3D RECONSTRUCTION OF INDOOR CORRIDOR MODELS USING SINGLE IMAGERY AND VIDEO SEQUENCES

ALI BALIGH JAHROMI

A DISSERTATION SUBMITTED TO
THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN EARTH AND SPACE SCIENCE AND ENGINEERING
YORK UNIVERSITY
TORONTO, ONTARIO

AUGUST 2019

ABSTRACT

In recent years, 3D indoor modeling has gained more attention due to its role in decision-making process of maintaining the status and managing the security of building indoor spaces. A 3D indoor model is an abstract representation of an indoor environment which represents the existing state of indoor outlines, usually enriched with accurate semantic and geometric information. 3D indoor space models can play a great role in addressing various matters including Building Information Modeling (BIM), augmented reality, occupant’s safety and health, situational awareness and emergency response, etc. Yet, they have not been abundantly provided for many buildings and man-made structures. Thus, 3D indoor representations need to be generated for such structures to allow various analysis that decisions are based upon.

In this thesis, the problem of continuous indoor corridor space modeling has been tackled through two approaches. The first approach develops a modeling method based on middle-level perceptual organization using an image. The second approach develops a visual Simultaneous Localisation and Mapping (SLAM) system with model-based loop closure using video sequences.

In the first approach, the image space was searched for a corridor layout that can be converted into a geometrically accurate 3D model. Manhattan rule assumption was adopted, and indoor corridor layout hypotheses were generated through a random rule-based intersection of image physical line segments and virtual rays from each of three orthogonal vanishing points. A scoring function is designed that considers volumetric aspect of created hypotheses along with their correspondences to physical edges, orientation map and geometric context of an image. This approach provides physically plausible solutions while facing objects or occlusions in a corridor scene.

In the second approach, a new technique was developed called Layout SLAM. The architecture of Layout SLAM has two major components, known as the "*front-end*" and the "*back-end*" together dealt with image observations and respective inferences of Extended Kalman Filtering (EKF) framework. Real time camera localization was performed by Layout SLAM while both corridor layout corner point features and normal point features were mapped in 3D. A new feature matching cost function was proposed considering both local and global context information. In addition, the introduced rotation compensation variable makes Layout SLAM robust against camera's orientation errors accumulations. Moreover, layout model matching of keyframes insures accurate loop closures that prevent miss-association of newly visited landmarks to previously visited scene parts.

To evaluate the proposed methods, a new dataset was generated. This dataset includes single images and video sequences acquired by hand-held cameras as well as Applanix TIMMS laser point clouds collected from various indoor corridors at York University. Geometrically accurate reference 2D (less than 3 image pixels accuracy) and 3D (about 2cm accuracy) layout models were associated with this dataset. Accordingly, different experiments were conducted including Root Mean Square Error (RMSE) calculation for generated image models, geometric comparison of the estimated 3D layouts, and evaluating Layout SLAM camera trajectories. For instance, the comparison of generated single image-based 3D models to ground truth models showed that average ratio differences in widths, heights and lengths were 1.8%, 3.7% and 19.2% respectively. Thus, the proposed method can successfully generate 3D indoor corridor models compared to its major counterpart. Moreover, Layout SLAM performed with the maximum absolute trajectory error of 2.4m in position and 8.2° degree in orientation for approximately 318m path on RAWSEEDS data set. Also, loop closing was performed strongly for Layout SLAM which provided 3D indoor corridor layouts with less than 1.05m displacement errors in length and less than 20cm in width and height for approximately 315m path on York University data set. Therefore, Layout SLAM performs robustly and produces very limited orientation errors. In future, I plan to extend Layout SLAM with less constrained geometric models.

To God

And to my family

ACKNOWLEDGMENTS

My PhD journey could not be successful without the help and support of many people and organizations. I wish to extend gratitude to all of them for their important roles throughout my research.

I would like to specially thank my supervisor, Dr. Gunho Sohn, who inspired me with his excellency in guidance and passion towards science. His ability to understand the core concepts of many academic problems and provide novel solutions is unique. Without taking his precious time for many hours, this research could not achieve its goals let alone be presented in this dissertation. Working under his supervision is an honor that stands with me for the rest of my life.

I wish to extend gratitude to my committee members, Dr. Costas Armenakis and Dr. Petros Faloutsos as well as Dr. James Elder who together provided their valuable insights through multiple hours of discussions on my research. Having access to their knowledge on my research work improved the quality of this dissertation a lot. Also, I would like to thank my previous supervisors, Dr. Baoxin Hu and Dr. Jianguo Wang who made my presence at York University possible. Moreover, I must thank my examiners Dr. Kourosh Khoshelham, Dr. Zhen Ming (Jack) Jiang and Dr. Mojgan Jadidi as well as Dr. Bruno Scherzinger from Applanix who provided TIMMS datasets.

I must thank my GeoICT colleagues, Dr. Jaewook Jung, Dr. Jungwon Kang, Dr. Yoonseok Jwa, Dr. Heungsik Kim, Dr. Junjie Zhang, Dr. Connie Ko, Dr. Chao Lue, Dr. Kiin Bang, Dr. Pio Claudio, Kivanc Babacan, Abdel-Hadi Hor, Kunwoo Park, Phillip Robbins, Eros Gulo, David Recchia, Leihan Chen, Afnan Ahmad, Aman Ullah Usmani, Maryam Jameela, Razieh Ramak, Zahra Arjmandi, Hyun Sun Park, Kang Zhao, Jacob Sunghwan Yoo, Harry Hyeonseok Kim, Brian Diep, Muhammad Kamram, Sowmya Natesan, Roman Seidel, Lena Albert, Tristan Faure, Yi-Chen Chen, Solomon Chan, Larry Wang, Duy Tran, and Dr. Andreas Wichmann. I must thank Dr. Mozhdeh Shahbazi, who collaborated on my research. Also, I specially thank Dr. Elder's laboratory members, Bob Hou and my friends, Tiana Asperjan, Sheri Godda, Rose Toung, Fiorella Pennano, Olivia Clarke, Tom Bradshaw, Tyler Hagemann, Adam Bergquist, Reza Safari, Hamid Reza Zolfagharinia, Mojtaba Eslamian, Reza Fasihiani, Kamran Ghazi, Babak Zeini, Keyhan Kanshlo, Shahab Pirnia, Majid Mosaheb, Hamed Jalalirad, Shahin Feyz, Dr. Ali Sepehri, Dr. Maryam Hariri, Dr. Iman Owrangi and Dr. Amir Salimi.

I would like to thank Ontario government for financially supporting my research through Ontario Trillium Scholarship. Moreover, I wish to thank Natural Science and Engineering Research Council of Canada (NSERC) Discovery, NSERC Collaborative Research and Development (CRD), NSERC CREATE Data Analytics and Visualization (DAV), Ontario Research Fund – Research Excellence Project – Intelligent Systems for Sustainable Urban Mobility (ISSUM), and York University for supporting my PhD research.

Beside all, my everlasting gratitude and heartfelt goes to my family members specially my parents and my brother for their encouragement, support, patience and unconditional love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
BIM	Building Information Model
BSP	Binary Space Partitioning
CRF	Conditional Random Field
CDF	Cumulative Distribution Function
DEM	Digital Elevation Model
DSM	Digital Surface Model
EKF	Extended Kalman Filtering
FAST	Features from Accelerated Segment Test
GNA	Gauss–Newton Algorithm
GNSS	Global Navigation Satellite System
HT	Hough Transform
ICP	Iterative Closest Point
IMU	Inertial Measuring Unit
IOP	Interior Orientation Parameter
LMA	Levenberg–Marquardt Algorithm
LOD	Level of Details
LSD	Line Segment Detector
MAR	Mobile Augmented Reality
MRF	Markov Random Field
MVS	Multi-View Stereo
RANSAC	RANdom SAmple Consensus
RAWSEEDS	Robotics Advancement through Web-publishing of Sensorial and Elaborated Extensive Data Sets
RGBD	Red Green Blue Depth
RMSE	Root Mean Square Error
RTK	Real-time Kinematic
SFM	Structure from Motion
SLAM	Simultaneous Localization and Mapping
SVM	Support Vector Machines

TIMMS	Trimble Indoor Mobile Mapping Solution
TLS	Terrestrial Laser Scanning
WGS84	World Geodetic System 1984 (WGS84)

Chapter 1
Introduction

1.1 Motivation

Rapid changes in human's lifestyle exacerbated their evolvement as indoor habitants. This imminent development is influenced by the world's population that will be on the rise for the coming years (Lutz et al., 2017). The increasing population rate would increase the rate of urbanization as well (International Herald Tribune, 2008). Urbanization has drastically impacted both our lives and our environments. Note that urbanization growing rate necessitates more constructions to be accomplished. Thus, cities will become larger and consequently the urge to plan, monitor, operate, modernize, manage, and make decisions for updating and analysing urban infrastructure would be undeniable.

Recently, Building Information Model (BIM) has become a significant part of building construction (Azhar, 2011). BIM is a digital representation of a facility and its physical and functional features. BIM is involved with high level of management and it can be a source of knowledge to support a facility construction and maintenance related decisions. Not only does BIM make building construction more productive and lucrative, but also supports the facility from the early stages of design and continues through its entire operational life. However, many of the existing urban infrastructures and facilities have been constructed based on traditional designing methods. Thus, new BIMs need to be generated for these urban infrastructures. Note that with new changes in existing infrastructures and the expansion of buildings, their respective representations must be regularly updated as well.

In the past years, many researchers in photogrammetry, computer vision and robotics fields dedicated their time and efforts to provide accurate representations of different building entities (Lee et al., 2010; Endres et al., 2012; Schwing and Urtasun, 2012; Valero et al., 2012; Chao et al., 2013). Geometrically accurate 3D building models are expressed to be the most important feature in representation of digital cities (Fuchs et al., 1998). Note that primitive based geometric models of city entities are appropriate inputs for managing, updating and analysing urban infrastructures. However, the generalization of these representative primitives would be a challenge (Xiong et al., 2013; Henry et al., 2014; Diaz et al., 2015; Whelan et al., 2015 and Bueno et al., 2018). Primitive based geometric representations would enhance the level of understanding in urban structure management compare to other types of representations including mesh models. For instance, indoor spaces where we spend most of our time, are the most important city environments that can be represented by primitive based geometric models.

Earlier efforts towards accurate representation of various city entities started by employing remotely sensed data to extract building models (Grün et al. 1995, 1997). Thereafter more studies have been presented on recognition, detection and reconstruction of building indoor spaces (Hähnel et al., 2003; Hedau et al., 2010; Schwing et al., 2013; Zhang et al., 2014; Liu et al., 2015; Tang at al., 2016; Zhu et al., 2016; Huang et al., 2017 and Wang et al., 2018). Note that the task of 3D modeling of indoor spaces has been associated with navigation issues and autonomous systems from the beginning. Various applications for indoor mapping and navigation services have been developed by companies including Google, Microsoft and Apple (Tóth et al., 2015). Accurate 3D indoor models are essential for various spatial information-based applications such as indoor security, indoor positioning and navigation (Li et al., 2013; Ochmann et al., 2016 and Lehtola et al., 2017). Moreover, new technologies like Mobile Augmented Reality (MAR) provide a platform for using 3D indoor models to interact with surroundings through a computer or mobile device. For instance, indoor space related information can be displayed on a mobile device upon a query entered by the user using the respective 3D indoor model.

Various data gathering technologies are available which can be used for 3D indoor space modeling. Images have been used as a common data source for modeling. Early efforts in indoor modeling include manually digitizing images to detect indoor layout. Recently, the computer vision related approaches provide the base for automatically reconstruct indoor models. Simultaneous Localization and Mapping (SLAM) and Structure from Motion (SFM) are notable techniques for recovering indoor layout from a collection of images. Using image data, occlusions, shadows and low level of contrast may disrupt indoor modeling cues extraction and enforce human's interactions. Laser scanners can directly provide accurate dense 3D point clouds and improve the level of automation in reconstruction of geometric models (Jung and Sohn, 2019). At an indoor scene, they can provide precise plane information. Yet, they cannot precisely identify layout boundaries (images are more accurate) due to their irregular point distribution. Laser scanners and cameras together can provide a full description of 3D indoor models. Note that registration between data sources is needed to achieve accurate performance in this regard.

In the past few years, many indoor modeling and reconstruction methods are presented that differ in terms of data sources (multi-data source vs. single data source), adopted data processing strategies (generic, parametric, or hybrid) and levels of automation (semi-automatic vs. full-automatic). Yet, developing a new method to generate geometrically accurate indoor models in a fully-automated way is still a big challenge. Sohn and Dowman, (2007) mentioned the critical factors that should be considered while developing a new modeling algorithm. These factors include: a) scene complexity, b) sensor dependency and c) incomplete cues. First, indoor spaces are containing various information of non-layout objects (e.g., tables, chairs, paintings, and other clutters) in addition to layout parts (e.g., floor, ceiling and walls). Moreover, indoor scenes have various formats and structures that cannot be described by a single standard type. Therefore, complex indoor scenes must be simplified to achieve a suitable interpretation. Second, sensors have their own characteristics in terms of data acquisition mechanism. These characteristics will have an impact on the reconstructed models and must be thoroughly understood prior to modeling. Third, missing data problem is quite common, and occlusions and objects overlap may cause disintegration problem in captured data. For instance, waxed floors may reveal redundant or spurious cues in images that cause confusion and ambiguity in modeling process.

Thus far, different methods have been proposed for reconstructing 3D indoor layout using various data sources (Yang et al., 2018). Yet, the proposed methods have limitations due to inherent sensor dependency, modeling accuracies, levels of automation and ability to solve missing data problems. A promising approach for solving some of these problems would be to take advantage of both model-driven and data-driven strategies using images. Indoor modeling using a single image have been exploited by other researchers (Hedau et al., 2010), since modeling cues can be extracted from a single image. Compared to laser point clouds, optical images can provide geometrically accurate and semantically rich information. However, laser point clouds have weakness in detecting layout edges, even though they can provide 3D information of planar patches. The cost of gathering high accuracy indoor laser point clouds is more expensive than capturing images. Also, post processing laser point clouds are very time consuming and labour intensive. Since high-quality images can be taken by low-cost cameras and plenty of images have been abundantly shared through the internet, the idea of crowd sourced modeling seems more achievable using images. Note that existing 3D indoor models can also be updated using newly captured images taken from different cameras.

Regardless of which data source is applied, the selected strategy for reconstruction of the indoor layout is very important. The accuracy of the adopted strategy has a great impact on the quality of the reconstructed layout. The adopted strategy should establish a robust relation between captured data and the general perception of the indoor layout. In addition, the interaction between existing layouts and newly captured data should be expressed in terms of continuous indoor modeling. Even though different strategies that deal with layout complexity have been proposed, the reconstruction of 3D indoor layouts over a large-scale area (indoor corridors) has been investigated relatively less. Thus, more research on developing methods for reconstructing indoor layouts from single or set of images are required. This notion provides a base for the idea of continuous indoor modeling.

1.2 Research Objectives

As mentioned previously, primitive based geometric representations of city entities are important for urban structure management. Moreover, geometric representation of indoor layouts would increase the level of understanding in preparing building information. Hence, the reconstruction of 3D indoor space geometric models (layouts) is essential for analysing and updating building information. The main objective of this thesis is to address critical steps towards creation of continuous 3D indoor space models, which includes 3D indoor corridor layout reconstruction (single view), indoor corridor mapping (visual SLAM) and update of the estimated corridor layouts, visual SLAM loop closing and quality evaluation. To accomplish these goals, the following issues need to be addressed:

First, the proposed 3D indoor layout reconstruction method should provide robust and accurate 3D indoor corridor layouts. The accuracy of reconstructed 3D layouts should meet urban planning and indoor navigation level accuracy to support indoor navigation and urban structure management. Regardless of indoor layout complexity (connected corridors) and the configuration of indoor clutters, the methods should produce accurate topologic and geometric 3D indoor corridor layouts. Second, the proposed methods must generate regularized indoor corridor layouts, abiding to standard regularities such as planes parallelism, planes orthogonality and global symmetry. Since the reconstructed layouts should represent the regular properties of real indoor corridor structures, applying these types of regularities is essential. Third, a reliable loop closing accuracy must be achieved while applying visual SLAM for continuous indoor corridor mapping. A revisited scene should be accurately and robustly registered to the existing 3D indoor corridor layouts (map) through recalling visited scenes by visual SLAM system. The accumulated mapping errors of the estimated indoor corridor layouts should be corrected. Thus, the estimated indoor corridor layouts must be updated effectively.

1.2.1 General Research Framework

Figure 1.1 presents the overall workflow of the proposed method for continuous indoor space modeling and the interrelation between the main parts of this thesis (major contributions are highlighted in colorized boxes). First, 3D indoor corridor models are reconstructed using single images (Chapter 4). The proposed method reveals the steps for generating a 3D indoor corridor model from a single image. These steps start from low level image processing and finish with 2D to 3D conversion of best fitting layout hypothesis. The proposed method includes following steps: 1) edge detection and straight line extraction, 2) orthogonal vanishing points estimation, 3) major box hypotheses creation (topology construction), 4) extraction of features (edge correspondences, volume maximization, orientation map, geometric context), 5) major box hypotheses scoring using Artificial Neural Network (ANN), 6) side box hypotheses creation, 7) hypotheses scoring (major box with its side boxes) by ANN, 8) selection of best fitting hypothesis and 9) 2D to 3D conversion using vertical vanishing point.

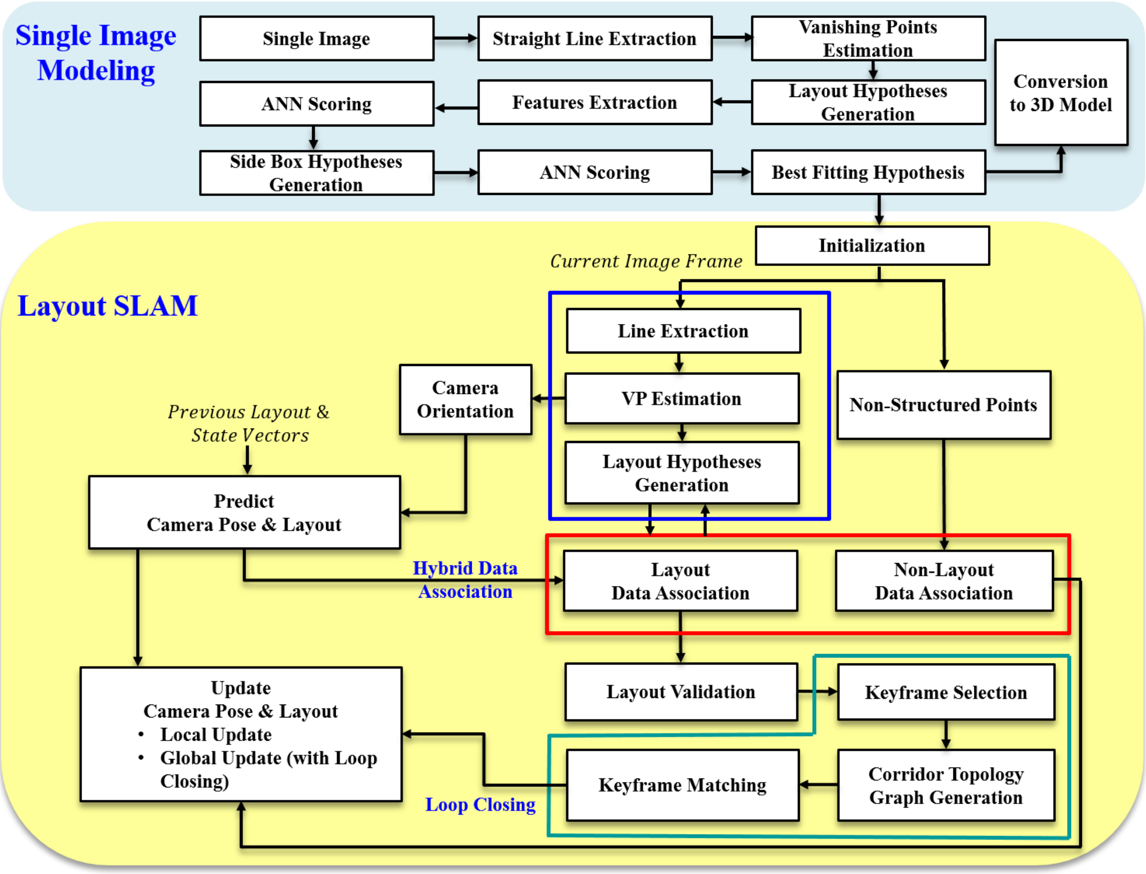


Figure 1.1 Proposed workflow for continuous indoor space modeling.

Regarding the process of modeling cues extraction and hypotheses creation, this dissertation explains how the combination of both detected straight line segments and virtual rays of vanishing points can effectively handle cluttered indoor scenes and create models even with incomplete layout evidences. For topology construction, the indoor layout is considered to follow the Manhattan rule assumption and the major corridor has 5 regularized faces including ceiling, floor, front wall, right wall and left wall. As part of this study, scoring hypotheses by considering both geometric and semantic features is examined. In the proposed linear cost function, the combination of geometric context and orientation map is tested for better layout hypotheses scoring. Finally, weight parameters were automatically determined through ANN while normalized features contributed to the network as inputs.

Second, a visual SLAM method called Layout SLAM is proposed (Chapter 5). The proposed method consists of following steps: 1) the system initializes by introducing the scene layout and its structural corner point features at the first frame, 2) for the rest of frames, straight line segments are detected and vanishing points estimated to generate layout hypotheses, 3) the best fitting layout hypothesis to the scene is identified through layout features matching process, 4) rotation compensation variable is introduced to eliminate the effect of orientation errors accumulation, and 5) finally an online sparse map of the indoor corridors layouts is built. Note that the proposed feature matching cost function considers both local and global context information by measuring pixel to pixel orientation differences of matched junctions and examining angle differences of directly connected corners. The proposed method can deal with the presence of few geometrical features and absence of texture in the scene through introducing indoor layout. Also, the amount of rotations that should be compensated at each step is estimated by consecutive vanishing points matching on a unit sphere.

Third, this dissertation reveals a new model-based loop closing technique that associates current scene parts to the previously visited landmarks (Chapter 6). Both layout information (topology and geometry of reconstructed layout models) and image information (photometric features) are used to match layout models of various keyframes. The proposed method imposes a geometric constraint on the global layout model consistency and reduces matching ambiguity by considering topological context. Homogenously textured corridors can be easily handled, since introducing layout compensates the imposed matching limitations. Different types of modeling errors such as orientation error, boundary displacement and shape deformation can be reduced as well.

Forth, to evaluate the performance of the proposed indoor space modeling algorithms a new type of dataset is prepared (Chapter 3). The prepared dataset acquired by hand held cameras including single images and video frames covering indoor corridor places. Also, laser point clouds were acquired, and ground truth 3D models were manually generated for evaluation of the results. This dataset is prepared to compensate for limitations of existing datasets.

1.2.2 Contributions

As discussed previously, in this thesis a solution for 3D reconstruction of indoor models is provided. The proposed approach considered and incorporated various study areas to shape the idea of continuous indoor space modeling. Here, major contributions of this thesis are summarized:

- Proposing a method to reconstruct 3D indoor corridor models from a single image: To compensate limitations of existing single image-based indoor modeling methods, that mainly define the scene layout as a single box primitive, the proposed method represents the indoor scene layout through hypothesizing-verifying multiple box primitives. Using middle-level perceptual organization and finding the ground-wall and ceiling-wall boundaries, multiple layout hypotheses can be generated intersecting detected straight line segments and virtual rays of vanishing points. Also, an edge-based objective function is proposed for evaluating layout hypotheses and finding the best fitting layout.

- Proposing the combination of geometric context and orientation map for image-based indoor layout evaluation: Clutter presence at indoor scenes causes shadows and occlusions that hinder a comprehensive interpretation of edge relations. Orientation map reveals the local belief of region orientations and geometric context can reveal the likelihood of possible label for the same region. Evaluation of indoor layout hypotheses by combining both, will incorporate image geometric and semantic information in hypothesis selection process and improve the results.
- Introducing Layout SLAM method for real-time indoor corridor layout estimation: The system is initialized using layout corner point features detected in the first frame and performs real time indoor corridor layout estimation and camera localization. Layout SLAM can reduce the effect of abrupt camera rotations by introducing rotation compensation variable to visual SLAM architecture. Vanishing directions of consecutive video frames are matched to estimate the amount of relative camera orientations. Moreover, layout structural corner points are matched using features that are invariant under scale, translation, and rotation. Layout Feature matching cost function considers both local and global context information.
- Proposing a new loop closing technique based on layout topology graph: Incorporating this loop closing method to the proposed layout SLAM algorithm will make it robust against error accumulations and miss-association of newly

visited scenes to the previously visited layouts. Both layout information (topology and geometry of reconstructed layouts) and image information (photometric features) are utilized to confirm a loop closing incident. This method imposes a metric constraint on the global layout consistency to adjust the mapping scale drifts and reduces matching ambiguity in the homogenously textured indoor corridors.

- Building a new dataset covering Manhattan type indoor corridors at York University based on both camera and laser scanner data: This dataset is used to evaluate the performance of the proposed continuous indoor space modeling method. The acquired data using hand held cameras includes both single images and video frames. Also, 3D laser point clouds are incorporated into this dataset using Applanix TIMMS and used to manually generate ground truth 3D layout models.

1.3 Thesis Outline

This thesis is arranged in 7 chapters. Here is the overview of the presented chapters:

Chapter 1 presents the motivation of this thesis and introduces the proposed methods and strategies for solving research questions.

Chapter 2 provides background information for better understanding this thesis, and gives a literature review regarding indoor modeling strategies, main data sources and model representations.

Chapter 3 introduces newly generated dataset and reveals the characteristics of the generated 3D ground truth models. This new dataset is generated to effectively assess the quality of the proposed continuous indoor space modeling algorithm.

Chapter 4 presents a method to reconstruct a 3D indoor corridor model using a single image. Manhattan rule assumption is applied to impose geometrical regularity on 3D indoor corridor models. Both physical line segments and virtual lines of vanishing points were used to generate multiple corridors layout hypotheses. Not only orientation map and geometric context of an image are combined, but also an artificial neural network is designed to evaluate and find the best fitting layout hypothesis.

Chapter 5 proposes a new visual SLAM algorithm for estimation of indoor corridors layouts and camera poses at the same time. This method matches vanishing points of consecutive video frames on a Gaussian sphere to estimate relative camera orientations and reduce the angular drifts in the system.

Chapter 6 introduces a new loop closing method using layout topology graph. Topology and geometry of reconstructed layouts and photometric features together assist loop closure occurrence. This method adjusts layouts scale drifts and reduces matching ambiguity where indoor corridors have low textures.

Chapter 7 gives a conclusion for this study and provides recommendations for future works.

Chapter 2

Background

Over the years, we have improved our lifestyles and progressively become indoor creatures. Thus, understanding the 3D space of an indoor environment and studying the relation of humans' activities to these places has become an important research topic. In recent years, for addressing related issues of indoor space data gathering, processing and modeling, many researchers have contributed in various fields. For instance, indoor space modeling is very well studied in Computer Vision and Robotics for navigation, recognition, and reconstruction. Moreover, spatial information of the indoor environments can be applied in many applications. Public security can be facilitated via accurate 3D indoor models that paired with current navigation technologies by the time of an incident. This chapter aims to review some of the most influential literatures related to indoor modeling. Here, we scrutinize data-driven and model-driven processing strategies in indoor modeling, and further discuss the current main sources of data in this field. Moreover, we reviewed current model representation techniques, and deepen our redaction by overviewing both metric and semantic representations.

2.1 Indoor Modeling

Indoor modeling can be defined as a procedure to generate digital models which represent interior structures with a physical valid form. In this process, the input data will eventually be transformed into structured archetypes illustrating high geometric and semantic information. Indoor models are increasingly gaining importance due to their various applications in architectural planning, indoor navigation and tracking, energy and lighting analysis, crisis management and emergency route planning, etc.

In recent years, the generation of 3D indoor space models has gained a lot of attention (Rassia, 2017). Thus, various indoor modeling techniques with respect to available data sources have been introduced to photogrammetry and computer vision communities. Categorizing indoor/outdoor building modeling and reconstruction algorithms into various classes may vary based on one's special inference. Yet, this categorization would be more meaningful by considering different aspects of these algorithms. Factors which may influence this categorization may include: a) applied data, whether having single or multiple data sources, b) strategies adopted for data processing, either generic (data-driven) or parametric (model-driven) and c) to what extent human must intercept, either fully automatic, semi-automatic or completely manual (Jung and Sohn, 2019). In this section, the existing indoor space modeling strategies (section 2.1.1) and major indoor reconstruction data sources (section 2.1.2) will be reviewed.

2.1.1 Parametric or Generic

To comprehensively understand the existing indoor modeling and reconstruction algorithms and to develop new ideas in this regard, paying special attention to their adopted data processing strategies would be an excellent choice. Generally parametric methods can reconstruct indoor models through fitting parameterized primitives to the acquired data. The main reason that these methods can be successful is that many manmade indoor places have common structure or shape. These common structures either cubical or cylindrical can represent indoor structures if they considered as the standard reconstruction primitives. Thus, ordinary indoor places can be simply represented as regularized indoor models. It should be noted that applying pre-defined parameterized primitives is crucial when encountering with low density or missing data problem. However, the proper selection of influential primitives is cumbersome. Moreover, complex indoor places cannot be fully represented with a set of basic primitives.

Quattoni and Torralba (2009) expressed that indoor scenes recognition problem can be addressed through a model which exploits local and global discriminative information. Hence, they proposed a method to recognize indoor scenes through a prototype-based model that can successfully combine both global spatial properties and local objects of the scene. Xiong and Huber (2010) created semantic indoor 3D models based on context. Xiong and Huber (2010) encoded laser point cloud in a voxel data structure and assume that they can be modeled using a set of planar patches. Contextual relationships between patches and local features together classify planar patches using a Conditional Random Field (CRF) model. Thus, classifying planar patches extracted from laser data leads to semantic 3D models of buildings. Understanding the 3D geometry of indoor scenes is possible via modeling the geometry and location of specific objects (Del Pero et al., 2012). Del Pero et al. (2012) defined rooms as box and windows, doors and pictures as rectangular frames. Also, they considered object characteristics meaning beds are shorter than they are wide, and cabinets are more likely to be tall and narrow. They used a statistical model which integrates objects with their specific prior on size, locations and relative dimensions to geometrically model indoor scenes.

Wang et al. (2013) addressed the problem of indoor scene understanding by introducing latent variables to account for clutter. Hence, the observed scene is jointly explained by recovered furniture layout and room geometry. Note that model parameters are learned from training images labeled with only room layout (considered as parametric box). Ikehata et al. (2015) reconstructed an indoor 3D model from panoramic RGBD images. A graph represents the scene geometry and the graph nodes correspond to either rooms, walls or objects. Scene graph can be manipulated by a structure grammar which drives a principled algorithm for new reconstructions. To recover a structured model, the grammar rules should be sequentially applied.

There are plenty of other methods that apply model-driven strategies for scene recognition or estimating indoor models (Hedau et al., 2010; Lee et al., 2010; Hedau et al., 2012; Schwing et al., 2012; Schwing and Urtasun, 2012; Chao et al., 2013; Schwing et al., 2013; Zhang et al., 2014; Diaz et al., 2015; Liu et al., 2015 and Zhu et al., 2016). However, contrary to parametric methods, generic ones do not make any assumptions about the shapes of indoor environments. Hence, they should be able to handle all types of indoor environments. Yet, these methods might cause irrefutable amount of deformations due to presence of outliers in the data. Note that such approaches usually require a regularization step during their reconstruction process. Generally, generic approaches start by extracting indoor modeling cues like primitives of surfaces, lines, outer boundary lines and intersection lines followed by indoor model reconstruction. Lee et al. (2009) used geometric reasoning to recover an indoor layout from a single image. They create scene plausible interpretations from a collection of extracted line segments. Geometric reasoning and verification applied to find the best fitting model hypothesis. Lee et al. (2009) proved that geometric constraints on bunch of segments can be described by a set of rules which facilitate scene hypotheses interpretation.

In generic approaches, extraction of surface primitives can be performed through Segmentation. Segmentation divides the available data into homogeneous regions. Popular segmentation algorithms based on Random Sample Consensus (RANSAC) (Tarsha-Kurdi et al., 2008) and region growing (Rottensteiner et al., 2005, Kada and Wichmann, 2012) can be used to segment indoor planes. Shao et al. (2012) proposed the semantic modeling of indoor scenes. Their method segments input images into semantical regions and replace the incoming segments by similar predefined three-dimensional models. This method can progressively reconstruct the whole scene based on captured RGBD-images. Silberman et al. (2012) proposed a method to interpret the main surfaces and objects at indoor scenes by parsing them into different regions and recovering support relationships. They calculated surface normal and aligned it to room dominant orthogonal direction. Using RANSAC, planes are fitted to RGBD data points and segmented based on color gradients and depth.

Silberman et al. (2012) reveal a better understanding of cues that can provide a 3D interpretation of structures.

After a data-driven method extracts surface primitives or segments data into homogeneous regions, indoor modeling cues can be formed based on topological and geometrical relationships of the segmented regions. Note that line intersections can be simply achieved by intersecting two adjacent segmented regions or surfaces. After all indoor modelling cues are gathered, 3D indoor models can be reconstructed by combining these cues. As expressed previously, the generic approaches and parametric ones have different strategies in the modeling process. Wang and Gupta (2016) took advantage of both these strategies and proposed a generative image modeling method that consider images to be comprised of an underlying 3D structure and textures which are mapped onto it. They offered a 3D model and texture Generative Adversarial Network that generates a surface normal map which used for generation of a 2D image. Their method can generate rather realistic images.

2.1.2 Data Sources

As it can be inferred from the previous section, multiple sensors have been used to construct 3D models of building indoor spaces. The most popular data sources include: a) RGB-D cameras, b) laser scanners and c) perspective cameras in form of stereo, monocular or omnidirectional vision. In the following sections a summary of the researches that used these sensors for indoor space modeling will be presented.

2.1.2.1 RGB-D Cameras

By the advent of new technologies in recent years, color and depth (RGB-D) images have become widely available. The RGB-D image is acquired by the combination of an RGB image and its corresponding depth image. Different techniques have been developed to capture RGB-D data (Chen et al., 2015). In stereoscopic camera pairs, the disparity between captured images provide depth information and in some other cameras light is emitted to help the calculation of depth. Also, various public RGB-D datasets have been introduced to computer vision community covering indoor scenes (Koppula et al., 2011; Silberman and Fergus 2011; Silberman et al., 2012; Anand et al., 2013; Xiao et al., 2013; Lai et al., 2014; Mattausch et al., 2014).

Since RGB-D cameras can swiftly acquire the 3D digital representation of indoor places, they can be used as a major data provider to indoor scene modeling research. Yet, RGB-D cameras often provide noisy or distorted depth information and interior objects usually have complicated geometry (Chen et al., 2014). Thus, various methods have been proposed to address these problems while modeling indoor places using RGB-D images. Early attempts started by registering a set of RGB-D images into a single reference frame and volumetrically representing point cloud that can be converted into mesh-based 3D models later (Valentin et al., 2013). Thus, traditional registration and fusion algorithms are used to develop new geometric indoor modeling methods. Note that these methods must consider the quality of RGB-D data that might be low in many cases.

Izadi et al. (2011) and Newcombe et al. (2011) proposed the Kinect Fusion system that provides level-of detail (LoD) scanning and modeling. Heredia and Favier (2012) used volume shifting in larger scale environments to extend the Kinect Fusion framework further. It should be noted that RGB-D SLAM provides a more complete framework in this regard (Endres et al., 2012). In RGB-D SLAM, robust point correspondences between frames will be provided through both depth image shape-features and features of RGB image that complement each other. Note that RGB-D SLAM applies either sparse mapping or dense mapping techniques. In sparse mapping, limited number of keyframes will be sparsely selected for coarse reconstruction while in dense mapping the complete RGB-D stream is utilized to achieve detailed reconstruction (Endres et al., 2012; Li et al., 2013; Henry et al., 2014; Whelan et al., 2015; Tang at al., 2016; Huang et al., 2017).

