an integrated color camera which gives the possibility to overlay the range data with the intensity information.
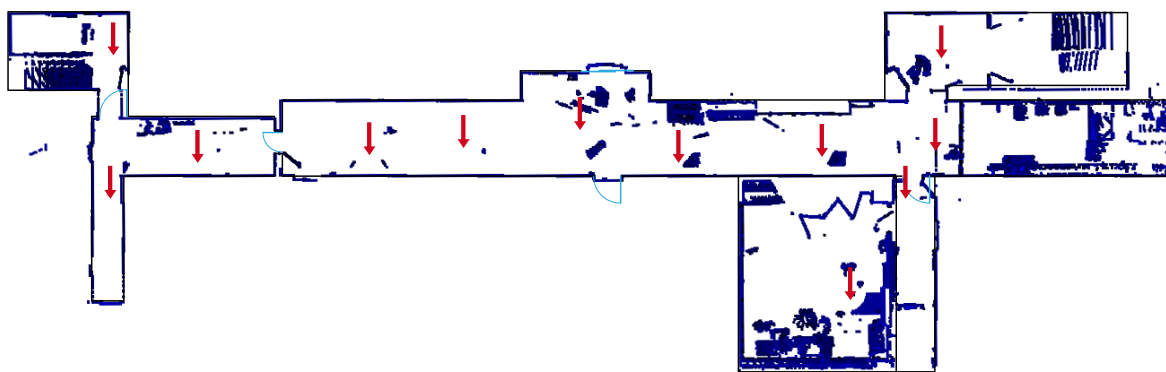
Figure 5.2: Orthographic view of a point cloud to show an example of measurement network geometry: the red arrows indicate the position of the standpoints. Each pair of standpoints shares at least three tie-points.

5.1.2 Registration with Cyclone

As already mentioned in Chapter 2, a registration of all point clouds is necessary for the estimation of their relative pose and the consistent computation of a common reference system. Once the clouds are registered, they are all optimally aligned in a way that minimizes the distances between homologous points in the new coordinate system. The registration is performed semiautomatically with the help of the 3D point cloud processing software Cyclone from Leica Geosystems (see results in Figure 5.3). There are mainly two types of registration that can be operated by Cyclone. The first approach to registration is constrained by the alignment of pairs of corresponding objects, each one in its initial coordinate system. At the end of the process, the two objects will have the same coordinates in the new reference system. The second option is to apply an overlap of the scan data and set so-called cloud constraints. To allow for an effective application of the cloud constraints, two critical factors should be considered, namely the number of points within the overlapping areas and the geometry of the overlapping clouds (in case of poor cloud geometry the problem may result underconstrained). If necessary, the two approaches – target registration and cloud constraint registration – can be also combined.

(a) *(b)*
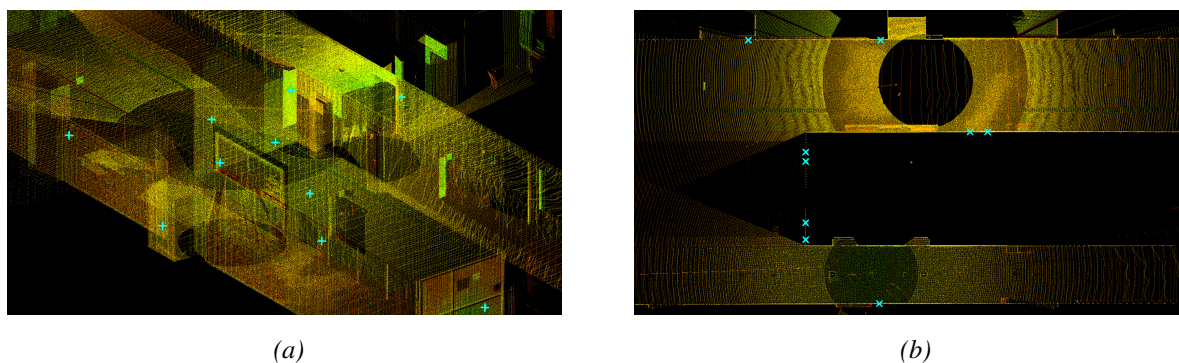
Figure 5.3: Point cloud alignment results in Cyclone: the blue crosses correspond to the tie-points. Snapshots of (a) a perspective view and (b) a top view.

In the case studies the registration is mainly based on the alignment of static tie-points which are identified by artificial targets specifically placed in the indoor test scene. So far, this kind of registration represents the standard method used by the main companies dealing with TLS. Diverse types of targets are allowed for the correct alignment of the clouds into a common coordinate system. Spheres (or half spheres) are 3D targets with magnetic mounts that have the advantage of being visible to the instrument at high distances. However, they are expensive and not easily portable. As an alternative to them, planar (flat) targets are able to support an accurate registration at lower costs and better flexibility. Besides, they are automatically recognized by the scanner during the acquisition, thus reducing the need for manual intervention while scanning. The disadvantage is that they are responsible for imprecise measurements from long distances or from critical angles. Nevertheless, this inconvenience can be easily solved by an accurate study of the global target layout, that means by providing a standpoint with enough targets within a close radius of its field of view. The type of targets chosen for the indoor scene registration in the case studies is a flat black and white target printed on a paper of size A4. Once the targets have been acquired automatically by the laser scanner, it is recommended to check manually for their accuracy since this will influence the accuracy of the registration result. Furthermore, the targets can be labeled to be distinguished uniquely among different point clouds so as to allow the software to perform the correct alignment. For this reason, it is necessary

that the labels assigned by the user are matching correspondent targets. During registration, the software Cyclone automatically creates a constraint for every pair of targets characterized by the same label.

The registration process assigns to all point clouds one global coordinate system that can be selected among those of the single clouds according to the user preferences. The output is a 3D representation of the interior determined by the coordinates of 3D points with a given spacing that depends on the sampling density of each scan. For the acquisitions, an average point spacing between 15 and 20mm has been set. Beside the X, Y and Z coordinates, other information can be associated to the points depending on the file format used to export the cloud. Just a continuous point coordinate listing (with no indication of the begin of a new cloud) is provided by the XYZ format. The survey file format SVY exports a list of coordinates preceded by a numerical point ID. Additionally, the first line of the file reports column headings. Most common exporting formats are PTX and PTS which also give account of the separation of the data in dependance on the number of individual scans contained in the registered cloud. The PTX file is a powerful file format that provides enough information for reconstructing the entire scan session. It contains implicit knowledge about individual scan locations as well as the points in their original reference system before the registration transformation. In the first two lines of the file, the number of scan columns and the number of points per column are stored (the HDS3000 can measure up to 5,000 points/column). Then translational and rotational parameters are provided before the X, Y, and Z coordinates, intensity values and color information. With the information retained by PTX files, point clouds can be also edited in other software packages such as Pointools.

In the present work it has been decided to create from the registered point clouds a file format that is smaller than the typical PTX file format since the only inputs necessary for the interior reconstruction are the 3D point coordinates. For that purpose, the point cloud is stored as a PTS file. The latter holds a stream of X, Y and Z coordinates (intensities and RGB colors) separated into individual scan acquisitions (for every individual scan, a point list is stored separately in the same file). At the top of each list of points, the amount of the points contained is also printed. This kind of file structure requires a further optimization step if all individual scans are expected to be unified, that means merged in a single list of points. A PTS point cloud that is unified in Cyclone has the advantage of maximizing the viewing and loading efficiency. Nevertheless, one more iteration is needed after unifying the clouds to allow for reading the input data with the automatic modeling software: since the original PTS data comes in ASCII format, the objective is to further reduce the file size by writing the PTS as a binary stream. This will significantly speed up all file I/O operations. Finally, the last type of file format that can be exported by Cyclone is TXT which provides the cloud coordinates in a text file that can be directly customized by the user.

Accuracy after registration

The degree of accuracy achieved by the registration is given by the mean absolute error of the global cloud alignment. The alignment is computed using both tie-points (vertex constraints) and cloud constraints. When the registration is successful, that means enough suitable constraints have been added, its status is labeled by Cyclone as "valid". In order to obtain maximum accuracy, the registration results are analyzed and the constraints are adjusted. If a certain constraint exhibits a relatively high average error, it implies that the target used to set that constraint was not aligned correctly to its correspondent tie-point in the adjacent scan cloud. In this case, it is recommended to disable the constraint affected by the error (after checking that the related target pair is actually misaligned). Hence, the mean absolute error that is meaningful in the evaluation of the registration results is only the one connected to the enabled constraints. There can be several reasons why a target leads to an erroneous constraint: first, the target label has not been reported

properly to the software. Second, the target was not clearly visible from one standpoint – for instance, the view angle was too small or the light conditions were unfavorable – or also, the sampling density has been set too low during the acquisition.

In one of the case studies, the registration process ended with a mean error for enabled constraints of 1mm whereas the disabled constraints produced an error of about 25cm, which as explained above has no influence on the global alignment. Two cloud constraints have also been added successfully so as to increase the global registration accuracy. The second case study is a more complex registration example due to the amount of points involved and the topology of the interior. The registration results is characterized by a mean absolute error for enabled constraints of 2mm, which is comparable with the error of 5mm for disabled constraints. In this case, three cloud constraints were included, all declared by Cyclone to be aligned.

5.2 Automatic modeling results

This section describes how the automatic modeling method has been tested on concrete examples of interiors. In particular, two main data sets with a considerable number of points have been acquired with the Leica HDS3000 on purpose to provide inputs that fit into the Manhattan-world reconstruction domain. From the data available, sub-clouds of interest are extracted and tested individually. The two main point clouds and the sub-clouds are presented in the following paragraphs together with the related modeling results. Afterwards, additional data sets are considered, which have not been acquired with the scanner HDS3000 but with a laser scanner prototype and a range camera, respectively. The test results of the automatic modeling with the above input data prove the robustness of the developed algorithm.

5.2.1 Control variables

A certain number of variables are set to control the outputs of the automatic modeling program. The optimization of such variables is supported by the analysis of the point cloud characteristics as well as by a test of the results in real time. Iterative updates of the control variables bring the solution to convergence. Once the value of each variable is properly adjusted via the user control, the algorithm automatically computes the 3D model in response to that specific input. A wrong parametrization of the control variables may lead to an incorrect model estimation. In the following list, all variables and their influence on the outcomes are discussed.

Consensus distance

As it appears in (4.5), the consensus distance is the maximum distance that allows a point to be considered in the proximity of the sweeping plane. The more the points collected by a sweeping plane at a given position, the higher the probability that the plane corresponds to a primitive-surface. The value of the consensus distance has a double influence on the output since it is also strictly connected to the step width Δd. This represents the separation between two consecutive sweep positions and is calculated as half of the consensus distance:

$$\Delta d = \frac{d_c}{2} \tag{5.1}$$

where d_c is the consensus distance. In the context of the strategy based on hypothesis-and-test, the step width defines the hypothesis while the consensus distance represents the test acceptance criteria. The relation
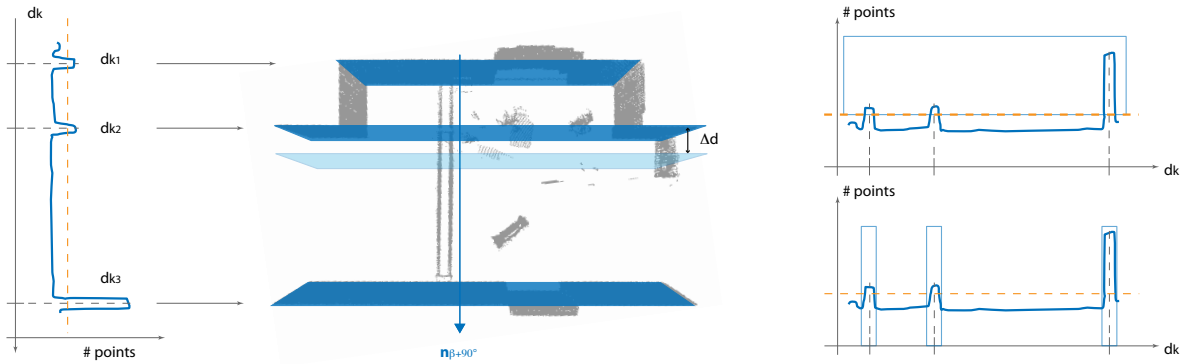
Figure 5.4: Histogram computation through the horizontal plane sweep: each histogram peak refers to a wall (left). A non-maxima suppression algorithm defines the suppression threshold and the suppression interval for peak localization (right).

expressed by (5.1) ensures that all points will be taken into account for the test since the space between two subsequent planes is included within d_c.

The consensus distance is given as an input for the calculation of the linear sweep and the following vertical segmentation too. It should be chosen accordingly to the sampling density of the point cloud, therefore high-sampled data allow for a wider distance (and a coarser sweep step).

Sweep vector

The X, Y and Z components of the sweep vector are given (Figure 4.4). They determine the directions of the linear sweeps, which coincide with the normal vectors to the sought primitive-surfaces. Throughout the reconstruction process three linear sweeps are performed as three dominant directions characterize the Manhattan-world interior. In particular, one sweep vector is aligned with the Z axis and the other two are expressed as trigonometric functions of the angle that determines the orientation of the walls. The sweep vector that is parallel to Z is given as an input directly by the user while the horizontal sweep directions are automatically computed by the program accordingly to the formulas (4.12) and (4.13). The unknown variable β is the output of the function which performs the rotational sweep. By fixing the values of the sweep vector components for the linear sweep, the computation of one specific type of dominant structure within the indoor volume is guaranteed (floor, ceiling or wall).

Suppression threshold and suppression interval

In Chapter 4 it has been described how the point cloud segmentation is supported by the computation of histograms. The histogram peaks localize the presence of surfaces in the interior. This concept is well represented in Figure 5.4, where the reference data consist of an example extracted by one the following case studies. A non-maxima suppression algorithm is responsible for multiple peak extraction with the suppression threshold as a reference for the minimal number of points expected at each peak. The suppression threshold is a specific variable given as an input to the linear sweep. In Figure 5.4, it is represented by the dashed dark yellow line.

With regard to the rotational sweep, the threshold is needed dependently on the strategy implemented for detecting the angular peaks. One strategy implies that only the angle corresponding to the maximum peak is computed while the measure of the second angle is inferred by adding 90 degrees (due to the Manhattan-world assumption). This calculation does not require a threshold value but just a function that returns the maximum point distribution of the histogram. On the other hand, if both peaks are computed in parallel, the processing steps for maxima extraction are the same as those for peak detection in the linear histogram, therefore the suppression threshold should be also defined as a control variable.

One can think that the denser the point cloud, the higher the threshold value allowed. However, it must be considered that the value of the suppression threshold is related to the step width too, thus to the maximum number of points countable in the proximity of the sweep plane at each step. Also, a special consideration should be done with respect to the threshold value for the rotational histogram. During the rotational sweep in fact, the search for points is reduced to a cylindrical domain centered in the rotation axis. The process is iterated and several rotations are computed so as to add up all histograms of single rotations to a cumulative histogram. It results that the suppression threshold is dependent on both the radius of the cylindrical domain and the number of iterations, which determines varying accumulation values in the sum of the rotational histograms.

The suppression interval is the second control variable that has influence on the non-maxima suppression algorithm output. It is concerned with the extent of the interval in which the maximum histogram peak is looked for. Its value remains almost constant throughout the execution of the case studies due to the fact that the histogram peaks exhibit always a regular shape.

Correction interval

A reshaping of the histogram peaks is needed in order to detect the positions of the primitive-surfaces with a higher accuracy. Due to the discretization of the sweep steps, each peak is approximated to a slanted rectangular shape, whose width stands for the double distance between two subsequent sweep positions. That means, the peak does not correspond to a well-defined position but has an extent that is equal to the consensus distance. Therefore, if the consensus distance is set to 5cm, a peak that is 5cm wide is expected. By applying a correction based on the average of the position values within the extended peak, the error due to the uncertainty on the determination of the peak coordinates is reduced of one order of magnitude. The correction interval defines the total number of position values involved in the computation of the average value. It strictly depends on the sweep step and on the point cloud noise level.

Rotational sweep variables

Two control variables are typical of the rotational sweep. One is the number of iterations which fixes the total number of full-range rotations from zero to 180 degrees. The second variable sets the rotational step number that specifies how many discrete angular steps control each single rotation. The number of iterations should be constrained to a given minimum in order to assure a robust detection of the wall orientations. In the case studies, it is never allowed to set an iteration number lower than 50 for point clouds with up to two million points. On the other hand, a very high number of iterations might reduce the software performances in terms of execution speed.

The total number of steps within one full-range rotation automatically fixes the angular increment. By setting 110 steps, an angular increment that is smaller than two degrees is obtained.