Low-quality RGB-D sparse data is suitable for modeling indoor scenes using semantic modeling techniques. Yet, segmenting an indoor scene into various semantic regions and separating each object from its surroundings is a challenging problem. Often in semantic indoor modeling the prior knowledge is applied in the form of contextual rules and interior objects shapes are usually known a priori. Chen et al. (2014) proposed to semantically model indoor scenes based on contextual information. They used a 3D scene database and investigate the co-occurrence of contextual information to not only ensure semantic compatibility but also constrain modeling. Li et al. (2011) proposed an iterative constrained optimization method to model objects by combining primitive shapes. Mutual relations such as placement, equality and orientation are considered to globally integrate locally fitted primitives. Also, the interior layout is commonly assumed as a box while walls and floors/ceilings are fitted with vertical and horizontal planar primitives, respectively (Xiao and Furukawa, 2014). Sanchez and Zakhor (2012) applied convex hull and alpha-shape algorithms to form non-rectangular polygonal shapes through determination of each planar primitive spatial extent.

2.1.2.2 Laser Scanners

The use of laser scanner offers rapid high-resolution capture of surface elevation data suitable for a large range of applications. The commercial use of laser scanners in the last few years has upstretched as more reliable and accurate systems are produced. Laser scanners can widely be incorporated in different fields such as architecture, engineering and construction domain as well as generation of 3D models of building and indoor facilities (Hähnel et al., 2003; Valero et al., 2012; Xiong et al., 2013; Ochmann et al., 2016; Lehtola et al., 2017). Yet, generated models are not often constructed automatically, and laser scanners data processing is labor-intensive and tedious. Early researches on modeling and mapping indoor spaces is performed using robots and terrestrial laser scanners (Maas and Vosselman 1999; El-Hakim, 2000; Frueh et al., 2005). However, fully automation of the modeling procedure is still a big challenge and it has gained a lot of attentions in past years (Pu and Vosselman 2009; Ripperda and Brenner 2009; Xiong et al., 2013; Xiao et al., 2015; Gimenez et al., 2016; Macher et al., 2017; Bueno et al., 2018; Wang et al., 2018).

As mentioned, a wide range of different algorithms have been proposed to model 3D laser scanner data. Hahnel et al. (2003) proposed a probabilistic method to tackle the problem of map generation with mobile robots. They applied the Expectation-Maximization method to extract dynamic objects from 2D and 3D data obtained with laser-range scanners. More intelligence-based approaches in this regard applied Random Sample Consensus (RANSAC) and Iterative Closest Point (ICP) to achieve better results (Rusu et al., 2007; Nüchter and Hertzberg 2008; Nüchter et al., 2010). Coughlan and Yuille (1999) mentioned that most of manmade indoor structures are constructed based on a Manhattan three-dimensional grid which can restrict the topology of indoor layouts. Adán and Huber (2010) proposed a method to reconstruct indoor spaces where clutters and occlusion are present in range data. They considered volume of indoor space to recognize significant surfaces through Support Vector Machines (SVM) learning technique and recover surface shapes in an acceptable condition. Budroni and Böhm (2010) also used laser scanners and followed the Manhattan world assumption to reconstruct a volume sweep of the indoor space. A cell decomposition approach was used to extract contours of the ground plan, and only suitable cells were added to the ground shape. Finally, walls were raised from the floor to the ceiling level. Armeni et al. (2016) proposed a hierarchical approach to semantically parse 3D laser scanner data obtained at a building. They defined the semantically meaningful spaces considering rules applied in Manhattan structures formation and further parsed the spaces into building elements such as walls and columns. Following Manhattan rule assumptions provides strong 3D prior for discovering building elements. Murali et al. (2017) generated building information models (BIMs) of houses using laser data. They used scans of Manhattan type indoor floors and created room layouts through detecting walls and reasoning on their relations. Xie et al. (2018) proposed a method to regularize building boundaries obtained from noisy point clouds. They detected planar structures and locally consolidated boundary points of planes and grouped them into piecewise smooth segments. Parallel and orthogonal segments are globally regularized through a labeling process formulated as a Markov Random Field (MRF) which solved by graph cut.

Not all of studies are considering Manhattan rules to simplify the formation of various building structures. Dealing with laser scanner data there would be other approaches to extract indoor structure elements. Recognizing structures in laser scanner data was in focus in many researches where a special attention is payed to the extraction of smooth planar surfaces (Vosselman et al., 2004). Boulaassal et al. (2007) proposed a method that delivers planar facade segments from Terrestrial Laser Scanner (TLS) data by applying RANSAC method to fit geometric primitives automatically. Nüchter et al. (2003) semantically analyzed the scanned indoor places. They transformed a 3D volumetric model into a very precise compact 3D map to generate semantic descriptions. Later, the scanned 3D environment was matched against a rough semantic description of overall indoor environments. Matching was accomplished by a Prolog program compiled from 3D laser data. Biber et al. (2005) acquired a realistic 3D model using a mobile robot which had a laser scanner and a panoramic camera on board. Walls were extracted using 2D laser scans and textures were built from the panoramic images. Shukor et al. (2011) attached a laser scanner to a mobile platform and proposed a knowledge-based method to reconstruct planar surfaces from 3D laser scanner point cloud data. Their algorithm is based on a computational geometry approach. Chen and Cho (2016) created an on-line incremental 3D environment map via orthogonal pair of laser scanners mounted on a mobile platform. The horizontal scanner provides information for the estimation of platform’s position and orientation using SLAM solution and the building structure is recovered using vertical scanner. Tomljenovic et al. (2016) used a model-driven mapping approach to represent building roof outlines that extracted as 2D polygons inside laser data. They generated both Digital Surface Model (DSM) and Digital Elevation Model (DEM) from laser data and generated objects with respect to point statistics from the linked laser point cloud. Using class modelling techniques, they could generate the ultimate class of objects representing buildings. Although laser scanners are very keen in capturing 3D geometric information from indoor/outdoor spaces, they are not good at providing texture. Therefore, fusion of laser scanner and camera can be helpful to obtain complete 3D information of indoor/outdoor places. There are some hybrid approaches which combine images and point cloud for building reconstruction. Brenner (2005) provided a brief review on early attempts towards this fusion where the underlying principles of hybrid approaches has been introduced. However, the need to fuse and calibrate laser scanner and camera together as well as data registration procedure make these approaches less favorable.

2.1.2.3 Cameras

When it comes to human cognitive, it is very easy to interpret the 3D structure of an indoor or outdoor environment in an image. However, teaching machines to automatically recognize different structures in an image and achieving the same quality as human’s brain is tedious. Early attempts in computer vision is involved with line drawings. Guzmán (1968) was one of the first researchers who interpreted line drawings to distinguish between different parts of objects. Clowes (1971) classified lines of polyhedral objects into convex, concave, and occluding to recover the 3D structure of the objects. The concept of gradient space has been introduced by Macworth (1973) and combined with some surface-based constraints. Waltz (1972) put a step forward and let line drawings to have missing edges, and Sugihara (1984) used an algebraic optimization for line drawing interpretation. Kanade (1980) considered Origami world that includes hollow shells and planar sheets as well as imposing heuristics indicating that parallel lines in image are parallel in the space. Since images are excellent sources for providing both geometric and semantic information of the environments, image-based modeling is very much appealing (Kress and Van Leeuwen, 1996). In recent years, plenty of 3D modeling and building reconstruction approaches using multiple or single images have been proposed. Yin et al. (2009) generated three-dimensional building models based on architectural drawings. They used images as the input and converted them into CAD files to generate the 3D models automatically.

Using images alone, Multi-View Stereo (MVS) algorithms can construct highly detailed 3D models of the environments (Vanegas et al., 2010). Furukawa et al. (2009) stated that multi-view stereo algorithms can produce high accurate data comparable to laser range scanners. They proposed MVS-based method which can perform in homogeneous places where piece-wise planar surfaces are abundant. They extracted dominant plane directions from planar surfaces to create plane hypotheses and recovered depth maps using MRF. Seitz et al. (2006) and Schöning and Heidemann (2015) quantitatively compared several multi-view stereo reconstruction algorithms. Furukawa and Hernández (2015) introduced efficient MVS optimization algorithms considering robust implementations of photometric consistency measures. Langguth et al. (2016) presented a multi-view reconstruction method that transits between stereo-matching and Lambertian shape-from-shading using image gradients. Ulusoy et al. (2017) applied object-level shape priors to propose a probabilistic model which integrates 3D shape information from multiple objects with image evidence from multiple views. Accurate objects 3D poses, and dense 3D reconstruction of the environment is inferred from their proposed model. Rebecq et al. (2018) proposed a solution for event-based multi-view stereo 3D reconstruction. They considered the ability of event cameras to provide semi-dense geometric information and continuous measurements while the sensor is moving.

Apart from multi-view stereo techniques, plenty of building reconstruction approaches have chosen single images as their primary data source (Lee et al., 2010; Schwing et al., 2013; Chao et al., 2013; Zhang et al., 2014; Diaz et al., 2015 and Zhu et al., 2016). When dealing with a single image, collection of line segments which can be detected by low level image processing can be a source to recover the building structure (Hedau et al., 2010). However, not all the detected lines can be helpful in the process of structure recovering. Some edges may lie on walls or objects surfaces which are not part of the original structure (Schwing and Urtasun, 2012). Information extraction from line segments of images is important specially where orthogonality constraints and Manhattan world assumption are applied. Košecká and Zhang (2002) proposed a method which can recover vanishing points and camera parameters in a single image. Their method is using line segments related to Manhattan world structure. Vanishing points were estimated using dominant rectangular structures in images by Košecká and Zhang (2005), Micusik et al.

(2008) and many other researchers. Han and Zhu (2009) found rectangles (grid or box patterns) in images from line segments that were aligned with vanishing points.

It should be noted that multiple images are needed to extract 3D information of the scene, unless a reference is present in the image (Criminisi et al., 2000). For example, the ground plane can be treated as a reference in a single image. Delage et al. (2006) suggested that using Bayesian methods can help to efficiently recover 3D information. They could recognize the floor-wall boundary, since visual cues in the image were combined with some prior knowledge about the scene geometry. Given a single image, their trained model can be used for 3D reconstruction. Hoiem et al. (2005) focused on semantical information and proposed a method to recover the 3D structure of indoor/outdoor spaces from an image. They estimated orientations (e.g., horizontal, vertical) in an image using statistical methods considering color, texture, orientation of the edges, position, etc. Saxena et al. (2009) proposed a method to connect regions based on connectivity or co-planarity so that there would be no need to consider any assumption about the ground plane. Saxena et al. (2006) considered image properties to estimate the absolute depth in a single image. Nedovic et al. (2007) divided an ordinary scene into some limited stages of 3D scene geometry. The information extracted from these stages can be a guide for a better depth estimation and image interpretation. Lee et al. (2009) applied geometric reasoning to estimate the physical possible interpretations of an indoor layout in a single image. Hedau et al. (2009) and Wang et al. (2010) applied structural learning approach to estimate the best fitting layout to an indoor image. Parameterizing the indoor structure by a cubic that aligned with orthogonal directions is one solution for indoor modeling (Schwing et al., 2013). Objects can impose physical constraints which can be employed to estimate the room layout (Pero et al., 2012). Zhang et al. (2014) utilized objects for reasoning about the scene layout. Nevertheless, the scene layout can be used to improve detection of objects in the scene (Fidler et al., 2012). Liu et al. (2015) applied prior knowledge of the scene and its semantics to resolve some of objects and layout relation ambiguities in an image. Liu et al. (2017) acquired a relatively accurate normal map from a single image to interpret scene geometry. They represented object as a normal-based graph and applied graph matching to retrieve similar object model from the database. Yang et al. (2018) recovered the indoor layout and shapes of the objects from a single 2D panorama by extracting geometric and semantic cues. A constraint graph on image line segments and detected planar super-pixels impowered their layout inference. Note that super-pixels are usually defined as image patches that better aligne with edges compare to a rectangular patch.

2.2 Model Representation

With the advent of new technologies and the increasing growth rate of indoor location-based service applications, there is much more demand for indoor space models especially from the new generation who requires more assistance indoors. Since indoor spaces including airport, hospitals, schools and shopping malls have quite complex structures, selection of a suitable model representation scheme will impact the indoor modeling research areas. Cadena et al. (2016) categorized model representation by considering two major aspects. First, *metric representation* which focuses on how to model geometry via a symbolic structure. Note that a geometric model encodes the geometry of an environment.

Second, *semantic representation* that considers geometric entities in an environment and associates them with semantic concepts. In this section, the major methods for geometric representation of models (section 2.2.1) and semantic representation of models (section 2.2.2) will be discussed.

2.2.1 Metric Representation

Three-dimensional modeling of the environment through encoding geometry has been studied in different research fields including photogrammetry, computer vision, robotics and computer graphics (Requicha, 1980; Shapiro, 2002; Flint et al., 2011). Cadena et al. (2016) mentioned different metric representations including: a) Landmark-based sparse representations, b) Low-level raw dense representations, c) Boundary and spatial-partitioning dense representations, and d) High-level object-based representations. In this section a brief review of these metric representation techniques will be provided.

Observed discriminative features in different places such as corner points and lines can be represented as a set of sparse landmarks by applying various localization and mapping techniques (Mur-Artal et al., 2015). Many of the proposed SFM or SLAM methods provide the same landmark-based representation (Triggs et al., 1999; Ackerman, 2014). In such methods, geometric aspects of the distinguishable landmarks will be measured. Since landmark-based representation is not suitable for visualisation and rendering purposes, low-level raw dense representations which can provide high-resolution geometric models are better choices in this regard. Unstructured set of point clouds can describe 3D geometry of an environment and have been successfully used in conjunction with RGB-D and stereo cameras (Nüchter, 2008). Even though these types of representations are visually attractive, they have poor quality in giving high-level topologic information which is necessary for providing geometric descriptions.

To overcome the weaknesses of using unstructured sets of low-level primitives, boundary and spatial-partitioning dense representations are introduced. Boundary representations are suitable for explicitly representing surfaces and volumes. They can detect objects surface boundaries which together define the objects. Various types of boundary representation methods are presented which include curve-based representations, implicit surface representations, plane-based models and surface mesh models (Whelan et al., 2012; Lu and Song, 2015; Whelan et al., 2015). Spatial-partitioning representations take advantage of contiguous non-intersecting primitives to define an object. In this regard, decomposing the 3D space into regular grid-based voxels is a tangible example which is applied in spatial-occupancy enumeration. Other methods for partitioning the environment include Binary Space-Partitioning (BSP) tree, octree and Polygonal Map octree (Everingham et al., 2010; Flint et al., 2011).

Apart from the abovementioned metric representations, higher-level representations have been mentioned in literatures. These types of object-based representations explicitly encode real objects in 3D and include solid shapes in their roster as well (Curless and Levoy, 1996; Salas-Moreno et al., 2013; Cieslewski et al., 2015). Note that modeling objects as solid entities enables the association of mass and volume to those objects. Cadena et al. (2016) categorized solid representations into three groups including: a) Parameterized Primitive Instancing, b) Sweep representations, and c) Constructive solid geometry. Families of objects (e.g., cubic, sphere, cylinder) are considered in parameterized primitive instancing methods, and a set of parameters (e.g., width, height, radius) are defined for uniquely identifying the objects for each family. Sweep representations consider the sweep of an object through space to define a solid entity. Leveraging symmetries, both translation sweep and rotation sweep are suitable to reason on the scene occluded areas (Bibby and Reid, 2010; Phillips et al., 2015). Complex solid entities are defined by constructive solid geometry methods applying Boolean operations on primitives (Requicha, 1980). These types of metric representations can model complex geometry by storing an object as a tree which primitives are its leaves and operations are the edges.

2.2.2 Semantic Representation

Some geometric models encounter with memory problems due to using many parameters such as points, lines and voxels to encode the entire 3D environment (Nießner et al., 2013). Using these parameters, they cannot provide high-level understanding of the geometric space. Thus, purely geometric models could not be the ultimate representation solution, and this notion opens the doors for creating semantic models of the environments. Generally, semantic models are created by adding semantic concepts to geometric entities of the environment (Bajcsy, 1988; Salas-Moreno et al., 2013). Various approaches for semantic modeling are proposed. Yet, these approaches are mainly different with respect to their adopted types of semantic concepts and their ways of associating these concepts with surrounding entities. Labeling various places (rooms) or segmenting known objects in the environment would be a tangible example of different approaches in semantic modeling (Pronobis and Jensfelt, 2012; Pillai and Leonard, 2015). Thus, semantic modeling focuses on classifying the environment according to semantic labels. Semantic parsing is one of the earliest approaches that mentioned in the literatures. Basic level of semantic parsing defines the matter as a classification problem where a relation between the predefined semantic concepts and the captured data is established.

The question about the semantic concepts relation and their numbers is normally answered by a task-driven decision which considers the organization and the level of semantic concepts. Thus, considering these two aspects a semantic representation can be built: a) level of semantic concepts, and b) organization of semantic concepts. Here, level of concepts means how much details must be considered in a representation. For example, categorizing rooms, corridors and doors in data is different from categorizing tables, chairs and books. It should be mentioned that semantic concepts are not exclusive since the number of concepts or properties of an entity is unknown. For example, an object can be stable or in motion. Moreover, both blackboard and table are pieces of woods sharing the same property but having different usability. Note that these points must be considered while arranging multiple semantic concepts.

Chapter 3

York University Indoor Datasets

Data gathering and processing are essential part in evaluation of any newly proposed algorithms. Although different kinds of datasets have been prepared to assess the quality of image based indoor models, most of these datasets were covering single rooms. Hence, for evaluating the specific performance of an algorithm which is related to indoor corridor modeling, a new type of dataset is needed. This dataset must be adapted to our research purpose so that it becomes suitable for performing the necessary assessments. In this chapter, we describe our newly generated datasets which used to evaluate the performance of our proposed methods. In the first half of this chapter, we describe datasets acquired by hand held cameras covering indoor corridor places. We collected single images and video frames datasets by crawling through different indoor corridors at York University campus area in Toronto, Canada. Different places such as Behavioural Science, Petrie Science, Osgoode Hall, Chemistry and Ross buildings are included. The two main selected test sites are: 1) first floor Petrie Science Building and 2) first floor Ross Building. For each site, acquired data types and their characteristics are thoroughly explained. Also, reference indoor corridor layouts and their respective orientation maps are provided through manually identifying corridor layouts in the image space (structural corner points positional errors are less than 3 image pixels). In the second half of this chapter, the acquired laser point clouds from the above test sites is introduced. To generate the laser benchmark dataset, we used the Trimble Indoor Mobile Mapping Solution (TIMMS). To improve TIMMS positional accuracy and geo-referencing the generated laser point clouds, several indoor control points (planar accuracies about 5mm) were delicately identified inside the selected buildings interiors through precise indoor surveying. The accuracies of TIMMS collected laser point clouds are close to 1cm relative to TIMMS positional accuracy. The prepared 3D laser point cloud is used to generate individual ground truth 3D models which further help to evaluate different aspects of our results.

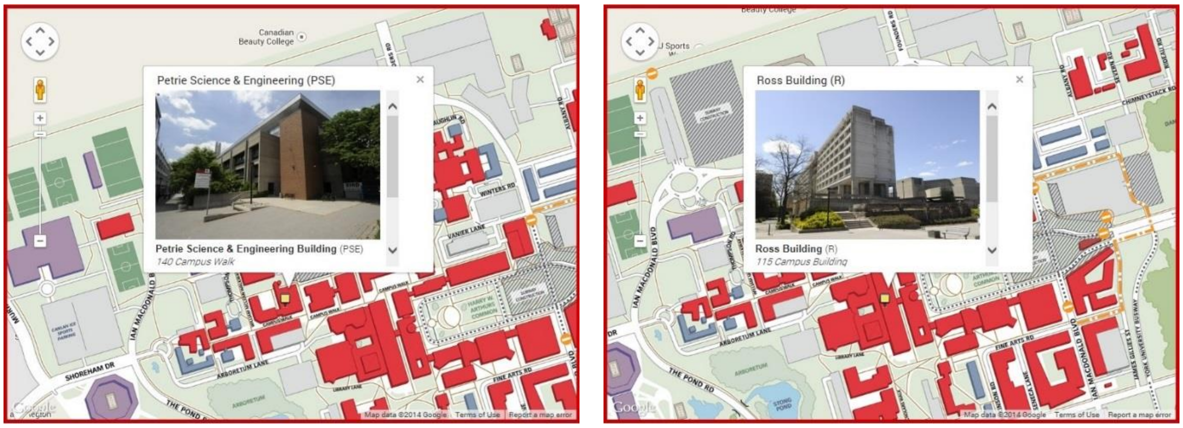
3.1 Introduction

One question might be on how to measure the quality of the generated indoor corridor models. To answer this question and to evaluate the estimated indoor corridor models by our proposed algorithms, the geometrically accurate 3D indoor corridor models should be reconstructed. Here, we introduce our newly generated datasets and describe how they captured by two different cameras and present their specific features in detail (section 3.2). Moreover, we introduce our prepared indoor laser point cloud data, a source for 3D ground truth model generation, which is used to assess the performance of our proposed Layout SLAM algorithm also presented in this thesis (section 3.3).

3.2 Imaging Datasets

In 2014 and 2016, the GeoICT research laboratory at York University initiated the generation of benchmark test datasets on 3D indoor corridor modeling. This benchmarking project supported by GeoICT research laboratory at York University, Natural Science and Engineering Research Council of Canada (NSERC) and Applanix Corporation Company that provided state-of-the-art indoor laser point clouds. The generated datasets can be used by avid individuals at GeoICT research laboratory to test their own algorithms on indoor corridor modeling and visual SLAM (Baligh Jahromi et al., 2017). By having this test dataset, we could conduct different analysis on our proposed algorithms in a less data-sensitive manner.

In this thesis, the benchmark datasets provided by the GeoICT research laboratory were utilized for the performance assessment of our proposed methods. Independent benchmark datasets were acquired over different indoor corridors at York University in Toronto, Canada. The main test sites are the first floors of both Petrie Science, and Ross Buildings. These benchmarks encompassed multi-sensor data including single images, hand held recorded videos and indoor laser scanning point cloud. In addition, we provided reference datasets, which include manually labelled indoor corridor layouts and their respective orientations in the image space. Both 3D indoor corridor models reconstructed using the estimated orthogonal vanishing points and extracted models in the object space from 3D laser point cloud, are included in this dataset as well. The GeoICT research laboratory also provided ground truth camera’s positions and orientations to facilitate the evaluation of the results produced by our proposed algorithm over the benchmark datasets. Figure 3.1 shows the location of the two main test sites at York University.



(a) (b)

Figure 3.1 Main test sites: (a) Petrie Science Building and (b) Ross Building

3.2.1 Dataset 1: Single Images

The first benchmark dataset consisted of single images covering corridors of Behavioural Science, Petrie Science, Life Science, Atkinson, Assiniboine, Osgoode Hall, Chemistry and Ross buildings at York University. We chose these buildings due to their free accessibility over time, abundancy of indoor corridors aligned with Manhattan frame structure and availability of their geometric floor plans. We walked through these buildings after working hours and inspect many corridors to take some images showing main corridors and accessory hallways with a clear view and enough resolution. To acquire images, we used two different cameras: a) Apple iPhone 4s (cell phone) back camera, and b) GoPro Hero5 camera. Table 3.1 presents some of the specifications of these two cameras. Both two cameras went through on the field calibration procedure. Hence, all the captured images were corrected for distortions. The calibration procedure will be explained later in this chapter.

Table 3.1 Specifications of digital cameras used for our dataset generation

Camera	*Image Format*		*Field of View (degree)*	*Focal length (mm)*	*Video Format*		*Frame rate (per sec)*
	Row (pixel)	*Col (pixel)*			*Row (pixel)*	*Col (pixel)*	
Apple iPhone 4s	2,448	3,264	56.423˚	4.28	1,080	1,920	30
GoPro Hero5	3,000	4,000	149.20˚	16.80	2,160	3,840	30

Statistic-wise, images taken from the selected places are covering in total 297 corridors, 1283 walls, 206 doors, and 53 windows. The number of images for each corridor ranges from 1 to 9, with the total number of selected images in our first dataset being 78, not counting the single room images. Figure 3.2 depicts some of the images in this dataset captured by iPhone 4s camera.

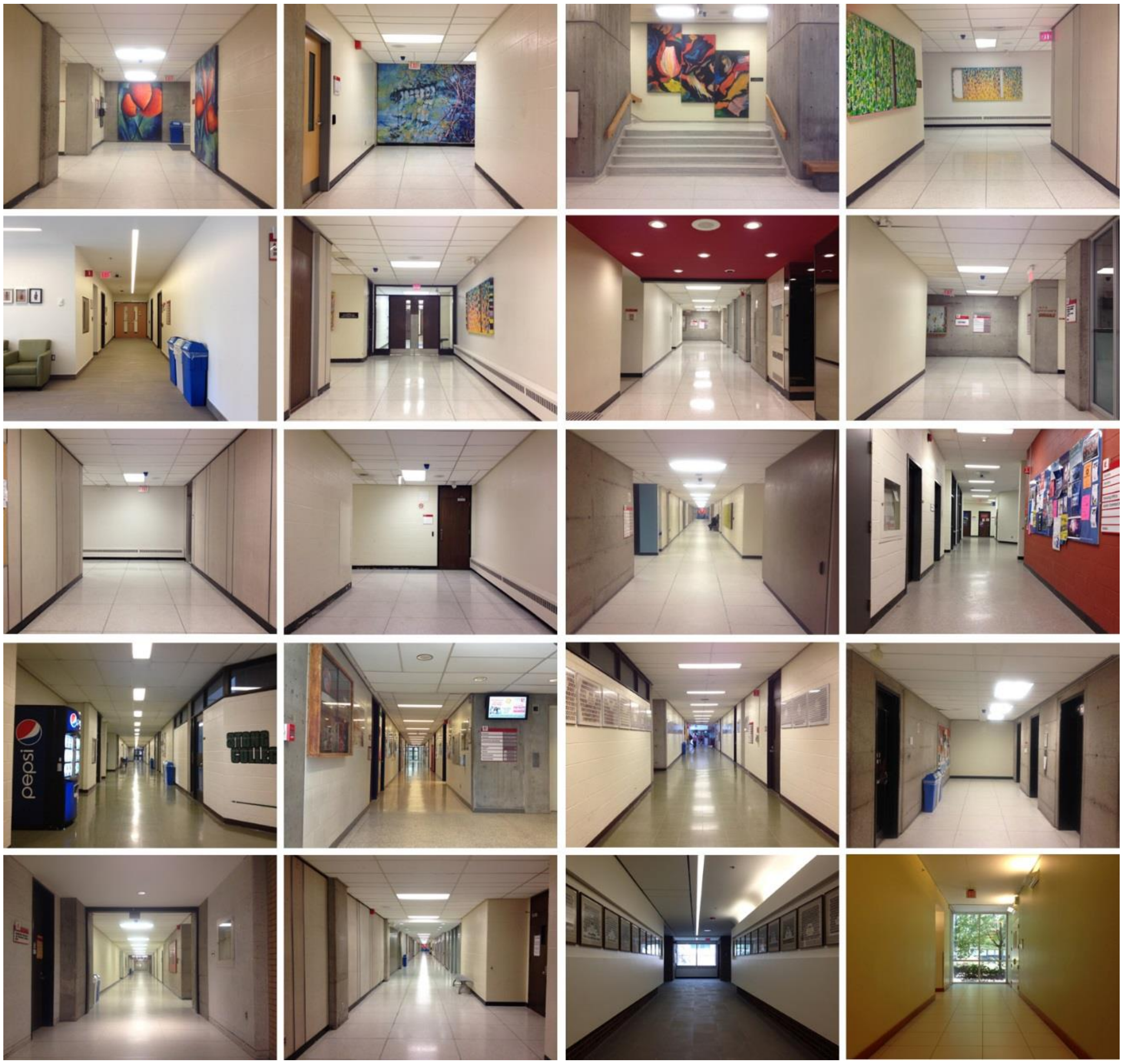
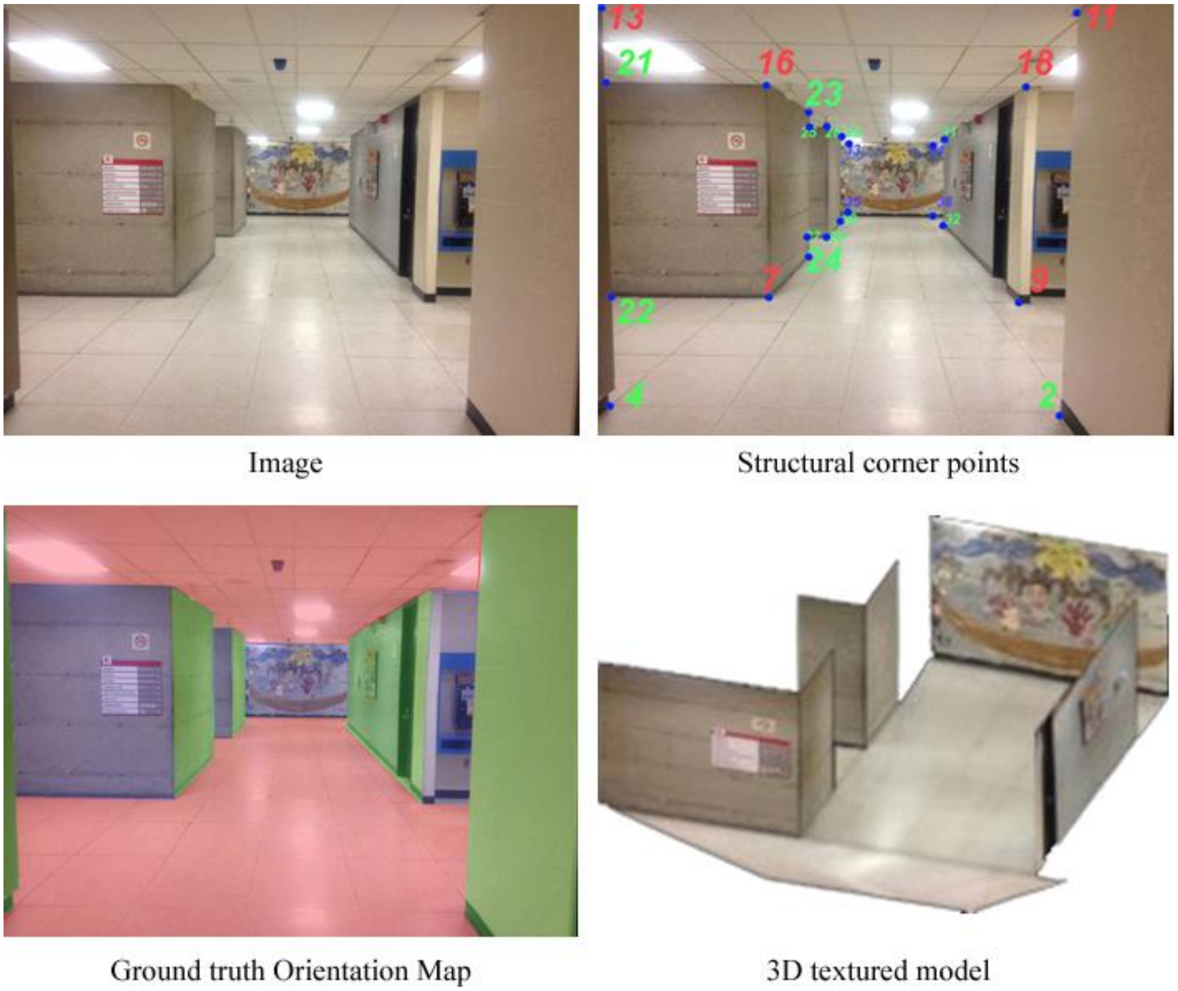


Figure 3.2 Sample images from our dataset taken by an iPhone 4s camera.

Not necessarily all indoor corridors have a rectangular layout. However, considering indoor corridors with a complex polygonal shape is beyond the scope of this study. Hence, only corridors having simple and rectangular outline were included in this dataset to conform the Manhattan frame assumption. Moreover, the prepared dataset is accompanied by different types of ground-truths including corridors layouts, orientation maps and 3D reconstructed corridor models. It should be noted that camera calibration is performed to estimate intrinsic camera parameters and image observations are undistorted prior to ground truth data generation. Also, 3D reconstruction is performed using orthogonal vanishing points while assuming camera height is 1, due to inability to measure absolute distances from single images. In the coming sub-sections these two subjects will further be explained.

In all images, the ceiling, floor, front, left, and right walls of the main corridor are identified as well as the ceiling, floor, right or left walls of each accessory hallways (side corridors) visible in the image. To identify these structural polygons and planes, the image coordinates of their respective structural corner points manually pinpointed with less than 3 pixels error using MATLAB R2009a software. If the respective pinpointed corner points are connected clockwise, they can identify each polygon’s boundaries in the image space. The structural corner points coordinate of each image were saved in different individual files. Later, these structural corner points coordinates were used while creating the ground truth orientation maps and indoor corridor layout 3D models. Figure 3.3 shows a sample image from the prepared dataset along with identified structural corner points, ground truth orientation map, and its respective 3D textured corridor model.



Image

Structural corner points

Ground truth Orientation Map

3D textured model

Figure 3.3 Ground truth orientation map and indoor corridor 3D model associated with a sample image from the prepared dataset.

It should be noted that in some cases pinpointing the structural corner points in one image could be a very challenging task. This challenge rises from the complex polygonal shape of the corridor’s indoor layers. In most cases the problem could be resolved by considering semantics such as scene type, presence of doors or windows, and moreover presence of furniture and their locations. Yet, there were still cases where the correct corridor layout could not be completely identified in the image space. Rationally, we allowed ourselves to remove the ambiguous examples from this dataset. The available images in the prepared dataset can be categorized by considering corridors lengths which directly affect the corridor scene complexity. Even though explicitly expressing the impact of corridors lengths on the overall corridor scene complexity is ambiguous, nobody can neglect the affect of this factor while generating ground truths for the prepared dataset. Hence, the dataset is partitioned into two image categories that cover long and short-range corridor lengths.

3.2.1.1 Camera Calibration

In this study, the appropriate model for the applied cameras is a pinhole camera model. This model recognizes the camera by a flat image plane and a perspective centre (a light-barrier hole). For every image point a ray of light can be estimated which highlights the optical path. This ray can be reconstructed while knowing the intrinsic camera parameters. Camera calibration is a procedure for the estimation of intrinsic camera parameters. Intrinsic camera parameters are revealing the internal characteristics of a camera. Generally, these parameters include principal point coordinate, focal length, skew and image distortion. Knowledge about the intrinsic parameters of a camera is an essential first step for 3D reconstruction. Through the camera’s intrinsic parameters, the structure of a scene in Euclidean space can be estimated. Also, lens distortion can be removed to improve the accuracy of this estimation. To perform the camera calibration, a fully automated assisted calibration can be applied. In this thesis, cameras are calibrated using MATLAB R2009a calibration toolbox (Bouguet, 2004).

3.2.1.2 Converting Corridor Layouts to 3D

Two-dimensional corridor layouts in image space can encode valid 3D models applying a few assumptions. Here, 3D coordinates can be computed sequentially without ambiguity. First the coordinates can be computed for floor and then for walls applying the constraint (walls and floor are attached) and ceiling at the end. If assuming the camera's height (distance between the camera and the floor plane) is equal to 1, then all metrics units will be in camera height. This assumption is needed since absolute distances can not be observed in single images.

Applying the homogeneous coordinate system, 2D and 3D coordinates are represented by small and capital letters, respectively. For example, vertical vanishing points $v_1= (x_1, y_1, 1)^T$ and $V_1= (X_1, Y_1, Z_1, 1)^T$ are representing the vertical directions in 2D and 3D spaces, respectively. Also, camera intrinsic parameter matrix is represented by *K* and a point in image space can be represented by *p*. Thus, having coordinates of three vanishing points (x_i, y_i) in image space, a ray and the three main axes normal directions can be expressed as:

$$\begin{gathered} P = \beta K^{-1}p \quad , \quad \beta > 0 \\ v_i = (x_i, y_i, 1)^T \\ V_i = \frac{K^{-1}v_i}{||K^{-1}v_i||_2} \end{gathered} \tag{3.1}$$

In the above equations, β is the scale factor and (x, y) are representing image coordinates. If normalizing the height to 1, the 3D coordinate of a point on the floor could be:

$$P = \frac{K^{-1}p}{V_1^T K^{-1}p} \tag{3.2}$$

Also, the height h between two image points p_1 and p_2 while p_1 is a point residing on the floor can be calculated. Here, we assume that in 3D space P_1 and P_2 are vertically aligned. Therefore, p_1, p_2 and v_1 must somehow be in line while applying the following equation:

$$\begin{gathered} P_2 = \beta K^{-1}p_2 = P_1 + hV_1 \\ = \frac{K^{-1}p_1}{V_1^T K^{-1}p_1} + hV_1 \\ [-V_1 \quad K^{-1}p_2] \begin{bmatrix} h \\ \beta \end{bmatrix} = \frac{K^{-1}p_1}{V_1^T K^{-1}p_1} \end{gathered} \tag{3.3}$$

Note that h can be calculated through solving least-squares. A recovered 3D model is depicted in figure 3.3.

3.2.2 Dataset 2: Video Frames

This dataset covers first floors of both Petrie Science and Ross buildings at York University. Video frames were acquired by Apple iPhone 4s back camera in 2014 and by

GoPro Hero5 camera in 2016. All video frames were taken using hand-held shooting technique in which the camera is placed in the operator’s hand while he is passing through the corridors with normal speed. Thus, some frames were shaky and not as stable as the frames that could be captured by tripod-mounted cameras. Videos were captured at both day and night times and always starts and ends at the same location forming a close loop. This prepared data set contains five individual videos that their duration varies from 3 to 12 minutes.

3.3 Indoor Laser Point Cloud Dataset

Point clouds are valuable materials for generating accurate 3D models. Theoretically, a 3D point cloud consists of a set of individual points in a three-dimensional reference frame. Point clouds are normally obtained from laser scanning or digital imagery. Even though imagery solutions can provide 3D point clouds through Structure from Motion (SFM) or visual Simultaneous Localisation and Mapping (SLAM) techniques, surfaces with no textures and problematic lighting conditions in indoor places make laser scanning to be the most promising technique for point cloud generation.

Laser scanners obtain dense point clouds from range measurements. Normally, a laser scanner emits a laser beam which is reflected by a rotating mirror to obtain a profile of the environment. Since the scanner is simultaneously rotating, the 3D scan of the environment can be generated. If the scanner is mounted on top of a tripod on the ground, terrestrial laser scanning (TLS) is possible which is the most precise way to obtain point clouds. To obtain full coverage of the surrounding space by TLS, multiple scanning locations may be needed due to occlusions. In such cases, several tripod scans must be combined which necessitates human intervention in the post processing stage.

Indoor mobile scanning can provide more cost-effective solutions since it performs faster than the TLS technique. Note that range measurements by mobile scanners cannot be used individually and it is mandatory to know the origin and direction of the measurements. Usually the scanner is sitting on top of a platform and its position and attitude are known. So, scanning and moving happen simultaneously which may reduce the accuracy of the point cloud because estimating the scanner pose might be inaccurate itself. Hence, the challenge is to improve the accuracy of the estimated platform's pose to reach the accuracy level that is suitable for indoor modelling. This scanner pose determination should be continuous even though satellite signals may not be available at indoor places. The solution is to incorporate auxiliary data and acquire platform trajectory in relative manner by correlating new measurements with earlier ones acquired at the beginning of the scan.

3.3.1 Trimble Indoor Mobile Mapping Solution (TIMMS)

To generate the indoor laser point cloud benchmark dataset, we used the Trimble Indoor Mobile Mapping Solution (TIMMS). This laser point cloud dataset in total covers corridors of Petrie Science (first, second and third floors), Chemistry (first and second floors) and Ross (first floor) buildings at York University. The point cloud acquisition from these buildings accomplished in a very short time. The whole data acquisition on 6 floors of these three buildings took approximately 4.5 hours. TIMMS needed 20 minutes initialization time at each building to fine-tune its Inertial Measuring Unit (IMU) for calculating the prior drift information. Hence, the actual scanning time for each building was approximately 1 hour.

TIMMS is the combination of various technologies for acquiring high precision indoor laser point clouds. TIMMS mounts on a cart moved by an operator. It primarily includes two sideway laser scanners which can acquire point clouds orthogonal to the direction of moving. The mounted laser scanners on the cart were Faro Focus X130 laser effective up to 130m with ranging error of ±2mm. Type 1 laser class with 1550nm wavelength. With a maximum scanning frequency, the vertical field of view is 300° and 360° in horizontal. Also, a very precise IMU contributes to the position estimation improvement through identifying the drift vectors pitch, roll and yaw on every movement. The integration of IMU heavily impact the amount of drift error and the accumulated error remains low over relatively long corridors. Thus, the accuracy of collected laser point clouds was close to 1 cm relative to position accuracy (root mean square derived by comparison of TIMMS with static laser scan). Also, spherical camera is mounted on the platform which collects images of the environment. Figure 3.4 shows the TIMMS on a mission at York University.

Note that initializing TIMMS at an indoor place without providing information about the global reference frame, enforce the incoming scan to be defined in an arbitrary coordinate system. Moreover, the other scans can be referenced to the coordinate system of the primary scan through tie points which results in a common coordinate system for the dataset. To address the geo-referencing problem and provide the absolute coordinates to the incoming TIMMS laser point cloud, indoor space control points in the global reference frame are needed.

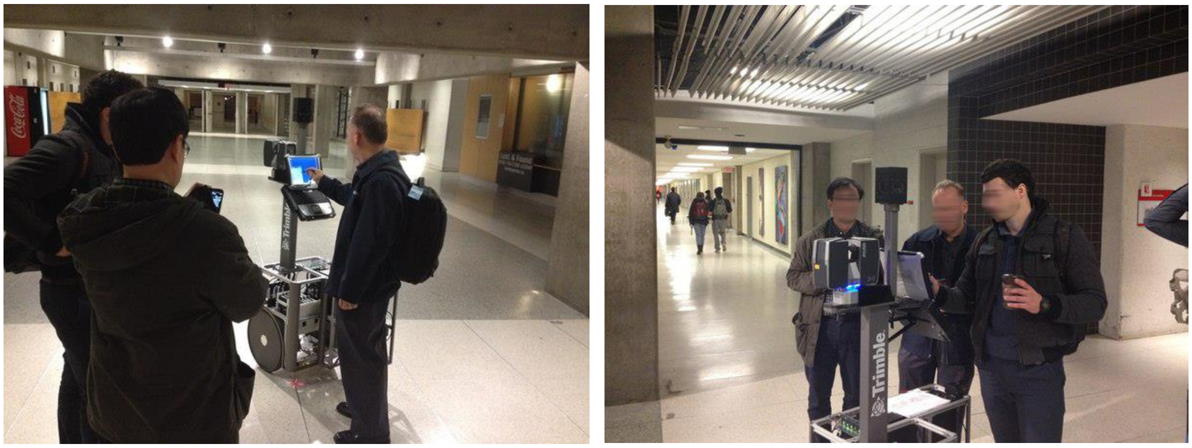
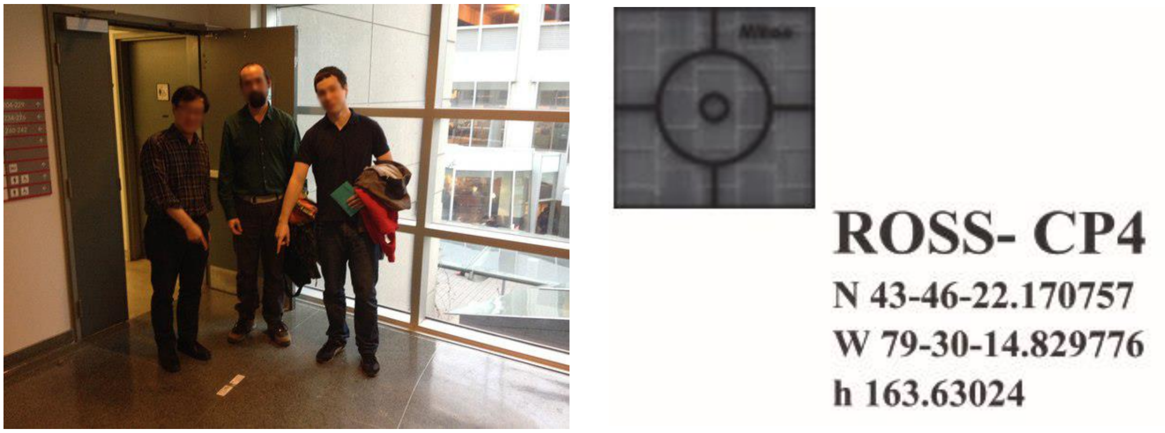


Figure 3.4 TIMMS on a mission at York University.

3.3.2 Indoor Space Control Points

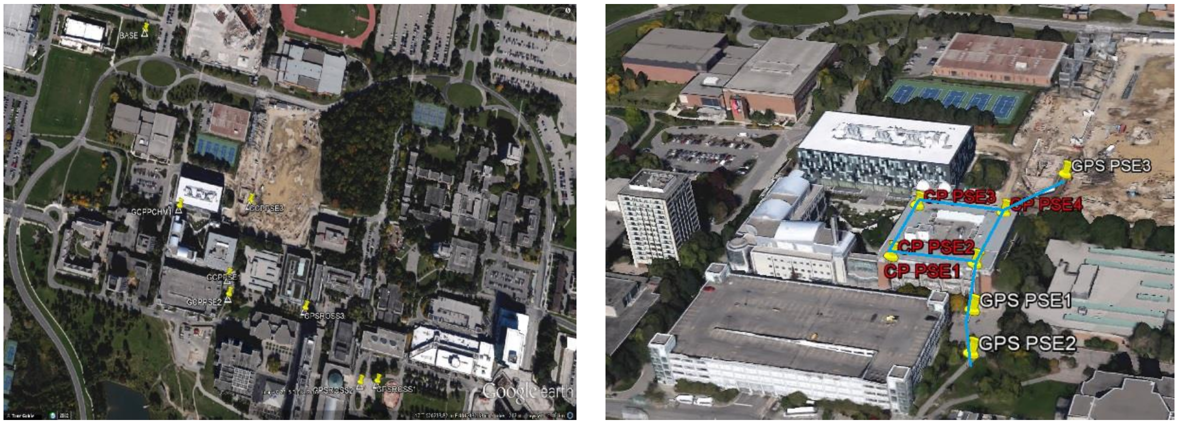
The geo-referencing of TIMMS laser point clouds into some global coordinate systems necessities the presence of indoor space control points related to those global coordinate systems. To acquire high accuracy laser point cloud through TIMMS, reference points that could be tied to some outdoor Global Navigation Satellite System (GNSS) points were delicately identified inside the selected buildings interiors. Here, the related indoor control points (planar accuracy about 5mm) were collected prior to TIMMS data acquisition through precise indoor surveying. Precise indoor surveying is encountered with various challenges in comparison to traditional field surveying including network visibility design, path planning, scheduling and logistics. To provide GNSS related control points inside the buildings, 29 photogrammetric targets in total had been placed on the selected floors (Figure 3.5).



(a) (b)

Figure 3.5 (a) An indoor control point on second floor of Chemistry building and (b) The applied photogrammetric target and its attached global coordinate at Ross building.

Overall, two sets of control points are provided inside and outside of the selected buildings. The first layer of survey control network was established outside the selected buildings which consists of seven GPS observed points. Note that the minimum number of GNSS reference points for establishing the survey control network was seven. Figure 3.6(a) shows the distribution of these control points inside York University Keele campus area.



(a) (b)

Figure 3.6 (a) GPS observed control points inside York University Keele campus area and (b) The first-floor traverse network of Petrie Science building.

For each selected building, the two most visible GPS points from the stablished control network were used as the end points of the indoor corridors traverse network (Figure 3.6(b)). The traverse is the second layer of the established network which was also adjusted by holding the two GPS end points as known. The traverse enabled the creation of various stations which were used as set up points to measure coordinates of the installed photogrammetric targets on the floors. To determine the coordinates of the local network GPS (Trimble R8) and Total Station (Leica TCA 1800) were used. All the observations were done in the double run to get the better accuracy. Total station was used to determine the vertical and horizontal angles along with the slope distances inside the buildings and GPS was used to calculate the baselines and the exact coordinates of the outdoor survey control points. In the end coordinates of the photogrammetric targets on the floors were determined, considering the WGS84 datum as the reference.

3.3.2.1 GPS Data Processing

To establish the GPS control network, the Real-time kinematic (RTK) positioning technique is applied. The RTK positioning is a technique for improving the position data precision derived from satellite-based positioning systems like GPS. RTK uses the signal information content and relies on a single base-station to provide real-time corrections while measuring the signal's carrier wave phase. Here, the applied RTK system comprised of a single base-station receiver near Tait McKenzie Centre at York University Keele campus area, and 7 mobile units close to the selected buildings. Theoretically, the base station observes the phase of the carrier and re-broadcasts it to the mobile units. The mobile units compare their own phase measurements with the one received from the base station. Comparing the phase measurements of the mobile units to the one provided by the base station, enhancement of the position data precision is achieved.

The collected data was processed through Trimble R8 Office program which computed the relative baseline. Here, the baseline between the two receivers is not directly measured from the satellite observations. Instead the baselines are derived from the measured coordinates of each station. The processing of GPS data to form baselines involves forming linear combinations of the phase observables and their subsequent adjustments. Having baseline computed, a-posteriori reference variance, estimated integer ambiguities, residual plots and percentage of the rejected data can be used to indicate the quality of the baseline. The estimated standard deviation and covariance matrix of the parameters can be used as well to indicate the quality of the solution. Finally, the coordinates can be returned in geodetic (λ,ϕ,h) form on WGS84 ellipsoid.

Note that the cut off angle was set at 15 degrees which effectively blocks any satellite signals received at low elevation angles. Signals at low elevation angles are less accurate due to their further travel through the atmosphere. Also, the broadcast ephemeris is used for locating the position of the satellite relative to the receiver. The precise ephemeris data obtained to achieve a more accurate solution by performing a later data processing. Here, Hopfield model is used for the tropospheric delay corrections and the ionosphere corrections were automatic.

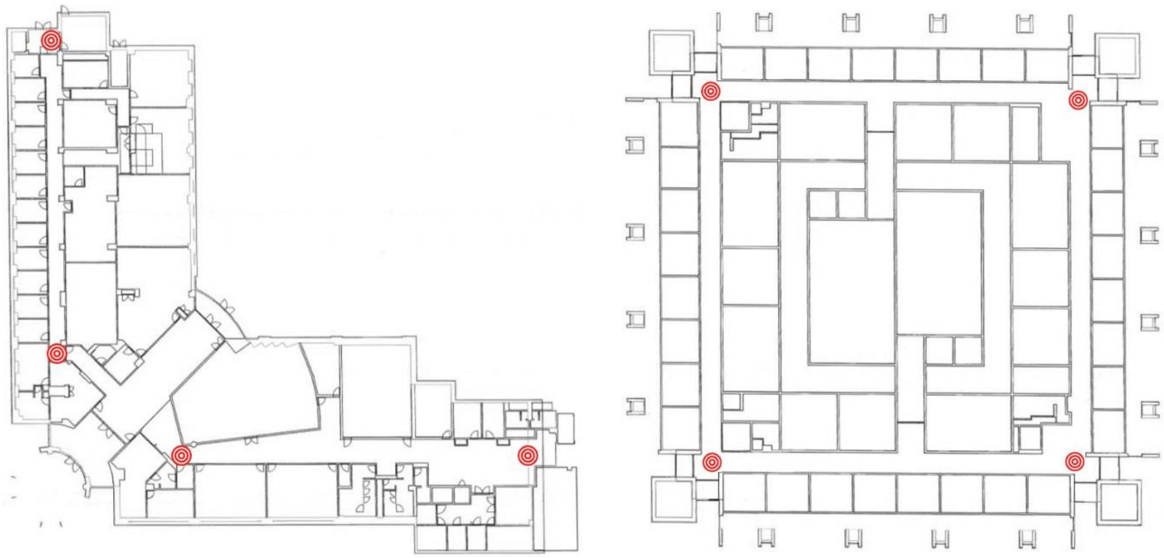
Table 3.2 Calculated geodetic coordinates of the observed GPS mobile units.

Station	*Latitude*	*Longitude*	*Horizontal SD*	*Ellipsoidal height*	*Vertical SD*
GPS PSE1	N 43-46-24.060740	W 79-30-23.602200	0.001 m	162.58480	0.002 m
GPS PSE2	N 43-46-23.297980	W 79-30-22.932540	0.005 m	162.03980	0.011 m
GPS PSE3	N 43-46-27.087990	W 79-30-24.678450	0.001 m	163.46880	0.002 m
GPS ROSS1	N 43-46-23.986550	W 79-30-09.748780	0.001 m	163.82580	0.002 m
GPS ROSS2	N 43-46-23.450670	W 79-30-10.869080	0.001 m	163.57880	0.002 m
GPS ROSS3	N 43-46-24.979450	W 79-30-17.161850	0.001 m	163.29480	0.002 m
GPS CHM1	N 43-46-25.650920	W 79-30-29.564730	0.005 m	159.67480	0.011 m

It should be noted that the GPS observation periods were about 12 hours for the base station and more than 30 minutes for every mobile units. After identifying the reliable observations, least squares adjustment is performed to adjust the independent baseline in the network. Finally, the incoming results of the adjusted GPS coordinates and their standard deviations are used as an input to the second layer network which was established inside the selected buildings. Table 3.2 represents the calculated geodetic coordinates for the observed GPS mobile units.

3.3.2.2 Traverse Network

The traverse was designed for the second layer network of control points which was supported by the first layer control network created by the GPS mobile units. The traverse network was processed in a similar way to the 3D Control Network and similar checks and corrections were applied to the observations. It should be noted that this process is completely performed using the latest version of the Columbus software in October 2014. The coordinates of GPS control points along with their respective reference directions achieved through the control network adjustment, were introduced to the Columbus software. Note that GPS control points are considered as “fixed” points in the traverse network. In the next step, the Total Station observations were added to the GPS control pints inputs in a specific order (To, From, Zenith, Direction, Distances, and standard deviations). Once all data is inputted, the Columbus software was able to perform the adjustment and provide coordinates to the unknown traverse points accompanied by their standard deviations. Figure 3.7 shows the positions of the unknown traverse points (photogrammetric targets) on the floor plan of the Chemistry and Petrie Science buildings first floors.



(a) (b)

Figure 3.7 (a) Places of indoor traverse points on the first floor of Chemistry building and (b) Places of indoor traverse points on the first floor of Petrie Science building.

As mentioned before, three different buildings at York University were selected for TIMMS indoor scanning which two of them had to be scanned on different floors (more than one). Hence, four different traverse networks were established and each of them were adjusted individually using the GPS control points as the reference.

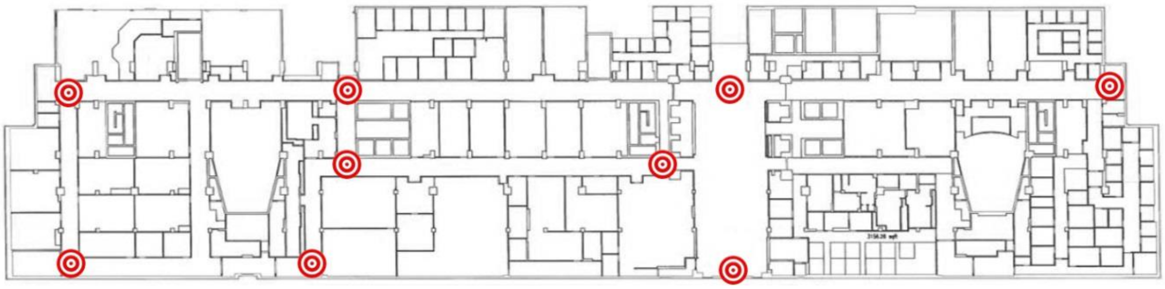


Figure 3.8 Positions of the traverse points on the floor plan of the Ross building first floor

Table 3.3 reveals the adjusted MTM zone 10 coordinates for the traverse points inside Ross building. Ellipsoidal heights are considered at this network so that control points could be easily used by TIMMS. It should be noted that Ross building traverse network was the most complex traverse that was set up for the buildings. Figure 3.8 depicts the positions of some traverse points on the floor plan of the Ross building first floor.

Table 3.3 Adjusted MTM zone 10 coordinates of traverse points inside Ross building.

Points	*North*	*North SD*	*East*	*East SD*	*Ellipsoid height*	*Height SD*
CP1	4848121.73733	0.00175	304513.15381	0.00195	163.65874	0.01563
CP2	4848111.21423	0.00287	304480.25050	0.00265	163.63225	0.02030
CP3	4848101.35738	0.00348	304466.73639	0.00324	163.63255	0.02184
CP4	4848096.12341	0.00337	304468.37515	0.00336	163.63024	0.02215
CP5	4848046.85161	0.00271	304483.90065	0.00451	163.63033	0.02562
CP6	4848041.44321	0.00305	304486.62510	0.00474	163.63750	0.02597
CP7	4848046.03893	0.00302	304501.19422	0.00454	163.63724	0.02706
CP8	4848008.58484	0.00288	304512.97212	0.00553	163.63790	0.02946
CP9	4847999.37359	0.00290	304486.88583	0.00556	163.63128	0.02909
CP10	4848043.44458	0.00269	304473.01517	0.00424	163.62788	0.02535
CP11	4848103.02003	0.00291	304454.24657	0.00281	163.64091	0.02224
CP12	4848163.58345	0.00294	304435.13507	0.00212	163.67731	0.01702
CP13	4848158.76405	0.00270	304420.35663	0.00198	163.66806	0.01538

3.3.3 TIMMS Data

TIMMS scanning at York University was added by the prepared indoor control points inside the selected buildings. TIMMS initialized itself using one of the indoor control points inside the buildings and managed the occurred drifts through regularly visiting the rest of indoor control points on the go. Since the prepared indoor control points were linked to the outdoor GPS control points, the incoming TIMMS laser point clouds had absolute coordinates in World Geodetic System 1984 (WGS84) datum as well. Hence, there would be no need to register the incoming laser point clouds of selected buildings together since they have already registered in a common coordinate system.

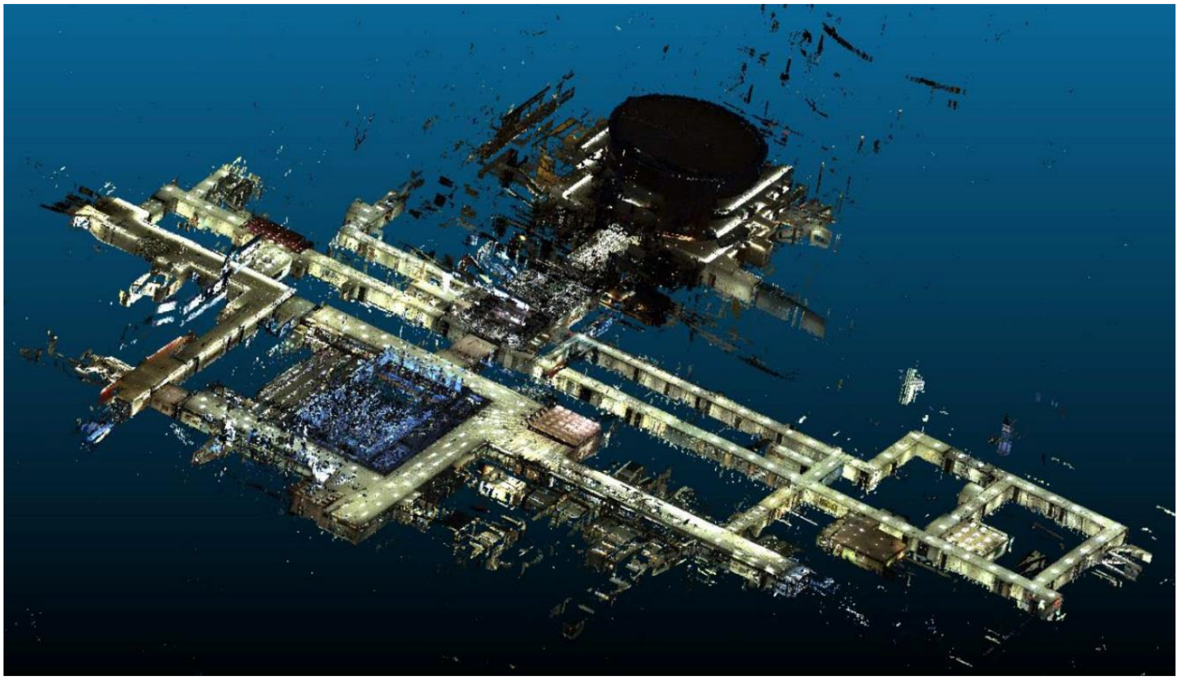


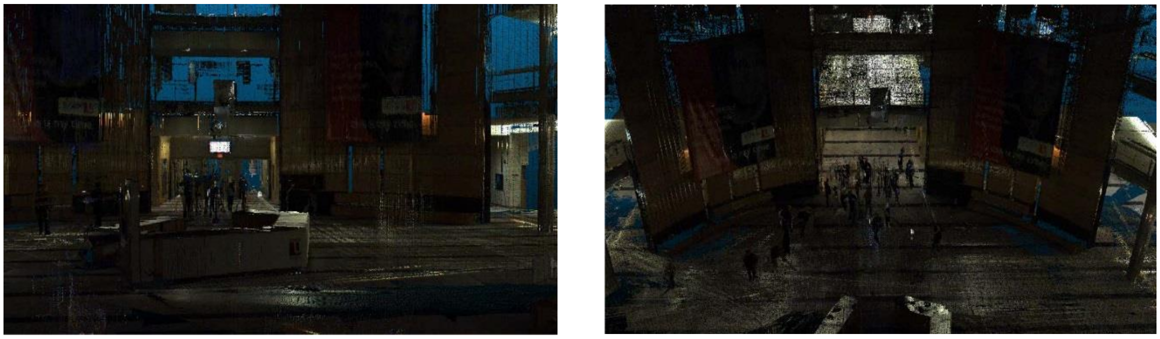
Figure 3.9 Raw laser point cloud acquired by TIMMS at Ross building

The incoming TIMMS laser point cloud data was pre-processed by Applanix Company technicians and provided to GeoICT laboratory in "LAZ" format. The data's volume is approximately 10 gigabytes and it was delivered with 4 different resolutions including 0.5, 1, 5- and 20-centimeters resolutions. Because the data's volume is too large, it was partitioned into the small sets of point clouds. This data partitioning is necessary due to the large size of the selected buildings and abundancy of lengthy indoor corridors. Partitioning raw data into many pieces introduced advantageous in detection of noises which are imposed by the long-range laser scanners. Figure 3.9 shows a snapshot of the raw laser point cloud acquired by TIMMS at Ross building.

3.3.3.1 Data Structural Units

As mentioned before, three different buildings at York University Keele campus area are individually scanned by TIMMS. Each building can be considered as a combination of different structural units together encompassing the objects. The structural units together can form corridors, hallways, offices etc. The acquired TIMMS laser point cloud includes several different structural units such as floors, ceilings, walls, doors, windows, stairs, dome etc. Also, it includes different objects such as tables, chairs, cabinets, bookshelves, computers, monitors etc.

The incoming TIMMS laser point cloud includes many points which are captured from the indoor corridors. In this dataset, indoor corridors are the most fundamental blocks of the buildings along with individual class rooms. These corridors provide the accessory networks inside the buildings. In this dataset, corridors of Chemistry and Petrie Science buildings are rectangular and approximately occupy cubic spaces. However, Ross building has some variations with larger corridors connecting bigger spaces including hallways and lecture halls. In Ross building laser point cloud, the connectivity is more between open spaces like corridors and halls rather than enclosed spaces like class rooms. Ross's main corridors are together connected to a 3-floor structure circling all over the center (Vari Hall). TIMMS could scan this huge one-piece space with no difference to a narrow corridor, using the 130-metre range laser scanner. Figure 3.10 shows TIMMS data covering the major parts of the Vari Hall.



(a) (b)

Figure 3.10 (a) Vari Hall TIMMS data ground view and (b) Vari Hall TIMMS data oblique view.

Note that most of the entrances in this dataset are covered by glass doors, with Ross and Chemistry buildings having the largest main entrances of all. Having glass doors at the entrances increase the chance of laser lights escaping from the indoor spaces and producing many outlier points which together they form a conical shape. This necessitates a more in-depth pre-processing procedure to be performed on the raw data to remove the outliers from the outcoming laser point cloud.

Beside corridors and halls, the prepared TIMMS laser data includes one Graduate Student Lounge (attached to Ross Building south corridor) and several individual office units directly connected to the main corridors. These places were scanned from inside by occupant's permission. This enables the prepared laser dataset to become more suitable for applying room modeling algorithms, beside being an enrich set for employing indoor corridor modeling algorithms. Figure 3.11 shows an oblique view of the laser data captured by TIMMS from Graduate Student Lounge at York University.

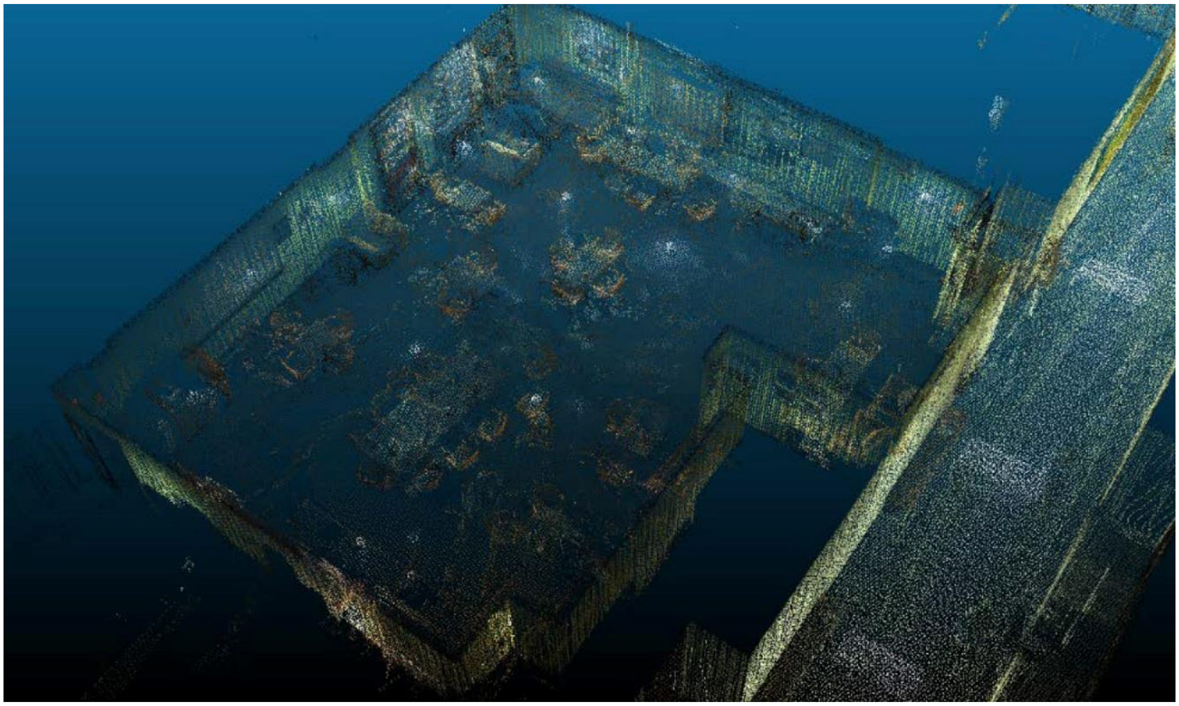


Figure 3.11 TIMMS data captured from Graduate Student Lounge at York University.

3.3.3.2 Ground Truth Corridor Models

The TIMMS scanning project at York University on 3D indoor space modeling led by Applanix Company in a collaboration with GeoICT laboratory provides precise and dense point clouds of indoor corridors. This dataset can be used as a valuable source for creating ground truth indoor corridor models. Since our video frame dataset is mostly covering the same areas as TIMMS laser data, ground truth 2D and 3D models can be generated using both datasets.

Chapters four and five in this thesis are introducing single image indoor modeling and layout SLAM algorithms, respectively. To evaluate the results obtained from the proposed algorithms, ground truth corridor models were manually extracted from available datasets. Here, first floors of both Petrie Science and Ross buildings were chosen for this task. At the first stage, planes are fitted to TIMMS laser data with 1-centimeter margin using Cloud Compare software and its plane fitting algorithm. Next, the corridors constructive planes are manually identified, and the rest of planes are discarded. Next, the remaining planes were intersected to identify the true junctions of the corridors. The 3D coordinates of the corridor junctions together construct the 3D ground truth corridor models of the selected places.

Having identified the 3D ground truth corridor models, several keyframes (9 and 32 frames from Petrie Science and Ross buildings, respectively) were selected from the video frame dataset. For all these keyframes, the ground truth 2D corridor layouts were manually identified in the image space with less than 3 pixels accuracy. The respective 2D junction coordinates were preserved in separate text files. Figure 3.12 shows a sample keyframe image with manually identified constructive lines of the 2D layout model and the reference 3D laser data of Ross building south section.

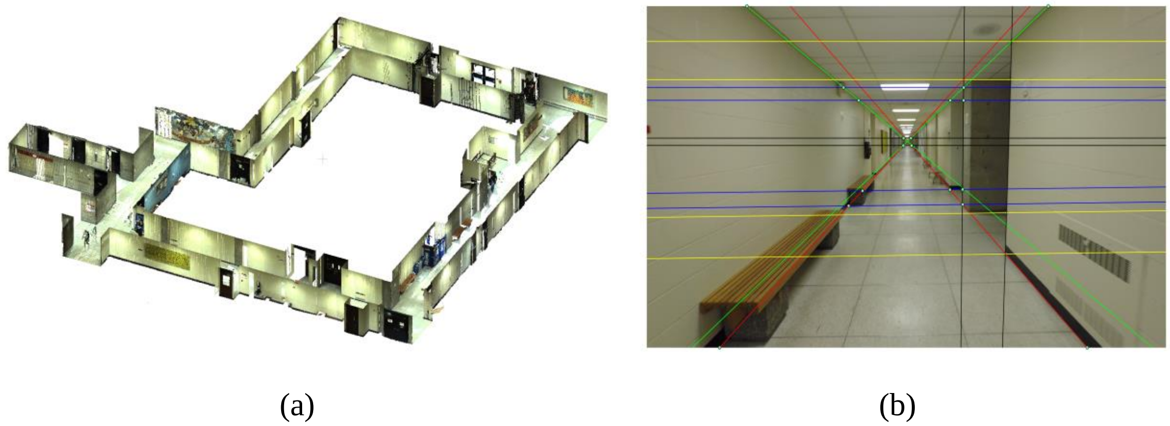


Figure 3.12 (a) Reference 3D laser data of Ross building south section and (b) A sample keyframe image with manually identified constructive lines of the 2D layout model.

Having identified both 2D and 3D coordinates of an indoor corridor’s junctions, the camera trajectories (camera positions and orientations) were estimated through photogrammetric space resection. The created ground truth corridor models were used to evaluate the proposed Layout SLAM algorithm.

3.4 Summary

In this chapter, we presented the prepared datasets and their characteristics which used for assessing our proposed algorithms. The detailed descriptions of imaging datasets were given in the first half of this chapter. The second half summarized TIMMS laser point cloud and the generated ground truth corridor 2D and 3D models. Using the 2D and 3D ground truth corridor models, the camera trajectories could be estimated through space resection. These datasets were fully used to evaluate the proposed image based indoor modeling and Layout SLAM algorithms through assessing different quality aspects of the generated 3D corridor models produced by these algorithms. In addition, we introduced new indoor space control networks in a global reference frame. These networks can be used as a benchmark for testing indoor space navigation algorithms. Even though this benchmark was not directly used to evaluate our Layout SLAM related results, in future works could be used to explicitly assess trajectory inconsistencies of robot-based SLAM algorithms.