Cell decomposition variables

The variables related to the the cell decomposition are used to control the computation of the floor plan. It has been already explained in Chapter 4 how the cell decomposition works and which are the criteria for the specification of full or empty cells. Such criteria are determined by the dynamic threshold for the sampling density (see equation (4.27)) and by the shrink difference between the contours of the original and the resized cell. The dynamic threshold is defined for each cell in accordance with its area. Two threshold values are given: one computed on the basis of the minimum estimated point density and the other in dependence on the maximum estimated density.

A further variable establishes how much the cells need to be shrunk to avoid point overflow from neighboring rectangles in the overall full cell classification. Theoretically, new cell contours are created by the shift of the original cut-lines toward the inner part of the cells. The shift generates a buffer region between the actual cell and the resized cell. The width of the buffer is given as a control variable. For noisy or large data, a relatively thick buffer is chosen with the condition of not allowing to generate a cell that become void, thus, implicitly, an upper limit to the buffer width is also set. Considering the average sampling density of the input data, a 15cm wide buffer has been used in most of the following examples.

Door width

A proper value for the door width is necessary as a control variable of the door extraction process. The reason for that is the flexibility of the extraction method, which is based on the detection of missing data along a selected point stripe. By "flexibility" it is meant the chance of detecting standard interior doors of different widths as well as other kind of doors such as those of an elevator. Thus, depending on the type of target, a specific width is given. According to the basic dimensions of a single door, the minimum standard width is 62.5cm whereas the maximum is reached at one meter (especially for outer doors). However, most interior doors are 75 or 87.5cm wide (from [Rabausch and Volz-Graetz, 2010]). An initial door width of 80cm is set and the process of searching gaps in the data is iterated by a direct visualization of the results. The point gaps are generated by possible misalignments between the wall plane and the door surface. When this does not happen, the requirement is that the door is left open during the acquisition phase.

5.2.2 Case study: long underground hallway

The first case study that will be analyzed is concerned with the scans of a long hallway (a part of it is shown in Figure 5.5). The hallway is located at the basement of a university building, thus no conventional windows – whose glass may generate holes in the laser data – are expected. Besides, university constructions have often in common with office spaces a rectangular layout that best fits the problem requirements. The main area of the hallway is occupied by a long corridor divided in two parts by a door (left open during the acquisition). The staircases are positioned at the corridor extremities while the elevator is accessible from the center of the hallway. The latter is also the largest point of the hallway with a width of 4.4m. In length, the scanned area of the corridor measures more than 45m, however, in total only 36m are considered for the reconstruction. Scan data of a side room adjacent to the hallway are also used to test the automatic modeling approach in presence of furniture, which will be filtered as noise by the algorithm. The whole data set includes 7,938,819 3D points acquired from 12 standpoints. After unifying the clouds, the amount of points is slightly reduced to 7,925,195. In the following sections, it is described how 3D models are automatically achieved from sub-clouds extracted from the hallway scans.

Example A

The case study analysis starts by testing a very simple but significative sub-cloud with about eight hundred thousand points. This corresponds approximately to the portion of the data shown in Figure 5.5. Such a data set does not include any particular characteristic that can be an issue for the reconstruction but at the same time it represents in a good way all the attributes of a typical Manhattan-world environment. The vertical segmentation results in the correct detection of both floor and ceiling level. It should be also observed that the primitive-surface assigned to the floor is detected at a negative position along the sweep direction, which confirms the laser source's standpoint as the origin of the reference system (the scanner is actually mounted on a tripod at a certain hight from the ground). However, during segmentation a certain number of wall points is classified incorrectly. Points that belong to the walls but are directly below the ceiling will be added to the ceiling points, whereas all wall points just above the floor will be automatically selected as floor points. This error is systematic and it is due to the finite value of the consensus distance for the assignment of the 3D points to a certain sweep plane. Consequently, the maximum distance at which a wall point is erroneously segmented – and thus it is assigned to the floor or the ceiling – is equal to the consensus distance. Despite this point-to-plane deviation, the correct positions of the horizontal surfaces are not affected by the outliers since they are just computed in a following step by uniquely detecting the peaks of the point density histogram along the vertical sweep. It results that the floor and the ceiling surfaces are respectively detected at -1.581 and 0.944m.

The rotational sweep that determines the orientation of the vertical structures outputs an angle of 171.82 degrees. As explained in the theoretical discussion of the modeling method in Chapter 4, this value gives the angle between the normal vector to a wall and one of the coordinate system axis, let us assume the X axis. The result of the rotational sweep strongly depends on the localization of the initial random points that are taken as starting reference for the iterations. Each random point extracted by the algorithm belongs to a wall which is exclusively perpendicular either to the vector in (4.8) or (4.9). In the case of this example, it is considered that most of the initial random points are localized on walls with normal vectors aligned to (4.8). Then, the cumulative result of the rotational sweeps indicates 171.82 degrees as the angle between the direction (4.8) and the X axis. On the other hand, if the majority of random points was chosen on walls with normal vectors parallel to (4.9), the alternative – and equally correct – value of the rotation angle would be 81.82 degrees, i.e. 171.82 minus 90.

This observation becomes clear in Figure 5.6 where two different sub-clouds extracted from the same case study are shown. Sub-clouds within the same case study are referred to a unique Cartesian coordinate system

(a)   *(b)*
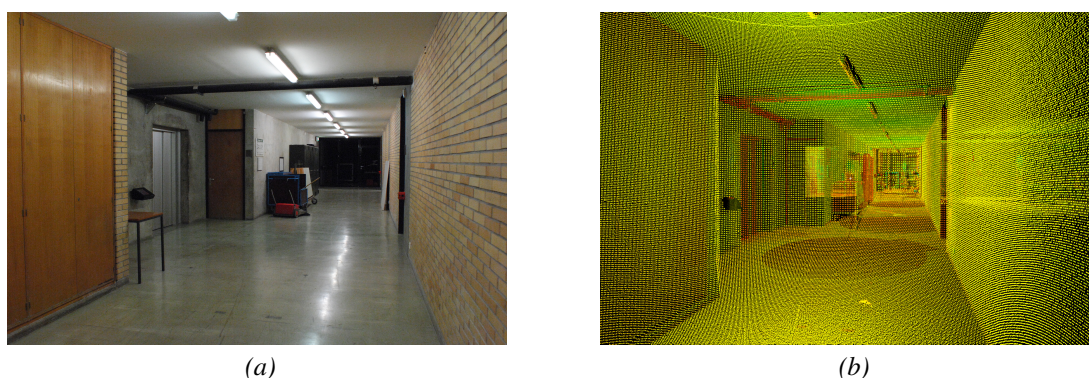
Figure 5.5: Views of the hallway in different representations: (a) a color image and (b) the corresponding point cloud.
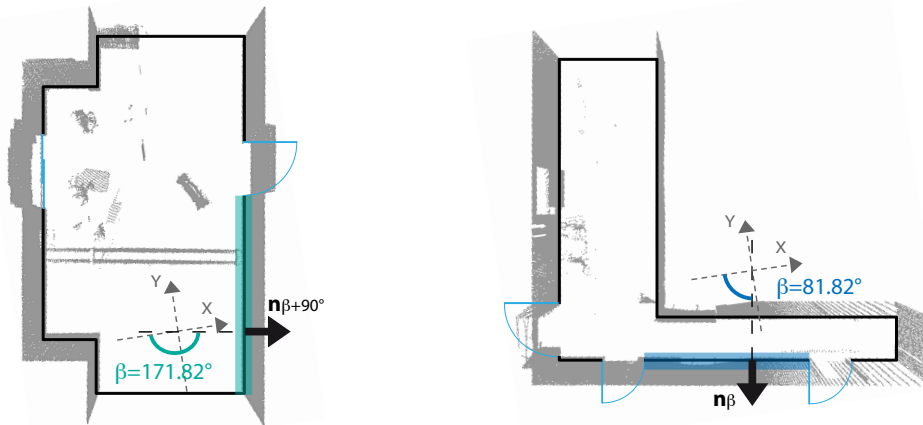
Figure 5.6: Different point distributions, which refer to the same coordinate system, generate different angles as outputs of the rotational sweep.

as they are the result of a global registration process. Depending on variable acquisition parameters such as the number of standpoints, the distance instrument-target and the sampling density, two portions of the same registered point cloud might exhibit different point distributions. By the random selection of points that are the inputs for the rotational sweep, the probability is higher to extract a point within an area of the data with higher density. Therefore, the dominant direction computed by the rotational sweep is presumably the direction along which the densest acquisitions were performed. Since the densest point distributions relative to the two clouds in Figure 5.6 are orthogonal to each other, the angles found by the rotational sweep (separately for each cloud) differ of 90 degrees.

Once the vertical plane orientation angles are given, two more linear sweeps which determine the wall positions along two separate directions can be performed. The number of walls and wall surfaces detected along one sweep vector corresponds to the number of cut-lines along that vector. As a final result in this example three and four cut-lines are found respectively aligned to the two perpendicular directions. Their intersection generates a total amount of six rectangular cells and is visualized in Figure 5.7. Here it can
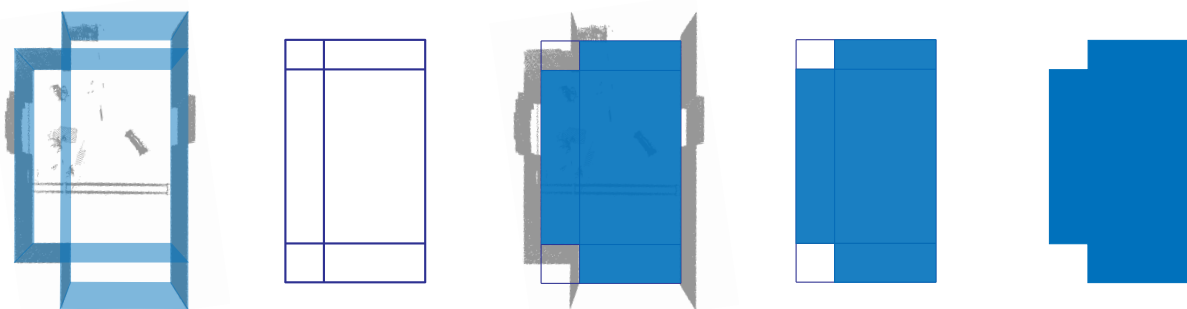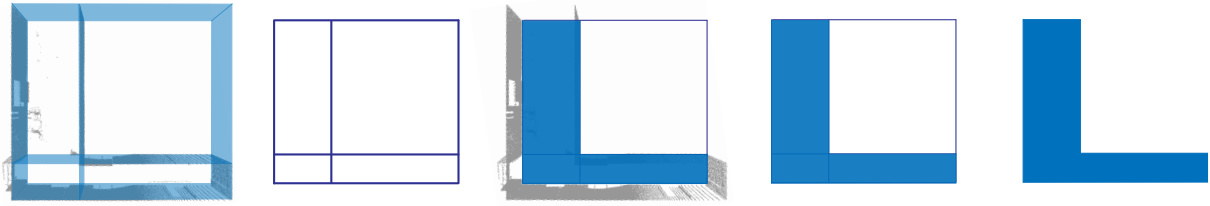

Figure 5.7: Floor plan cell decomposition by cut-line intersection in Example A. The cut lines are localized by the sweep planes along the wall directions and the ground plan is computed by the union of floor cells.

also be observed how artificial cut-lines were automatically added at the two extremities of the data, where actually no vertical structures are present. This is a useful procedure that aims at creating the model of a closed volume also when the input data is a subset with missing wall measurements. Figure 5.7 represents the whole process of generating the floor cells towards the ground plan computation. In particular, the figure
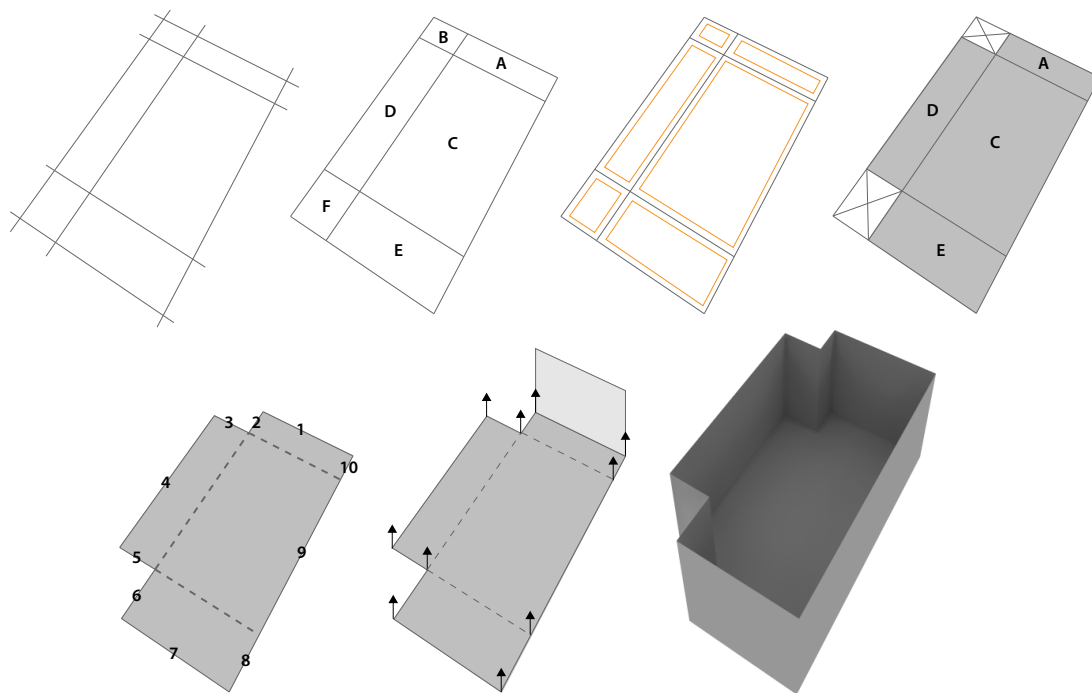
Figure 5.8: Steps towards the computation of the 3D model: from the cut-line intersection to the 3D representation of the point cloud in Example A.

highlights how, taking the sweep planes as references, the cut-lines are localized in correspondence to the planes that match the walls and the extremities of the sub-cloud. The orthogonal grid obtained as a result of the cut-line intersection is afterwards compared to the original data and the ground plan is calculated by the union of the cells "full" of floor points. A more detailed illustration of the algorithm used for the determination of the floor cells and the following wall computation is given in Figure 5.8. In the upper-left corner of the image the process starts with the cut-line intersection whose result is a list of 2D points (the points of intersection). In order to describe the floor as a union of quadrilateral cells, the points obtained from the intersection are clustered four by four, so that only neighboring points belong to the same cluster. Each cluster of four neighboring points represents a floor cell, named from A to F in Figure 5.8. The data structure that stores the cluster points contains a list of coordinate points in anticlockwise order. After the area of every cell is computed, the cell contours are shrunk like it is shown by the dark yellow lines in the third step of the figure. The shrunk contours allow for a correct separation of the cells full of material from the empty ones according to the threshold defined in (4.27). The separation between empty and full quadrilateral cells results in the upper-right representation of Figure 5.8, in which the empty cells (overlaid by a cross) are discarded and the full cells (grey-colored) are merged together. The next step is the construction of a polygon with the same shape of the ground plan being equivalent to a polygon whose perimeter corresponds to the ground plan outer boundary. To achieve that, the four sides of the full quadrilaterals are compared to each other in pairs: if the side of a cell overlaps the side of an adjacent cell, that side will not be part of the outer boundary, hence it will be eliminated (dash-lines in the figure). In order to demonstrate that two sides overlap, their extremities should coincide. For each pair of quadrilaterals, 2^4 side comparisons are needed. In the current example, the ground plan is defined by four cells, therefore the computation of its outer boundary requires in total 2^4 by five comparisons. The resulting outer sides are labelled with numbers from one to 10 in Figure 5.8.
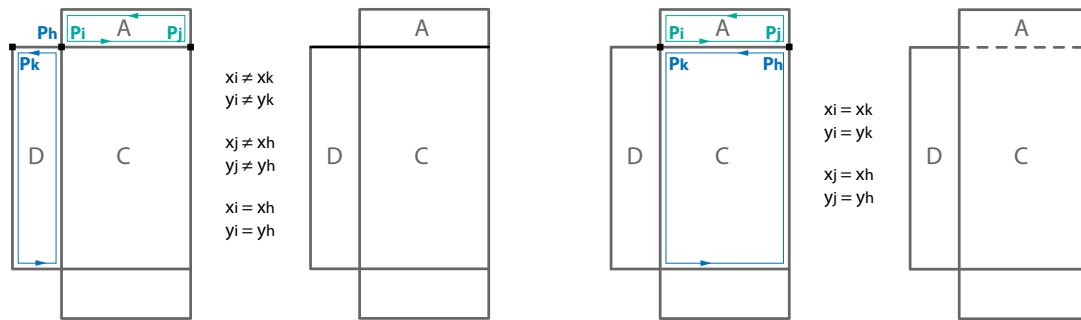
Figure 5.9: Process of removal of the cell inner contours: the ground plan perimeter is exclusively defined by non-overlapping sides.