Chapter 4

3D Reconstruction of Indoor Corridor Layout Using Single Image

In this chapter, the problem of indoor space modeling from a single image is tackled through middle-level perceptual organization. The search is for the corridor layout that can be translated into a physically plausible 3D model. Considering the Manhattan Rule Assumption, the stochastic approach is adopted to sequentially generate many physically valid layout hypotheses from image line segments. Each generated hypothesis will be scored for finding the one that best matches the scene. Finally, the best created hypothesis will be converted to a 3D model. The main contribution of the proposed method is providing an approach to create layout of indoor corridors in a hybrid way using both detected line segments and virtually generated rays from vanishing points. This method is beneficial for two main reasons. First, the hybrid way of generating scene layout provides a realistic solution when dealing with objects or occlusions in the scene. Moreover, it is well-suited to describe most corridor spaces. Since virtual rays used for layout creation are usually deviating from the true layout in long corridors due to the inaccuracy of the estimated vanishing points. Note that only using physical line segments for scene layout generation would be inefficient due to their inability to handle occlusions. Second, we considered different scoring functions to score the created layout hypotheses. These functions consider the volumetric aspect of the created hypotheses along with their correspondences to real edges, and compatibility to the orientation map and geometric context. These scoring function finds the most fitting solution in a linear way.

4.1 Introduction

Over the course of time, humans have changed their lifestyles and increasingly become indoor creatures. Thus, studying human's indoor activities and related issues in health, security and energy consumption is an important research field (Rassia, 2017). Usually, researchers in this field need to have access to spatial information of indoor spaces. However, unlikely in outdoor environment, not much spatial information of indoor spaces is available.

In recent years, spatial information of indoor spaces provided in the context of Building Science and Building Information Model (BIM) which include semantically rich and geometrically accurate indoor models has gained a lot of attention not only in the architectural field but also in other engineering communities. The generation of an indoor space 3D model needs a proper implementation of sensors as well as selecting a proper algorithm to reconstruct 3D models from the incoming data. Considering the available data gathering sensors (laser scanners, single and stereo cameras, RGBD cameras, etc.) and paying special attention to data processing time and sensor's cost, single cameras providing single images could be one of the reliable sources. Usually, a single image can cover a limited field of view and a large-scale environment may not be handled with a single image. Even though recovering the 3D model from a single image is inherently an ill-posed problem, single images are still suitable for modeling well-structured indoor corridor environments.

The early attempts on understanding the scenes start by recovering vanishing points and camera parameters from an image using straight line segments (Kosecka and Zhang, 2002). Considering the Manhattan World Assumption, rectangular surfaces aligned with main orientations were detected using vanishing points (Kosecka and Zhang, 2005; Micusik et al., 2008). Han and Zhu (2005) applied top-down grammars on detected line segments for finding grid or box patterns aligned with vanishing points in an image. Vanishing points were also used by Yu et al., (2008) to infer the relative depth-order of partial rectangular regions in the image. Parameterized models of indoor environments which were fully constrained by specific rules introduced to guarantee physical validity (Lee et al., 2009). Possible spatial layout hypothesis is sampled from collection of straight-line segments, but the method is not able to handle occlusions and fits room to object surfaces.

Vanishing points are not the only cues for understanding the scenes. Statistical learning showed to be an alternative to rule-based approaches (Hoiem et al., 2005; Delage et al., 2006; Hoiem et al., 2007). The statistical methods on image properties were used to estimate regional orientations and vertical regions “popup” considering the estimated orientations (Hoiem et al., 2005). Having a new image, the list of extracted features should be evaluated. The associations of these features with 3D attributes can be learned from training images. Therefore, the most likely 3D attributes can be retrieved from the memory of associations.

Although scene understanding is somehow feasible through applying statistical learning or rule-based approaches, fully inferring 3D information from a single image is still a challenging task in computer vision. Yet, prior knowledge about the scene type and its semantics might help resolve some of the ambiguities (Liu et al., 2015). The first method to integrate local surface estimates and global scene geometry used a single box to parametrize the scene layout (Hedau et al., 2009). Appearance based classifier was used to identify clutter and visual features were only computed from non-clutter regions. Hedau et al., (2009) used the structural learning approach to estimate the best fitting box to the image. Another approach like this has been proposed which does not need the clutter ground truth labels (Wang et al., 2010).

There are some other approaches related to the 3D room layout estimation from single images (Hedau et al., 2010; Lee et al., 2010; Hedau et al., 2012; Pero et al., 2012; Schwing et al., 2012; Schwing and Urtasun, 2012; Schwing et al., 2013; Chao et al., 2013; Zhang et al., 2014, and Liu et al., 2015). Most of these approaches parameterize the room layout with a single box and assume that the layout is aligned with the three orthogonal directions defined by vanishing points (Hedau et al., 2009; Wang et al., 2010; Schwing et al., 2013; Zhang et al., 2014, and Liu et al., 2015). Some of these approaches utilize objects for reasoning about the scene layout (Hedau et al., 2009; Wang et al., 2010, and Zhang et al., 2014). Presence of objects can provide some physical constraints such as containment in the room and can be employed for estimating the room layout (Lee et al., 2010; Pero et al., 2012, and Schwing et al., 2012). Moreover, the scene layout can be utilized for better detection of objects (Hedau et al., 2012, and Fidler et al., 2012).

Given a single image from a well-structured corridor, our goal is to reconstruct the corridor scene in 3D. That is, given only a monocular image of a corridor scene, we can provide a 3D model allowing the potential viewer to virtually explore the corridor without having to physically visit the scene. This adds another dimension to static GIS at indoor places and is particularly convenient for buildings where direct search in those places is particularly time consuming.

In this chapter, we take the room layout estimation one step further. Our goal is to estimate a layout for a corridor which might be connected to the other corridors from a monocular image. Therefore, there would be no single box constraint for the estimation of the scene layout. We phrase the problem as a hypothesis selection problem which makes use of middle-level perceptual organization that exploits rich information contained in the corridor. We search for the layout hypothesis which can be translated into a physically plausible 3D model. Based on Manhattan rule assumption, we adopt the stochastic approach to sequentially generate many physically valid layout hypotheses from both detected line segments and virtually generated ones.

Here, each generated hypothesis will be scored to find the best match to the image features. Finally, the best generated hypothesis will be converted to a 3D model. The main contribution of the proposed method is the creation of corridor layouts which are no more bounded to the one single box format. The generated corridor layout provides a more realistic solution while dealing with objects or occlusions in the scene. Hence, it is well-suited to describe most corridor spaces, and it outperforms the methods which are restricted to one box primitive for estimating the scene layout. Also, we propose a scoring function which takes advantage of both orientation map and geometric context for scoring the created layout hypotheses. Since no suitable data exists for this task, we used our own image dataset taken from York University Campus buildings. We collected images from various buildings, resulting in the total of 78 single images. We labeled our data with rich annotations including the ground-truth layout and the floor plan of each corridor within the buildings. In the following section an overview of the proposed method will be provided.

4.2 Indoor Layout Estimation

While indoor space modeling is possible through applying either top-down or bottom-up approaches, it would be naive to choose any of these approaches without considering their pros and cons. Top-down approaches can be labelled as deterministic, and this labelling could be justified by their dependency on employing strong prior. Hence, top-down approaches are usually more robust to the missing data problem. An example of applying top-down approach is the indoor modeling method presented by Hedau et al. (2009). While top-down approaches are very much deterministic in employing strong priors, bottom-up approaches usually make use of weak priors. Therefore, in bottom-up approaches the perception forms by data. This basically means that if you adopt a bottom-up approach for indoor space modeling, then you expect the created model to be more flexible compare to a model created by applying a top-down approach (Baligh Jahromi and Sohn, 2015).

Most of the time, indoor modeling using a single image must deal with the presence of clutters and occlusions in the scene. Hence, missing data problem could be a major issue in using single images for indoor modeling. Since top-down approaches are more robust to the missing data problem, they could be better approaches to be chosen for indoor modeling based on a single image. The proposed method in this chapter is more inclined to a top-down approach, and it is governed by this strong prior that the indoor scene layout must have a cubic formation. Yet, what makes this method different from the others is that this method does not restrict the indoor scene layout to be comprised of only one box. The proposed method relaxes the strong prior that indicates the indoor layout is comprised of only one box and let the incoming layout to be comprised of multiple connected boxes. This advantage of the proposed method targets the modeling of somewhat occluded parts of the layout structure in the scene.

Figure 4.1 shows the overall workflow of the proposed method which is as following; 1) Edges are extracted in the single image and grouped into straight line segments. 2) Line segments will be grouped based on parallelism, orthogonality, and convergence to common vanishing points. 3) Many physically valid major box layout hypotheses will be created using detected line segments and virtual rays of vanishing points. 4) The created major box layout hypotheses are scored using a scoring function, the parameters of which are optimized through artificial neural network (ANN) learning. 5) Only, 15% of layout hypotheses that get higher scores remain in the hypothesis generation pool and the rest are discarded. 6) The remaining major box layout hypotheses are deformed by sequentially introducing side box hypotheses to their structure. Note that the maximum number of side box hypotheses that can be integrated to a major box hypothesis is five. 7) Generated side box hypotheses are also scored using the same scoring function. 8) Finally, the best layout hypothesis is selected by comparing scores and this hypothesis is converted to a 3D model.

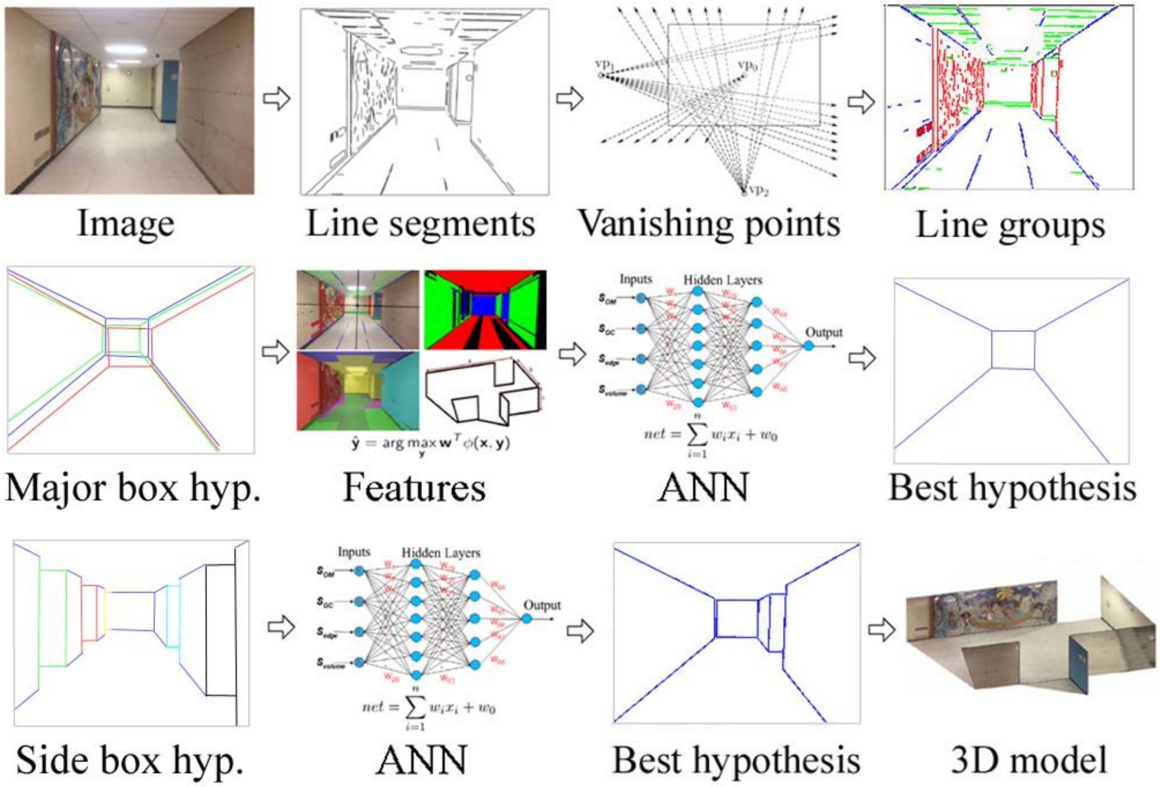


Figure 4.1 The proposed method's workflow for generating a 3D model from a single image.

4.2.1 Vanishing Point Estimation

Straight parallel lines in 3D space can be projected onto the 2D image plane, and they will intersect at a point called a vanishing point. In most of the manmade structures there are bunch of parallel lines which can provide orthogonal vanishing points (Kosecka and Zhang 2002, and Denis et al., 2008). Vanishing points have special geometric attributes which can be employed in many computer vision applications, such as camera calibration (Kosecka and Zhang 2002; Cipolla et al., 1999; Caprile and Torre 1990, and Tardif 2009), estimation of rotation angles (Kosecka and Zhang 2002; Antone and Teller 2000, and Denis et al.

2008), and more importantly 3D reconstruction (Parodi and Piccioli 1996, and Criminisi et al., 2000). To find vanishing points, different methods of straight-line clustering are available (Bazin et al., 2012). There are four main categories for these methods based on: 1) Hough Transform (HT), 2) Random Sample Consensus (RANSAC), 3) Exhaustive Search on some of the unknown entities, and 4) Expectation Maximization (Bazin et al., 2012).

Here, straight line segments were extracted in the image space using Line Segment Detector (LSD) method (Grompone von Gioi et al., 2010). LSD method can be used on digital images for line segment extraction and it is a linear-time line segment detector which can provide sub-pixel accurate results without tuning the parameters. The original idea of LSD is coming from Burns, Hanson, and Riseman's method (Burns et al., 1986), which makes use of a validation approach based on Desolneux, Moisan, and Morel's theory (Desolneux et al., 2000; and Desolneux et al., 2008). After the extraction of straight-line segments, recovering vanishing points is possible using RANSAC based algorithms. In this approach two straight line segments will be randomly selected and intersected to create a vanishing point hypothesis and then count the number of other lines (inliers) that pass through this point. The drawback of RANSAC based algorithms is that they do not guarantee the optimality of their solutions by considering the maximum intersecting lines as inliers. Here we follow Lee et al. (2009) to find three orthogonal vanishing points. In Lee et al. (2009) the coordinates of the RANSAC solution are fine-tuned using nonlinear optimization with the cost function proposed in (Rother 2000). Having estimated the three orthogonal vanishing points, the available line segments can be grouped into four different classes. Three of these classes are represented by the estimated vanishing points. The last class contains the line segments which are not related to the estimated vanishing points.

4.2.2 Scene Representation and Line Grouping

An indoor scene complexity may be that much to provide various vanishing points in one image which makes the recognition and modeling of the scene more difficult. Here, we tried to simplify the indoor scene as much as possible. For example, to modify the structure of an indoor scene, walls would be at the primary interest rather than windows or doors. Following the Manhattan rule assumption, the structure of the incoming indoor model should have a box/cube like formation. If the indoor scene is not bounded to only one room or one corridor, then there must be a major/key box to represent the scene along with some other side boxes which are intersecting with the major box to form the scene layout. Hence, the whole structure of an indoor scene would be represented by a single box or the integration of different single boxes. It should be noted that this representation of the scene layout only allows us to legitimize the estimation of three orthogonal vanishing points in the image space.

Normally, many edge pixels can be extracted from a single image. The intention is to link the extracted edge pixels into straight line segments based on predefined criteria. The criteria may include the proximity of edge pixels in the image space and similarity of the edge pixels gradients. Moreover, the straight-line segments can be grouped into line groups based on parallelism, orthogonality, and their orientation. The straight-line segment orientation can be identified based on its convergence into an estimated vanishing point.

As mentioned, only orthogonal vanishing points which are complying with the Manhattan World Assumption are valid to be detected in the image space. Thus, there could be only three different valid line groups identified in the image space. All the other detected line segments which do not converge at any of the three-estimated orthogonal vanishing points will be discarded. Consequently, vertical walls in the scene can only have two different orientations (facing the camera or being almost parallel to the camera line of sight (in case of having a vanishing point inside the image space), and floor plane and ceiling would have the same orientation. In other words, we can define 3 different surface planes in the scene which in the Cartesian coordinate system they might belong to: a) X-Y plane, b) X-Z plane, and c) Y-Z plane.

4.2.3 Layout Hypotheses Creation

Hedau et al. (2009) proposed a method for creation of a single box layout hypothesis by sampling pairs of rays from two furthest orthogonal vanishing points (V_x, V_z) on either side of the third vanishing point (V_y). They evenly spaced the image with these vanishing point rays. However, the positions of layout hypothesis junctions would be affected by ray spacing resolution and the estimated coordinates for vanishing points. Also, their approach may not provide acceptable results when dealing with long corridors due to the higher position uncertainty of the estimated vanishing points compare to ones estimated for small rooms.

In our proposed approach the layout is not going to be created completely by sampling rays from vanishing points. Though, sampling rays will be created if their presence is necessary for completing the process of layout hypothesis creation. In other words, these sampling rays will be employed if their presence is justified by physical line segments. For example, the intersection of two image physical line segments that can form a junction on the ceiling will provide information about its corresponding (directly connected) junction on the floor plane. Thus, a sampling ray of the vanishing point can be employed to connect these two junctions.

The structure of a scene layout hypothesis (h) can be represented as a set of corridors (cubic) $h = \{C_i | i = 1, 2, \dots n\}$ while each corridor consists of multiple faces (different sides of a corridor) $C = \{F_j | j = 1, 2, \dots m\}$. Here n and m denote the number of corridors and faces respectively. A major corridor ($\mathrm{C_{main}}$) is always represented by five faces $C_{main} = \{F_{front}, F_{left}, F_{right}, F_{top}, F_{bottom}\}$ while a sub-corridor (side corridor) has three faces $C_{sub}^{left} = \{F_{left}, F_{top}, F_{bottom}\}$ or $C_{sub}^{right} = \{F_{right}, F_{top}, F_{bottom}\}$.

Here, a set of scene layout hypotheses $H = \{h_i | i = 1, 2, \dots n\}$ will be sequentially created. For example, in Figure 4.1, the scene layout is created by the integration of three different corridors. The camera in standing in the major corridor at the time of exposure while there are two other corridors (side corridors/accessory hall ways) locating at the right and left side of the major corridor. Here, a major corridor hypothesis $h = \{C_{main}\}$ is generated first. More formally, let $PL = \{pl_i^O | i = 1, 2, \dots n\}$ and $VL = \{vl_j^O | j = 1, 2, \dots m\}$ be the set of physical line segments and virtually generated rays of orientation O, where $O \in \{X, Y, Z\}$ denotes one of the three orthogonal directions. Also n and m reveal the number of physical line segments and virtually generated rays, respectively.

A major corridor layout hypothesis h is created by intersecting sample lines from PL and VL where the minimum number of selected line segments from PL is 4, and the total number of line segments needed for this creation is 8. As mentioned before, a major corridor layout hypothesis is comprised of five polygons. The front wall has four junctions together forming a rectangular shape. Note that the layout structure in the object space is orthogonal. Thus, by identifying two junctions at the end of the front wall diagonal and with the help of vanishing point rays, a layout hypothesis can be created. Each junction can be created by intersecting two lines with different orientations. Figure 4.2 depicts an example of a major and side corridor layout hypothesis creation after identifying two of the opposite front wall junctions. This figure shows that layout hypothesis creation starts by inputting a single image into the system (a); Orthogonal vanishing points are extracted and physical straight line segments are classified (b); Physical lines of different classes are randomly selected (minimum 4 line segments) and intersected to generate potential layout junctions (c and d); Major corridor layout hypothesis generated by intersecting physical straight line segments (solid lines) and virtual rays (dashed lines) from vanishing points (e). Figure 4.2 also shows the creation of a side box hypothesis using both physical line segments and virtual rays from vanishing points. In this figure, virtual rays partition the right-side plane (in green) into small spaces (f); A partition that receive supports from the physical line segments will be preserved and more partitioning will be accomplished using virtual rays (g); Consequently a planar region (in blue) facing the camera is identified (h); Moreover, other regions representing floor and ceiling parts will be identified (i).

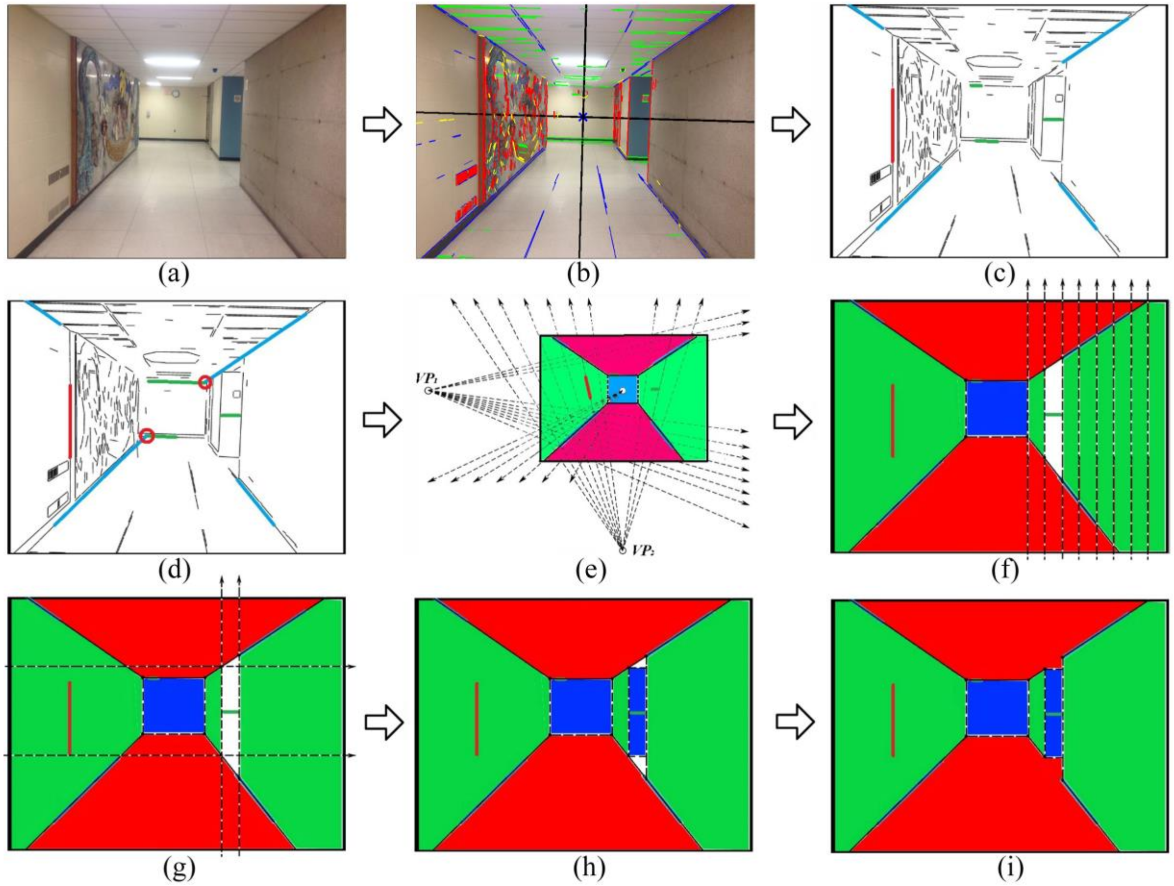


Figure 4.2 Creation of a major and a side corridor layout hypothesis by intersecting physical line segments and virtually generated rays from orthogonal vanishing points.

As mentioned above, in the proposed method line segments with different orientations are randomly selected and intersected to form the major scene layout hypotheses. It should be noted that very short line segments will be ruled out or merged with their adjacent line segments if their proximity is less than a predefined threshold (here the threshold is 5 pixels) while having the same edge gradients. Note that an image gradient is defined by the intensity/color directional change in an image. Eventually, hypothesis creation will start with line segments with longer length. The overall process is described in Algorithm 1. In this algorithm the general workflow of generating major corridor hypothesis has been described. This will lead to the generation of many different hypothetic major corridors in the scene. Only physically valid corridor hypotheses should be accepted in this process. Hence, the number of created major corridor hypotheses will be reduced to some extent.

Algorithm 1: Generating major corridor hypotheses

```
Set $H_1$ ← 0, where $H_1$ is the set of major corridor hypotheses;
    for all pair of line segments and virtual rays $(l_i, l_j)$ that intersect below horizon do
        if $(l_i \wedge l_j)$ are having different orientations then
            add floor junction point $P(l_i, l_j)$ to $F_{bottom}$
        end if
    end for
    for all pair of line segments and virtual rays $(l_m, l_n)$ that intersect above horizon do
        if $(l_m \wedge l_n)$ are having different orientations then
            add ceiling junction point $P(l_m, l_n)$ to $F_{top}$
        end if
    end for
    for all $P_i \in F_{bottom}$ and $P_j \in F_{top}$ do
        if $(P_i \wedge P_j)$ are residing on different sides of the image then
            connect $(P_i \wedge P_j)$ to $(V_x)$ and $(V_y)$ and $(V_z)$ via line segments or virtual
            rays and add scene with 1 major corridor $C_{main}(P_i, P_j)$ to $H_1$
        end if
    end for
return $H_1$
```

Having created the major corridor hypotheses, the presence of the side boxes (accessory hallways on the sides of the major corridor) will be examined. This process will be accomplished by employing either physical line segments or virtual rays. For example, if any of the major corridor's side-walls contain a line segment that has the same orientation as the wall itself, this line segment can be used for creating a side box hypothesis. Having the same orientation means the line segment must be perpendicular to the major corridor's side-wall. This is a tangible hint for having a side box in the scene. This process is described in Algorithm 2.

Algorithm 2: Generating side corridor hypotheses

```
Set $H_2$ ← 0, where $H_2$ is the set of layout hypotheses having side corridors;
    for all line segments ($pl_i^x \wedge \{F_{left}, F_{right}\} \in C_{main}$) where ($C_{main} \in H_1$) do
        if ($pl_i^x$) is inside $\{F_{left}, F_{right}\}$ then
            connect the end points of ($pl_i^x$) to ($\{F_{left}, F_{right}\} \in C_{main}$) borders via
            virtual rays of ($V_z$) to make ($\{F_{left}, F_{right}\} \in C_{sub}^{left,right}$) and add scene
            with side corridor $C_{sub}^{left,right}$ to $C_{main}$
        end if
     Set $H_2$ ← $H_2$ $U$ $C_{main}$
    end for
return
```

In the above algorithm the general workflow of generating side corridor hypotheses has been described. Hence, many hypothetical corridors may be generated on the sides of each major corridor hypothesis. It should be noted that in this process duplicate side corridor hypothesis will be deleted and overlapping hypothesis will be merged. Moreover, only physically valid hypotheses will remain in the hypothesis creation pool. Therefore, the number of valid hypothesis will be reduced, and the remaining ones are the final scene layout hypotheses. Figure 4.2 describes the core part of this process intuitively.

4.3 Selecting Features

As mentioned in the previous section, the proposed method sequentially creates the complete scene layout hypotheses through generation and integration of box form structures in the image space. This process is performed in the image space using classified line segments and virtual rays of vanishing points. Following this rational, many scene layout hypotheses will be created. To find the best fitting layout hypothesis to the scene, different features must be taken into consideration. These features should optimally characterize different qualities of the created hypothesis. In other words, these features together encode how well the created layout hypothesis represents the corridor scene in the image space.

Here, we considered four different layout characteristics in the image space to specify the desirable set of features. The first characteristic is the layout feasibility in terms of covering the whole scene. We call this as the Volumetric Reasoning feature. The second characteristic is the layout interaction with the detected straight-line segments in the image. We call this as the Edge Correspondences feature. The third characteristic is the layout structure in terms of surface orientations. We call this as the Orientation Map feature. The last one is the layout formation in terms of not being affected by the clutters. We call this as the Geometric Context feature. In the coming sections, these features will be introduced.

4.3.1 Volumetric Reasoning

Lee et al. (2010) imposed some volumetric constraints to estimate the room layout. They model the objects as solid cubes which occupy 3D volumes in the free space defined by the room walls. Following the same rational, here the containment constraint is taken into consideration which dictates that every object should be contained inside the layout. We interpret this constraint as the search for the maximum physically plausible created volume among all the created layout hypotheses. In other words, the layout hypothesis which covers a larger area is more probable to contain all the objects in the scene. Hence, the volumetric reasoning about the created layout hypothesis plays the role of a feature here.

4.3.2 Edge Correspondences

In an image, edges can be introduced as intensity discontinuities between the adjacent pixels. Hence, edges can identify the boundaries between various textures, an indication of a higher frequency in the image space. In some of the corridor scenes, enough and reliable layout features are not available due to the presence of objects with no texture or homogeneous texture. However, in such cases edges are readily available and where textures are homogeneous, edges could be valuable features most of the time. Usually, the intersection lines of the indoor corridor dominant planes can define the overall geometric structure of the layout. Normally, these planes have different textures or colors and their boundaries can be identified by a collection of straight-line segments in the image space. Therefore, straight line segments, a by product of combined edges in the image space, can provide a powerful cue about the indoor corridor layout structure. Here, we identified the layout and image edge-correspondences as a valid feature which can provides a clue on the quality of a created layout hypothesis.

4.3.3 Orientation Map

Although single images are a reliable data for indoor space modeling, automatic recognition of different structures from a single image is very challenging. Lee et al., (2009) presented the orientation map for evaluation of their generated layout hypotheses. The main concept of the orientation map is to define which regions in an image have the same orientation. An orientation of a region is determined by the direction of the normal of that region. If a region belongs to the XY plane, then its orientation is Z.

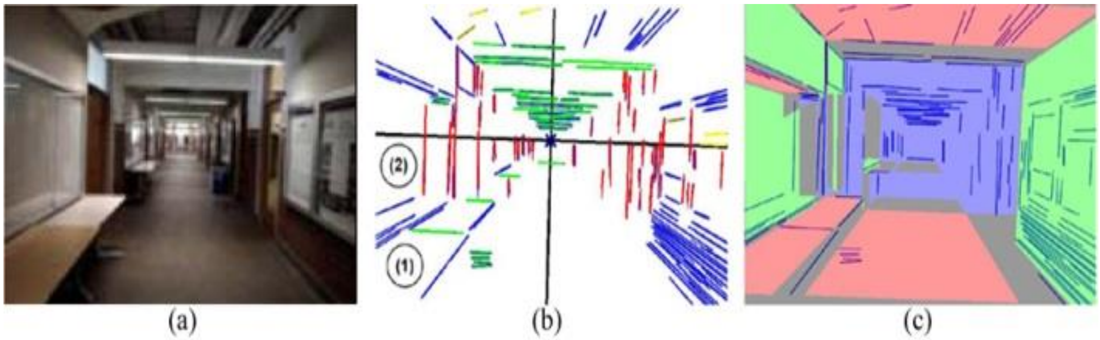


Figure 4.3 (a) Single image, line segments and orientation map. (a) Single image. (b) Detected straight line segments, vanishing point at centre, and two of vanishing lines in black. (c) Orientation map; regions are colorized according to their orientations.

Orientation map is a map that reveals the local belief of regional orientations computed from line segments (Figure 4.3). If a pixel is supported by two different line segments that have different orientations, then this would be a strong indication that the pixel orientation is perpendicular to the orientation of these two lines. For example, in figure 4.3(b), pixel (1) can be on a horizontal surface because a green line above it and two blue lines to the right and left are supporting pixel (1) to be perpendicular to the orientation of both lines. Also, pixel (2) seems to be on a vertical surface because blue lines above and below and red line to the right are supporting it. Also, there is a green line below pixel (2), but its support is blocked by the blue line between the green line and the pixel. Therefore, the support of a line will extend until it hits a line that has the same orientation as the normal orientation of the surface it is supporting. It means that a line cannot reside on a plane that it should be perpendicular to it. Here, we consider the image orientation map as one of the selected features.

4.3.4 Geometric Context

Hoiem et al., (2007) labeled an image of an outdoor scene into coarse geometric classes which is useful for tasks such as navigation, object recognition, and general scene understanding. Usually the camera axis is roughly aligned with the ground plane, enabling them to reconcile material with perspective. They categorized every region in an outdoor image into one of three main classes. First, surfaces which are roughly parallel to the ground and can potentially support another solid surface. Second, solid surfaces those are too steep to support an object. Third, all image regions which are corresponding to the open air and clouds.

Theoretically a region in the image could be generated by a surface of any orientation. To determine which orientation is most probable, Hoiem et al., (2007) used available cues such as material, location, texture gradients, shading, and vanishing points. It should be noted that some of these cues, are only helpful when considered over the appropriate spatial support which could be a region in a segmented image. The common solution is to build structural knowledge of the image from pixels to super-pixels.

Hoiem et al., (2007) solution was to compute multiple segmentations based on simple cues. Generally, they sampled a small number of segmentations which were representative of the whole distribution. They computed the segmentations by grouping super-pixels into larger continuous segments. Note that different segmentations provide various views of the image. To find the best segmentation, the likelihood that each segment is good or homogeneous must be evaluated. Also, the likelihood of each possible label for each segment must be evaluated. Finally, combination of all the estimates produced by different segmentations would be possible in a probabilistic fashion. Note that a segment could be homogeneous if all the super-pixels inside that segment have the same label. Hoiem et al., (2007) estimated the homogeneity likelihood using all the cues and boosted decision trees.

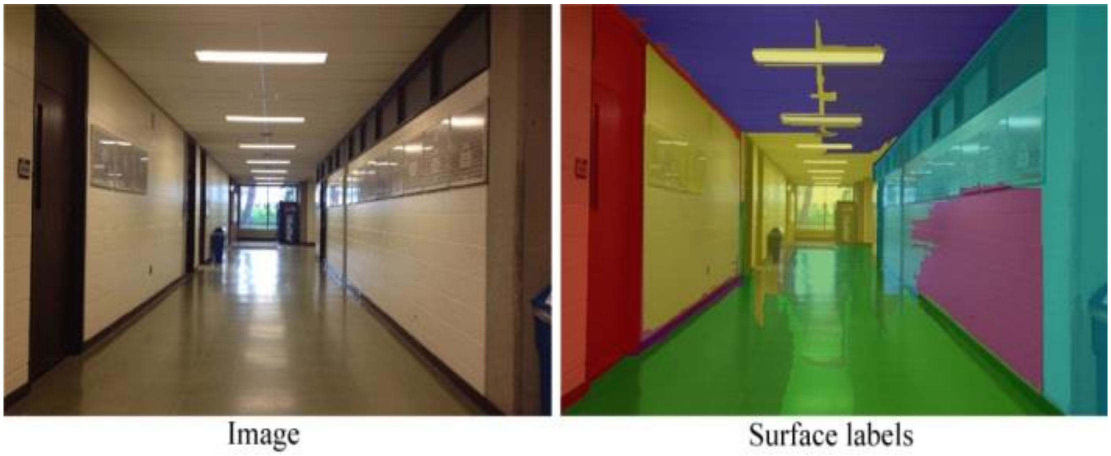


Figure 4.4 Single image of an indoor corridor and the estimated surface labels shown by different colors.

Hedau et al., (2009) used the same idea for labeling surfaces in an image, but this time the focus was on indoor places and recovering the spatial layout of cluttered rooms. They tried to achieve an overall estimate of where the objects are, to get a more accurate estimate of the room layout. To estimate the room layout surface labels including the objects, they use a modified version of Hoiem et al., (2007) surface layout algorithm. The image is over-segmented into super-pixels, and in the next step partitioned into multiple segmentations. Color, texture, edge, and vanishing points are the main cues which were computed over each segment. A classifier (boosted decision tree) is used to estimate the likelihood that each segment contains only one type of label and the likelihood of each of possible labels. Further, over the segmentations these likelihoods would be integrated to provide label confidences for each super-pixel. Figure 4.4 shows an indoor corridor image with its estimated surface labels. Here, we choose the geometric context as the last feature to be considered.

4.4 Evaluating Layout Hypotheses

As mentioned before, in the proposed method the complete scene layout hypothesis is sequentially created through generation and integration of cubic structures. This process is performed in the image space using classified line segments and virtual rays of vanishing points. Following this rational, many scene layout hypotheses will be created. Therefore, the created layout hypotheses must undergo an evaluation process for selection of the best fitting hypothesis. Figure 4.5 shows only floor plans of several layout hypotheses (pink regions) in the image space where the best fitting hypothesis is needed.

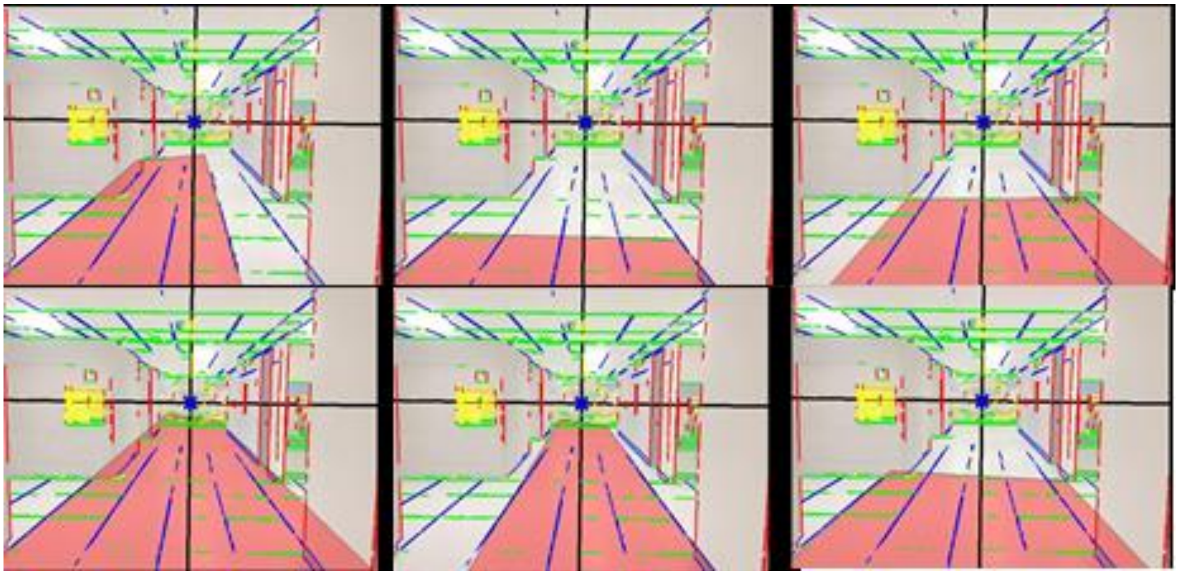


Figure 4.5 Several floor plans of layout hypotheses in the image space along with the classified straight-line segments and the estimated vanishing lines.

To perform the evaluation process, a linear scoring function can be defined to score each hypothesis individually. Given a set of created layout hypotheses in the image space

$\{h_1, h_2, ...h_n\} \in H$, we wish to do the mapping $S: H \rightarrow R$ which is used to define a score for the automatically generated candidate layouts in an image. For the proposed scoring function, we examined different type of features along with weight parameters. In the coming sub-sections these variations will be explained.

4.4.1 Evaluating Hypotheses by a Linear Scoring Function

The proposed scoring function must take some independent factors into consideration. Here, we considered the "Volume Maximization", "Edge Correspondences" and "Orientation Map" as the main features affecting the layout hypotheses scores. Hence, the expected value of the proposed scoring function "S" can be decomposed into the sum of three different functions, which characterize different qualities of the created hypothesis. These functions together encode how well the created layout hypothesis fits the corridor scene. We thus have

$$S(h_i) = W_1 \times S_{volume}(h_i) + W_2 \times S_{edge}(h_i) + W_3 \times S_{Omap}(h_i) \qquad (4.1)$$

Where h_i = candidate hypothesis

S = total scoring function

S_{volume} = scoring function for volume

S_{edge} = scoring function for edge correspondences

S_{Omap} = scoring function for orientation map

$W_{1,2,3}$ = equal weight values

As it can be seen in the above equation, the outcome of three different functions are combined to create the proposed total scoring function. Here, each function is focusing on a specific layout feature. These features are supposed to represent different qualities of the created layout hypothesis as close to the reality as it could be. Considering these features, three different functions can be defined to score each quality of the hypothesis. The final score of a candidate hypothesis will be defined by summing the outcomes of these three functions with equal weights. Note that equal weight values are imposed at this stage due to the lack of information about the importance of each function with respect to the others.

First, the containment constraint should be taken into consideration that dictates every object must be contained inside the corridor. As mentioned before, we interpreted this constraint as the desire for the maximum plausible volume for the layout hypotheses. Therefore, we decide to give a higher score to the layout hypothesis which has a larger volume. Hence, the volume scoring function *(S_{volume})* gives the highest volume score (score one) to the layout hypothesis which creates the largest valid volume. Also, it gives the minimum volume score (score zero) to the layout hypothesis which has the smallest valid volume. Hence, the incoming score of a candidate layout hypothesis will be a positive real number between zero and one. The volume score of a candidate hypothesis *(h_i)* can be calculated from the following equation:

$$S_{volume}(hi) = \frac{V_i - V_{Min}}{V_{Max} - V_{Min}} \tag{4.2}$$

Where S_{volume} = scoring function for volume

h_i = candidate layout hypothesis

V_i = calculated volume for hypothesis h_i
V_{Min} = minimum calculated volume among all the created layout hypotheses
V_{Max} = maximum calculated volume among all the created layout hypotheses

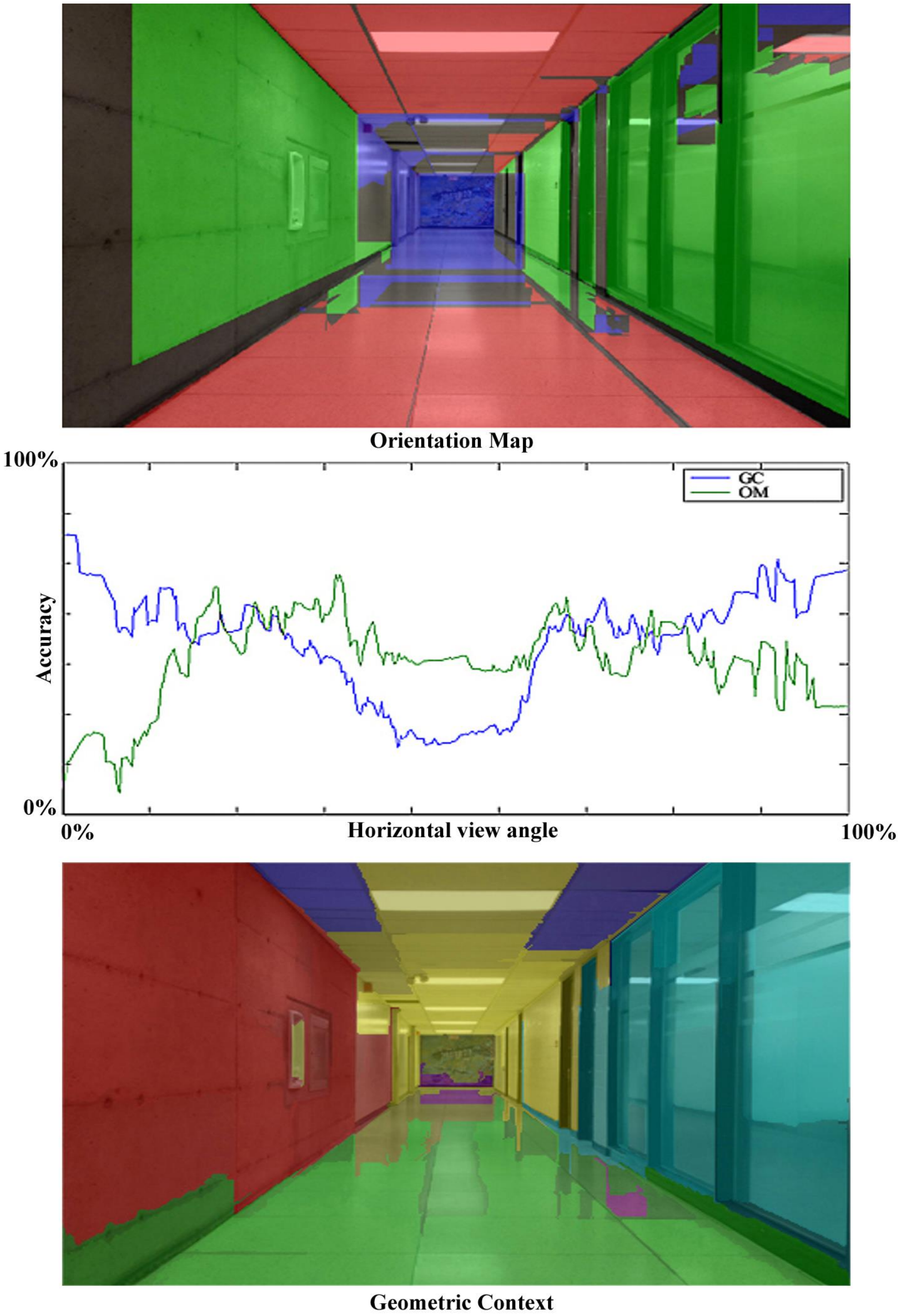
Considering the above equation, the other two functions which score edge-correspondences quality of the created layout hypothesis and the compatibility of the created layout hypothesis to the orientation map are also defined in the same way. The defined function gives the highest edge-correspondences score to the layout hypothesis which has the maximum positive edge-correspondences to the actual detected line segments. Here, the positive edge-correspondences are defined by counting the number of edge pixels which are residing close enough (here, less than 5 pixels) to the structural lines of the created layout hypothesis. Therefore, the layout hypothesis which has the biggest number of detected edges close enough to its borders will get the highest score from the proposed function *(*S_{edge}*)*.

The compatibility of the created layout hypothesis to the orientation map is calculated pixel by pixel. The created layout hypothesis will provide specific orientations to each pixel in the image, and the orientation map is also suggesting orientations to the image. Therefore, by comparing these two (pixel by pixel) the compatibility between the created layout hypothesis and the orientation map can be calculated. Here, the number of pixels which get the same orientation from the created layout hypothesis and the orientation map are going to be counted. The proposed function *(*S_{Omap}*)* gives the highest orientation map score (score one) to the layout hypothesis which has the most pixel-wise compatibility to the orientation map and lowest score (score zero) to the one which has the least compatibility.

Considering the combination of these three functions as depicted above, each hypothesis will be examined individually, and gets score based on the above-mentioned functions. As discussed, the incoming scores will be normalized based on the maximum and minimum incoming values. The normalized scores will be integrated and the hypothesis with the maximum score will be selected as the best fitting hypothesis.

4.4.2 Evaluating Hypotheses by OM and GC Combination

Zhang et al., (2014) applied both orientation map and geometric context on overlapping perspective images. In their paper, they expressed that the geometric context can provide better surface normal estimation at the bottom of an image, while the orientation map works better at the top of an image. Hence, they combined the top part of the orientation map image and the bottom part of geometric context image and used the incoming result to evaluate the room layout. This drastic variation in the performance of orientation map and geometric context from the top to the bottom of the images is explainable. Since most of the images in their dataset were captured from single rooms, either this variation is due to the presence of clutters in most rooms, or because their model was trained based on images looking slightly downwards.



Orientation Map

Geometric Context

Figure 4.6 Orientation Map and Geometric Context accuracy changes by changing the horizontal viewing angle.

Unlike single rooms which are usually small in size and full of clutters, corridors are usually less occupied with clutters and have longer length. Therefore, we examined the horizontal view angle in the image to evaluate the performance of orientation map and geometric context in the corridors. Figure 4.6 shows the changes in the accuracy of orientation map and geometric context compare to the ground truth training images (here, 34 images). As it can be seen in this figure, by changing the horizontal view angle from left to the right side of the image, the orientation map and geometric context performances are varying to a considerable extent. The geometric context is outperforming the orientation map around both sides of the images, while the orientation map is outperforming the geometric context around the center of the images.

Hence, we decided to use this valuable information for evaluating the layout hypotheses through combining both orientation map and geometric context. The combination of orientation map and geometric context is performed by considering their respective performance curves with respect to the horizontal view angle. The combination of these two looks to be a very simple task, yet orientation map and geometric context have little differences in their representation. Hence, their representation must be standardized before this combination would be possible.

On one hand, the orientation map is not numerically expressed, and on the other hand the geometric context is expressed by likelihood of each possible label for all super-pixels in the image. As mentioned before, the orientation map is a map that reveals the local belief of region orientations in an image. These local orientations are assigned to the image regions through examining their supporting line segments. Usually the orientation map is colorized with four different colors which are red, green, blue, and black. The first three colors are representing a specific orientation in 3D space which is either X, Y, or Z. Also, the black color represents the unsupported regions in an image. Since there is a possibility that some regions in a specific image could not get complete support from line segments, those specific regions would be colorized as black and officially would not be assigned with any orientation.

A specific orientation can be assigned to a local region in an image, and the assigned orientation can be expressed numerically. In other words, it is possible to say how good the assigned orientation is. To express the orientation map numerically for every region in an image, the supporting line segments should be in focus. Here, image pixels *($I_{x,y}$)* will get a value between zero and one for their assigned orientation ($OM(I_{x,y})$). This value is assigned by comparing the length of supporting line segments to the lines created by the intersection of these line segments with respect to their distance to a pixel. In other words, when a region is fully supported by the complete line segments in an image, it will get a value of one for its assigned orientation. Also, when a region is supported by some truncated line segments, it may get the value of zero for its assigned orientation. Figure 4.7 shows how this assignment can be interpreted.

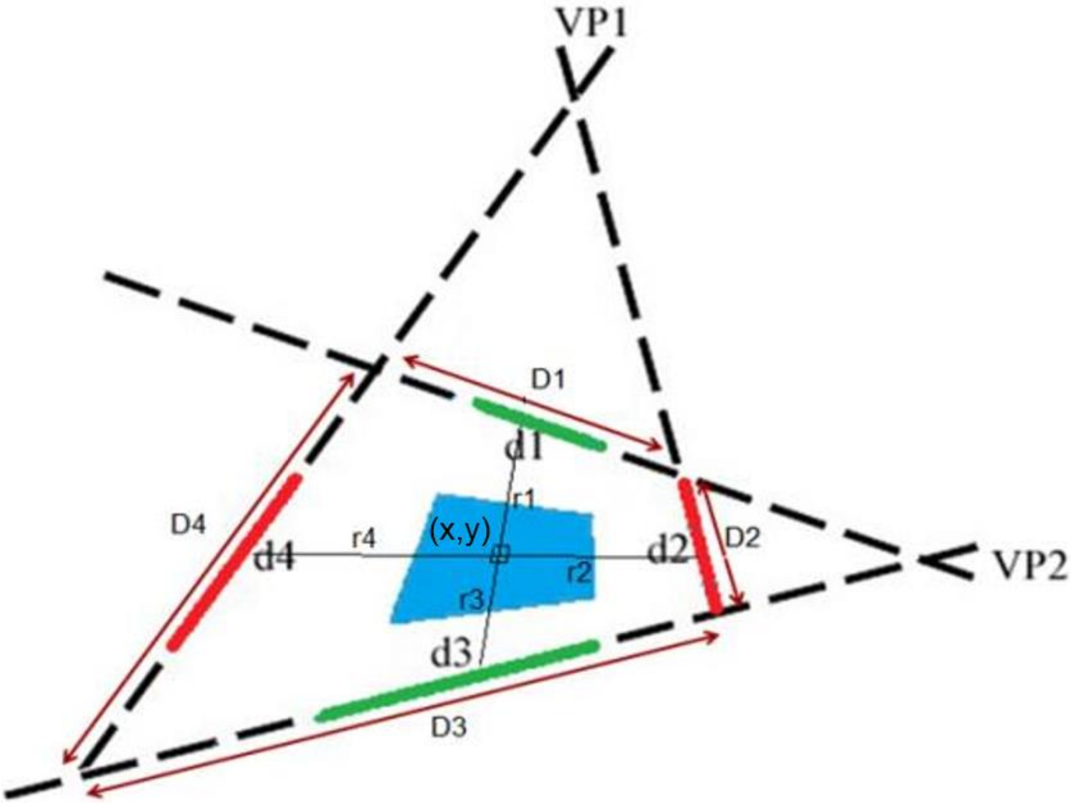


Figure 4.7 An orientation of a region supported by classified line segments.

$$OM(I_{x,y}) = \frac{r_3 \times \left(\frac{d_1}{D_1}\right) + r_1 \times \left(\frac{d_3}{D_3}\right) + r_4 \times \left(\frac{d_2}{D_2}\right) + r_2 \times \left(\frac{d_4}{D_4}\right)}{r_1 + r_2 + r_3 + r_4} \tag{4.3}$$

Finally, the assigned values will be normalized and all the pixels in the orientation map image will get a value between zero and one for their respective orientation. It should be noted that in single images a small line segment might be longer in real world than what it looks in the image due to the perspective effect. Therefore, we used the vanishing points and project all the detected line segments to the image borders to suitably compare their lengths.

After assigning different values to the pixels in the orientation map image, we had to express the surface labels of geometric context as orientations. Therefore, we expressed the geometric context by the same three orientations which are suggested by the orientation map. Here, we choose the highest value of the surface label probabilities for each pixel as the assigned orientation value to that pixel. Finally, the assigned values will be normalized, and it is possible to combine the incoming results of the orientation map and geometric context with respect to the horizontal view angle in the image. Formulas below are showing how these values can be used for evaluating an individual layout hypothesis:

$$k_{x,y} = \frac{a_{x,y}}{a_{x,y} + b_{x,y}} \tag{4.4}$$

$$p_{x,y} = k_{x,y} \times OM(I_{x,y}) \tag{4.5}$$

$$q_{x,y} = (1 - k_{x,y}) \times GC(I_{x,y}) \tag{4.6}$$

$$I_{x,y}\ (OM, GC) = max\ (p_{x,y}\ ,\ q_{x,y}\ ) \tag{4.7}$$

$$S_{OM\&GC}\ (h_i) = 1 - \frac{1}{n \times m} \times \sum_{x=1}^{n} \sum_{y=1}^{m} \left(\left| I_{x,y}(OM,\ GC) - J_{x,y}(hi) \right| \right) \tag{4.8}$$

Where a_{xy} = accuracy of Orientation Map at pixel (x,y)

b_{xy} = accuracy of Geometric Context at pixel (x,y)

h_i = candidate hypothesis

$J_{x,y}\ (h_i)$ = hypothesis “*hi*“ orientation value at pixel (x,y)

$I_{x,y}\ (OM, GC)$ = *OM* and *GC* integration value at pixel (x,y)

$OM(I_{xy})$ = orientation map outcome at pixel (x,y)

$GC(I_{xy})$ = geometric context outcome at pixel (x,y)

$S_{OM\&GC}\ (h_i)$ = scoring function for *OM* and *GC*

The compatibility of the created layout hypothesis to the orientation map and geometric context is calculated pixel by pixel. The created layout hypothesis will provide specific orientations to each pixel in the image space, and the orientation map and geometric context are also conducting the same task. Therefore, by comparing layout hypothesis orientation to the orientation provided by the combination of orientation map and geometric context, the better pixel-wise evaluation of the layout hypothesis can be achieved. Consequently, the best fitting hypothesis is selected through the following scoring function applying equal weights:

$$S(h_i) = w_1 \times S_{volume}(h_i) + w_2 \times S_{edge}(h_i) + w_3 \times S_{OM\&GC}\,(h_i) \qquad (4.9)$$

4.4.3 Evaluating Hypotheses by ANN

As mentioned in previous sections, different features are used for evaluating the created layout hypotheses. However, the defined scoring function were applying equal weights for all features. Here, we defined this weight optimization problem as a nonlinear classification problem. Common classifiers are categorized into linear classifiers and nonlinear classifiers. In our case, selected features are complicated, and classes are not linearly separable. Therefore, a classifier which produce nonlinear discriminates is needed. Artificial Neural Networks (ANN) are powerful nonlinear classifiers while input features and classes are too complex. Basic components of an artificial neuron include: A) connecting links that provide weights, w_j, to the input values, x_j, for $j=1 \ldots m$. Weights are designed to minimize mean square error through training data set. B) an accumulator, summing the weighted input values to provide the input to the activation function. Here, w_0 is called the bias, a numerical value associated with the artificial neuron, as in equation (4.10). C) an activation function *g*, which is a consistent monotone function that projects from *net* to *g(net)*, the output value of the artificial neuron.

$$net = \sum_{i=1}^{n} w_i \, x_i + \, w_0 \tag{4.10}$$

Here, to optimize the influence of selected features (their respective weights) we applied a multilayer feed-forward network with back propagation. The introduced ANN input layer has 4 nodes and the input values are the normalized incoming values of the previously mentioned individual feature scoring functions (e.g. S_{volume}, S_{edge}, S_{Omap} and S_{Gc}). The input layer accepts the input values and the outputs of each layer's artificial neurons are the inputs to the artificial neurons in the next layer until they reach the output layer. Here, different number of layers and neurons are tested applying 10-fold cross-validation technique, and the optimal ANN was found to have four layers, with 4, 7, 5, and 1 neurons in each layer. There are two hidden layers between the input and output layers, each has 7 and 5 nodes respectively and the final layer has only 1 node. The classification between the two classes is made by defining a threshold to the output value at the final node.

In the prepared ANN system, the activation function is a sigmoid function. A back-propagation algorithm was used to optimize weight values. Thus, weight values (vector w) were estimated optimally through ANN trained by 161 and 206 positive and negative layout samples respectively. These samples are synthetic data manufactured based on training data set mentioned earlier in this chapter. Finally, 100 epochs were executed to train the ANN. All input data was normalized from 0.1 to 0.9, and an optimal threshold of 0.7 (empirically defined) was chosen for output results classification. This ANN is used on the provided testing dataset for classifying generated layout hypotheses to valid and invalid hypotheses. Eventually, the hypothesis with the highest output value is chosen as the best hypothesis. Further details about artificial neural networks and their applications can be found in Bishop (1995).

4.5 Experimental Result

The performance of the proposed method was evaluated over the prepared dataset at York University. As mentioned in chapter 3, the ground truth layouts and orientation images were provided for York University dataset. Here, the York University dataset is divided into two categories of training set and testing set. Out of 78 images (corridor scenes) in the dataset, 53 images were chosen for testing and the rest of 25 images were chosen for training. The training set is used for accomplishing two tasks; a) calculating the accuracy curves with respect to accuracy changes in horizontal viewing angles for both orientation map and geometric context; b) providing the training set for ANN along with the synthesized data. Since, the ground truth orientation images were provided for each image in the dataset, the comparison between estimated layouts and ground truth layouts is possible through test images. Both qualitative and quantitative assessments for the proposed indoor layout estimation method were conducted. In the following sub-sections, the incoming results will be presented.

4.5.1 Qualitative Assessment

Qualitatively, around 58% of images returned acceptable layouts. Here, two different criteria are set to evaluate the estimated layouts in the image space. If the estimated layout conforms with these two criteria, then it is considered as an acceptable estimate. These two criteria together evaluate the topologic and geometric information of the estimated layout. The topologic acceptance is occurred when both the estimated layout and the ground truth layout can be represented by the same topological graph. This means that the number of present corridors (including major, right side, and left side corridors) in an image are the same for both estimated layout and the ground truth layout. The geometric acceptance is examined by considering layout structural junctions. If all Euclidean distances between corresponding junctions of the estimated layout and the ground truth layout are less than a predefined threshold (= 5 image pixels in experiments), then the estimated layout is considered geometrically acceptable in the image space.

It should be noted that even when floor-wall boundary was partially occluded by the objects or could not be detected through middle-level perceptual organization, the scene layout was successfully recovered in some images (Figure 4.8). When the physical line segments cannot be detected from the image, virtual rays created through vanishing points can play the same role as the physical line segments. In these cases, the key cube hypothesis will be created using both physical line segments and virtual rays. It should be noted that virtual rays cannot be always helpful, specially when the corridor length is very long. In these cases, the estimated vanishing points may not have enough accuracy.

Therefore, the created virtual rays will be deviated from the real layout borders as the ray gets closer and closer to the camera.

As shown in Figure 4.8, the incoming results show that almost similar levels of model quality were achieved for both linear (equation 4.9) and ANN (equation 4.10) hypotheses scoring. However, the model-based evaluations indicate that the model quality for the ANN hypotheses scoring is better than the one for linear hypotheses scoring. This is mainly related to depth errors which occur more in long corridor scenes. We observed that many front corridors planes (corridors front faces) in the linear scoring scheme were misplaced and resided closer to the camera. As a result, the corridor lengths estimated from linear hypotheses scoring caused a low success rate of corridor depth estimation.

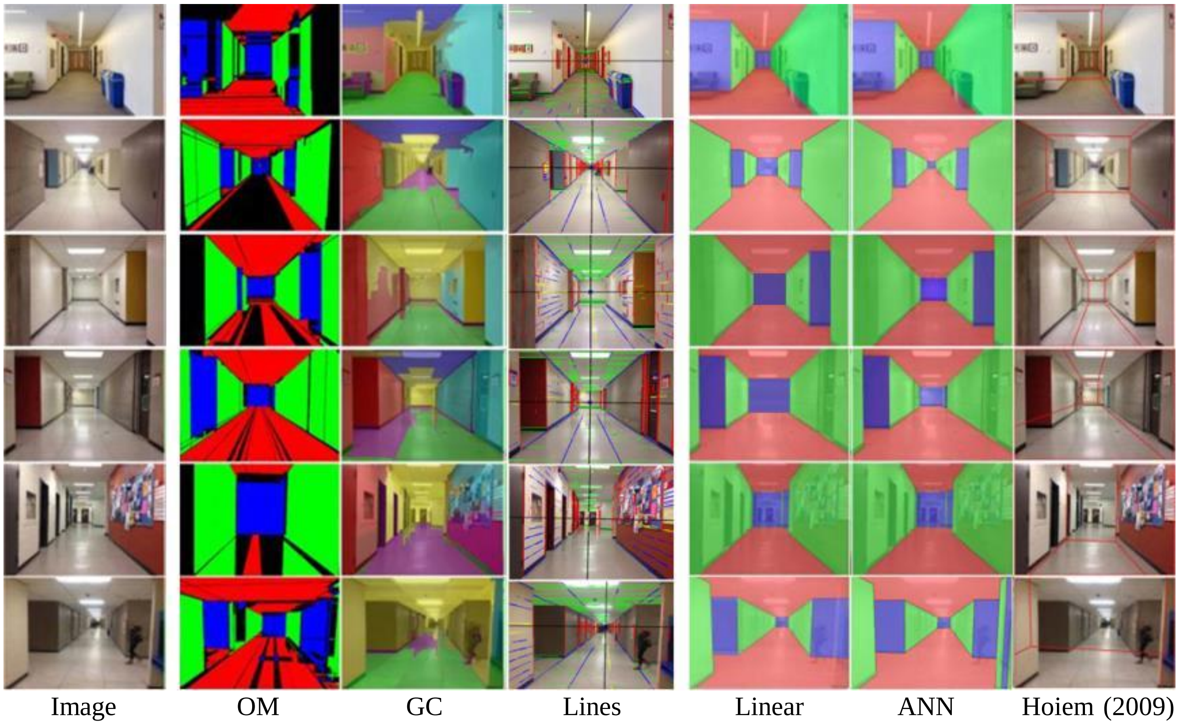


Figure 4.8 Examples of the created layouts which can be successfully convert to 3D.

4.5.2 Quantitative Assessment

We compared the evaluation results of both ANN and linear hypotheses scoring with those generated by Hedau and Hoiem (2009) algorithm. Here, this algorithm is chosen for comparisons due to its novelty and good performance compare to other proposed algorithms. Also, its source code is available to the public which facilitates quantitative comparisons without dealing with implementation difficulties compare to the other available algorithms. Note that Hedau and Hoiem (2009) source code has already been trained on a large dataset that includes single room and corridor images which makes it suitable to be compared to the proposed algorithm.

Here, area-based evaluation results are considered (Table 4.1). In this experiment, the estimated layout for a test image is compared to its correspondent ground truth layout in the image space. To conduct this comparison, adopted orientations by both estimated layout and ground truth layout will be compared at pixel level for all test image. Results of this evaluation demonstrate that our method can outperform Hedau and Hoiem (2009) algorithm in terms of the completeness and quality, specially when accessory hallways are present in the scene. When corridors with accessory hallways were considered, our proposed method showed more accurate results. It should be noted that both methods are assessed only for our dataset which mainly includes indoor corridors. In terms of robustness, our proposed method outperforms the Hedau and Hoiem (2009) algorithm, since the accuracy of our method is better than Hedau and Hoiem (2009) evaluated method. Considering that the accuracy is above 58% for comparing our method to the ground truth data while the accuracy of Hedau and Hoiem (2009) method is only 17%.

Table 4.1 Area-based evaluation of the proposed method compare to Hoiem (2009).

Dataset	*Proposed method*				*Hedau and Hoiem (2009)*	
	Linear hypotheses scoring (average)		*ANN hypotheses scoring (average)*		*Single corridor (average)*	*Multiple corridors (average)*
	Single corridor	*Multiple corridors*	*Single corridor*	*Multiple corridors*		
GeoICT research laboratory dataset	*77%*	*71%*	*85%*	*76%*	*61%*	*49%*

Here, for each test image a quantitative table was produced to examine the proposed indoor modeling algorithm. Sample tables are presented here (Tables 4.2 and 4.3) which are presenting the quantitative results of the wrongly estimated layout in Figure 4.9. Table 4.2 reveals the orientation difference between this estimated layout and the ground truth layout. This table can be used for evaluating the overall performance of the generated layout.

Here, a comparison between the ground truth orientation and the orientation suggested by the created layout is performed. It should be noted that this comparison is accomplished based on measuring pixel to pixel correspondences. Therefore, if two correspondent pixels on the ground truth image and the created layout image having the same orientation, then it shows that the proposed method could correctly estimate the layout orientation at that pixel. Each image pixel can accept only one orientation out of three (O_1, O_2 *or* O_3). The orientations are colorized by “Red”, “Green”, or “Blue” in Figure 4.9. Table 4.2 reveals the pixel to pixel orientation correspondences for the created layout in Figure 4.9.

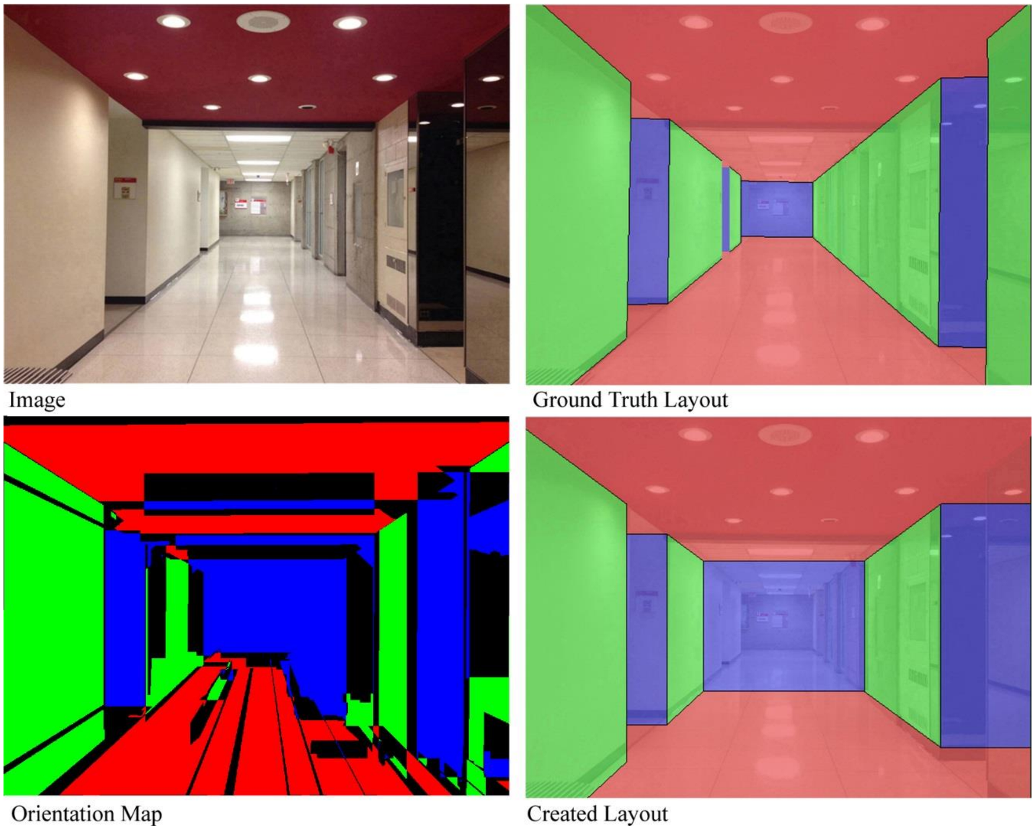


Figure 4.9 The created layout and the ground truth layout visualized in the image space.

Table 4.2 Pixel to pixel correspondences based on orientation.

Figure 4.9	*Floor*	*Ceiling*	*Front Walls*	*Right Walls*	*Left Walls*
Floor	*172223*	*0*	*28448*	*0*	*4613*
Ceiling	*0*	*265663*	*20131*	*0*	*0*
Front Walls	*3478*	*4262*	*122469*	*1483*	*1060*
Right Walls	*15833*	*21729*	*87519*	*78750*	*0*
Left Walls	*0*	*11271*	*23933*	*0*	*217135*

Table 4.2 reveals valuable information. The (i, j)-th entry in this table represents the number of pixels with ground truth label i which are estimated as label j, over the test image. As it can be seen in this table, floor, ceiling and wall estimates are partially correct

and some specific regions were wrongly oriented. This can be explained by the dependence of the method on the creation of the true major box hypothesis (slightly deviated in Figure 4.9) and the great impact of the linear scoring function in the selection of the best hypothesis. If the major box hypothesis is wrongly estimated at the first step, the method could not correct this false estimation and will end up in awkward result. Therefore, a true estimation of the major box provides a very strong condition to the success of our method.

4.5.3 Layout Comparison to 3D Model

Geometrical errors in length, width, and height of the estimated layouts can be assessed through evaluation of the 3D reconstructed layouts. Here, 3D reconstruction is performed following the proposed approach in Lee et al. (2009). To simplify the problem, three different parameters (λ_x, λ_y, *and* λ_z) are defined for each part (main corridor or accessory hallways) of the layout in the object space. Considering an arbitrary 3D coordinate system defined by the estimated orthogonal vanishing points in the object space, the ground truth and the estimated corridors (both reconstructed in this coordinate system) can be compared using these three parameters. For example, λ_x could be defined as the width of the 3D reconstructed main corridor divided by the width of the ground truth main corridor. With the same rational λ_y and λ_z could be defined as the ratio of length and height of the 3D reconstructed layout to the length and height of the ground truth layout. In table 4.3, width, length and height of the created layout (Figure 4.9) are compared to the ground truth layout in 3D space.

Here, the main reason that we chose λ_x, λ_y, *and* λ_z parameters for accomplishing this comparison is that both 3D ground truth layout and the 3D estimated layout are generated using estimated orthogonal vanishing points. Since, reconstruction of 3D model from a single image is an ill posed problem, the true scale factor remains as unknown. Therefore, comparison of the relative distances between the structural planes would be more logical than comparing the coordinates of the correspondent vertices in the layout through RMSE (measuring Euclidean distances in 3D space).