The technique used for the side comparison is based on the analysis of vertex coordinates. Let us consider two sides: one goes from P_i to P_j (cell A in Figure 5.9) and the other one goes from P_h to P_k (cell D or C). Since the vertices of the cells are stored in anticlockwise order, the sides coincide if $P_i \equiv P_k$ and $P_j \equiv P_h$, that is if the x- and y- coordinates of P_i and P_j are equal to the coordinates of P_k and P_h, respectively. Figure 5.9 illustrates additionally how the sides are eliminated if the above conditions are satisfied (left context) and how instead they cannot be removed if their extremities do not match properly (right images).

Once the sides that characterize the outer boundary are determined, the final model is computed by extruding the vertical walls from the floor contour up to the ceiling level. By extruding it is meant computing additional cells on the wall planes. In practice, so many quadrilateral cells along the wall directions are calculated as the number of outer contour sides obtained. The final model achieved by the automatic reconstruction is illustrated both in Figure 5.8 and Figure 5.11a. All models in the following representations will be shown without the ceiling for ease of visualization.
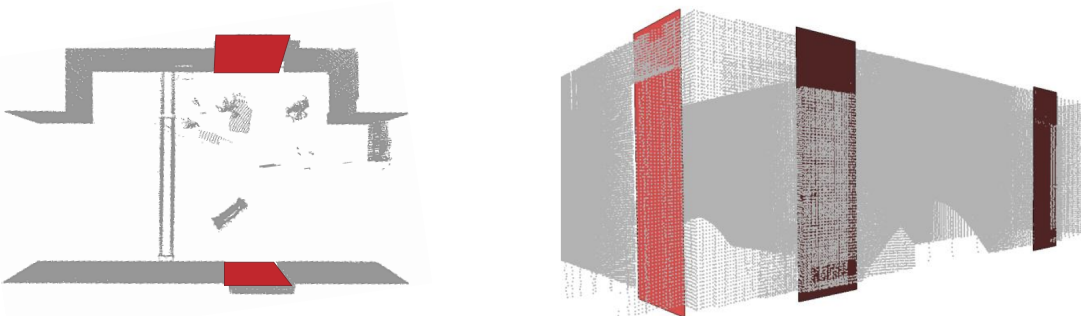
Figure 5.10: Results of the door extraction method in Example A (left) and Example B (right).

The 3D model of the above described sub-cloud is also enhanced by the additions of doors in correspondence to the data gaps detected by the door extraction method. In particular, a room door, which was close during the scans but is slightly receding with respect to the wall plane, is correctly localized. The second door detected is that one of an elevator. As explained in the previous section, the only control variable to be given as an input to the door extraction function is the minimal width of the gap coincident to the door. In this particular example, a width of 80cm is found to be appropriate for the extraction of both doors. The results of the door computation are visible in Figure 5.10, where also a detailed view of the doors in the following

example is shown. The doors are visualized within the point cloud as patches defined by the door width and the door position in the acquired scene.

Table 5.1 shows a summary of the modeling results for the described sub-cloud (Example A) as well as for other sub-clouds that will be considered in the following examples of the current case study. The value of the floor heights is not constant due to systematic variations of the instrument alignment, which has to be carried out manually. However, the fluctuations of such values are contained within a range that is comparable to the horizontal accuracy calculated by the method explained in the following Section 5.3. Also, an observation of the ceiling values reveals how just one of such values is placed at an average distance of more than 35cm from the other ceiling levels experienced in the same case study. The reason for this divergence from the mean height is that the ceiling in the Example C belongs to a room that is separate from the rest of the hallway and thus it is characterized by its own structural features. Finally, it should be noticed that the last column in Table 5.1 lists the number of valid floor cells in relation to the total amount of (full and empty) cells found by the cut-line intersection.

Example B

The second example related to the case study of the long hallway corresponds to a sub-cloud extracted from the bottom left extremity of the data overview in Figure 5.2. In particular, it includes the 3D measurements of a corner area of the hallway. Because of the orthogonality conditions spread over different wall pairs, the topology of a corner area represents an ideal case study to test the reconstruction of a Manhattan-world interior. Although the extent of the interior volume in Example B is comparable to the volume of the Example A, the number of points contained in the sub-cloud of the current example is slightly smaller. This is due to a part of the data in which the point density of the acquisition is lower than the mean density of the whole case study. Such a low density area can be observed in the right image of Figure 5.6 and is due to the presence of a single standpoint in a relatively narrow corridor. The point cloud of the Example B contains

(a) Example A   *(b) Example B*
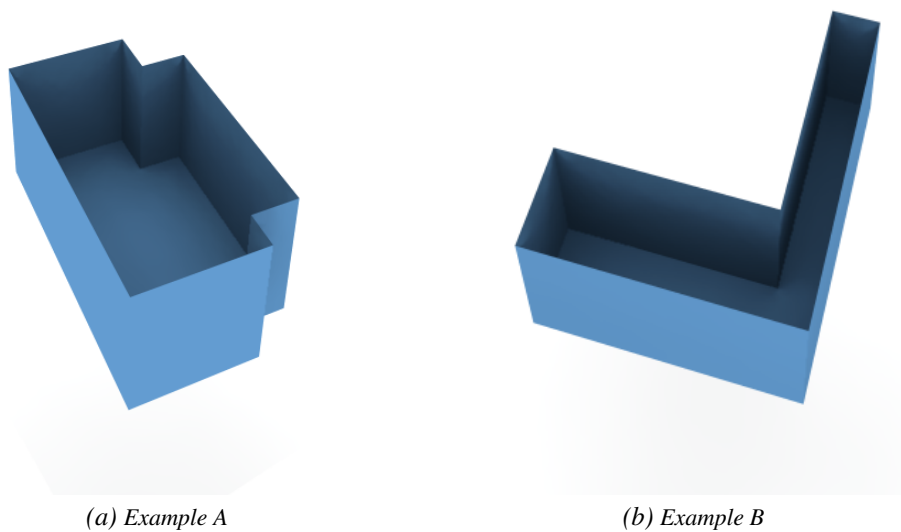
Figure 5.11: Models obtained through the application of the automatic reconstruction technique using as inputs the sub-clouds described in Example A (a) and B (b).

Figure 5.12: Floor plan cell decomposition by cut-line intersection in Example B. The cut lines are localized by the sweep planes along the wall directions and the ground plan is computed by the union of floor cells.

646,514 points and is the smallest sub-cloud examined in the case studies that have as inputs data acquired with the Leica HDS3000.

By using the same control variables already tested in Example A (some of them are listed in Table 5.2), the experimental results of the algorithm applied to the Example B show consistency in the 3D models obtained. Indeed, the two sub-clouds belong to the same registered point cloud which is characterized by a uniform sampling density (except for the extremities of the hallway) and a regular topology, therefore most of the control variables can remain unchanged. Only the value of the door width needs to be modulated for a correct extraction of the doors. The similarity with the former control variables allows for a correct computation of the positions of floor, ceiling and walls by keeping the value of the consensus distance to 10cm during the execution of both the vertical and the horizontal sweep. In particular, floor and ceiling are found at -1.617m and 0.983m, respectively. It results that the orthogonal distance between the ceiling plane and the floor is 2.6m, which is a reasonable measure for the ceiling height of a modern office building. However, the separation between ceiling and floor calculated in the Example A is 2.525m, that means 7.5cm smaller than the one calculated in the Example B. This result is caused, on the one hand, by the systematic error due to the instrument alignment and, on the other hand, by the step discretization of the sweep computation. It can be observed anyway that the variation of the room heights among the two examples does not exceed the consensus distance.

The module that calculates the angle between the wall directions and the coordinate axes is the rotational sweep function. By setting in Example B the same rotational sweep variables used for all examples in this case study, the orientation of the walls is given by an angle of 81.82 degrees. In practice, this result is achieved by selecting 50 different random points so as to iterate over separate rotations, each articulated in 110 steps. The diversity of the angle values in Example A and B has been already discussed in the previous paragraph and is visualized in Figure 5.6. In brief, different angles just refer to dominant walls with perpendicular orientations.

Example	Points	Floor Height (m)	Ceiling Height (m)	β (degrees)	Ground Cells
A	778,982	-1.581	0.944	171.82	4/6
B	646,514	-1.617	0.983	81.82	3/4
C	2,415,768	-1.624	0.951	0	13/30
D	1,501,630	-1.568	1.331	3.27	4/4

Table 5.1: Summary of the modeling results for different sub-clouds extracted from a single case study.

The number of cut-lines derived by the horizontal sweeps and localized by the histogram peaks is three for each one of the two Manhattan-world directions. In accordance with the computed cut-lines, the ground plan is partitioned into four rectangular cells, of which only one is not accepted to build the floor plan as it results to be "empty", thus outside the floor contour. The cells originated by partitioning the floor are shown in Figure 5.12, where it is possible to observe the whole pipeline of the ground plan computation starting from the 3D point cloud to the planar representation of the union of floor cells. The model computed from the point cloud in Example B is shown in Figure 5.11b.

By further processing the wall points, the door extraction method localizes four gaps. Three of them are doors which are successfully detected by the algorithm while the fourth gap is the result of a false detection. The error in the final output is due to an ambiguous interpretation of the data causing an inconsistency between the existing gap and the type of structure identified. In fact, while the stripe of missing points is correctly computed on the wall plane (there is actually a gap in the data), the object responsible for the point gap cannot be properly classified (as noise) by the software and is simply added to the list of detected doors. In conclusion, every convex or concave object deforming the wall profile could cause a linear gap on the wall, which may be reconstructed as a door. However, the method is robust and the doors are located with an accuracy that is affected only from measurements errors (Figure 5.10). The value of the door width used in Example B is of 0.5m.

Example C

This example is intended to be an extension of the model described in the Example B. The input point cloud includes the same points contained in the sub-cloud of the Example B plus additional measurements of the hallway corridor. Precisely, the whole data set consists of more than two million 3D points from both sub-clouds in Example A and Example B. Because of the considerable size of the point cloud under examination and especially the large amount of the wall points, some of the control variables should be adjusted with respect to their previous values in order to achieve the expected results. In particular, the consensus distance is decreased from 10cm to 8cm within the function that executes the horizontal linear sweep (in the top view of Figure 5.13, sweeping from up to down). Due to the related reduction of the step width, a higher number of cut-lines allows for the localization of all relevant vertical structures contained in the point cloud. The suppression threshold is also modified in a way that less points should be collected in correspondence to the peaks to accept them as walls. A comparison of the new control variables in Example C with reference to the variables used in the previous Example B is shown in Table 5.2.

The execution of the vertical sweep shows how the heights of floor and ceiling are very close to the values computed in the previous examples. In particular the floor is found at -1.624m – about 4cm difference with the result in A and only 7mm away from the floor in B – while the ceiling results in being positioned at

Example		Step Width (m)	Suppression Threshold (points)
B	HS 1	0.1	25,000
	HS 2	0.1	11,000
C	HS 1	0.1	11,000
	HS 2	0.08	500

Table 5.2: Comparison of control variables for the two horizontal sweeps in Example B and C.

0.951m – that is 7mm above the ceiling in A and about 3cm below the ceiling found in B. These values demonstrate the robustness of the sweep method from an experimental point of view.

At the next step of the algorithm, the rotational sweep computes the direction of the dominant walls, which is indicated with a good approximation by the same vector already calculated in the Example A. In theory, the two vectors found in Example A and Example C are supposed to have the same direction and either the same versus or opposite versus. However, differently from Example A, the vector obtained from the rotational sweep in C is rotated by 172 degrees. This indicates that the program detects an angle of eight degrees (180 minus 172 degrees) between the two point clouds, which are expected to be completely aligned, instead. Such an output might be explained by the wide extent of the walls along the hallway corridor in the Example C. It is reasonable in fact that the error in the angle is originated by the computations of the rotational sweep in C (not in A) as the results given by the two previous examples A and B already proved to be consistent with each other, thus they are assumed to be correct. In conclusion, it can be observed that a misalignment of eight degrees is negligible in this conditions since it is distributed on a long distance of about 30m (this is the length of the portion of the hallway considered in this example).
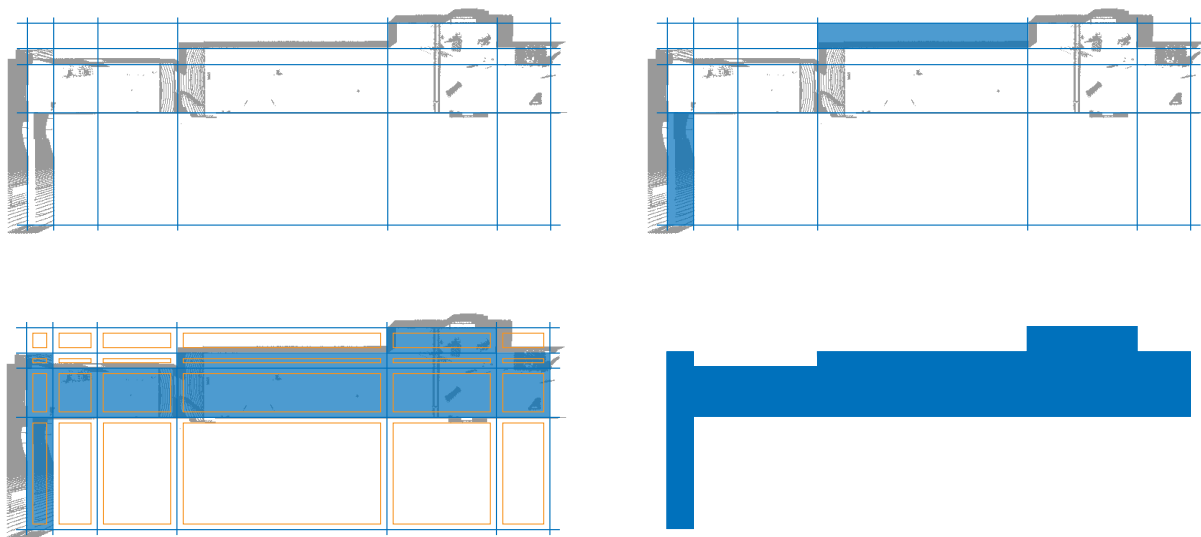
Figure 5.13: Cut-line intersection and ground plan computation after shrinking the quadrilaterals.

By the histogram peak extraction process six cut-lines are detected along the shortest hallway direction and seven along the orthogonal direction. Their intersection generates 30 cells, of which 13 are valid and are used for building the floor plan. These results are observable in Figure 5.13, where also the technique of shrinking the cell contours is depicted. Actually, only five cut-lines are shown along the shortest direction of the point cloud in Figure 5.13. The fifth and sixth cut-lines are found very close to each other, thus they generate an extremely narrow stripe of cells, which is not visible in the figure (the stripe contains one valid cell and five cells that do not belong to the ground plan). In the image at the upper right corner of the figure the error in evaluating the floor cells is shown by emphasizing two incorrect detections. Such an erroneous separation between full and empty cells occurs before operating the cell contour shrinking. What happens is, on the one hand, that the larger blue-colored cell is selected as a valid cell and used to build the ground plan, whereas it is obviously outside the floor outer boundary. On the other hand, the second cell highlighted in the same image is instead discarded from being part of the floor, therefore it is initially left out the ground plan contour. That computation results in a final model that is not consistent with the input point cloud. As already explained in Chapter 4, there are two events which cause this inconsistency. The first one (related to

the error in the upper cell) is triggered by a point overflow from a full cell to an adjacent cell which actually contains only a limited number of points, not sufficient alone to overlay the suppression threshold. That is, once the points from a neighboring cell are added to an "empty" cell, this automatically becomes valid in accordance with the threshold value. Besides, the second event leading to a miscalculation of a valid cell is the lack of measurement points, due to a low sampling density in the acquisition phase (see the smaller blue cell in the figure). In that case, the number of 3D points counted inside the cell is not high enough to activate the threshold although the cell belongs to the floor. By shrinking the cell contour and counting only the points within the new boundary, the distinction between full and empty cells proves to be correct. The final model obtained by the protrusion of the walls from the ground plan is shown in Figure 5.14 (left image).

A sub-cloud that is part of Example C was also tested separately. It refers to the area that includes almost the whole corridor except the left corner area (it includes the data set in Example A, too). The comparison of the modeling results of this sub-cloud with the whole point cloud C shows continuity in the values obtained. The floor level is localized in both examples at -1.624m while the ceiling has an offset of about 2cm. The rotational sweep defines the dominant direction at zero degrees, which confirms the correctness of the angle computation in a critical situation characterized by extremely long walls. A comparison of the two reconstructions is observable in Figure 5.14, where the 3D models are shown. In addition, in order to set the proportions of the two sub-clouds with reference to the whole hallway, the Figure 5.15 depicts a top view of the cell decomposition results of the current example next to the same results obtained for the sub-cloud in the following test. The two areas appear in the real scene aligned like they are visualized in the figure.