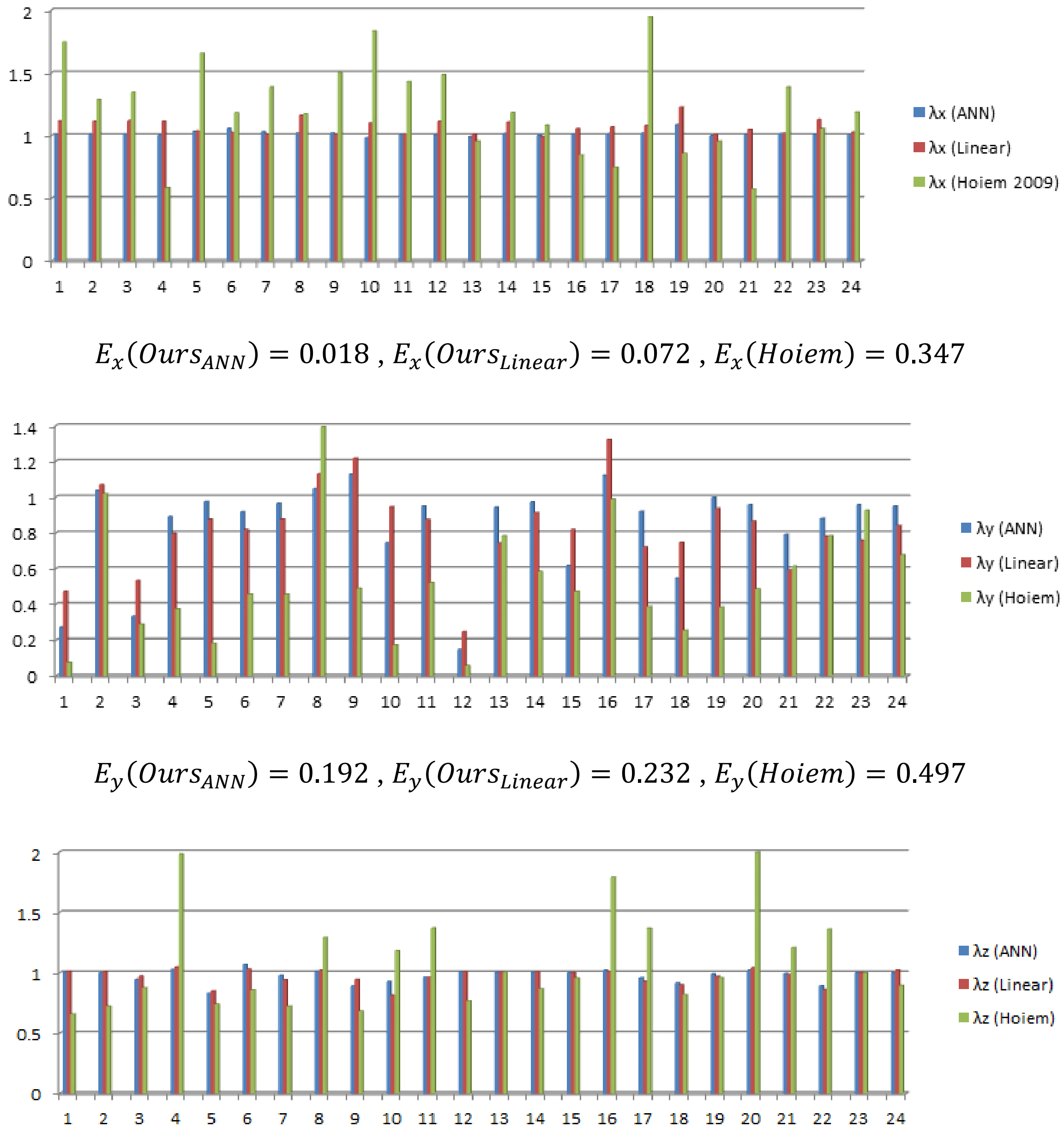
Table 4.3 Ground truth layout and created layout comparison by scale ratios.

Figure 4.9	λx	λy	λz
Major Corridor	*1.1219*	*0.3599*	*1.1934*
Right Corridor	*1.6164*	*2.0545*	*0.9976*
First Left Corridor	*0.9233*	*0.9494*	*0.9460*
Second Left Corridor	*0*	*0*	*0*
Third Left Corridor	*0*	*0*	*0*

As it can be seen in the above table that the number of accessory corridors in the scene is wrongly estimated and the length of reconstructed accessory hallway at the right side of the scene is almost 2 times longer than its correspondent ground truth. Hence, this table can give a better understanding of the geometrical errors of the proposed method.

Figure 4.10 shows the results of comparison between the reconstructed major corridor layouts and their correspondent ground truth layouts in 24 different test images using the abovementioned scale ratios. Note that only 24 ground truth 3D models were available for this comparison that do not cover the whole testing data set. Hence, the generation of more 3D ground truth models are in focus for future works which helps to expand such comparisons. Here, $E_{x,y,z}(M) = \frac{\sum_{i=1}^{n} |\lambda_{x,y,z} - 1|}{n}$ is the average scale ratios differences in either X, Y or Z directions for the applied method "M" and n is the number of test images. The selected images have almost the same scene complexity, so that the comparison of their reconstructed layouts seems rational. Notice that scene complexity by itself is a subjective term which may cause confusion. To facilitate the understanding of this subject, scene complexity is defined as a function of four major factors which are: a) Type of scene layout or the number of structural planes, b) Presence of objects, c) Presence of occlusions, and d) Depth of the corridor.

As it can be seen in Figure 4.10 the proposed method is providing better results compare to the method of Hedau and Hoiem (2009). However, our proposed method was more successful in the estimation of scene layout width and height (λ_x and λ_z are close to 1) over the test images. While, it has more problems in estimation of the true length of the corridors [$E_y(Ours_{ANN}) = 0.192$]. This is a very critical issue which must be scrutinized. A typical explanation for this may directly emerge from the selection of features for scoring layout hypotheses. It should be noted that the performance of selected features is very much sensitive to the presence and detection of straight-line segments. Basically, they lose their accuracy while the straight lines are hard to detect. This issue become more critical when the front face of the major corridor is residing far from the camera. Hence, more study must be performed on this subject in the future.



$$E_x(Ours_{ANN}) = 0.018 \ , \ E_x(Ours_{Linear}) = 0.072 \ , \ E_x(Hoiem) = 0.347$$

$$E_y(Ours_{ANN}) = 0.192 \ , \ E_y(Ours_{Linear}) = 0.232 \ , \ E_y(Hoiem) = 0.497$$

$$E_z(Ours_{ANN}) = 0.037 \ , \ E_z(Ours_{Linear}) = 0.046 \ , \ E_z(Hoiem) = 0.292$$

Figure 4.10 Scale ratios differences between the 3D reconstructed ground truth layout and the created layout. *X* axis is showing the image index and *Y* axis is showing the scale ratio.

4.5.4 Failure Cases

As mentioned previously in the algorithm of the proposed method, layout hypotheses are generated through intersection of both physical line segments and virtual rays created through vanishing points. However, fragmented straight-line segments and virtual rays together may cause the created hypotheses to deviate from the true layout borders specially when the length of a corridor is too long. This is one big reason in increasing failure cases. Either poorly estimated vanishing points or truncated line segments may result in poorly generated layout hypotheses. In these situations, even though some parts of the generated layout are aligned with the true layout borders, the other parts will be deviated from the reality when layout lines are getting closer to the image borders.

The other reason for failure would be the occlusion of the layout borders through the presence of objects, furniture or people. When the real layout boundaries are fully occluded on the floor, the physical line segments detected from the ceiling-wall boundary along with the virtual rays of vanishing points may not be enough to identify the underlying scene layout. Also, there are some other failure cases which are mostly because of inability to identify orthogonal vanishing points, detection of wrong line segments on glass surfaces or waxed floors, misaligned boundaries, no lines supporting down the corridor or fully occluded side-wall boundaries. Note that if orthogonal vanishing points could not be estimated in an image while straight lines are abundant in the scene, one solution would be to add more tilt to the camera’s rigid body at exposure time. Figure 4.11 depicts some exemplar failure cases.

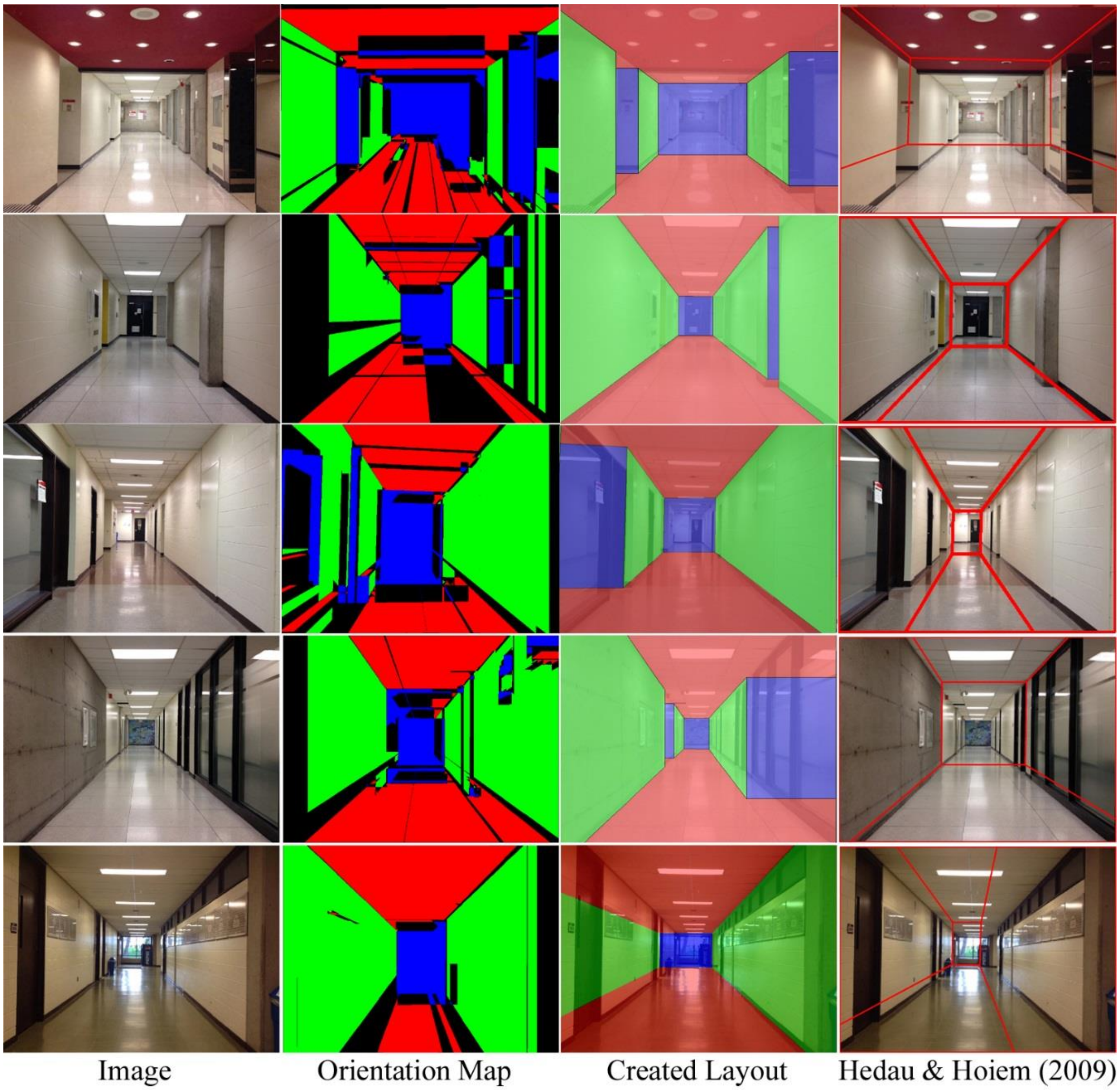


Figure 4.11 Example of the failure cases due to wrong major corridor estimation, wrong corridor depth estimation or missing/misplacement of side cubes.

In Figure 4.11 the created layout hypotheses are deviated from the actual scene layout. The most conspicuous problems in the above images are: a) wrong depth estimation for the major box hypothesis, b) wrong side cube generation. Both issues are interesting problems which can be tackled in the future researches.

4.5.5 Limitations

Overall the algorithm could manage to estimate acceptable major corridor and select the correct number of side corridors in most of the images. However, the proposed edge-based layout estimation method can only be applied to Manhattan framed corridors and it is not able to estimate the indoor corridor layout when the orthogonal vanishing points cannot be estimated in the scene. Also, the proposed method could not filter out inaccurate edges, specially edges detected on waxed and glass surfaces which cause anomalies in both orientation map and geometric context. Moreover, the identified features for selecting the best hypothesis are not perfect and could not fully represent the reality of the scene. Hence, the proposed algorithm cannot precisely estimate the depth of the corridors in some cases. This is the main limitation of the proposed method caused by wrong estimation of the major corridor front face. Also, the other limitation is that the method does not consider line segments with small length while there is not enough boundary support for small length lines. This will reduce the number of generated layout hypotheses in the pool and consequently limits the possible layout choices.

4.6 Summary

In this chapter, we proposed an automatic 3D indoor corridor layout estimation method from a single image which covers a full chain of single image modeling. The focus of this chapter is on proposing a method which utilizes middle-level perceptual organization, which relies on finding the ground-wall and ceiling-wall boundaries using detected line segments and the orthogonal vanishing points. 3D modeling of indoor spaces is not a trivial task, and it involves with major problems. These problems may directly inherit from the modeling approach itself, or the adopted data gathering technique. Here, the proposed indoor corridor layout estimation approach is following the Manhattan rule assumption to simplify the structure of the indoor corridor layouts. What makes the proposed method more conspicuous than the other methods is that the incoming estimated layout is not restricted to only one corridor. We addressed the indoor corridor layout estimation problem by hypothesizing-verifying multiple box primitives. The proposed method relies on both detected line segments and virtual rays created by orthogonal vanishing points to estimate indoor corridor layouts. This method can easily handle the presence of accessory hall ways and occlusions in corridor scenes even the objects were occluding some parts of the floor-wall or ceiling-wall boundaries. This feature beside the compatibility of the estimated layout to the combination of orientation map and geometric context are the main advantages of the proposed method. The proposed method shows that by applying a prior knowledge (knowing camera's distance to the floor), the 3D layout of an indoor scene can be successfully recovered using a single image. A very interesting future problem would be the integration of the created individual indoor layouts which is a huge step towards complete indoor space modeling.

Chapter 5

Layout SLAM for 3D Indoor Corridor Reconstruction

In this chapter, a recently developed visual Simultaneous Localization and Mapping (SLAM) technique, known as Layout SLAM, is introduced. This real time indoor corridor layout estimation method complies with the Manhattan World Assumption at indoor spaces. The proposed Layout SLAM architecture has two major sections, known as the "*front-end*" and the "*back-end*" together dealing with the captured video frames and the Extended Kalman Filtering (EKF) inference. The system initializes by introducing the scene layout and its structural corner point features which permit the real time camera localization on the run. The proposed method detects straight line segments and estimates orthogonal vanishing points to introduce physically plausible layout hypotheses at every instance. The created layout hypotheses for every video frame must go through a matching process to find the best fitting layout hypothesis of the scene. Hence, a layout structural corner points matching scheme is introduced with a feature matching cost function which considers both local and global context information. The proposed cost function consists of a unary term, which measures pixel to pixel orientation differences of the matched junctions, and a binary term, which examines the angle differences between directly connected layout junctions. Consequently, Layout SLAM can build an online sparse map of the indoor corridors layouts which enables the system to deal with the presence of few geometrical features and absence of texture in the scene. Layout SLAM is robust against error accumulations caused by sudden changes of camera orientation through introducing a rotation compensation variable. The system estimates the amount of rotations to be compensated through consecutive layout model and vanishing points matching on a unit sphere. The experiments performed on York University and the RAWSEEDS datasets. Results depict that Layout SLAM performs robustly while produces very limited orientation errors.

5.1 Introduction

Simultaneous Localization and Mapping (SLAM) is the ensemble of techniques for building the globally consistent map of an environment and localizing the moving platform within that environment. SLAM is feasible in a wide range of environments including indoor, outdoor, airborne and underwater places. Moreover, SLAM is a solution for many problems including navigation, 3D mapping and inspection with both autonomous and manned platforms. Multiple sensors are used to perform SLAM and the most popular ones include: a) 2D/3D laser scanners (range and bearing sensors); b) perspective cameras in form of monocular, stereo, omnidirectional vision (bearing-only sensor); c) sonar and radio frequency beacons (range-only); and d) depth (RGBD) cameras (range and bearing).

In this chapter, we combined our previously introduced single image indoor corridor layout estimation method with SLAM to “recognize” or “re-map” the observed indoor corridor layouts. Since the proposed method intends to map indoor corridor layouts, it is called “Layout SLAM”. Here, the focus is on visual SLAM (VSLAM) implemented using monocular vision. Compared to range sensors, monocular cameras have the benefit of gathering denser visual information from the environment using cheaper and lighter sensors. Also, real-time detection and recognition of objects are less challenging using images compared to sparse point clouds. As such, visual SLAM is extensively applied in indoor mapping, augmented reality and robotics applications. However, the main drawback of a monocular camera is its inability to perceive range directly; determining the 3D location of observed points requires at least two views as well as the knowledge of the relative motion of the camera between the views. Inability to measure range also results in scale ambiguity; that is, the built map will be defined up to an arbitrary scale.

The true scale can only be recovered using auxiliary sensors or external measurements from the scene (Engel et al., 2014). Another critical issue is the sensitivity of visual SLAM systems to irregular camera motions. For instance, if a camera is rotated substantially, tracking assumptions used in conventional visual SLAM will not hold true anymore. Therefore, we introduced a new technique based on matching orthogonal vanishing points on a unit sphere, which provides Layout SLAM with the ability to handle rapid camera motions. Matching vanishing directions of consecutive video frames on a unit sphere provides the ability to suppress the effect of rapid camera movements and mitigates the layout and features matching on the run.

Different types of features have been introduced to visual SLAM systems, with point and edge features being the most common (Civera et al., 2010; Davison et al., 2007; Eade and Drummond, 2009; Klein and Murray, 2008; Konolige and Agrawal, 2008; Nist´er et al., 2004; Sibley et al., 2010; Zhou et al., 2015). Point features can be extracted in different ways, such as SIFT key points (Lowe, 2004), Harris Corner detection (Harris and Stephens, 1988), or AGAST corner detection (Mair et al., 2010; Rosten and Drummond, 2006). Point features have favourable properties; they can be easily detected in the image and they can be simply matched, which are both properties suitable for many environments. Hence, systems based on point features are fast and reliable. However, sufficient, reliable point features are not available in the case of objects with no texture or homogeneous texture, e.g. unicolor walls; in fact, in many man-made structures point features are difficult to extract while edges are readily available.

In such cases where textures are homogeneous, edge features can replace point features and thus be valuable features. As well, straight line segments can pose additional constraints on the object-space coordinates and provide higher redundancy for camera pose estimation. Despite these advantages, edge-based SLAM has some serious challenges. A tangible example would be identifying edge correspondences between two consecutive images (Meltzer and Soatto, 2008). Moreover, an edge might be detected in one image as a unique line segment, while it might be sliced into various shorter line segments in another image. Therefore, only a subset of all detected edge features can be used for successful matching.

Considering the above information, Layout SLAM tries to take advantage of both points and lines to achieve robustness against the conditions of indoor corridor scenes. Hence, two types of features are incorporated into the Layout SLAM system which are the layout structural corner point features supported by image line segments, and normal point features that are conspicuous in the scene. Note that most of the current SLAM algorithms are not able to directly create 3D models of low-texture environments. Often, the incoming sparse map conveys little information about the geometric characteristics (geometric model) of the indoor corridor scene. Yet Layout SLAM with its incorporated features, not only can create real-time 3D models of the scene, but also can overcome most indoor corridor conditions where only a few distinct point features are present.

In general, we focus on using a single image estimated indoor corridor layout to initialize the Layout SLAM system, and estimate the camera pose and 3D positions of features in corridor scenes through EKF. The structural layout corner points feature, initialized in the Layout SLAM system tend to improve the robustness of both state and layout estimations on the run. The proposed method takes advantage of both point and line entities to compensate for the insufficiencies of current SLAM systems at indoor corridors. Note that the computational time was not considered as an influential factor in the abovementioned architect and real-time processing will be considered in future works.

First, the spatial layout of the scene is created by fully applying line features of the first video frame. Then, the layout structural corner point features are identified which are supported by line features aligned with the true orthogonal directions. These features are initiated in the Layout SLAM system and enforce it to be bounded to the identified orthogonal directions on the run. Also, vanishing directions of consecutive video frames are matched on the unit sphere helping the algorithm to find the accurate feature matching search space. To match layout structural corner point features, a new feature matching cost function is proposed which considers both local and global context information.

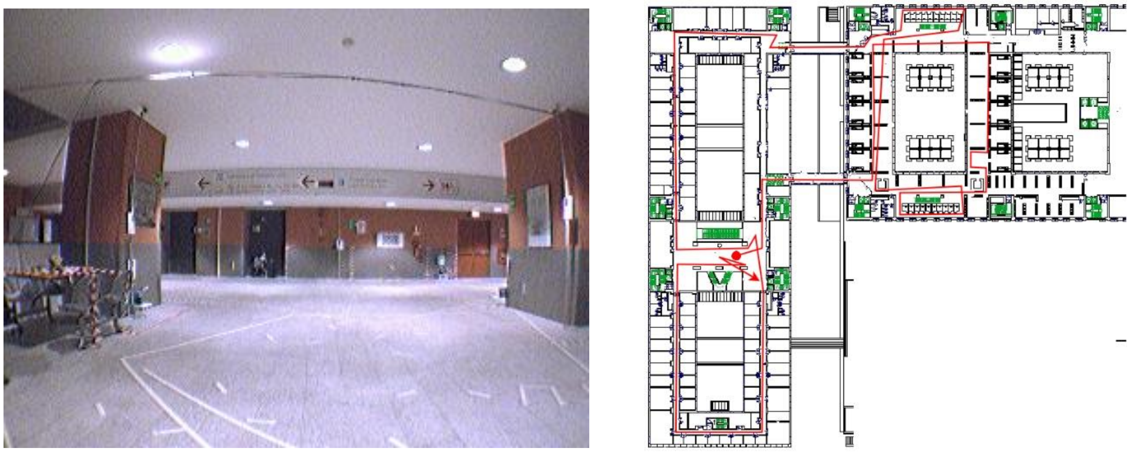
The proposed method can directly create 3D models while dealing with the presence of few geometrical features and absence of texture by benefiting from image based structural straight-line segments. This study demonstrates that layout understanding through straight line-segment detection and orthogonal vanishing point estimation could improve both camera state estimation and direct 3D modelling in visual SLAM system while dealing with low-textured environments.

5.2 Camera Model and Data

The pinhole camera model is the simplest model to describe the camera imaging process recognized by a flat image plane and a light-barrier hole (the camera perspective centre). Here, the pinhole camera model is considered for applying the mathematical equations which explain the camera observations in the proposed Layout SLAM system. Note that to reconstruct the rays of light that have created any image point (reversible optical path), the interior orientation parameters (IOPs) of the camera are needed. These parameters include the principal distance, offsets of the principal point, and other intrinsic camera parameters such as lens distortions. Having identified these parameters, the image observations can be undistorted to ensure the collinearity condition. Here, the interior orientation parameters are obtained by calibrating the cameras using MATLAB calibration toolbox (Bouguet, 2004), and all images undistorted accordingly.

To evaluate the performance of the proposed Layout SLAM algorithm in this chapter, several datasets are used. The first is the newly generated dataset which introduced in chapter 3. This dataset is associated with ground-truth corridor layouts and camera trajectories for keyframes. The second is the publicly available Robotics Advancement through Web-publishing of Sensorial and Elaborated Extensive Data Sets (RAWSEEDS) by Bonarini et al., (2006). This dataset is gathered from buildings at the University of Milano-Bicocca, in Milan, Italy. Many indoor locations explored by a robot that crawled through floors of an office building and its nearby places. Various architectural structures are detected such as: a) Narrow corridors with offices on their sides which their entrances have various depths and deeply recessed within the walls. b) Wide and narrow hallways of different types connecting to spaces with various dimensions and shapes occupied with chairs and tables. c) Narrow bridges with glass walls connecting buildings. d) A library with many open bookcases, computer desks, spaces with chairs and tables, and open halls. e) Different types of passageways and doors. Note that the floor is very smooth which brings more stability to the captured videos.

Although RAWSEEDS dataset is not only covering indoor corridors and most of its scenes have complicated outlines, it is chosen for evaluation of the Layout SLAM because of its available ground truth camera trajectories. Here, the Bicocca_2009-02-25b dataset of RAWSEEDS is selected due to the robot's path and the static environment which does not include moving people or objects. Figure 5.1 shows a sample video frame from RAWSEEDS (Bicocca_2009-02-25b) dataset and the robot's path on building floor plan.



(a) (b)

Figure 5.1 (a) a sample video frame from RAWSEEDS (the Bicocca_2009-02-25b) dataset and (b) the robot's path on the floor plan of the building.

The third dataset that utilized in this chapter is York Urban Line Segment database (Denis et al., 2008). This dataset is a collection of 102 images that only 45 of them are captured at indoor spaces. These images are mostly taken at York University campus buildings with a calibrated Panasonic Lumix DMC-LC80 digital camera (640×480 image size). For each image a set of straight-line segments and their corresponding three orthogonal vanishing points, are provided. Note that the provided vanishing points conform to 3D orthogonal frame of the indoor space.

5.3 Layout SLAM Architecture

The structure of a SLAM system is comprised of two main parts: the "*front-end*" and the "*back-end*". Together, these two components of the SLAM architecture are dealing with sensor data and the system inference about it. The front-end delivers the sensor data in form of models that can be used for estimation, while the back-end solely infers from the provided information by the front-end. The major parts of the front-end are "*feature extraction*" and "*data association*". Data association itself includes feature tracking in short term and identifying map parts that belong to the same environment in long term. In general, the back-end is dedicated to map estimation.

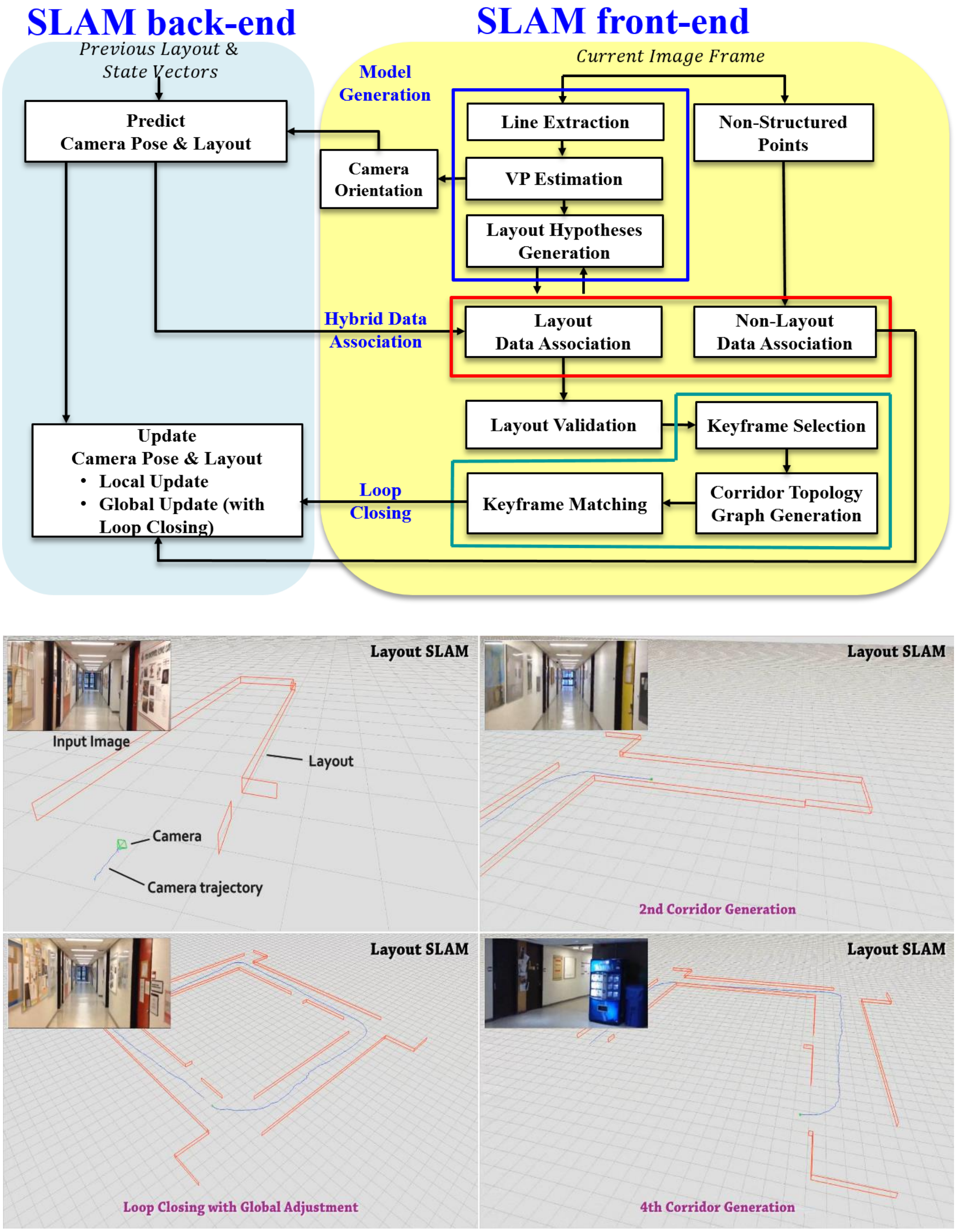


Figure 5.2 Layout SLAM architecture with model-based loop closure (top); Schematic view of Layout SLAM performance at first floor of PSE building, York University (bottom).

The above architecture of the SLAM system enables the introducing of new SLAM algorithms by altering and improving the current SLAM system's main components. In this thesis, we followed the mathematical concepts developed by Davison (2003) and modified the open-source MATLAB code provided by Civera et al., (2010) to introduce Layout SLAM algorithm. The key concept of the Layout SLAM method is the probabilistic layout estimation of indoor corridor scenes. Beside introducing a rotation compensation variable and a new technique for loop closing detection, both initialization scheme and feature selection and feature matching blocks of Davison's (2003) Mono SLAM algorithm are significantly altered in Layout SLAM algorithm. Figure 5.2 shows the overall workflow of the proposed Layout SLAM algorithm that explained thoroughly in both chapters 5 and 6 of this thesis. In the following sections, more details about the proposed Layout SLAM algorithm will be presented.

5.4 Layout SLAM Back-end

In a typical SLAM system, given a set of measurements, several unknown variables should be estimated through the back-end block. Usually, these unknown variables in a visual SLAM system are: a) the discrete set of camera poses (camera trajectory), and b) the position of landmarks in the scene. The proposed Layout SLAM method uses the Extended Kalman Filtering (EKF) as its core back-end block to perform the predictions and updates on the desired variables. The extended Kalman filter can linearize about the current mean and covariance estimate.

5.4.1 Layout SLAM State Vector

Layout SLAM provides the current estimate of the camera and all features states as well as the uncertainty of these estimates. The corridor scene layout is initialized through its layout structural corner points features at system start-up and grows dynamically as it is updated by the EKF. The created Layout SLAM is comprised of a state vector $\hat{x}$ and covariance matrix P. State vector $\hat{x}$ is composed of the camera state ($\hat{x}_v$), structural point features ($\hat{y}_s$) and normal point features ($\hat{y}_n$) state estimates.

$$\hat{x} = \begin{pmatrix} \hat{x}_v \\ \hat{y}_s \\ \hat{y}_n \end{pmatrix}$$

$$P = \begin{bmatrix} P_{xx} & P_{xy_s} & P_{xy_n} \\ P_{y_sx} & P_{y_sy_s} & P_{y_sy_n} \\ P_{y_nx} & P_{y_ny_s} & P_{y_ny_n} \end{bmatrix} \tag{5.1}$$

$$x_v = \begin{pmatrix} r^w \\ q^{wc} \\ v^w \\ \omega^c \end{pmatrix}$$

Here, the camera state vector *(x_v)* comprises of a three-dimensional position vector *(r^w)*, orientation quaternion *(q^{wc})*, velocity vector *(v^w)*, and angular velocity vector *(ω^c)*. The superscripts *W* and *C* represent the world frame and the camera frame respectively. Feature state $\hat{y}_s$ represents the three-dimensional position vectors of identified layout structural points which are the corridor junctions. Also, the feature state $\hat{y}_n$ represents the 3D position vectors of normal points which are randomly selected from a bunch of conspicuous corner points in the scene. Camera and features state estimates will be updated during camera motion and all features observation. As soon as a new layout structural corner point feature is observed, the layout will grow with new states. The mean estimates of the camera and features states as well as a first-order uncertainty distribution, associate the estimated scene layout over time. Note that the probability distribution over the mentioned parameters is approximated as Gaussian distribution.

5.4.2 Layout SLAM Features

One of the main differences of Layout SLAM system to the other SLAM algorithms like Mono SLAM is the incorporation of structural layout point features along with conspicuous corner point features in its state vector. The number of structural layout point features compare to the normal point features are very limited in a typical corridor scene. The corridor layout itself can impose a constraint on the map which has to be estimated and therefore the layout structural points together can assist accurate real-time localization. However, the structural point features are not always visible in a corridor scene, especially when the camera is turning from one corridor to the other. This necessitates the presence of normal point features in the layout SLAM state vector.

The critical role of maintaining features as long-term landmarks in visual SLAM systems has been irrefutably proven in SLAM research. Layout SLAM is using the Extended Kalman Filtering as its core back-end section. Yet, EKF is not suitable for maintaining many features for a long time in the system. Hence, the primary goal in layout SLAM is to capture a sparse set of high-quality layout structural corner points and maintain them in the system for as long as possible. These features can influence the correlation between the map and the camera poses estimates. These features may have position estimates which are uncertain in the reference frame, but they highly correlate the camera pose estimates in many sequences. Holding correlation information of these features in the system for a long time can improve estimates of the other related features and enables the system to recognize known areas after short periods of neglect. Therefore, the goal is to maintain the layout structural corner point features as long-term landmarks while maintaining normal point features for short term periods in the system to improve the camera pose estimates.

5.4.3 Layout SLAM Motion Model

The ability to measure the layout structural corner point features in many sequences is directly affected by the adopted motion model in the system. Introducing a motion model for a camera which is carried by a person walking inside a corridor is not fundamentally different from the motion model of a wheeled robot moving smoothly on a flat surface. Davison (2003) adopted a "constant velocity, constant angular velocity model" which assumes large accelerations to be unlikely and imposes smoothness to the camera motion. In the layout SLAM algorithm, the same motion model is adopted.

Here, the assumption is that at each time step an amount of unknown acceleration a^w and angular acceleration α^c with Gaussian distribution and zero mean, generates an impulse of the velocity and angular velocity in the system. Note that the uncertainty growth rate in the adopted motion model is determined by the covariance size of the noise vector n.

$$n = \begin{pmatrix} v^{w} \\ \Omega^{c} \end{pmatrix} = \begin{pmatrix} a^{w}\Delta t \\ \alpha^{c}\Delta t \end{pmatrix} \tag{5.2}$$

5.4.4 Layout SLAM Prediction and Update

The introduced state vector in the layout SLAM system is revolving in two alternating steps. The first step is called the prediction step, regarding the camera movements between the actual image capture. This important blind period movement must be predicted using layout SLAM adopted motion model. The second step is called the update step, regarding the update which is needed for the state vector after measurements of the features is attained.

Considering the constant velocity, constant angular velocity model in the layout SLAM system, and assuming the introduced noise vector n's covariance matrix is diagonal, the camera state prediction would be the same as the one proposed by Davison (2003):

$$f_{v} = \begin{pmatrix} r^{w}_{new} \\ q^{wc}_{new} \\ v^{w}_{new} \\ \omega^{c}_{new} \end{pmatrix} = \begin{pmatrix} r^{w} \ + (v^{w} \ + \ V^{w})\Delta t \\ q^{wc} \ \times \ q((\omega^{c} \ + \ \Omega^{c})\Delta t) \\ v^{w} \ + \ V^{w} \\ \omega^{c} \ + \ \Omega^{c} \end{pmatrix} \tag{5.3}$$

In the above equation, the notation $q((\omega^c + \Omega^c)\Delta t)$ denotes the quaternion defined by the angle-axis rotation vector $(\omega^c + \Omega^c)\Delta t$. Note that in the EKF framework, the new predicted camera state f_v is affecting the camera state uncertainty Q_v along with the noise vector *n*'s covariance P_n. Here, Jacobian calculations will help to calculate Q_v:

$$Q_v = \frac{\partial f_v}{\partial n} P_n \frac{\partial f_v}{\partial n}^T \tag{5.4}$$

As it can be inferred from the above equation, the uncertainty growth rate in this system is very much affected by the size of P_n. This is the main incentive to introduce a new variable to the system, which will be discussed in the following section.

Note that not only the camera movement must be predicted by layout SLAM but also the position of features in the image space must be predicted. This prediction is a critical step in the process of measuring a feature which already exists in the layout SLAM state vector. Moreover, to successfully apply the layout SLAM system, proper feature observations should be made. Davison (2003) used the pinhole camera model to predict the image position *(u, v)* of a 3D point feature. Here, we used the same procedure for predicting 2D positions of normal point features relative to the camera in the proposed Layout SLAM method.

$$h_L^c = R^{cw}\,(y_i^w - r^w)$$

$$h_i = \binom{u}{v} = \begin{pmatrix} u_0 - fk_u \dfrac{h_{Lx}^c}{h_{Lz}^c} \\ v_0 - fk_v \dfrac{h_{Ly}^c}{h_{Lz}^c} \end{pmatrix} \tag{5.5}$$

Here, *u0*, *v0*, *fku*, and *fkv* are the camera intrinsic calibration parameters, R^{cw} is the rotation matrix, and h_L^c is the vector connecting the camera projection center to the 3D point feature y_i^w in the camera frame *C*. Note that, R^{cw} can affect the position of the predicted point, and also R^{cw} itself is affected by the applied motion model in the prediction step of the EKF.

The uncertainty of the above prediction can be represented by the innovation covariance matrix S_i which includes the constant noise covariance *R* of measurements as well. Considering S_i, an elliptical search window in the image space can be introduced for the predicted point where its corresponding match should lie with high probability. It should be noted that the update phase in EKF can be performed after the feature matching is completed. More information on this will be presented in the coming sections.

While normal point features are being predicted in the image space, the structural point features (layout junctions) need to be predicted as well. Here, the procedure for predicting the structural point features involves the estimated orthogonal vanishing points. The orthogonal vanishing directions must be identified for consecutive video frames, projected onto the Gaussian sphere, and match the corresponding vanishing directions to estimate the amount of relative camera rotation between two frames. By applying the same amount of rotation to the estimated layout in the former video frame while projecting it into the current frame, the layout position in this frame can be roughly estimated. Note that the vanishing point estimation algorithm is assumed to produce zero error. Thus, the maximum value of the normal point features innovation covariance matrix S_i is chosen to represent the uncertainty of this prediction as well. The adopted symmetric 2 × 2 innovation covariance matrix identifies the shape of a two-dimensional Gaussian probability density function over predicted layout junctions in the image space. Setting the number of standard deviations at 3σ provides an elliptical search window around the predicted layout junction that its match probably resides there. Hence, uncertainty helps the algorithm to facilitate the layout matching while it is trying to find the identified structural features in the next video frame. In the following sections, more information on the above subject will be presented.

5.4.5 Layout SLAM Rotation Compensation

As mentioned before, the size of the noise vector covariance P_n is affecting the uncertainty growth rate in the layout SLAM motion model. Even though large covariance enables the system to cope with rapid accelerations, increasing the uncertainty in the system will affect the estimates and necessitates perfect measurements to be made at each time step to constrain estimates. It should be noted that accomplishing perfect measurements in a low textured corridor environment with a relatively narrow angle camera is unlikely. Therefore, small covariance which indicates a very smooth motion with small accelerations must be considered. Thus, a new variable must be introduced for enabling the system to cope with sudden, rapid camera movements. Here, we introduced a rotation compensation variable “δ” to the layout SLAM system to cope with sudden rapid camera movements:

$$\omega_{t-1}^{w} = \omega_{t-1}^{w\,\prime} + \delta_{t,t-1} \tag{5.6}$$

Here, $\omega_{t-1}^{w\,\prime}$ is the amount of rotation in the system after update phase has been accomplished in the EKF at step t-1. This variable will be replaced by ω_{t-1}^{w} in the system before making predictions for step t. Moreover, $\delta_{t,t-1}$ is the amount of rotation difference between two consecutive steps (t-1 and t), which is calculated independently using estimated vanishing points. Here, we assume that $\delta_{t,t-1}$ is free of error. Hence, its uncertainty is not needed to be considered in the EKF.

5.5 Layout SLAM Front-end

Expressing the camera measurements (intensity of each pixel) as the SLAM state analytic function would be the ideal task of the Layout SLAM front-end. Yet, it is extremely hard to write such a function. First, designing a scene representation which is tractable and general is not possible. Second, it is extremely hard to prepare an analytic function which can connect the representation parameters to image measurements even if this general representation exists. Hence, Layout SLAM frond-end module extracts sufficient point features from video frames since these point features observations can be simply modeled within the back-end. As mentioned before, the front-end module also associates each feature measurement to an individual 3D point in the scene which is known as data association. Data association module in the proposed Layout SLAM algorithm associates corresponding features in consecutive video frame measurements (feature tracking) in short term and associates new feature measurements to previous landmarks (loop closure) in long term. Note that the front-end module receives information from the back-end block to support validation and loop closure detection.

5.5.1 Layout SLAM Initialization

In the proposed Layout SLAM algorithm, the front-end module provides an initial guess for the unknown variables at the very beginning of the run. Camera position and its orientation along with the position of both layout structural point features and normal point features must be initialized at the first step. Beside camera position which is hypothesized to reside at the adopted coordinate system's origin, the camera orientation hypothesized to be aligned with the adopted coordinate system's axis. Since the actual positions of landmarks could not be estimated without triangulating from multiple views, their respective positions initialized on a unit sphere which its center resides at the camera projection center. Hence, the initial depth for all the landmarks would be 1. Note that landmarks depths could be estimated more precisely while more observations are achieved during the run. However, these initial guesses of the unknown variables are necessary for the nonlinear optimization process in the proposed Layout SLAM system.

5.5.2 Layout SLAM Feature Extraction

As mentioned before, two types of features (structural and normal points) are contributing to the Layout SLAM system for improving the camera pose and map estimates. Here, the front-end module is responsible for both extraction and tracking of these features. Normal point features are selected among the conspicuous corner points in the scene through applying Features from Accelerated Segment Test (FAST) corner point detection algorithm (Rosten and Drummond, 2006). Note that a conspicuous corner point in the scene can be found where at least two different and dominant edge directions are intersecting at a local neighborhood.

Although identifying normal point features in the scene does not need prerequisites, extracting structural point features requires the scene layout to be estimated in advance. Structural layout point features are directly extracted from the estimated layout in the scene. At the very first step of Layout SLAM initialization, the scene layout structural point features are introduced to the system through the same layout estimation algorithm proposed in chapter 4. Like what has been proposed in chapter 4, edges are extracted from the first video frame and then grouped into straight line segments. Orthogonal vanishing points are estimated, and different layout hypotheses are created. The generated layout hypotheses are evaluated by ANN and finally the best fitting layout hypothesis along with its structural corner points features are introduced into the Layout SLAM system. Note that introducing layout point features to the system may happen again if the system fails to efficiently track the scene layout during the run.

5.5.3 Layout SLAM Feature Tracking

Feature tracking module in the Layout SLAM system is aided by consecutive frames orthogonal vanishing points matching on unit sphere. As mentioned before, straight parallel lines in 3D space can be projected onto the 2D image plane where they can intersect and form a vanishing point. Vanishing points have special geometric attributes that can be utilized in rotation estimation and some other computer vision applications like camera calibration. Note that Layout SLAM is specifically designed for mapping indoor corridor places where Manhattan world constraint is applicable. This constraint allows vanishing points to match more easily in small-baseline video frame sequences. Moreover, this improves the robustness of the rotation estimation where noisy line segments are extracted from video frame sequences.

Representing vanishing points on a unit sphere enables the system to take advantage of the 3D space. Considering the adopted pinhole camera model, the unit sphere can be placed on the center of projection. Figure 5.3 shows 3D straight lines projected onto the image plane and represented by great circles on the unit sphere. These circles are created by the intersection of unit sphere and planes which contain both line and the center of projection. These great circles of parallel lines intersect at a specific point on the unit sphere, and the direction from the center of projection to this point is the vanishing direction (D). Here, the vanishing direction of line segments l_1, l_2 is estimated by intersecting their interpretation planes Q_1, Q_2 on the unit sphere. Vanishing direction in 3D space $d \in R^3$ can be defined through homogeneous coordinates $D \in P^3$. Note that in 3D space, the vanishing direction D in homogeneous coordinates can be transformed into another vector D' by a 4 × 4 rotation and translation matrix.

$$D = [d^T \ 0]^T = [X \ Y \ Z \ 0]^T$$
$$D' = \begin{bmatrix} R & T \\ 0_{1\times3} & 1 \end{bmatrix} D = \begin{bmatrix} Rd \\ 0 \end{bmatrix} \tag{5.7}$$

In the above equation, R is the rotation matrix and T is the translation vector. As shown, the transformed vanishing direction equals *Rd*. This indicates that the vanishing direction transformation is influenced by rotation only. Since a vanishing point on image plane is the projection of the vanishing direction, it can have the same property (Kroeger et al., 2015). Hence, a vanishing point is a translation-invariant feature and consequently rotation can be more accurately estimated through vanishing points. Considering this fact, the front-end module in the Layout SLAM system estimates the relative camera rotation between two consecutive video frames through identifying and matching vanishing points in these frames. Consequently, the feature tracking scheme is improved by real-time rotation estimation since it must search to find candidate feature matches in the image space.

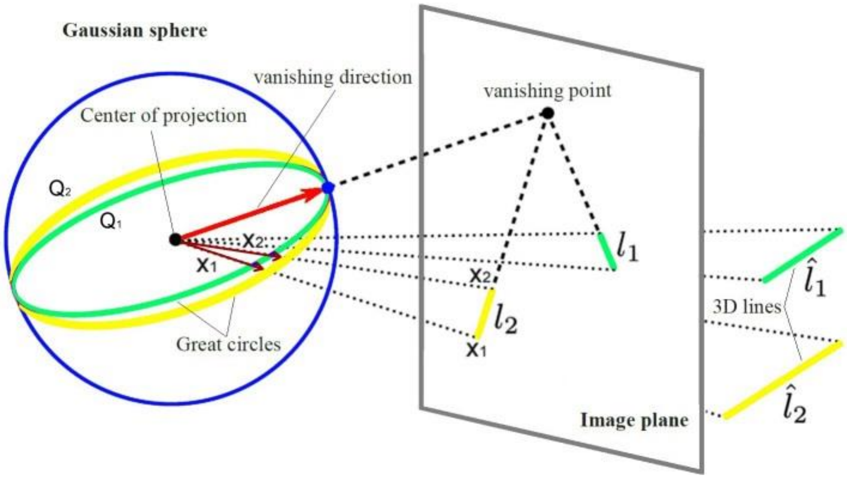


Figure 5.3 Image line segments l_1, l_2 of 3D parallel lines intersect at a vanishing point on the image plane. The same vanishing point can be parametrized as a vector pointing towards the intersection of great circles of lines l_1, l_2 and the unit sphere (original figure from Kroeger et al., 2015).

After the estimation of orthogonal vanishing points in each video frame, the corresponding vanishing directions are identified on the unit sphere. Since vanishing directions of two consecutive video frames are referring to the same indoor corridor layout, transforming from one to another will identify the amount of relative rotation between these two frames. Hence, having identified the relative rotation between consecutive vanishing directions, the same amount of rotation can be applied to the structural elements of the indoor layout (lines and corners) on the unit sphere. Therefore, the estimated layout in frame "*t-1*" can be back projected onto frame "*t*" with no rotation difference. This process enables the algorithm to remove the effect of abrupt camera rotation changes and facilitates the feature tracking of both normal and structural layout point features in the image space.

5.5.4 Layout SLAM Feature Matching

Feature matching in the data association module of the proposed Layout SLAM system is performing a very critical rule. As mentioned before, the Layout SLAM feature vector contains two types of features. Although both features are representing points, they will go through different schemes for selection and matching in the proposed algorithm. Normal point features follow the rules set by Davison (2003), while layout structural corner point features will benefit from local image orientations and global cues of indoor corridor structures for both selection and matching.

For normal point features, matching is done through a normalized cross-correlation search for the relatively large (11 × 11 pixels) template patch projected onto the current camera estimate. Note that 11 × 11 pixels template patch has proven to reveal enough information for matching (Davison 2003). The template should be scanned over the image to find a peak which indicates that a match is resided at that image location. Since the search for a match on the entire image space would be computationally expensive, the search space must be narrowed down to maximize efficiency. Here, Layout SLAM prediction module can help as an active approach for narrowing down the search space in the image.

For structural corner point features, matching scheme is quite different. Here, the local orientations for the junctions of the estimated indoor layout in video frame "*t-1*" will be compared to the orientations of their corresponding junctions in the created layout hypotheses of video frame "*t*". Also, the examination of the constructed angels between the connected junctions will enforce the global consistency in the matching scheme. Note that the transition vector of the camera’s center of projection between two consecutive video frames is assumed to be very small. Yet, this small transition vector will cause the projected layout of the video frame “*t-1*” not to reside on the exact layout for video frame “*t*”. Therefore, identifying a search region in the image space around every junction in question is critical for the system which can potentially reduce the number of valid layout hypotheses in video frame “*t*”.

Here, the maximum value of innovation covariance matrix for the normal point features helps to provide an elliptical search area around the layout junctions in question. The junctions of the matching candidate layout would reside there with high probability. Considering the extracted straight-line segments and the estimated vanishing points in the current video frame, several layout hypotheses are generated. Note that the estimated layout in the previous video frame provides valuable information about the structural corner points which should be matched to the ones in the current video frame. Figure 5.4 highlights a search area around one of the structural corner points of the projected layout and the potential junction matches from the created front-face hypotheses. This figure shows how some front-face candidate hypotheses can be simply ruled out by considering their positions with respect to the identified search region.

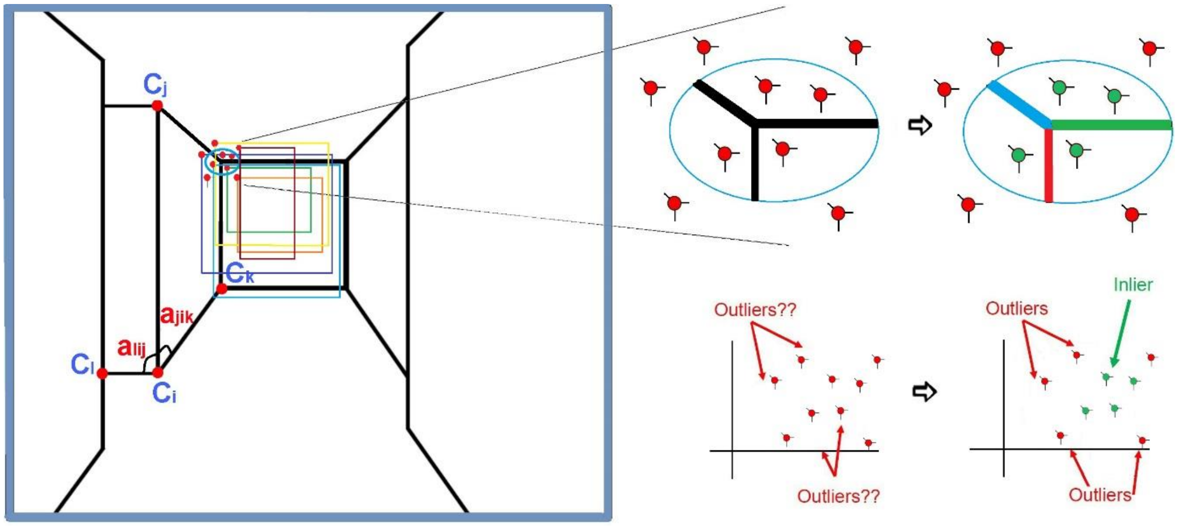


Figure 5.4 Left: several front-face candidate hypotheses forming rectangular regions in the image space and some angles between connected structural corner points of the projected layout. Right: the structural corner points of several front-face candidate hypotheses are ruled out by residing outside of the search region.

As mentioned before, a corner point feature reveals only the local information about the layout structure. Hence, the matching layout hypothesis cannot be identified in the current video frame only through the orientations of its junctions. Hence, beside orientation another feature must be included in the matching scheme to impart the global structure information. This feature is set by considering other structural corner points which are directly connected to the structural corner in question. In total, two types of features were adopted; a) the planar orientations which are defined by the type of a structural corner in question, and b) angles (α_i, α_j, α_k ...) between the lines connecting the corner in question (C_i) to the other directly connected corners (C_j, C_k, C_l ...). Figure 5.4 shows the angle features for structural corner C_i. The proposed cost function (S) consists of a unary term which measures the orientation differences of the matched corners, and a binary term which measures the angle differences between corresponding layout structural corners, as following:

$$S = \frac{\sum_{i=1}^{Q}\left[a \times \frac{\sum_{i=1}^{n} O(i)}{n} + (1-a) \times \frac{\sum_{i=1}^{m} P(i)}{m}\right]}{Q} \tag{5.8}$$

Here, Q is the number of junctions in the projected layout of the previous video frame and "*a*" is the weight value which balances the unary term and the binary term (*a = 0.5* in the experiments). Also, "*n*" is the number of image pixels in the search area (here 11 × 11 pixels) around the junctions in question and "*m*" is the number of other directly connected structural corner points to the candidate junction point. The unary term *O(i)* measures pixel to pixel orientation difference between the matched corner features derived from the estimated indoor layout of previous video frame and the current video frame:

$$O(i) = \left|O_i^M - O_i^F\right| \tag{5.9}$$

The binary term is designed to deal with relationships between neighbor features in terms of angles. It is calculated for all corner features which are directly connected to the matched corner features derived from the indoor layout and the current video frame. For angle differences, we start from the vertical vanishing point direction at each corner feature and count the corresponding angles clockwise. The angle difference $\left|\alpha_{jik}^M - \alpha_{jik}^F\right|$ between the lines connecting the matched corners (C_i^M, C_i^F) and the other directly connected corners

$(C_j^M, C_k^M, C_j^F, C_k^F)$ is valued by either 0 or 1. Note that the suggested layout corner match must maintain the orthogonality of the indoor structure. If the connected line between the candidate junction and the other corners converge to the estimated vanishing point, then the value of the angle difference would be 0 and 1 otherwise. To measure the convergence the bias of the vanishing point estimation error is considered. Moreover, if a junction is missing in the layout hypothesis, the hypothesis would be penalized by receiving the value of 1 for each missing angle.

$$P(i) = \left|\alpha_{jik}^M - \alpha_{jik}^F\right| = \begin{cases} 0 & if \quad \alpha_{jik}^M = \alpha_{jik}^F \\ 1 & \quad otherwise \end{cases} \tag{5.10}$$

For each video frame, a candidate layout and its corresponding layout junctions which minimize the proposed cost function is selected as the optimal layout match. Note that if the minimum cost is larger than a certain threshold (= 0.7), the matches are not considered as matched layouts. This threshold has an impact on the initialization of a new layout to the system if no layout matches could be found in several consecutive video frames. Hence, an experiment is performed for optimizing this threshold, which will be presented in the experiments section.

5.6 Experimental Results

To examine the performance of the proposed Layout SLAM algorithm, two main sites of the prepared York University dataset along with the dataset of *Biccoca_2009-02-25b* from

RAWSEEDS are used. The descriptions of York University sites (Ross and Petrie Science buildings first floors) are delivered in chapter 3. Overall, three different video sequences from the prepared dataset have been used along with the dataset of *Biccoca_2009-02-25b* from the RAWSEEDS. Table 5.1 provides more details about the selected videos.

Table 5.1 Specifications of video frames used for Layout SLAM validation.

Dataset	*Platform*	*Video Frames*		*Ground Truth*		*Data*	
		Camera	*Resolution (pixels)*	*Trajectory*	*3D Model*	*#Frames*	*length*
York University Petrie Science Building, one loop	*Human*	*Apple iPhone 4s*	*1920 x 1080*	*Only Keyframes*	*Laser point cloud*	*9,245*	*159m*
York University Ross Building, one loop	*Human*	*GoPro Hero5*	*3840 x 2160*	*Only Keyframes*	*Laser point cloud*	*3,627*	*112m*
York University Ross Building, two loops	*Human*	*GoPro Hero5*	*3840 x 2160*	*Only Keyframes*	*Laser point cloud*	*8,553*	*315m*
Biccoca_2009-02-25b of RAWSEEDS, several loops	*Robot*	*Unibrain Fire-i 400*	*320 x 240*	*The whole path*	*Null*	*52,695*	*774m*

Note that videos recorded by hand held cameras benefitted from the video stabilization quality and all of them were captured at 30 frames per second. For all the scenes, the cameras start their first motions from a position observing the indoor corridor from the mid part of the hallway. At the beginning, several layout structural corner point features including top and bottom corners of the connecting corridors as well as other salient point features such as points from paintings and posters are visible. In the following sections, experimental results would be presented which incorporates the tests on the performance of the rotation compensation variable as well as Layout SLAM trajectory results on different datasets, and evaluation of estimated layouts in image space.

5.6.1 Evaluation of Rotation Compensation Variable

Layout SLAM performance is highly dependent on geometric content of video frames acquired through straight line segments. Here, the main assumption is that an indoor corridor scene is bounded by several flat surfaces. Thus, straight line segments are used as low-level features to provide information about the scene layout. Straight line segments extracted from flat surfaces will serve as a basic tool to detect the layout structure.

As mentioned before, LSD method is adopted in Layout SLAM architecture to accurately detect straight line segments in the image space. The adopted LSD method is tested on York Urban dataset images and the average of 537-line segments per image were detected. Having detected straight line segments in the image space, the orthogonal vanishing points are estimated applying the adopted method described in chapter 4. Table 5.2 reveals the incoming results of the adopted vanishing point estimation method using all detected line segments. Note that results are obtained by implementation of methods in MATLAB software using a desktop computer with 3.4 GHz Intel Core i7 processor and 8GB RAM.

Table 5.2 Vanishing point estimation results on York Urban dataset images.

Dataset	*Line detection*		*Vanishing point estimation (average)*				*Focal length error (average)*	*Time*
	Method	*Average # of lines*	*Method*	*Frame error*	*Vertical error*	*Horizontal error*		*Running time per image*
York Urban dataset	*LSD*	*537.3*	*RANSAC based*	*1.6861°*	*2.3643°*	*1.1346°*	*7.6%*	*112.04s*

As it can be seen in the above table, the average running time is roughly 112 seconds per image (102 images in total) which is not suitable for an online application. To reduce the computation time and improve the vanishing point estimation results, the number of participant line segments is reduced. LSD itself detects local straight contours on image space based on the gray level changing speed. A line support region is created by grouping local pixels which share the same level-line angle up to a specific threshold. Hence, a rectangle is associated with the local group of pixels and covers the whole line support region. This rectangle could represent a potential line segment if its angle corresponds to the level-line angle of the inside pixels. Here, the same notion is applied to approximate fragmented local straight-line segments, which share the same angle up to a certain tolerance, with an individual rectangle. The angle of this rectangle is considered as the angle of a line which represents these local straight-line segments in vanishing point estimation process. A tangible example is depicted in figure 5.5 where local straight-line segments are shown to be approximated by a rectangle. Note that both thresholds for identifying the closeness and tolerance angle of line segments which candidate them to be approximated by a rectangle, are chosen empirically.

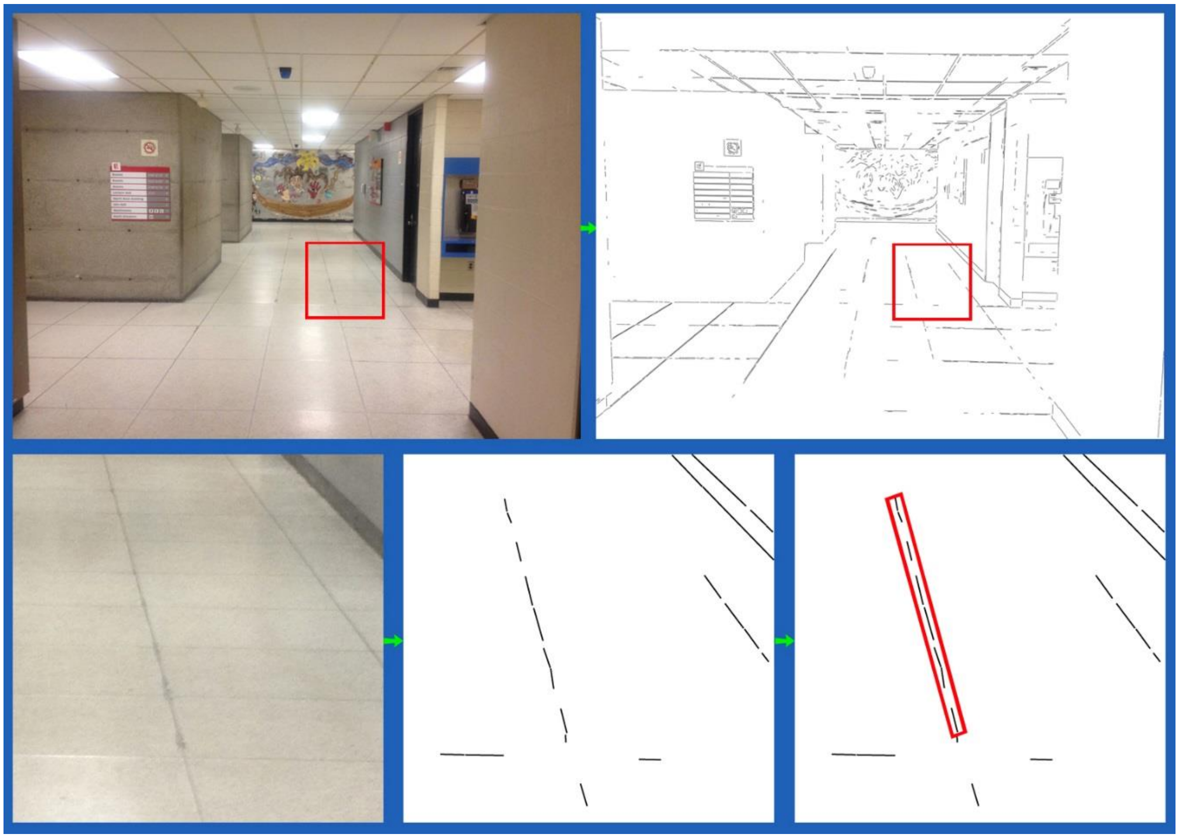


Figure 5.5 Top: an image and its extracted line segments through applying LSD method (area of interest is specified by a rectangle in red). Bottom from left to right: the area of interest, its extracted straight-line segments and local straight-line segments approximated by a rectangle in red.

Having applied the above-mentioned strategy, the number of line segments which can participate in the vanishing point estimation process will be reduced to some extent. Table 5.3 reveals the incoming results of applying the same vanishing point estimation method using the first longest 100-line segments. As it can be seen in this table the vanishing point estimation results are improved, and the average calculation time is reduced to 6.78 seconds per image.

Table 5.3 Vanishing point estimation results using 100-line segments detected in an image.

Dataset	*Line detection*		*Vanishing point estimation (average)*				*Focal length error (average)*	*Time*
	Method	*# of line segments*	*Method*	*Frame error*	*Vertical error*	*Horizontal error*		*Running time per image*
York Urban dataset	*LSD*	*100*	*RANSAC based*	*1.3969°*	*2.0011°*	*1.0093°*	*8.5%*	*6.78s*

Note that the vanishing point estimation results are improved, and their respective error ranges are identified. Thus, the effect of the proposed rotation compensation variable can be evaluated while considering the average accuracy of the estimated vanishing points. To perform this evaluation, small portions of a video recorded at the longest corridors of Ross building are considered. The camera starts recording at a position observing the long corridor from the mid part of it while several corridor junction points are visible. The camera swings somehow sharply to the left and right while it moves towards the end part of the corridor. Table 5.4 reveals the performance of the Layout SLAM method in tracking normal point features with and without applying the rotation compensation variable.

Table 5.4 Effect of applying rotation compensation variable in Layout SLAM architecture.

Dataset	*Rotation compensation variable*	*The average times point features successfully measured compare to times predicted in the image space (%)*					*Video*
		Every frame	*Every 3 frames*	*Every 5 frames*	*Every 7 frames*	*Every 9 frames*	*Total # of features*
York dataset (Ross building video, corridors 4&5)	*Not applied*	*86.32%*	*81.76%*	*73.25%*	*69.31%*	*56.14%*	*112*
	Applied	*93.52%*	*90.80%*	*87.50%*	*85.58%*	*79.62%*	*112*

As it can be interpreted from the above table, the incorporation of the rotation compensation variable into the layout SLAM architecture improves its performance while dealing with rapid movements of camera. Here, small covariance was adopted for the noise vector in Layout SLAM motion model which indicates small accelerations and smooth motions in the system. Yet, the rotation compensation variable tries to suppress the need for perfect measurements at each time step even though the system assumed to have smooth motions all the time. Note that the performance of this newly integrated variable will be influenced by the performance of both the adopted line segments extraction and vanishing point estimation methods.

5.6.2 Comparisons of Camera Trajectories

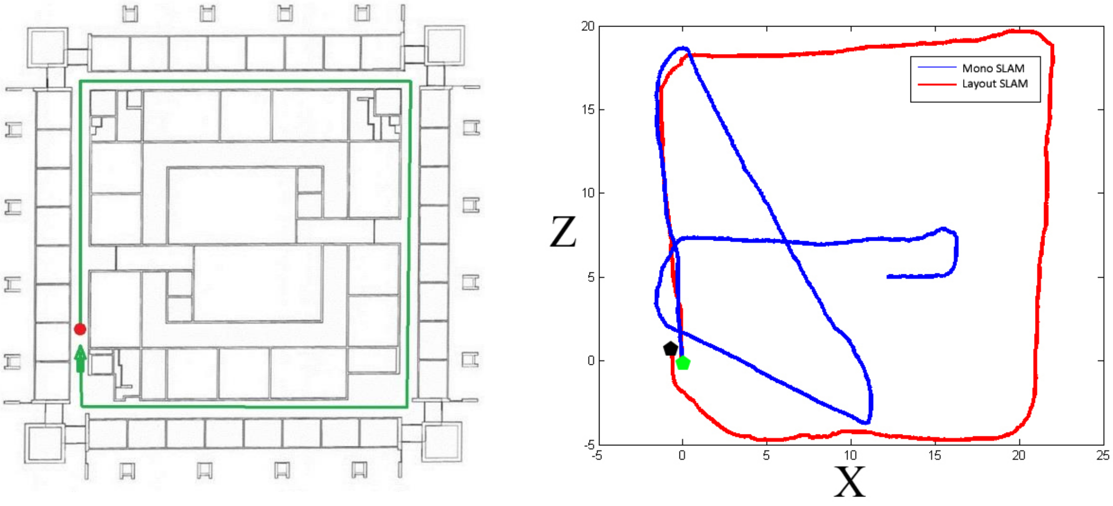
As mentioned before, the performance of the proposed Layout SLAM algorithm is tested on different datasets including the RAWSEEDS and York University datasets. Here, three different video sequences were used to examine the validity of the proposed Layout SLAM method for camera trajectory estimation. The descriptions about these videos are provided earlier in this section and Table 5.1. In the first video, consisting of 9245 frames, the handheld camera moved for about 159 meters and completed a relatively large loop of 4 connected corridors (Petrie building first floor). The second video, consisting of 3615 frames, one loop of 6 connected corridors was covered (Ross building first floor about 112 meters). The third video is from *Biccoca_2009-02-25b* of RAWSEEDS dataset, consisting of total 52695 frames, covering several loops.

5.6.2.1 Qualitative Comparison

Here, relatively challenging video sequences are considered, and camera trajectories estimated by Layout SLAM algorithm are compared to camera trajectories estimated by Mono SLAM algorithm (Civera et al., 2010). Here Mono SLAM algorithm is chosen because Layout SLAM architecture is reconstructed by altering some Mono SLAM core blocks and it is wise to compare its performance to the original work. Note that the comparison of the proposed Layout SLAM algorithm to recently proposed methods is not revealed in this thesis due to some technical difficulties and it will be revealed in future works. In this section the estimated camera trajectories are presented while no loop closing technique is applied. Figure 5.6 shows the camera trajectories estimated by both Layout SLAM and Mono SLAM algorithms at Petrie Science building first floor. The camera's trajectory starting and ending points are depicted with green and black circles respectively.

The validity of the resultant trajectories is undoubtedly clear from visual inspection. The demonstrated camera trajectory results depict that the proposed Layout SLAM algorithm outperforms the original Mono SLAM algorithm. Mono SLAM could not correctly estimate the camera orientations at the end part of the corridors while the camera is turning into the new corridor environment. This problem is mainly formed by the fact that the camera is getting very close to the corridor side wall while turning into the new corridor and the number of visible known features reduces a lot. This problem causes the Mono SLAM not to be able to estimate the correct amount of camera orientations while registering new features to the system. Yet, Layout SLAM can handle this situation very well. This is mostly due to the incorporation of vanishing point estimation results into the

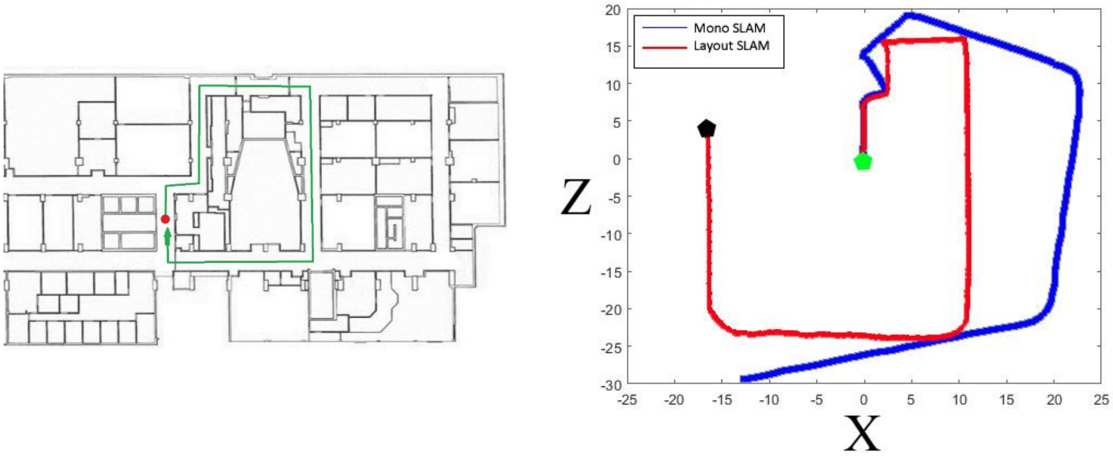
Layout SLAM algorithm. This incorporation prevents the wrong estimation of the camera orientations during the run and reduces angular drifts a lot.



(a) (b)

Figure 5.6 (a) Schematic view of camera's path on the first floor of Petrie Science Building and (b) Estimated camera trajectories by Layout SLAM (red) and Mono SLAM (blue).

Figure 5.7 also depicts the camera trajectories estimated by both Layout SLAM and Mono SLAM algorithms at one section of Ross building first floor. The camera's trajectory starting and ending points are depicted with green and black circles respectively. Here, the camera trajectory results depict the same quality of Layout SLAM algorithm in estimating the correct camera orientations at the turning points. Note that layout SLAM has encountered with scaling issues in this run which is mostly caused by the long length of the 5th and 6th corridors and low number of features on these corridors.



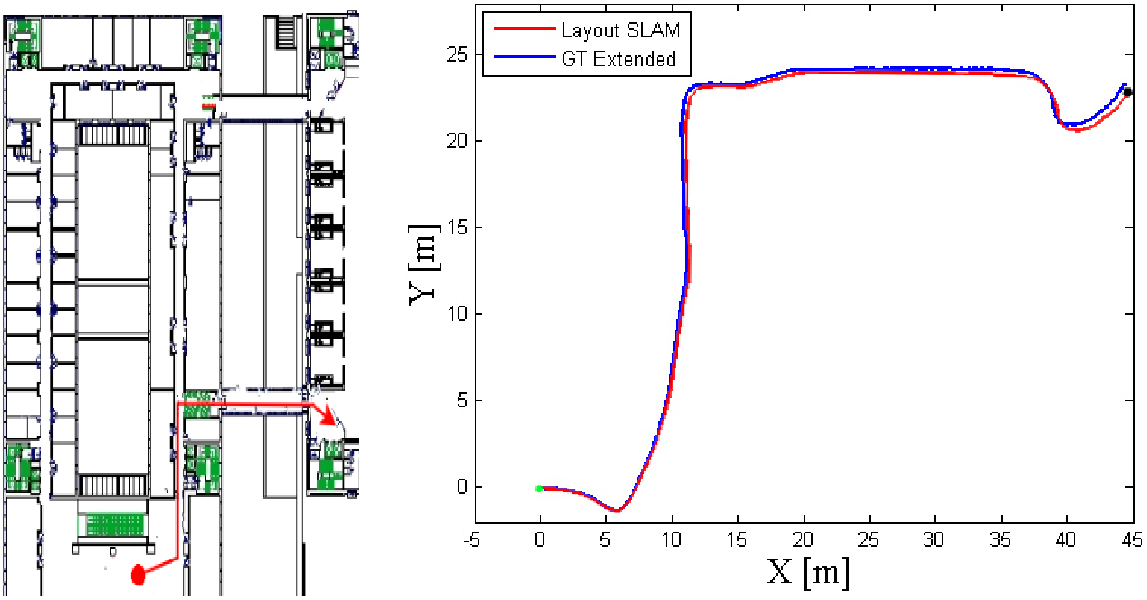
(a) (b)

Figure 5.7 (a) Schematic view of camera's path on the first floor of Ross Building and (b) Estimated camera trajectories by Layout SLAM (red) and Mono SLAM (blue).