Example D

The last example is slightly different from the others already analyzed in the same case study because of the characteristics of the point cloud. The target is a single room that is separated from the rest of the hallway and was acquired from two standpoints: one positioned in the center of the room and the second with the scanner directly out of the door. A top view of the geometry of the room together with the scanner standpoints is given in Figure 5.2. Precisely, the area interested in this example is that one at the bottom of the hallway: a quadrilateral area that measures about 7.8m by 5.8m.
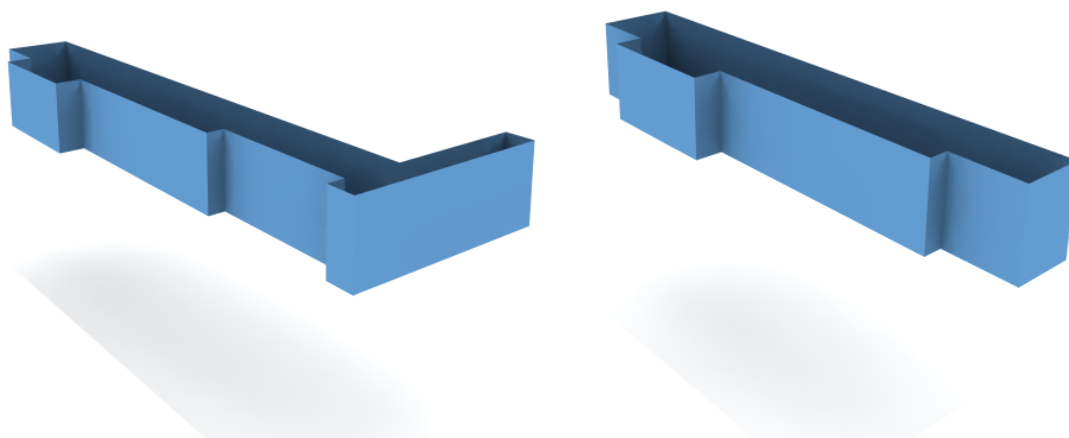
Figure 5.14: *Models obtained from the reconstruction of the sub-clouds in Example C: the test sub-clouds are respectively given by the union of the data contained in both Examples A and B (left) and by the sub-cloud in Example A plus additional measurements (right).*

Figure 5.15: Ground plan cell decomposition results for Example C (left) and Example D (right) visualized next to each others like they are positioned in the real scene.

The most interesting characteristic of the acquisition done in the room of Example D is the amount of noisy points due to the high quantity of different kind of objects lying on the floor. Consequently, together with the 3D points of floor, ceiling and walls, many 3D measurements are also associated to points that do not contribute to the computation of the correct structure of the room (Figure 5.17). In total, 1,501,630 points were measured during the acquisitions. The main goal of this example is to demonstrate the robustness of the automatic modeling method by successfully managing the 3D reconstruction despite the high number of outliers.

By a comparison with the previous examples, it is clear that floor and ceiling of the room are positioned at different heights along the vertical direction. They are computed by the vertical sweep at -1.568m and 1.331m. The ground plan geometry resulting from the cut-line intersection is a basic quadrilateral area. It is obtained by the union of four valid cells and is illustrated in detail in Figure 5.16. In the left image of the figure, the virtual intersection of the sweep planes is represented in combination with the point cloud (deprived of the floor points). Here it is easy to see how the point cloud is augmented with 3D measurements from diverse objects: in particular chairs, a table and some panels on the right side. All these objects contribute to the addiction of points to the sweep planes at the corresponding sweep positions, thus they are also responsible for the peaks that generate the cut-lines (the suppression threshold is influenced by the objects points as well). On the right side of the same image, on a plane behind the wall points, additional points are grouped into quadrilateral clusters. These are actually points belonging to the outer wall of the building, while the points that the algorithm is able to recognize belong to the inner wall of the room. The
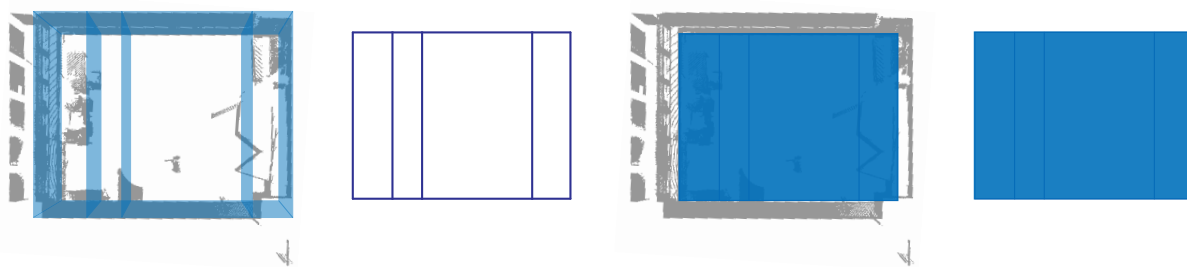
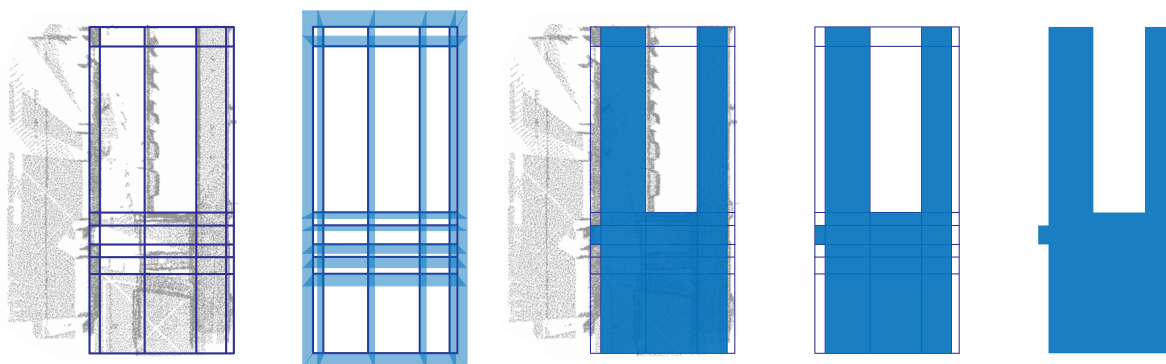*Figure 5.16: Floor plan cell decomposition by cut-line intersection in Example D. The cut lines are localized by the sweep planes along the wall directions and the ground plan is computed by the union of floor cells.*
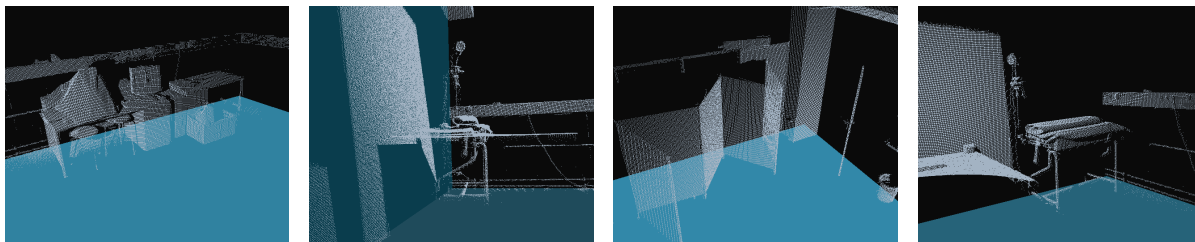
Figure 5.17: Detailed views of noisy measurements caused by pieces of furniture that were lying on the floor during the acquisition. The measurements refer to Example D.

outer points are visible by the scanner standpoint and therefore they are collected during the measurements since the room is provided with glass windows even if it is situated underground. Except for the high amount of outliers, the structure of the room is well suited for the reconstruction.

Together with the room in Example D, a similar example is also taken into consideration. This contains the sub-cloud of the room already described and a part of the adjacent corridor. The number of points given as an input to the software increases to 4,493,122. Both models are shown in Figure 5.18. The ground plans (and thus the models) of the two point clouds of Example D are originated by the intersection of not corresponding cut-lines. In particular, the ground plan of the larger scene includes a total of 24 full cells that are extracted among 35 quadrilaterals. The histogram peaks correspond to eight and six cut-lines, respectively along each one of the two sweep directions. The separation between full and empty ground cells is shown on the right image in Figure 5.15 where the union of the blu regions indicates the accepted floor plan while the cyan areas correspond to the cells that have been discarded.

(a)   *(b)*
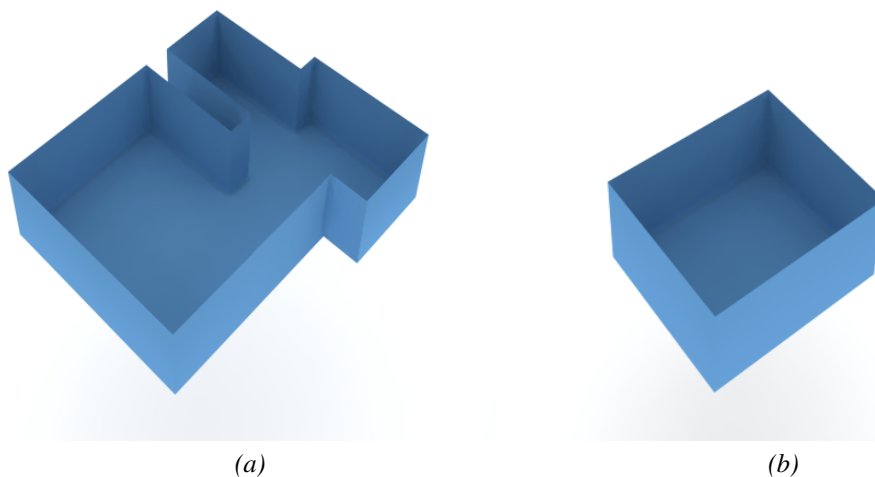
Figure 5.18: Models obtained from the point clouds in Example D. The model (a) refers to the sub-cloud of the Example D plus additional point measurements of the adjacent corridor.

5.2.3 Case study: complex and dense point cloud of a university building's floor

The second case study considered consists in a more complex point cloud acquired from the interior of another university building (shown in Figure 5.19). By "more complex point cloud" it is meant a larger data set of an interior with an elaborate Manhattan-world topology. The whole floor is mainly divided in two parts whose ceiling levels differ in hight. The elevators cavity is responsible for the separation between the two areas. The whole registered point cloud contains 16,737,097 points that have been measured from nine standpoints with a variable point spacing ranging from 10 to 20mm. Whereas the extent of the whole indoor space measures 57.3m by 9m, more than 80 square meters are occupied by the elevator cavity, thus resulting in a gap in the data. Full-length windows substitute the walls at the two extremities of the interior.

A sub-cloud is extracted from the registered point cloud that results from the above acquisitions in order to be used as the input for the next example. It consists in more than one third of the whole point cloud and corresponds to the portion of the data highlighted in Figure 5.20. The full-colored area is the result of the ground plan computation, which is overlapped with an image of the registered point cloud directly taken from the software Cyclone. Since the sub-cloud that is initially extracted out from the registered data contains more than 6,500,000 points, a voxel space is defined to decrease the point density and improve the software performance in terms of computing time. A discussion about how the voxel space is conceived is given in the following paragraph. The number of data points to be processed by the reconstruction algorithm is finally reduced to 85,211. The floor is detected at -1.65m, that is 3.53m away from the ceiling. However, this result is to be referred to the major ceiling alone. Due to the double-height ceiling, three planes should be detected by the vertical sweep (including the floor plane), whereas the algorithm is designed exclusively for the detection of two horizontal planes, that means for the reconstruction of rooms with a single ceiling level. In this example, the final ceiling height is determined either by the highest or the second highest peak in the histogram calculated during the sweep.

The value of the rotation angle is 44.18 degrees. The sequential steps of the ground plan computation are visualized in Figure 5.21. In the first image of the sequence, the cut-line cell decomposition is overlapped with the point cloud. The cut-line intersection results are then isolated in the second image, where they are represented by the sweep plane intersections to recall the correspondences between sweep planes in 3D and cut-line projections onto the floor plane. Due to the presence of objects on the floor of the main hall where the measurements took place, additional cut-lines are detected along the most extended direction, reaching a total of eight cut-lines. Along the perpendicular direction six more cut-lines are computed, two of which are automatically added by the software at the extremities of the sub-cloud in order to define the boundary lines between the sub-cloud and the rest of the data. The intersection produces 20 full cells that are accepted as ground cells and 15 empty cells that should be discarded having not reached the required value for the
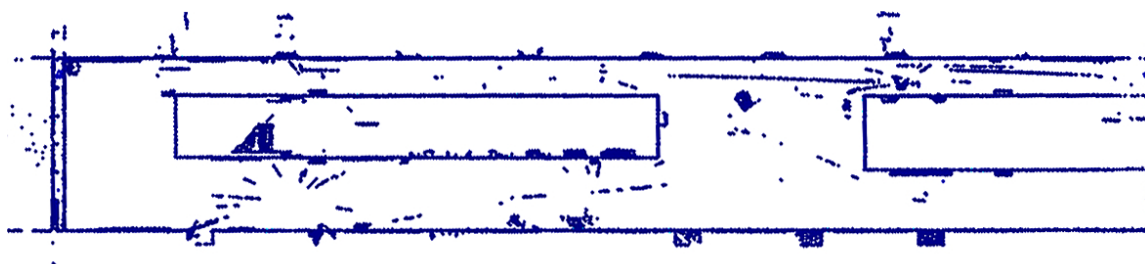
Figure 5.19: Top view of a dense point cloud of a university building's floor. The floor and ceiling points are hidden for ease of visualization.
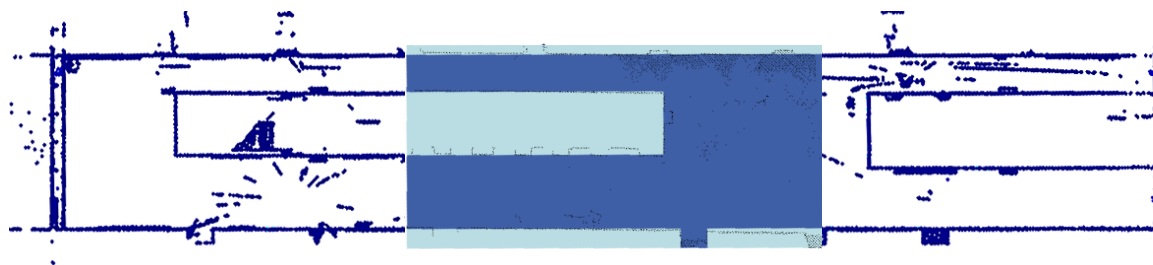
Figure 5.20: Ground plan of a test sub-cloud overlapped with the point cloud of the whole registered cloud.

dynamic threshold. On the first and third image of Figure 5.21, a part of the point cloud appears not to be included in the area covered by the ground cells. In reality, the points that exceed the ground plan on the right side belong to the ceiling of a neighboring room and were measured from the main hall through a stripe of glass panel at the top of the wall (however, the points that are involved in the determination of the ground plan cells are only the floor points, previously segmented by the vertical sweep). A similar situation, in which noisy points from an adjacent room are included in the measurement of the room under test, was also encountered in the Example D within the first case study.

The last three images of the series show how the final ground plan also includes one small-sized cell. This cell, which corresponds to a wall cavity hosting a door, has a reduced dimension compared to cells that are usually computed by floor plan cell decomposition. Therefore, the suitability of the algorithm is claimed also for the detection of detailed cells with a size up to less than 1.5 square meters among point clouds with a density of 10 points per cm. In order to obtain the desired result and optimize the outcome, a fine tuning of selected control variables must be configured. In particular, within the horizontal sweep the step width is set to 0.05m and the suppression threshold that determines if a peak is eligible for representing a wall or not only requires a minimum of 550 points to confirm a vertical structure. These extremely low values are justified by the transformation of the original sub-cloud into a voxel space that is created ad-hoc to reduce the high density of the laser measurements. The steps towards the computation of a voxel space from a dense point cloud are described below. The model generated from the sub-cloud transformed into a voxel space is shown in Figure 5.22.

Figure 5.21: Floor plan cell decomposition by cut-line intersection of a complex and dense point cloud. The cut lines are localized by the sweep planes along the wall directions and the ground plan is computed by the union of floor cells.

Voxel space

A voxel space is a domain built by a set of 3D cubic cells in a Cartesian coordinate system with axes defined by the vectors $\mathbf{x}_v$, $\mathbf{y}_v$ and $\mathbf{z}_v$. A voxel represents the unitary element of the voxel space and is computed after a transformation of the point cloud into a grid structure. Such a transformation assigns multiple 3D points to the same voxel in order to decrease the global point density of the initial data. The voxel side size is given to allow for the estimation of further voxel space parameters in the way explained in the following text.

The first step toward the calculation of the voxel space is to define the dimension of the point cloud and consequently the voxel space size as:

$$x_s = |x_{max} - x_{min}|, \quad y_s = |y_{max} - y_{min}|, \quad z_s = |z_{max} - z_{min}|$$

where the second terms of the above equations give the extents of the point cloud along the three main directions. Then, the number of cells relative to each axis is computed dividing the voxel space size by the single voxel side size s (constant in all directions) and taking the integer part:

$$n_x = [\frac{x_s}{s}], \quad n_y = [\frac{y_s}{s}], \quad n_z = [\frac{z_s}{s}].$$

In the second case study the value of s is set to 10cm, thus a voxel occupies a volume of 10cm^3. The new origin of the voxel space coordinate system coincides with the bottom left corner of the voxel space (Figure 5.23). Therefore, all the coordinates of the points in the point cloud become

$$x_v = x - x_{min}, \quad y_v = y - y_{min}, \quad z_v = z - z_{min}.$$

All points (x_v, y_v, z_v) are checked in order to assign each of them to a precise voxel in the voxel space. The indices that localize the voxel which a point belongs to are given by

$$i_x = [\frac{x_v}{s}], \quad i_y = [\frac{y_v}{s}], \quad i_z = [\frac{z_v}{s}]$$
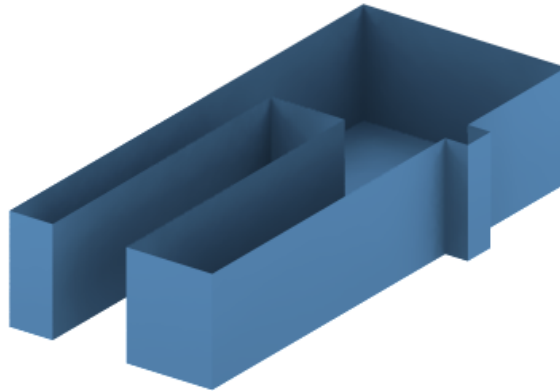
Figure 5.22: 3D model of the sub-cloud extracted from the dense point cloud representing a building floor. The whole point cloud was down-sampled through the computation of a voxel space.

along $\mathbf{x}_v$, $\mathbf{y}_v$ and $\mathbf{z}_v$, respectively. Likewise, an index in 3D is also computed for directly accessing all cells:

$$k = i_x n_y n_z + i_y n_z + i_z. \tag{5.2}$$

If at least one point is localized in a specific cell with index (5.2), that cell is designated to be the domain for a single 3D point in the new low-density point cloud. Usually, more than one point is indexed to the same cell with the result that several measurement points are allocated to the same voxel. However, due to the necessity of resizing the original data for reducing the point density, by definition the voxel should contain only one point. The coordinates of this point are calculated by approximating the coordinates of all points inside the voxel to the centre of the cell. The final model accuracy is affected by this approximation besides being influenced by the resampling of the point cloud, which causes a removal of a high percentage of points. A discussion about the accuracy evaluation is given in a dedicated section.

Figure 5.23 is an illustration of the voxel space and its parameters. The voxel space is shown with reference to the cartesian axes of the registered point cloud (in grey in the figure) and the new axes $\mathbf{x}_v$, $\mathbf{y}_v$ and $\mathbf{z}_v$. These are obtained with a translation of the coordinate origin to the bottom left corner of the voxel space. The last image represents the distinction between the cells that contain measurement points and the cells that are empty thus do not contribute to the generation of points for the sparse point cloud.

5.2.4 Variable density point clouds

This section is dedicated to special acquisitions that were performed by sensors different from a laser scanner. The point clouds obtained are not characterized by a dense sampling density as the data acquired by the laser scanner in the previous case studies. The imaging devices used to record the indoor scenes require a reduced preliminary network design and in general less effort in terms of time spent for the measurements. Therefore, the results of these particular acquisitions are very interesting for the extension of the automatic modeling method to further scopes and applications. For instance, the use of devices which can only deliver sparse data but are more portable and less expensive than a laser scanner could make the modeling process suitable for mobile applications in the area of indoor navigation. The assumption of considering a Manhattan-world domain still applies.
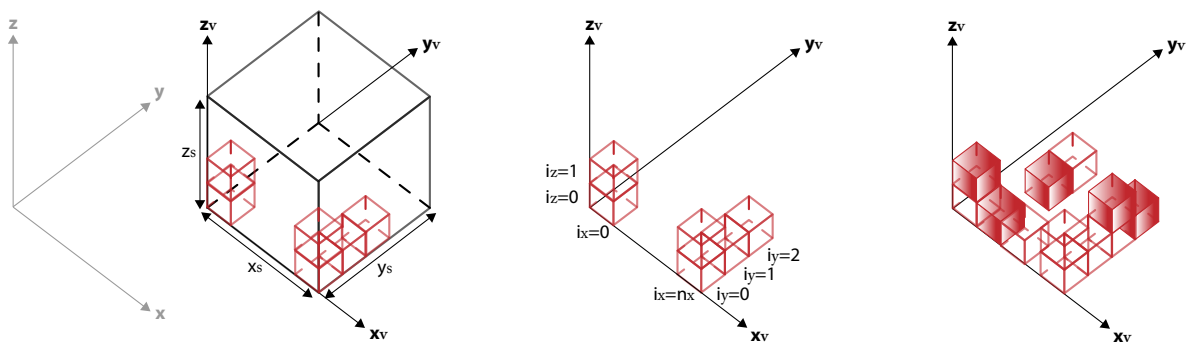
Figure 5.23: Representation of the voxel space and its parameters. Firstly, the voxel space is calculated in accordance with the volume of the point cloud. Secondly, the voxels that collect 3D points are selected for the determination of a low-density point cloud.

5.2.4.1 Point clouds generated by 2D camera acquisitions

In this paragraph it is described how the data acquired by a digital camera mounted on a mobile platform are used as inputs for the reconstruction algorithm. The results of the modeling process are shown. The camera is part of a mobile robot which also includes an electro-optical sensor and a computing unit that runs a reconstruction software. The system design as well as the image acquisitions have been conceived and executed within the work of Fietz [2011]. The whole system is moving within the interior scene according to the shortest possible route (traveling salesman problem) and capturing 2D images from a predefined number of standpoints. For the calculation of the standpoints, i.e. the camera poses, photogrammetric positioning techniques are applied. A picture of the mobile robot designed in [Fietz, 2011] is given in Figure 5.24b.

The camera is equipped with a Pentax Cosmicar Lens 4.2mm and is able to deliver color images detecting the scene within a field of view of 80 degrees horizontally by 65 degrees vertically. The integrated CMOS sensor has a size of 2,592x1,944 pixels, each pixel with a size of 2.2μm. The 2D intensity images captured by the digital camera contain feature points that become the inputs for a bundle block adjustment. By point correspondences, the best camera parameter estimates are calculated and the 3D positions of all points observed in the set of images is found [Fietz, 2011]. Consequently, a 3D point cloud is available for being tested by the automatic reconstruction software. Differently from the point clouds already tested in the previous case studies, the 3D data generated by the images taken from the mobile robot are characterized by a discontinuous point density. Areas with a high amount of points alternates with empty areas. That derives from the computation of the point correspondences: only the areas around the feature points are matched to be localized in the 3D space. Besides, only points relative to the wall measurements are available, while floor and ceiling points are missing (see Figure 5.25). Therefore, in order to enable the point cloud to become a suitable input for the modeling algorithm, synthetic points are added at the floor and ceiling heights. Alternatively, the values of the floor and ceiling level could be added manually in a pre-processing phase.
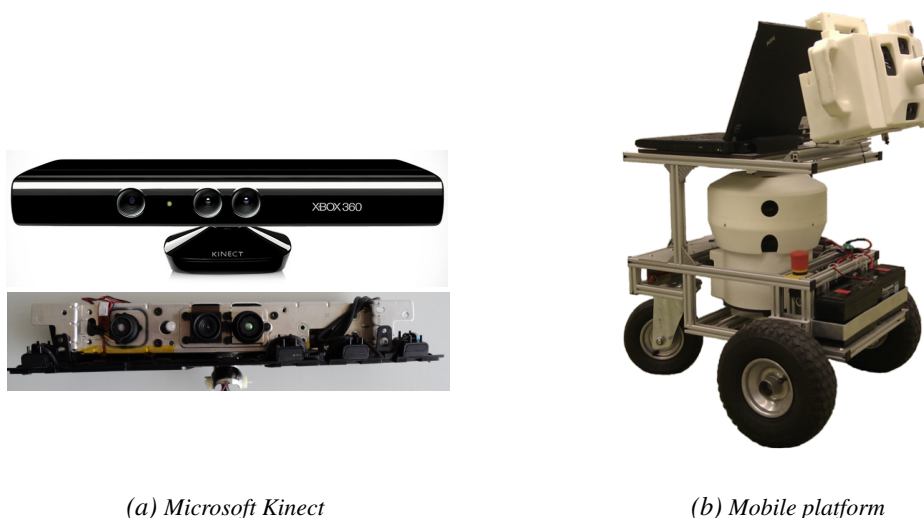
(a) Microsoft Kinect   *(b) Mobile platform*

Figure 5.24: Alternative sensors used for the acquisition of interior scenes: (a) the Kinect system by Microsoft as it is sold (top) and disassembled by Haala et al. [2011] (bottom); (b) a mobile platform equipped with a color camera developed by Fietz [2011].

The test data concern the 3D measurements of a single room. Including the points that are generated synthetically, the resulting point clouds consists of 21,637 points, of which only very few are distributed on the
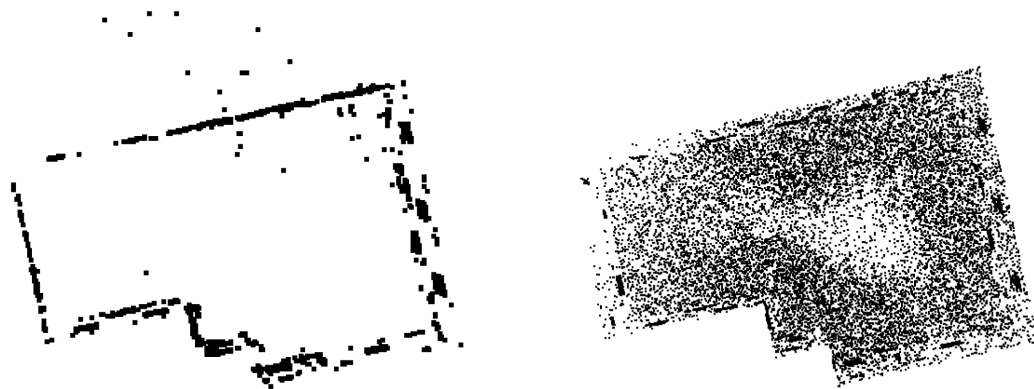
Figure 5.25: Top view of the point cloud generated by image correspondences (left) and same point cloud after the addition of floor and ceiling points (right).

walls. It is important to test this data set due to the particular point acquisition technique which differs from the standard acquisitions with the laser scanner. The analysis of the results starts with the computation of the rotational sweep since the necessity is eliminated to perform the previous vertical sweep for the calculation of floor and ceiling heights. Now these values are given as inputs, thus they are known a priori. The wall orientation angle is found at 103.09 degrees with respect to the camera system. The two horizontal plane sweeps that are performed separately along the wall directions partition the ground into 15 rectangular cells. In particular, four and six cut-lines are found, respectively. The ground plan is built from the union of 12 cells and the model in Figure 5.26 is computed accordingly. Since the point cloud is characterized by sparse measurements on the walls, some of the control variables of the horizontal sweeps should be fitted to the new data. First, the step width is lowered to 8.5cm, whereas the standard step width used in most of the case studies is 10cm. A lower step width allows for a finer detection of potential vertical structures. Second, the suppression threshold is lowered of one order of magnitude, since the number of wall points is also considerably reduced with respect to the amount of points available from the laser acquisitions. It can be observed that the values chosen for the control variables in this example are very similar to the settings adopted in the particular Example D of the hallway case study (see Table 5.2).

The last step towards the computation of the full model should be the detection of potential doors. Anyway, due to the clustered point distribution on the walls where the doors are expected to be localized, the search for data gaps in the data does not produce any results. A detailed representation of the sparse data points in correspondence to a wall can be observed in Figure 5.26, in which the cloud and the model are shown together.

5.2.4.2 Point clouds acquired by a low-cost sensor

In this section another special application of the automatic modeling is described. In particular, the point clouds that are the inputs of this case study are generated by range measurements with the low-cost sensor Kinect from Microsoft. The point cloud acquisitions are described within the work of Haala et al. [2011], whose main goal is the pedestrian indoor navigation supported by 3D models that are computed from photos of evacuation plans.

The Kinect sensor is provided by Microsoft as an imaging device that interfaces the user and its environment with the Xbox 360 video game console. It implements 3D motion capture beside voice recognition

features. In order to enable these capabilities, it is equipped with a two-side 3D camera, a central RGB camera and a multi-array microphone (see Figure 5.24a). The 3D camera serves as a depth sensor with two main components: an infrared (IR) projector and a monochrome CMOS sensor. The depth is captured by projecting a dot pattern of light onto the target similarly to structured light techniques mentioned in Chapter 2. The CMOS camera then reads the projected infrared dots at a frame rate of 30 frames per second and computes a 11-bit Video Graphics Array (VGA) image with a typical resolution of 640x480 pixels. Every pixel in the depth image is aligned to a pixel in the color images acquired by the RGB camera at 8-bit VGA resolution. The sensor has an angular field of view of 57° horizontally and 43° vertically. In combination with the Xbox software, it has a practical range in the interval from 1.2 to 3.5m.

The dot pattern projected by the IR source are matched with the corresponding pixels in the IR images captured by the CMOS camera. The correspondences are used to compute the 3D coordinates of the target object surface and thus to generate a 3D point cloud. However, this process can only generate point clouds that are limited in size by the field of view of the imaging sensor. The point cloud of a whole room should be computed as a combination of range images from multiple viewpoints. A discussion on the technique used for the co-registration of a sequence of point clouds by means of both RGB and range images can be read in [Haala et al., 2011].

Suitable point measurements that are very similar to laser scanner point clouds are provided by the output of the co-registration for being processed by the automatic modeling algorithm. Minor differences are remarkable between the two types of point clouds. First of all, since the acquisition with Kinect was intentionally performed just to collect points on the walls, the data resulting from Kinect range images, similarly to the point clouds acquired by the mobile robot, are not provided with floor and ceiling points. Therefore, after being aligned, they should be integrated by the addition of some extra points at the floor and ceiling heights, whose values are directly recovered on the basis of the coarse 3D model in [Haala et al., 2011]. It is assumed that the original evacuation plan from which the coarse model is computed was not subjects of changes over the time, thus providing correct values of floor and ceiling level. A further observation may be done regarding the orientation of the point clouds in the two situations. Before starting the measurements, the laser scanner is manually leveled so that it can measure – after registration – a point cloud which is already aligned with the Z axis of a Cartesian reference system. On the other hand, the point cloud collected by the range sensor is not leveled in the field, but only after its acquisition has taken place. With these
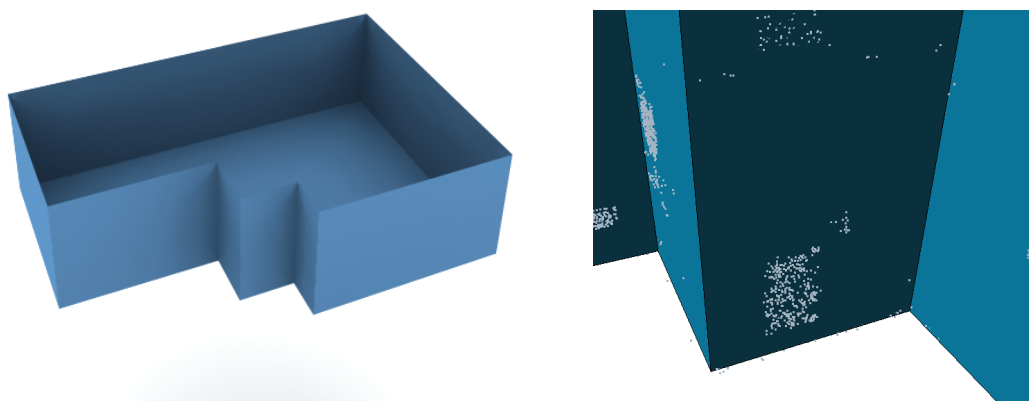
Figure 5.26: 3D model of indoor measurements generated by digital camera acquisitions (left) and particular of the same model overlapped with the point cloud (right). The detailed view shows how the points are sparsely distributed within small areas on the walls.
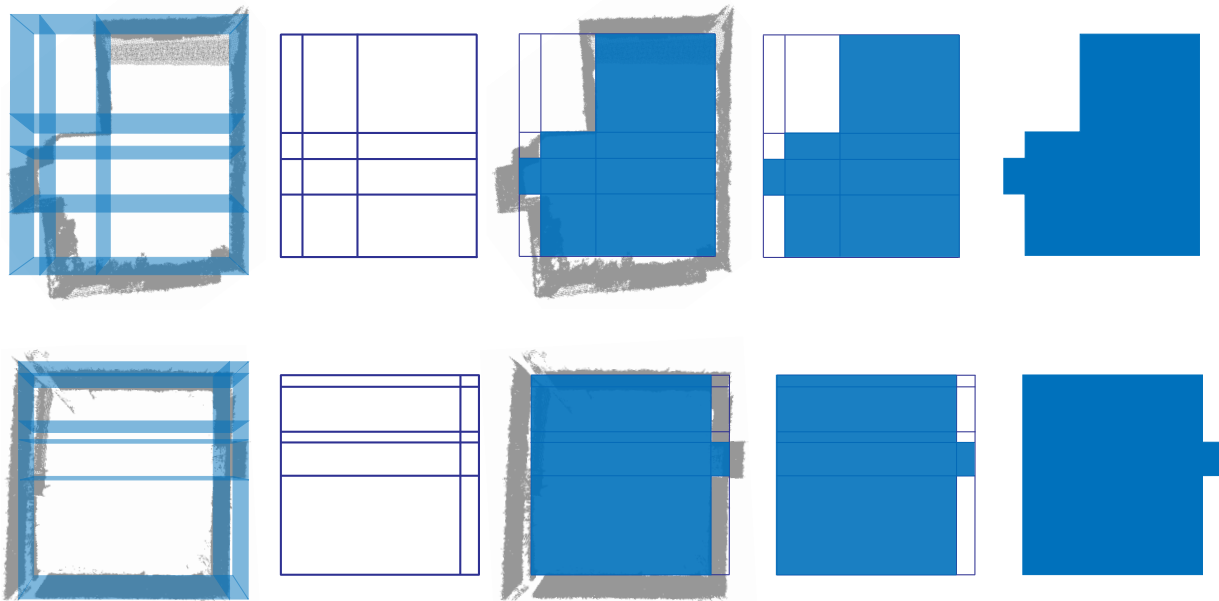
Figure 5.27: Floor plan cell decomposition by cut-line intersection of rooms A (top) and B (bottom). The cut lines are localized by the sweep planes along the wall directions and the ground plan is computed by the union of floor cells.

considerations, the reconstruction method can deliver optimal results from range data as well. The models obtained are comparable to those generated from laser point clouds. The full point clouds of two separate rooms are analyzed.
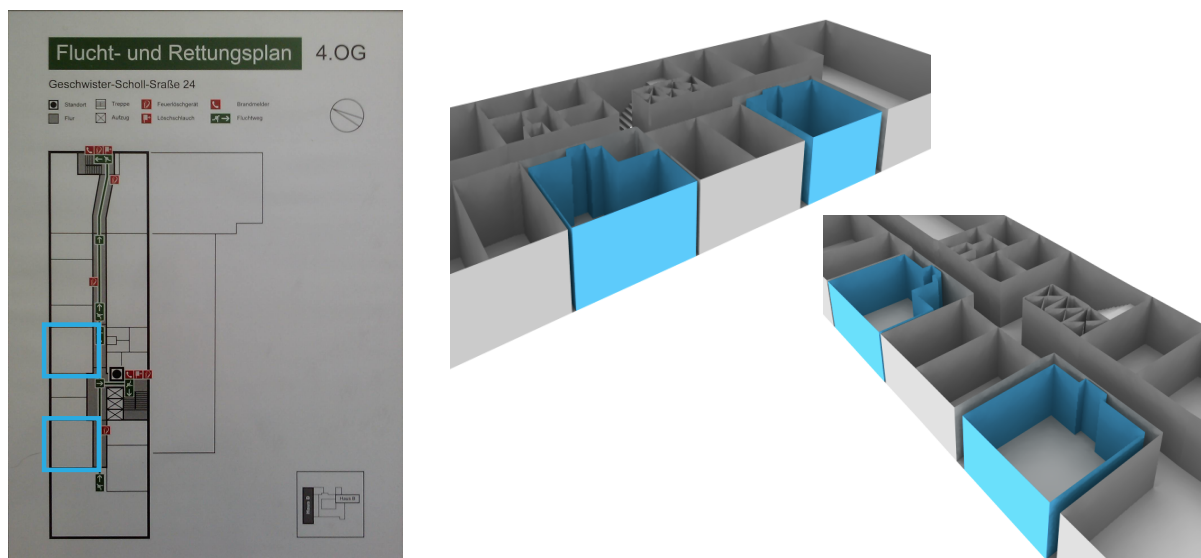
Figure 5.28: Automatically-generated models of room A and room B (in blue) embedded in the context of the comprehensive coarse model computed by Haala et al. [2011] (right). The coarse model is calculated on the basis of an intensity image of the evacuation plan (left).

Room A

Both room A and B belong to the same floor of a university building, therefore they are characterized by similar architectural features. The most interesting aspect is the presence of built-in cupboards whose panel doors hide inner walls making it impossible to extract and define the underlying as-built room geometry. Nevertheless, the results of the reconstruction based on plane sweep might be compared with the corresponding models generated by the analysis of evacuation plans described in [Haala et al., 2011]. Due to the similarity of the evacuation plans to the original design drawings, a model that is consistent with the elementary building structure is available. At the same time, the automatic modeling allows for a reconstruction of the interior building geometry relative to the current conditions, including potential structural modifications. By the comparison of the models obtained by the two techniques, apparently invisible changes in the indoor layout could be detected.

The point cloud of room A consists of about six hundred thousand 3D points. The floor and ceiling level are indirectly extracted from the coarse model at -1.119m and 1.5631m, respectively. The height of the room (similarly to the whole corridor) results 2.675m. The consensus distance set for the detection of the horizontal structures is 5cm, which corresponds to the smallest value adopted for such a variable in the case studies. A small consensus distance in fact ensures high accuracy in the detection of floor and ceiling points that are generated synthetically from the coarse model.

Since the detection carried out by the vertical sweep is supported by a pre-computed 3D model, the goal of this example mainly focuses on the analysis of the results of the following reconstruction step. The overall model computation is considered with particular attention to the wall positions defining the ground plan contour. The consensus distance is kept constant to 5cm. Theoretically, the horizontal sweep might be affected by a large number of missing points at the junction between the floor and the walls and also by several pieces of furniture present in the room during the acquisition. In practice, the computation of the wall positions and orientations yields good results and the walls can be detected correctly thanks to a relatively low value of the suppression threshold within the two horizontal sweeps. The minimum number of points required in order to specify a histogram peak as a real wall is six thousand along one direction and ten thousand along the perpendicular direction. This means, each wall is supposed to contain at least one percent of the points in the whole point cloud.

The rotational sweep computes the main linear sweep direction at an angle of 91.64 degrees. Although in the reality the scene has an orthogonal layout, the consistency with grid-like perpendicular walls is not proved in the measured point cloud due to alignment errors in the co-registration process. This errors are then eliminated during the model generation by the assumption of referring to a Manhattan-world domain.

The ground plan is defined by five and four cut-lines along the two Manhattan-world directions, respectively. Their intersection generates 12 ground cells but only eight of these are confirmed as ground cells. Figure 5.27 shows that the computed cut-lines "straighten" the whole room topology, which was not true to the reality of the scene in the aligned point cloud. The final model obtained is shown alone in Figure 5.29a, while it is inserted into the building floor coarse model computed by Haala et al. [2011] in Figure 5.28. This figure also shows the accuracy of the sweep model with respect to the architectural context to which it belongs. The wall orientation appears to be correct despite the misalignment between point cloud and cut-lines visible in the Figures 5.27 and 5.30.

Room B

The example B concerns the reconstruction of an office room. The data acquisition and pre-processing result in a point cloud of the same order of magnitude and the same density of the point cloud in A. Precisely, the total number of points involved in the reconstruction amounts to 590,378. The floor and ceiling levels are slightly higher than those in the previous example, but their distance is kept constant. However, similarly to room A, also for the reconstruction of the horizontal structures in room B the algorithm makes use of pre-computed data. The final model of room B can be observed in Figure 5.29b.

The complexity of the room contour is lower, thus a lower number of floor cells is found. The consensus distance assigned to the sweeps ranges from 5cm for the vertical sweep to 6cm. Besides, the suppression threshold set for the horizontal sweep along the main direction allows a minimum number of only hundred points for the acknowledgment of a vertical plane. Along the second direction, the value of the suppression threshold is two orders of magnitude higher requiring exactly ten thousand points in correspondence to a wall peak. From the rotational sweep, it results that the main direction is aligned at an angle of about 85 degrees to the co-registration axis. With the above control variables, three and six cut-lines are detected by the linear horizontal sweeps. That leads to the partition of the floor into ten cells, the half of which belong actually to the ground plan.

(a) Room A　　　　　　　　　　　　　　　(b) Room B
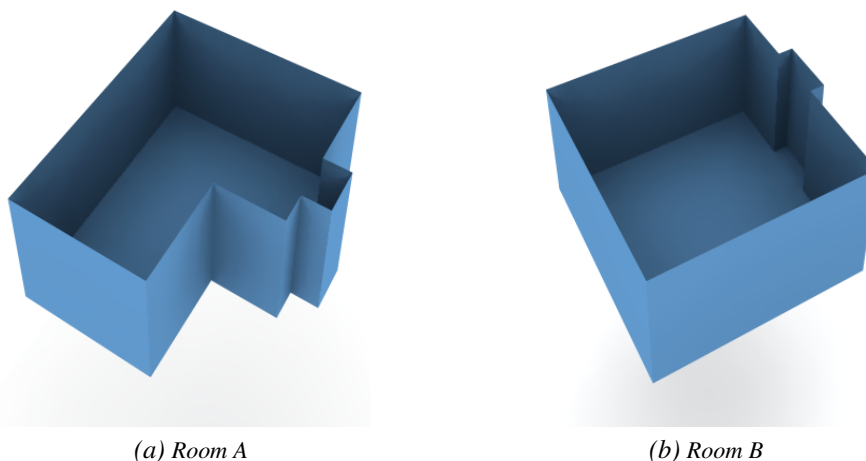
Figure 5.29: Models obtained through the application of the automatic reconstruction technique using as inputs the sub-cloud of the rooms (a) A and (b) B.

Once the rooms have been separately reconstructed with the above method, the integration of the two single models into the coarse model shows a seeming partial mismatch of the results in terms of the room sizes (Figure 5.28). Especially the volumes occupied separately by the two room models appear small-sized compared with the corresponding volumes in the coarse model. As mentioned before, this is due to updates in the interior building configuration such as the addition of integrated pieces of furniture that were not yet included in the original evacuation plan. Apart from the presence of built-in cupboards, which re-size the room geometry with respect to the former ground plan, the impossibility of reconstructing the wall depth prevents the two types of models from being fully corresponding.

The execution of this example also provides for the detection of a door. The latter coincides with the only door present in the office, namely the entrance door of the room. The wall where the door is localized is entirely furnished with cupboards with the result that, as explained above, the reconstruction takes place

Figure 5.30: Top views of the point clouds of room A and room B with the corresponding 3D models.

at the level of the cupboards and not of the walls. The door belongs to the same plane of the room walls, therefore a gap can be localized at few centimeters from the wall between the two cupboards also when the door is not open. The minimum width required for the detection of the door in this example is the standard width 80cm.

5.3 Evaluation

In order to provide a numerical evaluation of the method proposed, regression planes are computed through the moment method and compared to their corresponding sweep planes. The moment method uses a least squares optimization technique which minimizes the orthogonal distances between measured points and the model feature by applying an orthogonal distance fitting (ODF) [Ahn, 2004]. In particular, the moment method is a special case of ODF for linear problems. A comprehensive analytical overview of the moment method for plane fitting can be read in Appendix A, whereas just a brief explanation with reference to the automatic reconstruction is given below. The input of the fitting is a given number of selected 3D points.

Once the center of mass of the observation points is known, the calculation of a central moment matrix and its singular value decomposition (SVD) allow for an efficient minimization of the square sum of orthogonal distances (residuals). Three subsequent steps depicted in Figure 5.31 lead to the general solution of the fitting. Firstly, by a point segmentation that is driven by the knowledge of the plane positions, (planar) portions of the scanned data are extracted and their centers of mass are calculated separately. The centers of mass will be used for the localization of the regression planes since the best regression plane of a point
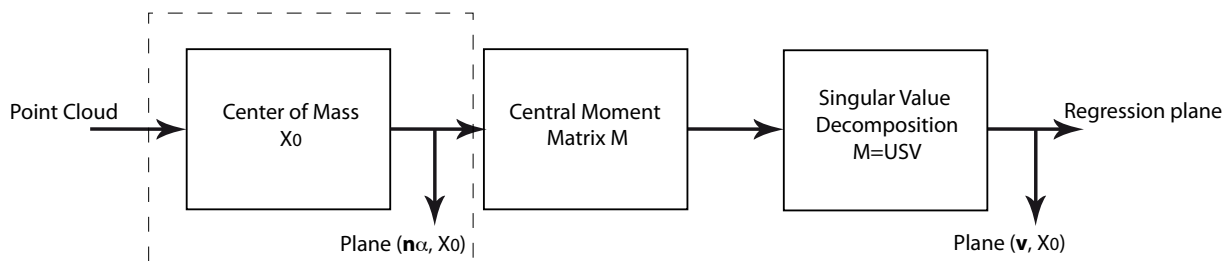
Figure 5.31: Visualization of the three main steps of the evaluation method based on SVD.

distribution always contains the center of mass. Secondly, the 3x3 central moment matrix is derived for each portion of the data by the evaluation of the variance of the point coordinates along the axes. Finally, the last step yields the factorization of the central moment matrix, calculated by SVD.

The SVD returns a 3x3 diagonal matrix with non-negative diagonal elements corresponding to the eigen-values of the central moment matrix. They are a measure of the stretching, i.e. the variance of the selected points along the principal directions. The columns of the two unitary matrices **U** and **V** in A.8 return an orthonormal basis of eigenvectors that coincide with the directions along which the point cluster is distributed. In particular, the eigenvector associated to the smallest eigenvalue gives the direction of the normal vector to the regression plane. In the following section, it is described how accuracy values of the vertical and the rotational sweeps are calculated with reference to the regression planes derived by ODF.

5.3.1 Accuracy of the horizontal sweep planes

In the case of the examples described in this chapter, the sweep planes that require an evaluation based on ODF belong to two separate categories: the horizontal planes and the planes of the walls. Let us first consider the horizontal – floor and ceiling – planes. Due to the assumption about the alignment of the instrument to the Z axis, the sweep vector for floor and ceiling segmentation is univocally determined by the vertical direction. That means, the accuracy of the sweep planes along Z should only be verified in terms of sweep position, whereas the orientation is fixed a priori. To do that, the difference in positions between the planes previously computed by the sweep and the regression planes is estimated along the vertical direction. Given the normal vector (that should not be calculated again through SVD), the regression planes can be computed by ODF from the center of mass of a selected point cluster. The points chosen for each ODF are selected in the proximity of either the floor or the ceiling plane. Table 5.3 and Table 5.4 show some results of the ODF in comparison to the fitting based on the linear sweep. The accuracy is computed as the orthogonal distance between the regression plane an its correspondent sweep plane and it is characterized in the optimal case by values in the order of millimeters. The values indicated in Table 5.3 refer to the examples from A to D described in the case study of the long underground hallway. The two sub-clouds additionally extracted in the examples C and D are not considered here since they are just extensions of the above examples, thus they present similar reconstruction characteristics. The results of the ODF of horizontal planes within the single sub-cloud tested in the case study of the complex university building are listed in Table 5.4.

Example	Floor	Orthogonal Distance (m)	Ceiling (m)	Orthogonal Distance (m)
A	-1.581	0.0208	0.944	0.0379
B	-1.617	0.0235	0.983	0.002
C	-1.624	0.0109	0.951	0.0322
D	-1.568	0.0214	1.331	0.0125
Mean	-1.598	0.019	1.052	0.0212

Table 5.3: Accuracy of the vertical sweep planes for the first case study. The examples from A to D are considered.

Example	Floor	Orthogonal Distance (m)	Ceiling (m)	Orthogonal Distance (m)
–	-1.653	0.01	1.872	0.0013

Table 5.4: Accuracy of the vertical sweep planes for the second case study. A single example is considered.

5.3.2 Accuracy of the rotational planes

The approach to the evaluation of the wall accuracy differs in several aspects from the accuracy estimation of floor and ceiling. The first difference concerns the parameter measured for the comparison of the sweep plane with the regression plane. Since an error originated by the rotational sweep is expected to affect the wall orientations, the accuracy is calculated as the angle between the sweep vector and the regression plane normal. Thus, it will be expressed by an angular measure – in degrees – and not by a distance, as before. A further point of dissimilarity is the additional computation required to obtain the normal vector to the regression plane. The direction of such a normal vector in fact is not known, therefore it should be recovered by SVD.

The last distinction between the two implementations of the accuracy evaluation is the method used for selecting the points for the ODF. The set of initial points, which is used to estimate the central moment matrix and to compute its factorization, is not chosen on the basis of predefined plane positions as this would cause potential sweep errors to be inherited by the resulting accuracy value. Hence, in order to select suitable wall points for the fitting, an approach similar to region growing is preferred. That is, a seed point is chosen randomly among the points previously measured on the walls so as to support a region-based extraction of initial 3D points. The region includes all points in the neighborhood of the seed point and defines the input for the calculation of the matrix of central moments. Since the random seed point might be positioned anywhere on the wall, a verification of the region extracted for the fitting is needed to avoid point distributions that could lead to an erroneous regression plane. For instance, if the region defined by the seed point neighboring area includes points on a pair of orthogonal walls, the regression plane resulting from the ODF is expected to be positioned at 45 degrees from the correct orientation (of one of the two walls), thus it would result in a wrong estimation of the final accuracy value. The results of the accuracy estimation in the case of the rotational planes are shown in Table 5.5 and Table 5.6 separately for the two case studies.

Example	Wall Angle β (degrees)	$\sin\beta$	$\cos\beta$	Angular Difference (degrees)
A	171.82	0.1421	0.9898	2.96
B	81.82	0.9898	0.1423	7.21
C	0	0	1	2.67
D	3.27	0.0569	0.9984	5.74
Mean	-1.598	–	–	4.65

Table 5.5: Accuracy of the rotational planes for the first case study. The examples from A to D are considered.

Example	Wall Angle β (degrees)	$\sin\beta$	$\cos\beta$	Angular Difference (degrees)
–	44.18	0.6969	0.7171	9.89

Table 5.6: Accuracy of the rotational planes for the second case study. A single example is considered.

5.3.3 Accuracy evaluation supported by Cyclone

Another test has been conducted in order to validate the robustness of the automatic reconstruction. This test concerns a further analysis on the correctness of floor and ceiling positions. In particular, the sweep planes that represent separately the floor and the ceiling are compared to synthetic planes generated with the help of the software Cyclone. Cyclone indeed is a modeling tool as well, thus it provides the user with a set of standard shape selection for different types of 3D fitting. By using Cyclone function for region growing, a patch can be (semi)automatically fitted over a selected surface. To implement that, three picked points that are supposed to lie on the same flat wall surface are manually provided to the software which computes automatically a least squares adjustment. The comparative outputs of the fitting calculated by Cyclone on the one side and by the sweep method on the other side are summarized in Table 5.7, where a sample sub-cloud containing more than 750 thousand points is considered. The analysis of the results is done by comparing

Fitting	Floor (m)	Ceiling (m)	Difference (m)
Cyclone LS	-1.599	0.988	2.587
Plane Sweep	-1.55	0.899	2.449
Correction	-1.6	0.975	2.575

Table 5.7: Comparative results of a least square fitting computed by Cyclone and the planes found with the sweep method.

an absolute measurement that is the difference in height between floor and ceiling. The output of Cyclone least squares exhibits values comparable with both outputs of the basic sweep and of the sweep enhanced by the peak correction. However, it should be noticed that the distances floor-ceiling computed respectively with the least squares and the correction have a reciprocal uncertainty of 1.2cm whereas an uncertainty of 13.8cm is observed in the case the sweep without correction is applied.

Chapter 6

Conclusions

The motivation for the presented dissertation originated from the intention of improving the degree of automation in the generation of interior building models. According to this purpose, the work developed aimed at investigating a new method for the automatic reconstruction of indoor scenes at LOD4. The thesis described the steps that lead to the computation of interior CAD models and the requirements for a successful reconstruction. In its initial development, the presented approach was supposed to model scenes that were exclusively acquired with a laser scanner. However, it was demonstrated with different practical examples that the modeling method also exhibits a reproducibility of optimal results in the case of input data provided by a low cost sensor. Whereas the laser scanner directly delivered a 3D point cloud, the low cost sensor provided range images that needed to be post-processed in order to give consistent 3D measurements of a whole room. The adaptability of the approach to diverse measurement systems might allow for applying the method to multiple scenarios.

For the implementation of the automatic modeling, two main phases were proposed, which were executed in sequence. The first phase consisted in the segmentation of the data based on plane sweep. The primitive-surfaces representing the indoor structures were matched to sweeping planes to recover each time the surface positions. Linear sweeps perpendicular to the Manhattan-world directions were computed to determine the position of the floors, ceilings and walls, while rotational sweeps supported the computation of the surface orientations if these were not defined in the problem domain. In a second phase, the floor plan was computed by cell decomposition according to the analogous technique for solid modeling. Half-space modeling was used to divide the floor plan into rectangular cells by splitting the floor surface using straight lines. Besides, interior doors were detected by additional computation. In order to recover the topology of the indoor scene, the only information available was carried by the coordinate information contained in the point cloud. The independence of the modeling program from extra inputs makes the approach suitable for architectural reconstruction of existing buildings only relying on on-site measurements.

The results of different case studies showed how, unlike least squares concepts, the developed modeling method is robust to the presence of outliers due to the automatic allocation of each measured point to a discrete surface. This allows a reliable computation of the room topology despite the unavoidable existence of noisy data.

6.1 Future work

The major open issue concerning the research carried out in this thesis is the specification of the reconstruction domain. The implemented algorithm is effective for Manhattan-world scenes presenting an orthogonal

layout. This requirement sets a priori the orientation of the walls to fixed perpendicular directions. Hence, as a further development of the automatic modeling, a generalization of the reconstruction domain would guarantee a higher efficiency among a wider variety of situations. A possible realization of the modeling method extended to a number n of arbitrary wall orientations consists in the improvement of the rotational sweep iterations. By increasing the numbers of rotations about given vertical axes and monitoring the histogram peaks, additional orientations could be directly detected from the histogram. Similarly, the reconstruction of multiple high ceilings might be also supported by an enhanced version of the vertical sweep.

Another potential development for the method proposed is the combination of adjacent interior models into a unified representation. So far, different sub-clouds extracted from the same registered point cloud are processed separately, thus they result in separate models. However, if these models correspond to adjacent portions of the data, it could be useful to provide a joint reconstruction that is computed from the union of the single models. The advantage of reconstructing separate models – and then merging them together if proximity constraints apply – would be the reduction of the rotational sweep error that is caused by the reconstruction of very long corridors.

In order to make the method completely independent from the type of architecture to be reconstructed, the modeling approach should be targeted to a wide range of different interiors, each one requiring an ad-hoc implementation based on the interior features. Such features could include for example arches, columns or other kind of round architectural elements. The latter could be stored in a feature library. The global reconstruction would then be responsible for distinguishing the type of interior and would activate the correspondent function that is specifically developed for the type of interior detected. The final model could be also enhanced with fine details and textures if intensity images are also available.

Appendix A

Moment method

With reference to the work of Ahn [2004], the moment method for plane fitting is reviewed. The moment method gives the particular solution to a general ODF problem in the linear case. ODF refers to a least squares model fitting with an orthogonal distance as an error measure: it evaluates the model error by computing the shortest distance – orthogonal distance – between the measurements and the model itself.

A.1 Plane fitting

The implicit form of the plane that passes through the point $\mathbf{X}_0$ and is perpendicular to the unit vector $\mathbf{n}$ is determined by the equation

$$(\mathbf{X} - \mathbf{X}_0)^T \mathbf{n} = 0. \tag{A.1}$$

Given a set of n measurement points in a three-dimensional space, the square sum of the orthogonal distances between each point $\{\mathbf{X}_i\}_{i=1}^n$ and the model described by (A.1) is

$$\sigma_0^2 = \sum_{i=0}^n \left[(\mathbf{X}_i - \mathbf{X}_0)^T \mathbf{n} \right]^2 = \mathbf{n}^T \left[\sum_{i=0}^n (\mathbf{X}_i - \mathbf{X}_0)(\mathbf{X}_i - \mathbf{X}_0)^T \right] \mathbf{n}, \tag{A.2}$$

where

$$\mathbf{X}_i = \begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix}, \quad \mathbf{X}_0 = \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix}.$$

In order to fix the value of $\mathbf{X}_0$ that determines the best model, the square sum of orthogonal distances (A.2) is minimized:

$$\frac{\delta \sigma_0^2}{\delta \mathbf{X}_0} = -2 \left(\sum_{i=0}^n (X_i - X_0), \sum_{i=0}^n (Y_i - Y_0), \sum_{i=0}^n (Z_i - Z_0) \right) \mathbf{n}\mathbf{n}^T = \mathbf{0}^T. \tag{A.3}$$

If $\mathbf{n}$ has an arbitrary direction that is not parallel to a coordinate axis, the solution of (A.3) becomes

$$\sum_{i=0}^n (X_i - X_0) = \sum_{i=0}^n (Y_i - Y_0) = \sum_{i=0}^n (Z_i - Z_0) = 0 \tag{A.4}$$

Equation (A.4) yields the coordinates of the point (X_0, Y_0, Z_0) (parallel axis theorem [Beer and Johnston, 1997]):

$$X_0 = \overline{X} = \frac{1}{n} \sum_{i=0}^{n} X_i, \quad Y_0 = \overline{Y} = \frac{1}{n} \sum_{i=0}^{n} Y_i, \quad Z_0 = \overline{Z} = \frac{1}{n} \sum_{i=0}^{n} Z_i, \quad (A.5)$$

which coincides with the centre of mass of the point cloud. Therefore, the best model results to be the plane that contains the centre of mass $(\overline{X}, \overline{Y}, \overline{Z})$. If $\mathbf{n}$ is parallel to one of the coordinate axes, the value of $\mathbf{X}_0$ may be set along this axis so that the solution (A.4) still applies.

By defining the distances between the i-th point and the center of mass

$$x_i = X_i - X_0, \quad y_i = Y_i - Y_0, \quad z_i = Z_i - Z_0$$

and setting accordingly the second central moments about $\mathbf{X}_0$ – variances between the points and $\mathbf{X}_0$ – along the axis

$$M_{xx} = \sum_{i=0}^{n} x_i^2, \quad M_{yy} = \sum_{i=0}^{n} y_i^2, \quad M_{zz} = \sum_{i=0}^{n} z_i^2,$$

$$M_{xy} = \sum_{i=0}^{n} x_i y_i, \quad M_{xz} = \sum_{i=0}^{n} x_i z_i, \quad M_{yz} = \sum_{i=0}^{n} y_i z_i,$$

the symmetric square matrix $\mathbf{M}$ (central moments tensor [Beer and Johnston, 1997]) is determined as follows:

$$\mathbf{M} = \begin{pmatrix} M_{xx} & M_{xy} & M_{xz} \\ M_{xy} & M_{yy} & M_{yz} \\ M_{xz} & M_{yz} & M_{zz} \end{pmatrix}. \quad (A.6)$$

Hence, the square sum of orthogonal distances (A.2) can be expressed by

$$\sigma_0^2 = \mathbf{n}^T \mathbf{M} \mathbf{n}. \quad (A.7)$$

A.2 Singular value decomposition

The SVD is used for the factorization of rectangular matrices of real numbers. It is used here for the decomposition of the square matrix $\mathbf{M}$, which becomes

$$\mathbf{M} = \mathbf{USV} \quad (A.8)$$

with

$$\mathbf{V} = \mathbf{U}^T \quad \mathbf{UV} = \mathbf{I}.$$

The result of the SVD is a product of matrices: the diagonal matrix $\mathbf{S}$ has the same dimension of $\mathbf{M}$ and contains nonnegative diagonal elements in decreasing order. It yields

$$\mathbf{S} = diag(\lambda_1, \ldots, \lambda_j) \quad | \quad \lambda_1 \geq \ldots \geq \lambda_j \geq 0$$

where $j = 3$, since $\mathbf{M}$ and $\mathbf{S}$ are 3x3 matrices in a 3D space. The elements on the diagonal of $\mathbf{S}$ are called singular values and correspond to the eigenvalues of $\mathbf{M}$. The columns of $\mathbf{U}$ and $\mathbf{V}$ are called left and right

singular vectors. The singular values and singular vectors satisfy

$$\mathbf{M}\mathbf{v}_j = \lambda_j \mathbf{u}_j, \tag{A.9}$$

in which $\mathbf{u}_j$ and $\mathbf{v}_j$ indicate the i-th columns of $\mathbf{U}$ and $\mathbf{V}$, respectively.

A.2.1 Regression plane

The eigenvalues can be interpreted geometrically as a measure of the point cloud distribution along the corresponding eigenvectors. The lowest distribution direction is related to the eigenvector with the smallest eigenvalue and coincides with the normal to the regression plane. Thus, the ODF problem is solved in 3D by choosing a regression plane that contains the centre of mass of the point distribution and has as a normal vector the eigenvector $\mathbf{u}_j$ associated with the smallest eigenvalue λ_j. Therefore, the regression plane resulting from the initial model (A.1) is defined by

$$(\mathbf{X} - \overline{\mathbf{X}})^T \mathbf{u}_j = 0. \tag{A.10}$$

A numerical example of the application of the moment method for plane fitting can be read in [Ahn, 2004].

List of Figures

List of Tables

Bibliography

Adan, A. and Huber, D. 2011. 3d reconstruction of interior wall surfaces under occlusion and clutter. In *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, pages 275–281.

Ahn, S. J. 2004. *Least-Squares Orthogonal Distance Fitting of Curves and Surfaces in Space*. PhD thesis, University of Stuttgart.

Alba, M., Fregonese, L., Prandi, F., Scaioni, M., and Valgoi, P. 2006. Structural monitoring of a large dam by terrestrial laser scanning. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36.

Alexander, C., Ishikawa, S., and Silverstein, M. 1977. *A pattern language*. Oxford University Press.

Amman, M., Bosch, T., Lescure, M., Myllyla, R., and Rioux, M. 2001. Laser ranging: a critical review of usual techniques for distance measurement. *Optical Engineering*, 40. doi:10.1117/1.1330700.

Autodesk 2011. Realizing the benefits of bim. In Autodesk Building Information Modeling. Document downloadable at `http://usa.autodesk.com/building-information-modeling/about-bim/`.

Bahmutov, G., Popescu, V., and Mudure, M. 2006. Efficient large scale acquisition of building interiors. In *Computer Graphics Forum*, volume 25(3), pages 655–662.

Becker, S. and Haala, N. 2008. Integrated lidar and image processing for the modelling of building facades. In *Photogrammetrie – Fernerkundung – Geoinformation PFG*, pages 65–81.

Beer, F. P. and Johnston, E. R. 1997. *Vector Mechanics for Engineers: Dynamics*. McGraw-Hill International. 3rd Edition.

Beraldin, A., Blais, F., Cournoyer, L., Godin, G., Rioux, M., and Taylor, J. 2003. Active 3d sensing for heritage applications. In *The e-Way into the Four Dimensions of Cultural Heritage Congress*.

Beraldin, J., Picard, M., El-Hakim, S., Godin, G., Valzano, V., Bandiera, A., and Latouche, D. 2002. Virtualizing a byzantine crypt by combining high resolution textures with laser scanner 3d data. In *Proceedings of the International Conference on Virtual Systems and Multi-Media*, pages 3–14. Gyeongju, Korea.

Bernardini, F. and Rushmeier, H. 2002. The 3d model acquisition pipeline. *Computer Graphics Forum*, 21:149–172.

Besl, P. and McKay, N. 1992. A method for registration of 3d shapes. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 14:239–256.

Besl, P. J. 1988. Active, optical range imaging sensors. *Machine Vision and Applications*, 1:127–152.

Biber, P., Fleck, S., and Duckett, T. 2005. 3d modeling of indoor environments for a robotic security guard. In *IEEE Conference on Computer Vision and Pattern Recognition – Workshop*.

Blackwell 2012 (accessed Mai 3, 2012). Constructive solid geometry. `http://www.cl.cam.ac.uk/~afb21/publications/masters/node23.html`.

Blais, F. 2004. Review of 20 years of range sensor development. *Journal of Electronic Imaging*, 13(1):231–240.

Brenner, C. 2004a. Building reconstruction from images and laser scanning. *International Journal of Applied Earth Observation and Geoinformation*.

Brenner, C. 2004b. Modelling 3d objects using weak csg. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume 35.

Budroni, A. and Boehm, J. 2010. Automated 3d reconstruction of interiors from point clouds. *International Journal of Architectural Computing*, 8:55–73.

Campana, S., Sordini, M., and Rizzi, A. 2009. 3d modeling of a romanesque church in tuscany: archeological aims and geomatics techniques. In *ISPRS Commission V Workshop 3D-Arch 2009*. Trento, Italy.

Cavagnini, G., Sansoni, G., and Trabeschi, M. 2009. Using 3d range cameras for crime scene documentation and legal medicine. *SPIE*. doi:10.1117/12.806191.

Chen, Y. and Medioni, G. 1991. Object modeling by registration of multiple range images. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 2724–2729. Sacramento, CA, USA.

Coughlan, J. and Yuille, A. 1999. Manhattan-world: compass detection from a single image by bayesian interference. In *IEEE International conference on Computer Vision*. Kerkyra, Greece.

CyArk 2011 (accessed December 1, 2011). Cyark. `http://archive.cyark.org/`.

Debevec, P., Taylor, C., and Malik, J. 2006. Modeling and rendering architecture from photograph: a hybrid geometry- and image-based approach. In *23rd International Conference on Computer Graphics and Interactive Techniques SIGGRAPH 96*. New Orleans, Louisiana USA.

Delage, E., Lee, H., and Ng, A. 2006. A dynamic bayesian network model for autonomous 3d reconstruction from a single indoor image. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Doellner, J., Kolbe, T., Liecke, F., Sgouros, T., and Teichmenn, K. 2006. The virtual 3d city model of berlin – managing, integrating and communicating complex urban information. In *Proceedings of the 25th International Symposium on Urban Data Management UDMS*. Aalborg, Denmark.

Eastman, C. 2009 (accessed April 20, 2012). Building information modeling. Digital Building Lab – Georgia Tech. `http://bim.arch.gatech.edu/?id=402`.

Eastman, C., Teicholz, P., Sacks, R., and Liston, K. 2008. *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors*. Wiley.

El-Hakim, S., Beraldin, A., and Blais, F. 1995. A comparative evaluation of the performance of passive and active 3d vision systems. In SPIE, editor, *St. Petersburg Conference on Digital Photogrammetry*.

El-Hakim, S., Beraldin, j., Picard, M., and Godin, G. 2004. Detailed 3d reconstruction of large-scale heritage sites with integrated techniques. *IEEE Computer Graphics and Applications*, 23(3):21–29.

Fietz, A. 2011. *Ableitung von Bewegungsstrategien zur automatisierten, vollstaendigen Vermessung von Innenraum-szenen auf autonom navigierender Plattform*. PhD thesis, University of Stuttgart.

Foley, G. D., van Dam, A., Feiner, S., and Hughes, J. 1996. *Computer Graphics: Principles and Practice*. Addison-Wesley.

Furukawa, Y., Curless, B., Steitz, S., and Szeliski, R. 2009a. Manhattan-world stereo. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Furukawa, Y., Curless, B., Steitz, S., and Szeliski, R. 2009b. Reconstructing building interiors from images. In *IEEE International conference on Computer Vision*.

Gibson, O. 2005 (accessed April 24, 2012). Gates unveils his vision of a future made of silicon. `http://www.guardian.co.uk/media/2005/oct/28/newmedia.microsoft`.

Gokturk, S., Yalcin, H., and Bamji, C. 2004. A time-of-flight depth sensor – system description, issues and solutions. In *Computer Vision and Pattern Recognition Workshop*. IEEE Computer Society.

Goodchild, M. 2011. Spatial thinking and the gis user interface. In *Procedia Social and Behavioral Sciences*, volume 21.

Gruen, A. 2000. Semi-automated approaches to site recording and modeling. In *XIXth ISPRS Congress*. Invited paper, Amsterdam.

Guidi, G., Remondino, F., Russo, M., Menna, F., and Rizzi, A. 2008. 3d modeling of large and complex site using multi-sensor integration and multi-resolution data. In *9th International Symposium on Virtual Reality, Archeology and Cultural Heritage VAST 2008*, pages 85–92.

Haala, N., Fritsch, D., Peter, M., and Khosravani, A. M. 2011. Pedestrian navigation and modeling for indoor environments. In *7th International Symposium on Mobile Mapping Technology*. Crakow, Poland.

Haehnel, D., Burgard, W., and Thrun, S. 2002. Learning compact 3d models of indoor and outdoor environments with a mobile robot. *Robotics and Autonomous Systems*.

Hartley, R. and Zisserman, A. 2003. *Multi View Geometry in Computer Vision*. Cambridge University Press.

Hsu, S., Acharya, S., Rafii, A., and New, R. 2006. Performance of a time-of-flight range camera for intelligent vehicle safety applications. *Advanced Microsystems for Automotive Applications*, pages 205–219.

Ikeuchi, K. and Sato, Y. 2001. *Modelling from Reality*. Kluwer Academic Publishers.

Ioannidis, C., Valani, A., Georgopoulos, A., and Tsiligiris, E. 2006. 3d model generation for deformation analysis using laser scanning data of a cooling tower. In *12th FIG Commission 6 Symposium*. Baden.

ISTI-CNR, V. C. L. 2010 (accessed November 25, 2010). Meshlab. `http://meshlab.sourceforge.net/`.

Johnston, M. and Zakhor, A. 2008. Estimating building floor plans from exterior using laser scanners. In *Proceedings of SPIE*, volume 6805.

Kolb, A., Barth, E., Koch, R., and Larsen, R. 2009. Time-of-flight sensors in computer graphics. In *Eurographics 2009*.

Kolbe, T. 2012 (accessed April 30, 2012). Exchange and storage of virtual 3d city models. `http://www.citygml.org/index.php?id=1523`.

Kolbe, T., Groeger, G., and Pluemer, L. 2005. Citygml – interoperable access to 3d city models. In Oosterom, Zlatanova, and Fendel, editors, *Proceedings of the first International Symposium on Geo-information for Disaster Management*, pages 21–23. Springer Verlag. Delft, the Netherland.

Kolbe, T., Nagel, C., and Stadler, A. 2009. Citygml – ogc standard for photogrammetry? In *Photogrammetric Week 2009*.

Kraus, K. 2007. *Photogrammetry*. Walter de Gruyter.

Kwon, S., Bosche, F., Kim, C., Haas, C., and Liapi, K. 2004. Fitting range data to primitives for rapid local 3d modeling using sparse range point cloud. *Automation in Construction*, 13:67–81. Elsevier, New York, USA.

Leica 2010 (accessed October 22, 2010). Scanstation c10. `http://hds.leica-geosystems.com/en/Leica-ScanStation-C10_79411.htm`.

Lindenbergh, R., Pfeifer, N., and Rabbani, T. 2005. Accuracy analysis of the leica hds3000 and feasibility of tunnel deformation monitoring. In *ISPRS Workshop Laser scanning 2005*. Enschede, The Netherlands.

Luhmann, T., Robson, S., Kyle, S., and Harley, I. 2006. *Close Range Photogrammetry*. Whittles.

Maes, S., Reinhardt, W., and Wang, F. 2006. Conception of a 3d geodata web service for the support of indoor navigation with gnss. In Abdul-Rahman, A., Zlatanova, S., and Coors, V., editors, *Innovations in 3D Geoinformation Science: Lectures Notes in Geoinformation and Cartography*, pages 307–316.

Matta, C. and Kam, C. 2007 (accessed April 27, 2012). Gsa's national 3d-4d-bim program. `http://www.gsa.gov/graphics/pbs/Flyer_2007_01_15_laser_scanning.pdf`.

Morse, E. 2009 (accessed April 25, 2012). An online case study resource for building information modeling in college education. `http://www.wpi.edu/Pubs/E-project/Available/E-project-090109-010641/unrestricted/EJM-BIMinEducation.pdf`.

Mortenson, M. 1997. *Geometric Modeling*. Jan Wiley.

Moussa, W., Abdel-Wahab, M., and Fritsch, D. 2012. An automatic procedure for combining digital images and laser scanner data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*,

volume 39, pages 229–334. Melbourne, Australia.

Mundy, J. 1993. The relationship between photogrammetry and computer vision. In *Integrating Photogrammetric Techniques With Scene Analysis and Machine Vision, SPIE*.

Nisbet, N. and Dinesen, B. 2010. *Constructing the business case Building Information Modelling*. British Standard Institution and BuildingSMART UK.

Nishijima, Y. and Oster, G. 1964. Moire patterns: Their application to refractive index and refractive index gradient measurements. *Journal of the Optical Society of America*, 54:1–4.

Nuechter, A., Surmann, H., Lingemann, K., and Hertzberg, J. 2003. Semantic scene analysis of scanned 3d indoor environments. In *Proceedings of the Eighth International Fall Workshop on Vision, Modeling, and Visualization*.

OGC 2008. Opengis city geography markup language (citygml) encoding standard. In Groeger, G. and Kolbe, T. and Czerwinski, A and Nagel, K., editors. Reference number of this OGC project document: OGC 08–007rl, version 1.0.0. Document downloadable at `http://www.opengeospatial.org/standards/citygml`.

Olivieri, P., Gatti, M., Straforini, M., and Torre, V. 1992. A method for the 3d reconstruction of indoor scenes from monocular images. In Sandini, G., editor, *ECCV*, volume 588 of *Lecture Notes in Computer Science*, pages 696–700. Springer.

Pfeifer, N. and Boehm, J. 2008. Early stages of lidar data processing. In Li, Z., Chen, J., and Baltsavias, E., editors, *Advances in Photogrammetry, Remote Sensing and Spatial Information Sciences: 2008 ISPRS Congress Book*, pages 169–184. CRC Press.

Pfeifer, N. and Briese, C. 2007. Laser scanning – principles and applications. In *3rd International Exhibition and Scientific Congress on Geodesy, Mapping, Geology, Geophysics, Cadaster GEO-SIBERIA 2007*, pages 93–112. Novosibirsk, Russia.

Pu, s. and Vosselman, G. 2006. Automatic extraction of building features from terrestrial laser scanning. In *Proceedings of the ISPRS Commission V Symposium Image Engineering and Vision Metrology*, volume 35. Dresden, Germany.

Pu, s. and Vosselman, G. 2007. Extracting windows from terrestrial laser scanning. In *Proceedings of the ISPRS Workshop Laser Scanning 2007*, volume 36, pages 320–325. Espoo, Finland.

Pu, s. and Vosselman, G. 2009. Knowledge based reconstruction of building models from terrestrial laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64:575–584.

Rabausch, K. and Volz-Graetz, A. 2010. *Innenraeume*. Meuller.

Remondino, F. 2003. From point cloud to surface: the modeling and visualization problem. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume 34. Tarasp-Vulpera, Switzerland.

Remondino, F. and El-Hakim, S. 2006. Image-based 3d modelling: a review. *The Photogrammetric Record*, 21(115):269–291.

Remondino, F., El-Hakim, S., Girardi, S., Rizzi, A., Benedetti, S., and Gonzo, L. 2009. 3d virtual reconstruction and visualization of complex architectures – the 3d-arch project. In *ISPRS Commission V Workshop 3D-Arch 2009*. Trento, Italy.

Remondino, F. and Fraser, C. 2006. Digital camera calibration methods: Considerations and comparisons. In *ISPRS Commission V Symposium Image Engineering and Vision Metrology*.

Remondino, F. and Zhang, L. 2006. Surface reconstruction algorithms for detailed close-range object modeling. In *ISPRS Commission III Symposium*, volume 36, pages 117–123. Bonn, Germany.

Ribarsky, W., Wasilewski, T., and Faust, N. 2002. From urban terrain models to visible cities. *IEEE Computer Graphics and Applications*, 22:67–81.

Riegl 2012 (accessed April 20, 2012). Riegl vz-400. `http://www.riegl.com/uploads/tx_pxprieg|downloads/10_DataSheet_VZ-4000_03-04-2012_PRELIMINARY.pdf`.

Rioux, M. 1984. Laser range finder based on synchronized scanners. *Applied Optics*, 23:2837–3844.

Rizzi, A., Voltolini, F., Remondino, F., Girardi, S., and Gonzo, L. 2007. Optical measurement techniques for the digital preservation, documentation and analysis of cultural heritage. In *VIII Conference on Optical 3D Measurement Techniques*, volume 2, pages 16–24. Zurich, Switzerland.

Rusu, R. B., Marton, Z. C., Blodow, N., Dolha, M., and Beetz, M. 2008. Towards 3d point cloud based object maps for household environments. *Robotics and Autonomous Systems*, 56:927–941.

Sabol, L. 2008. Building information modeling and facility management. In *IFMA World Workplace*. Design + Construction Strategies. The power of process in the built environment.

Saleh Uddin, M. 1999. *Digital Architecture*. McGraw-Hill.

Sansoni, G., Trebeschi, M., and Docchio, F. 2009. State-of-the-art and applications of 3d imaging sensors in industry, cultural heritage, medicine and criminal investigation. *Sensors*, 9:568–601.

Schmitt, G. 1999. *Information Architecture: Basis and Future of CAAD*. Birkhaeuser Verlag. Berlin.

Schumacher, S. and Boehm, J. 2005. Georeferencing of terrestrial laserscanner data for applications in architectural modeling. In *ISPRS Commission V Workshop 3D-Arch 2005*.

Shah, J. and Maentylae, M. 1995. *Parametric and Feature-based Cad/Cam Concepts, techniques and Applications*. Wiley.

Shamos, M. I. and Hoey, D. 1976. Geometric intersection problems. In *17th IEEE Annual Symposium on Foundations of Computer Science*. Huston, TX, USA.

Shapiro, L. and Stockman, G. 2001. *Computer Vision*. Prentice Hall.

Stroud, I. 2006. *Boundary Representation Modelling Techniques*. Springer.

Tsai, R. Y. 1987. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3:323–344.

Van Berlo, L. and De Laat, R. 2011. Integration of bim and gis: The developement of the citygml geobim extention. In *Advances in 3D Geo-Information Sciences*. Kolbe, T. and Koenig, G. and Nagel, C. Draft version.

Van den Hengel, A. 2010. Image-based modelling for augmenting reality. In *International Symposium on Ubiquitous Virtual Reality 2010*. Invited paper, Gwangju, South Korea.

Van den Heuvel, F. A. 1999. A line-photogrammetric mathematical model for the reconstruction of polyhedral objects. In *Proceedings of SPIE*, volume 3641, pages 60–71.

Van den Heuvel, F. A. 2001. Reconstruction from a single architectural image from the meydenbauer archives. *Photogrammetrie Fernerkundung Geoinformation*, 4:247–260.

Verma, V., Kumar, R., and Hsu, S. 2006. 3d building detection and modeling from aerial lidar data. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2213–2220.

Voltolini, F., Remondino, F., Pontin, M., and Gonzo, L. 2006. Experience and considerations in image-based modeling of complex architectures. In *ISPRS Commission V Symposium*, volume 36, pages 309–314. Dresden, Germany.

Vosselman, G. 2010. *Airborne and Terrestrial Laser Scanning*. Taylor and Francis.

Vosselman, G., Gorte, B., Sithole, G., and Rabbani, T. 2004. Recognising structure in laser scanning point clouds. In *Proceedings of the ISPRS Working Group VIII/2: laser scanning for forest and landscape assessment*, pages 33–38. Freiburg, Germany.

Wagner, W., Ullrich, A., Melzer, T., Briese, C., and Kraus, K. 2004. From single-pulse to full-waveform airborne leser scanners: potential and practical challenges. In *XXth ISPRS Congress Geo-Imagery Bridging Continents*. Istanbul, Turkey.

Wiley 2006 (accessed Mai 3, 2012). Autocad file extension. http://media.wiley.com/product_ancillary/39/04701099/DOWNLOAD/9780470109939appc.pdf.

Yin, X., Wonka, P., and Razdan, A. 2009. Generating 3d building models from architectural drawings: a survey. *IEEE Computer Graphics and Applications*, 29:20–31.

Zhang, Z. 1994. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13:119–152.

Zhang, Z. 2000. A flexible new technique for camera calibration. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 22:1330–1334.

Curriculum Vitae

Personal Data

Name	Angela Budroni
Date of birth	19 November 1980
Place of birth	Sassari, Italy

School

1994–1999	Secondary school "Liceo Scientifico G. Spano" in Sassari, Italy

University Studies

1999–2004	B.Sc. in Electronic Engineering at the University of Cagliari, Italy
2002–2003	Study abroad as an ERASMUS student at the TU Delft, the Netherlands
2004–2007	M.Sc. in Electronic Engineering at the University of Cagliari, Italy
2004–2005	Study abroad as a guest student at the TU Berlin
2008–2011	PhD Thesis at the University of Stuttgart

Work Experience

2006–2007	Student assistant at Fraunhofer Institute for Manufacturing Engineering and Automation IPA in Stuttgart
2012–2013	Research scientist at Fraunhofer Institute for Embedded Systems and Communication Technologies ESK in Munich

Acknowledgments