Note that Cumulative Distribution Function (CDF) of the trajectory error could be used for the above comparisons. CDF is normally used for measuring the precision of a system. Yet, it can be seen in figures 5.6 and 5.7 the proposed Layout SLAM system performance is superior to Mono SLAM and CDF trajectory error calculation seems unnecessary. Note that CDF trajectory error calculation would be considered in future comparisons of the proposed Layout SLAM algorithm to recently developed methods.

5.6.2.2 Quantitative Comparison

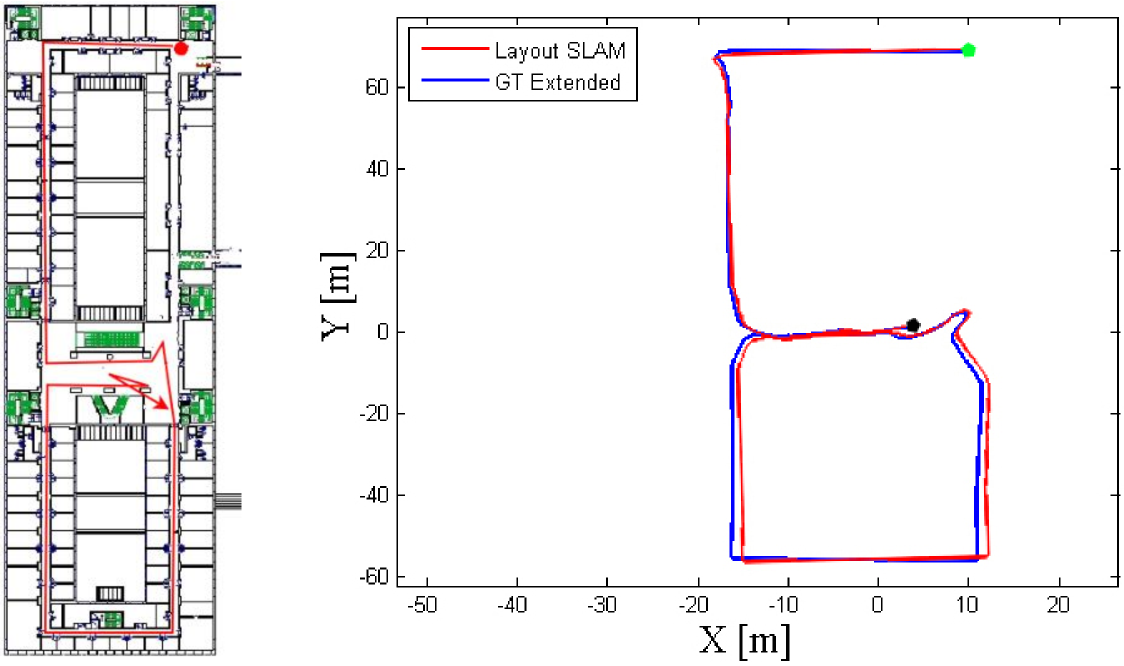
As mentioned before, the performance of Layout SLAM algorithm is tested on the dataset of *Biccoca_2009-02-25b* from RAWSEEDS as well. Although this dataset consists of various architectural features, only its indoor corridor environments are examined for experiments. The library scenes and part of connecting glass wall areas are excluded in experiments due to their non-corridor type structure and significantly challenging lighting conditions. Hence, the whole dataset is divided into two parts (Figures 5.8(a) and 5.9(a)) which are mainly covering indoor corridor scenes. To evaluate the Layout SLAM algorithm, the incoming trajectory results are aligned with the provided ground truth data of RAWSEEDS (Figures 5.8(b) and 5.9(b)). Note that the camera's trajectory starting and ending points are depicted with green and black circles respectively. The chosen scenes are quite challenging for having sharp turns in narrow corridors and including various featureless walls. It should be noted that the camera orientation errors increase at turns around featureless corners where the vanishing points cannot be estimated routinely. These errors will be accumulated on the run and lead to a position error at the end.



(a) (b)

Figure 5.8 (a) Schematic view of camera's path on the first part of *Biccoca_2009-02-25b* data and (b) Camera trajectories by Layout SLAM (red) and ground truth (blue).

As it can be seen in figures 5.8(b) and 5.9(b), the accumulated drift error is noticeable while plotted with the ground truth trajectory. Yet, the amount of position error is not large and the relative error on the run is less than 1% of the trajectory length.



(a) (b)

Figure 5.9 (a) Schematic view of camera’s path on the second part of *Biccoca_2009-02-25b* data and (b) Camera trajectories by Layout SLAM (red) and ground truth (blue).

As it can be seen in figure 5.9(b), there is an obvious position drift after the camera make a sharp turn towards the southern section of the building because of scene complicity and lacking enough features to support. Since Layout SLAM method incorporates the orientation information of the estimated vanishing points as a correction to the estimates, the orientation errors will be reduced and not accumulated yet bounded during the run. Table 5.5 provides the mean and maximum errors obtained in the above experiments. This table reveals the Layout SLAM performance with the maximum absolute trajectory errors of 2.4m in position and 8.2° degree in orientation for approximately 318m path.

Table 5.5 Layout SLAM absolute trajectory errors compared to ground truth data.

Dataset	*Approximate path length*	*Position error (m)*			*Orientation error (degree)*		
		Average	*Maximum*	*Maximum error over trajectory*	*Average*	*Maximum*	*Standard deviation*
Biccoca_2009-02-25b (part 1)	*75m*	*0.146*	*0.274*	*0.36%*	*0.531*	*4.255*	*0.496*
Biccoca_2009-02-25b (part 2)	*318m*	*0.873*	*2.389*	*0.75%*	*1.483*	*8.171*	*1.141*

To reduce the amount of scale drift error, the number of measured features in every frame should be increased. However, increasing the number of features will affect the computational cost of the adopted EKF framework and make it impossible to perform in real-time. Therefore, in these experiments time is not considered as a governing factor.

5.6.3 Evaluation of Estimated Layouts

Layout SLAM is designed to estimate the indoor corridor layout along with the camera trajectory in Manhattan frame structure. Figure 5.10 depicts the estimated layout in 3D space for Petrie Science building first corridor. In the previous section the camera trajectory comparisons are presented. Here, the estimated layouts in the image space should be validated as well. Hence, the accuracies of the estimated indoor layouts in the image space are on focus for the evaluation of the proposed Layout SLAM method. The ground truth layouts in the image space were manually created for a sparse set of video frames (only 12 keyframes) of Petrie Science building first floor.

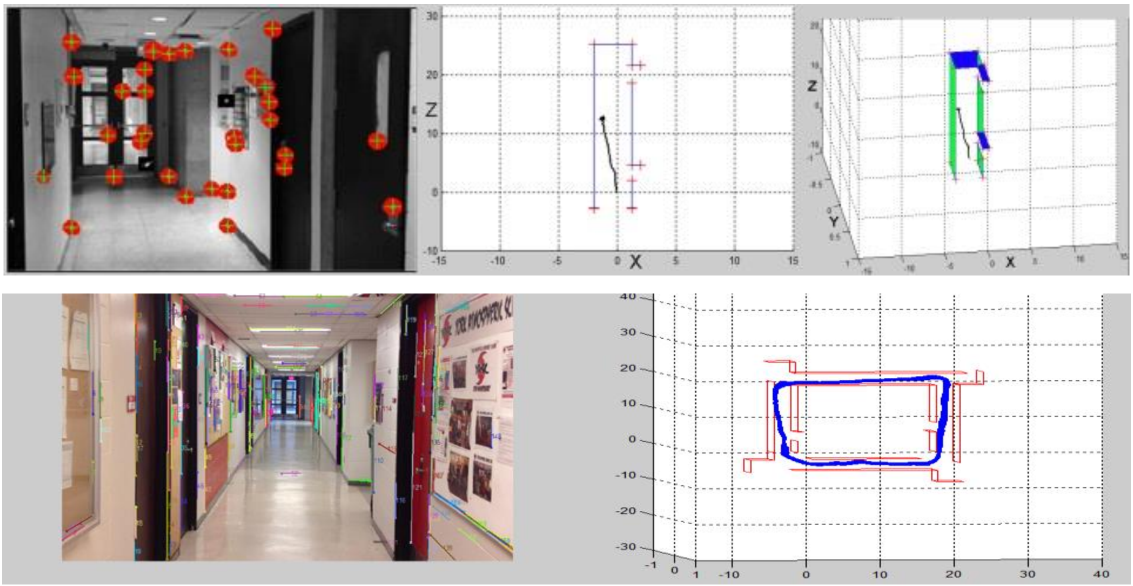


Figure 5.10 Top from left to right: a sample video frame and identified normal point features, camera trajectory and estimated layout in 3D space (top view) and current estimated layout in 3D. Bottom left: starting frame with detected straight-line segments. Bottom right: the estimated corridor layouts in 3D for Petrie science building dataset.

To perform the accuracy assessment, the RMSE of the estimated layout structural corner points in the image space is considered. Table 5.6 reveals the quantitative assessment of the estimated layout structural corner points. The Layout SLAM experiments with respect to manually digitized corridor layouts in image space reveals that the average difference in x and y directions on layout structural corner points of first corridor are -2.37 and 1.98 pixels, with RMSE of ±1.83 and ±1.51 respectively. Also, the result for the complete loop shows average differences in x and y directions are -4.91 and 3.42 pixels, with RMSE of ±1.97 and ±1.84 respectively.

Table 5.6 Quantitative assessment of estimated layout structural corner points (unit: pixel).

Dataset	*Approximate path length*	*Position error (x)*		*Position error (y)*	
		Average	*RMSE*	*Average*	*RMSE*
Petrie Science Building (first corridor)	*38m*	*-2.37*	*±1.83*	*1.98*	*±1.51*
Petrie Science Building (first loop)	*159m*	*-4.91*	*±1.97*	*3.42*	*±1.84*

As expressed in this chapter, the proposed cost function for finding the best fitting layout hypothesis is playing an important role in structural corner points matching. Hence, the acceptance threshold of the proposed matching cost function should be chosen carefully. Note that if the minimum cost is larger than this threshold, the layout hypothesis under question is not considered as a true match. To evaluate the impact of this cost function threshold, the overall matching performance is tested by assigning different values to this threshold. Once more the RMSE of the estimated structural corner points was measured in a sample subset of Petrie Science building video. Table 5.7 reveals the results of this experiment. As the minimum cost function threshold becomes larger, the overall number of matched layout corner points increases. Yet, the calculated RMSE shows that the overall cost function accuracy deteriorates as the threshold becomes larger. Note that the accuracy of the manually digitized layouts has an impact on these values as well.

However, the impact of the accuracies of both vanishing point estimation algorithm and the prepared ground truth data are neglected in the experiments. Moreover, if small values are assigned to the proposed cost function threshold the number of matched layout structural corner points would be too small to recover a physically plausible layout structure in the image space. Therefore, the optimal value for the cost function threshold which provide better results in matching a layout hypothesis to the image is 0.7 (table 5.7).

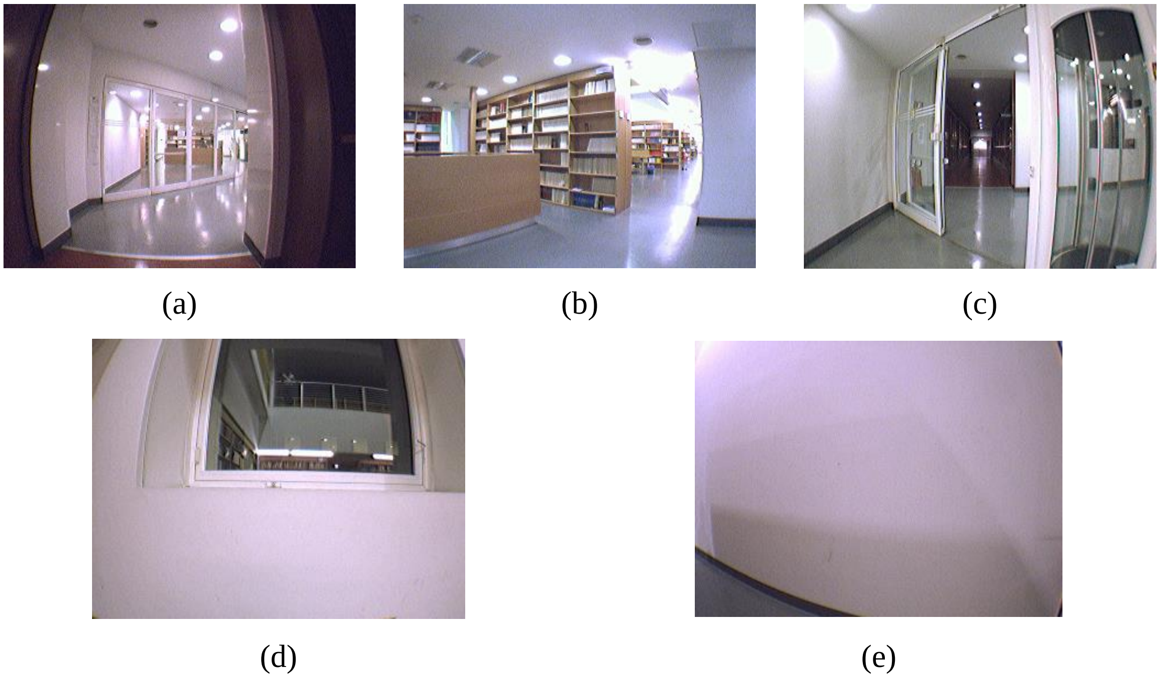
Table 5.7 Quantitative assessment of layout matching cost function threshold (unit: pixel).

Dataset	*Matching cost function threshold*	*Total number of matched corners*	*RMSE*	
			x	*y*
Petrie Science Building (first corridor)	*0.5*	*24*	*±2.08*	*±2.84*
	0.6	*32*	*±2.31*	*±2.12*
	0.7	*30*	*±1.83*	*±1.51*
	0.8	*33*	*±2.49*	*±2.01*
	0.9	*38*	*±2.90*	*±2.48*

5.6.4 Failure Cases

As discussed previously, the performance of Layout SLAM is tested on *Biccoca_2009-02-25b* from RAWSEEDS dataset. Layout SLAM is encountered with difficulties to correctly estimate the orthogonal vanishing points, generate the indoor layout and estimate the camera pose in the library section of RAWSEEDS dataset. These problems are directly related to the characteristics of the selected dataset. In the library scenes, there is a quite large space between the ceiling and side walls with surrounding bookshelves. This formation is not often suitable to be represented by a single cubic structure. Hence, Layout

SLAM is failed to reconstruct the indoor layouts in parts of the library scenes. In some occasions the camera approaches a very narrow corner and turns into a new space. In these cases, the image is often showing one or two plane walls which makes it impossible to extract enough line segments and estimate orthogonal vanishing points. Moreover, not enough point features can be measured in these types of scenes which makes the camera pose estimation inaccurate. Another problem with layout and camera poses estimation caused when the camera moves toward a window or a glass wall. In these cases, there are plenty of normal features visible through the glasses which will be detected in the scene yet residing outside of the indoor layout boundaries. Figure 5.11 shows some of the problematic video frames in the experiments.


(a) (b) (c)

(d) (e)

Figure 5.11 Problematic video frames of RAWSEEDS dataset: (a, b, c) hard to detect indoor layout, (d, e) having not enough good quality point features.

5.6.5 Limitations

Overall the proposed Layout SLAM algorithm could generate acceptable geometric and topologic results regarding both the estimation of indoor corridor layouts and camera poses. However, this method can only be applied to indoor corridor environments where Manhattan structure remains intact. If Layout SLAM method is applied to non-Manhattan frame environments where the orthogonal vanishing points cannot be estimated in images, then the algorithm retreats to its root and performs like Mono SLAM. Also, the proposed Layout SLAM algorithm would encounter with layout scale issues in some cases if the count of measured features is very low at texture-less corridors intersection.

It should be noted that Layout SLAM back-end is founded on EKF framework. Therefore, the computational time for the algorithm increases if the number of video frames and features increase as well. In such cases the P matrix becomes very large at every state, and consequently increases the computational time. One solution would be truncating the whole path into several smaller size paths and perform the experiments individually. Yet, the resultant scale issues must be addressed while stitching the small size paths together. Note that in the previous experiments this technique is not applied.

5.7 Summary

In this chapter, we proposed a new visual SLAM method which can estimate the indoor corridors layouts along with camera poses at real time. The proposed Layout SLAM method extract different type of point features in video frames and estimate their respective 3D coordinates in an arbitrary coordinate system. The estimated 3D point features together form a structure which can be modeled as an indoor corridor layout. Note that the proposed Layout SLAM system is using the Extended Kalman Filtering (EKF) for inference. EKF provides the opportunity to perform real time camera localization at every instance. This method also utilizes the orthogonal vanishing point estimation results to improve the camera orientation estimation on the run.

To find the best fitting layout to the scene, plenty of layout hypotheses are introduced to the system at every instance. Hence, a new cost function is designed to improve the feature matching scheme in Layout SLAM system. This cost function focuses on image based contextual information to improve the matching performance of the proposed method at indoor places. The experimental results reveal that the proposed Layout SLAM algorithm can successfully build an online sparse map of structural layout point features. Moreover, the proposed system is robust against orientation error accumulations via incorporation of a rotation compensation variable into the Layout SLAM architecture.

Chapter 6

Model-Based Loop Closure for Layout SLAM

In this chapter, a recently developed model-based loop closing technique is introduced. The proposed model-based method aims to accurately associate newly visited scene parts to the previously visited layouts. This loop closing technique refines the quality of layouts estimated by the existing Layout SLAM algorithm explained in Chapter 5. Layout SLAM benefits from this novel technique of loop closing by matching layout models of various keyframes. Both image information (photometric features) and layout information (topology and geometry of reconstructed layout models) are utilized to address a loop-closure detection. The novelty of using the layout-related information in the proposed loop closing technique provides two advantages. First, it imposes a geometric constraint on the global layout model consistency and, thus, adjusts the mapping scale drifts. Second, matching ambiguity will be reduced in the context of indoor corridors. The second advantage specially reveals when encountering with homogenously textured corridors, where extracting enough conspicuous point features is fairly a challenging task. The proposed model-based loop closing method is designed to compensate the limitations of the existing loop closing algorithms. Hence, it reduces three types of modeling errors including orientation error, boundary displacement and shape deformation that are often involved in estimation of indoor corridor layout models. To evaluate the proposed loop closing technique, the experiments were performed on wide-angle videos acquired by a handheld camera (introduced dataset in chapter 3). The achieved results depict that the proposed algorithm can successfully improve the estimates of Layout SLAM algorithm by detecting loop closing instances while incorporating very limited number of features.

6.1 Introduction

Layout SLAM is aimed at mapping the observed corridor environment progressively, localizing the camera with respect to indoor layout, and detecting loop-closures to avoid error accumulation. In general, SLAM has objectives like two other approaches namely visual odometry and optical/scene flow. Contrary to visual odometry, in SLAM the reconstructed environment map is used and updated over an extended period of time using loop closing. Moreover, contrary to the scene flow technique, the platform ego-motion is repetitively estimated in SLAM. Note that the scene flow technique is solely concerned with motions at any pixel.

The main objectives of applying a loop closing technique in SLAM include accuracy enhancement for localization and mapping results, uncertainty reduction, and suppression of locally accumulated errors in the global context. For instance, if the camera rotates largely between two consecutive video frames then a new environment would be captured with no overlap by the immediately previous observed scene parts. Hence, a new map part will be generated that must be linked to the previous parts. This “linking” task is an important element that brings consistency to the global map and allows association of new measurements to old “landmarks” which is realized through loop closure.

In general, loop-closure detection techniques are based on the principles of place recognition and can be divided into three different categories (Williams et al., 2009): i) image to image; ii) image to map; and iii) map to map. The following paragraphs shortly review some of the most common techniques of visual place recognition. Readers are referred to Lowry et al. (2016) for a comprehensive survey of visual place recognition techniques and their applications in SLAM loop closing.

Image to image (appearance based) techniques are mainly based on visual bag-of-features models (Ho et al., 2006; Cummins et al, 2008). A visual vocabulary is first built from previous key-frames (reference images). Constructing the vocabulary consists of three main procedures: extracting features and their descriptors from reference images, clustering the descriptors, and filling the vocabulary with the centroids of these clusters as visual words. Then, the features of the new image (query image) are matched against the visual words in the vocabulary and a histogram is built from the matching outcomes. The peak of the histogram determines the place correspondence. To make these techniques more robust to appearance and viewpoint changes, advanced techniques such as burstiness weighting (Sattler et al., 2016), spatial matching (Philbin et al., 2007), and convolutional neural networks (Sunderhauf et al., 2015) are proposed.

Image to map techniques perform 2D to 3D matching to identify the correspondences of the query image in the existing map (Williams et al., 2008). Loop-closure validation can also be performed through RANSAC based algorithms. As a result, these techniques deliver the relative 3D similarity transformation between two parts of the map (new part and old landmarks). To retrieve the scale, the camera is tracked for a while in both map parts (Fischler and Bolles, 1981).

Map to map techniques are actually extended versions of appearance-based techniques, where the relative geometric (spatial) distance between features is considered as additional constraints to make the matching procedure robust (Clemente et al., 2007).

Once the corresponding features are identified from two sub-maps, maps can be transformed to one another using a rigid body transformation. According to (Clemente et al., 2007), using five common features from different sub-maps is enough for closing a large loop.

When a loop closure is successfully detected and validated (at the SLAM front-end), it means that the camera has captured a part of the scene which was previously observed from a different perspective. Once this occurs, a pose-graph optimization or bundle adjustment (at the SLAM back-end) must be applied to adjust the accumulated errors of camera poses and map landmarks (Grisetti et al., 2010; Schneider et al., 2013).

In this chapter, a new loop closure detection method is proposed which relies on top-down knowledge of corridor layout, i.e., spatial decomposition of corridor face topology graph, to make a keyframe matching performance robust. This model-based loop closure detection method allows a global adjustment of indoor corridor model parameters generated by the proposed Layout SLAM method.

6.2 Side Corridor Model Detection

One of key features of Layout SLAM is the ability to generate multiple cuboid models representing not only a main corridor, but also side corridors. Side corridors intersect with the structural walls of the main corridor and provide open spaces inside their layout structure. In Layout SLAM algorithm, the presence of side corridors can be detected in the image space through comparing the geometric features of the previously estimated main corridor layout to the ones detected in the current frame. Significant amount of geometric differences between two corresponding image regions will trigger the process of side corridor model generation. The side corridor model generation process in the proposed Layout SLAM method is very much the same as the one explained in chapter 4. Yet, there is a difference in the generation of potential side corridor hypothesis. In the previous work by Baligh and Sohn, 2016, many side corridor hypotheses were generated using the extracted straight-line segments in the image space. These generated hypotheses will be examined later for finding the best fitting hypothesis to the estimated layout structure.

Here, there would be no hypotheses pooling and instead the side corridor layout will be generated directly by considering the appearance cues of current video frame. With given video frame, two sets of appearance cues must be extracted. First, geometric cues indicate orientation context of planar surfaces estimated for the previous main corridor layout (previous video frame). Second, geometric cues measure the same orientation context, but using straight lines extracted from the given video frame. If a combinatory integration of these two geometric cues from successive video frames indicates the excessive presence beyond coverage of one corridor, a layout model generation process which creates a secondary corridor is initiated. Note that this process can continue to create multiple side corridors until a certain termination condition is met.

The original method for side corridor creation in an image space is introduced by Baligh and Sohn, (2015). Here, the two generated orientation contexts will be overlaid to identify the regions with orientation conflict. The amount of orientation confliction is counted in pixels. Yet, the number of counted pixels should be more than a predefined threshold to trigger the side corridor creation process. Here, this threshold is chosen intuitively. A region with orientation confliction should have proper formation and resides inside a wall of the estimated main corridor layout. The new side corridor layout that fully covers this region will be created by intersection of vanishing points virtual lines with the main corridor layout borders. Note that the created side corridor layout would satisfy the volumetric maximization and the orthogonality of the estimated layout structure. Figure 6.1 shows an example of side corridor detection in a given frame.

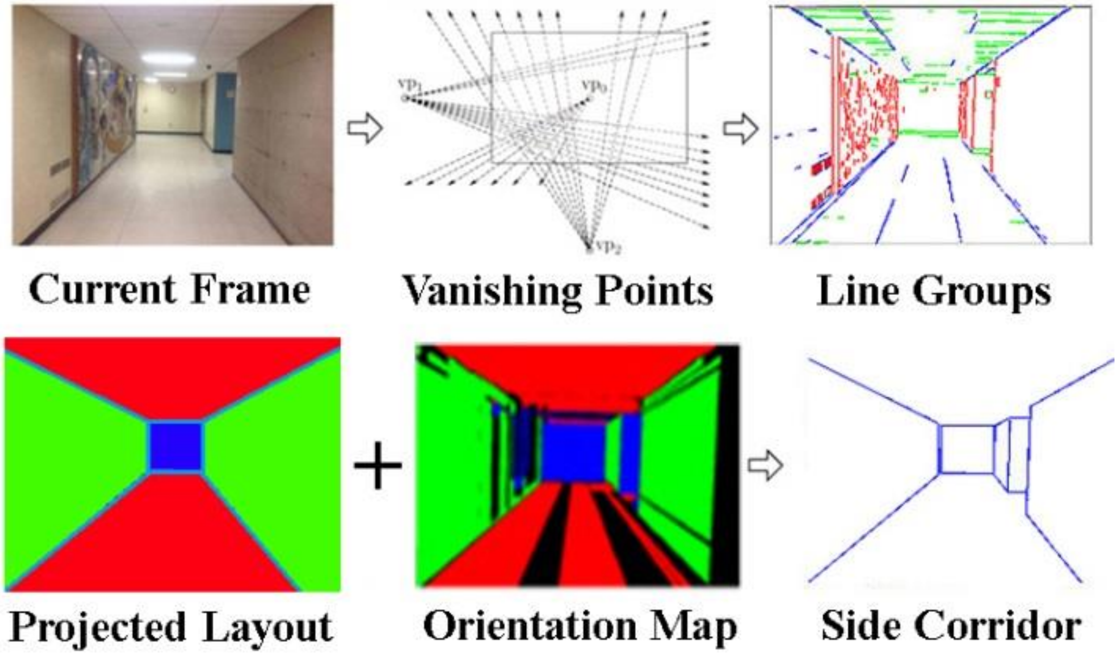


Figure 6.1 The orientation context of the projected scene layout (previous video frame) compared to the current frame orientation context to trigger side corridor generation.

The success of the side corridor generation method is highly dependent on the detection of orthogonal vanishing points which contribute to creation of the side corridor layouts. Therefore, considering the Manhattan rule assumption in the image space will play a great role in the success of this method. The applied method intends to simplify the scene layout by considering it to be formed of integrated cubical structures. Hence, this method only intends to form key structural planes in the image space and identify a cubical structure in right or left sides of the major corridor layout by intersecting orthogonal lines originated from vanishing points.

6.3 Model-Based Loop Closing

Loop closure detection is one of the main features of any SLAM system which makes it distinctive of the other similar systems such as visual odometry. Loop closure detection in visual SLAM systems is a big challenge especially in robotics applications, since camera is the only sensor in these systems. The classical loop closure problem can be defined as recognising when the SLAM system has visited a previously mapped environment. In such cases, two parts of the map are found to belong to the same environment. However, these two map parts may have incompatible position and orientation even by considering the map uncertainty estimate. Therefore, the SLAM system must apply the appropriate transformation which is required to align these two map parts and allegedly close the loop.

In this chapter, both model information (topology and geometry of reconstructed layout model in image space) and image information (radiometry) are used to address loop closure detection. Hence, the proposed model-based loop closure algorithm enables adjusting errors associated with indoor models generated by Layout SLAM through robustly detecting a global loop closure. The proposed loop closing method comprises of three steps: 1) selecting a keyframe which contains sub-corridors, 2) generating a corridor topological graph, spatially decomposing a keyframe with wall faces, and 3) matching paired keyframes for detecting a global loop closure. Note that measuring a degree of visibility of side corridors from a given video frame is one of the factors which influence the selection of a keyframe. In the previous section, a side corridor layout generation is discussed, while the contribution of this process to the keyframe selection will be explained in the next section.

To ease the problem of loop closing in the proposed Layout SLAM architecture, independent local maps were generated after detecting and closing each individual loop. The idea of hierarchical map creation by integration of independent local maps is proposed by Estrada et al. (2005). Since the back-end section of the proposed layout SLAM system is based on EKF, dividing the whole map of the environment into several local sub-maps provides benefits to both front-end and the back-end sections. One of the major benefits is related to EKF update processing time which increases when the number of map features increases too. The other benefit comes by limiting EKF cumulative linearization errors within the local map which happens through poor data association and leads to overconfident state estimates. The only issue arises here is the scale problem which is not observable through monocular vision. Hence, various local maps may have inconsistent scales which can be handled through a scale invariant matching scheme.

Here the main influential factor is to build accurate sub-maps after identifying and closing the loops for all corridors and then matching local sub-maps which may contain high or low localization uncertainty. To apply the method, once the camera enters a previously visited corridor and the loop closing is accomplished, the current map freezes and the next local map will be initialized. The next local map will use the last camera location as its initialization position. Here, the previous sub-map features which are currently visible in the scene should be initialized in the new sub-map through their image locations. These features will be common in adjacent sub-maps and they can provide information for integrating sub-maps. Through these common features the scale variations between adjacent sub-maps can be handled. It should be noted that for preserving the statistical independence among sub-maps, no other information will be inserted from the previous sub-map to the current sub-map.

6.3.1 Finding Keyframes

As mentioned before, the presence of a side corridor can be examined in the image space by comparing the geometrical features of both current video frame and the back projected layout from the previous video frame. As the camera moves forward in an indoor corridor scene, side corridors may appear gradually in many of the captured video frames. Side corridors are providing additional topological information to the current layout. Hence, this unique topological information along with the measured structural features can play a great role in identifying previously visited environments.

Obviously, searching all the captured video frames and pinpointing common layout features is not optimal for performing the loop closing task. Yet, identifying the optimal video frames for benchmarking the Layout SLAM trajectory would be very beneficial. Here, this optimal video frame is called the keyframe. To handle loop closing instances, we propose choosing keyframes which reduce the possibility of matching ambiguity and increase the efficiency of the structural point features detection. In other words, an optimal subset of reference video frames must be selected as keyframes which together they can approximate the whole corridor space.

Here, the selected video frames must contain as many salient structural point features as possible while having normal point features uniformly distributed in the scene as well. Thus, the problem is defined as following: given n number of video frames which side corridors are appeared in them $I = \{I_i | i = 1, 2, 3, \dots n\}$, the optimal keyframe set $F = \{I_k | k = 1, 2, \dots m\}$ must be computed that minimizes the cost function defined as $C(F, I)$. Here, the proposed cost function includes two terms: $C_c(F)$ which is modeling the completeness of the indoor corridor layout and $C_v(F)$ which is modeling the visibility of the same layout. Hence, the following equation can be defined:

$$C(F,I) = \alpha \times C_v(F) + C_c(F) \tag{6.1}$$

In the above equation, α is the weight value (α = 2). The visibility term is introduced to identify the optimum view of the side corridors in the video frames under question. The visibility term can be simply defined by comparing the number of pixels covering a side corridor P_S in the image space to the total number of pixels P_T; $C_v(F) = 1 - \frac{P_S}{P_T}$. When the number of pixels covering a side corridor goes higher, the visibility of this area would be more as well.

The completeness term is introduced to guarantee that the chosen keyframes contain the maximum number of structural point features (indoor layout corner points) and normal point features (FAST corner points) as possible. To improve the performance of the proposed Layout SLAM system, these features must appear in different video frames which lead to accurately localizing these features in 3D space. Here, the features which are matched during the data capturing procedure are grouped. The incoming feature groups can be denoted as Y, which represents a series of matched features in various frames; $Y = \{y_i | i \ \in g(Y)\}$ where $g(Y)$ represents the reference video frame set with respect to Y.

If $|g(Y)| = 0$, this means an initialized feature in one frame does not have any corresponding match in the other frames. Hence, a threshold is defined to guarantee that the selected features were appeared in at least a minimum number of video frames: $|g(Y)| \geq 35$. Considering this fact, the saliency of a feature $S(y)$ can be defined as the match count of this feature in the other video frames $|g(Y)|$ divided by the number of times the feature is predicted by EKF: $|p(y)|$; $S(y) = \frac{|g(Y)|}{|p(y)|}$. Finding insufficient matches for a feature may result to unreliable positioning of this feature in the environment.

The other factor which can be considered here is the distribution of features in the image space which affect the quality of feature real time tracking in the proposed Layout SLAM method. Density of a feature $d(y_j)$ can be defined by considering each surrounding pixel x in the image j. The density of a feature can be related to its position in the image space, examined by the number of pixels which are residing in a predefined window while the feature is at the center. If the feature is fully surrounded by image pixels in the predefined window, then the value of the respective density would be one, and zero otherwise. The size of this window can be adopted by considering the size of video frames (here window size is 61×61). Hence, we can define the density of a feature set as: $d(Y) =$

$\frac{1}{n \times |g(Y)|} \sum_{j \in g(Y)}^{|g(Y)|} \sum_{i=1}^{n} d(y_{ij})$, where $d(y_{ij})$ expresses the density of a feature y_i in image j and n is the number of features in a set. Eventually, the completeness term can be defined as:

$$C_c(F) = 1 - \left(\frac{\sum_{Y \in F} \frac{S(Y) \ + \ d(Y)}{2 + \gamma}}{\sum_{Y \in I} \frac{S(Y) \ + \ d(Y)}{2}} \right) \tag{6.2}$$

Here γ controls the sensitivity to feature saliency and density. Also F and I denote the keyframe set and the video frame set, respectively. The exact solution to the selection of keyframes would be an exhaustive search of all possible subsets of I in the reference video frames considering the above equation. However, in the case of Layout SLAM this approach would be computationally expensive. It should be noted that a constraint can be applied here, which bounds the maximum number of keyframes in a set. The maximum number is equal to the number of detected side corridors in the whole scene. For the selection of keyframe set, the procedure starts with an empty set and then the frames will be added progressively. At each step, a new keyframe will be added to the set if it produces the less cost for the system, and consequently it will be added to the keyframe set. The process stops when the incoming cost cannot be reduced any longer. Following this scenario, the complexity of the computations will be reduced to some extent. Considering the incoming results, keyframe based feature matching is possible which is essential for Layout SLAM loop closure algorithm.

6.3.2 Keyframes Matching

In the previous sections, the generation of side corridors and selection of keyframes were presented. These two tasks can play a great role in the proposed loop closing algorithm. Hence, the best solution would be matching a test frame to the set of available keyframes for examining the occurrence of a loop closure. It should be noted that the test frame itself is a keyframe and it would be the last keyframe created on the run. To examine the possibility of matching an individual test frame to any of the previously created keyframes, some specific definitive terms must be introduced first.

Considering the indoor corridor environments, a model M can be denoted as a set of corridors $\mathrm{M} = \{\mathrm{C_i} | \mathrm{i} = 1, 2, \ldots \mathrm{n}\}$ with n number of corridors. Each corridor consists of m numbers of faces $\mathrm{C} = \{\mathrm{F_j} | \mathrm{j} = 1, 2, \ldots \mathrm{m}\}$ representing front, left, right, top and bottom sides of a Manhattan type cubical corridor.

It should be noted that the main corridor (major corridor) is always represented by five faces $C_{main} = \{F_{front}, F_{left}, F_{right}, F_{top}, F_{bottom}\}$ while sub-corridor (side corridor) has three faces $C_{sub}^{left} = \{F_{left}, F_{top}, F_{bottom}\}$ or $C_{sub}^{right} = \{F_{right}, F_{top}, F_{bottom}\}$. Here, left and right are determined based on the attached position of the sub-corridor to the main corridor. Figure 6.2 shows a test frame with its specified corridors in the image space, and its respective corridor topological graph for the model. It should be noted that the sub-corridor numbering always starts at the furthest position with respect to the camera. Therefore, the same sub-corridors would have similar numbering for their graph representation in the other keyframes.

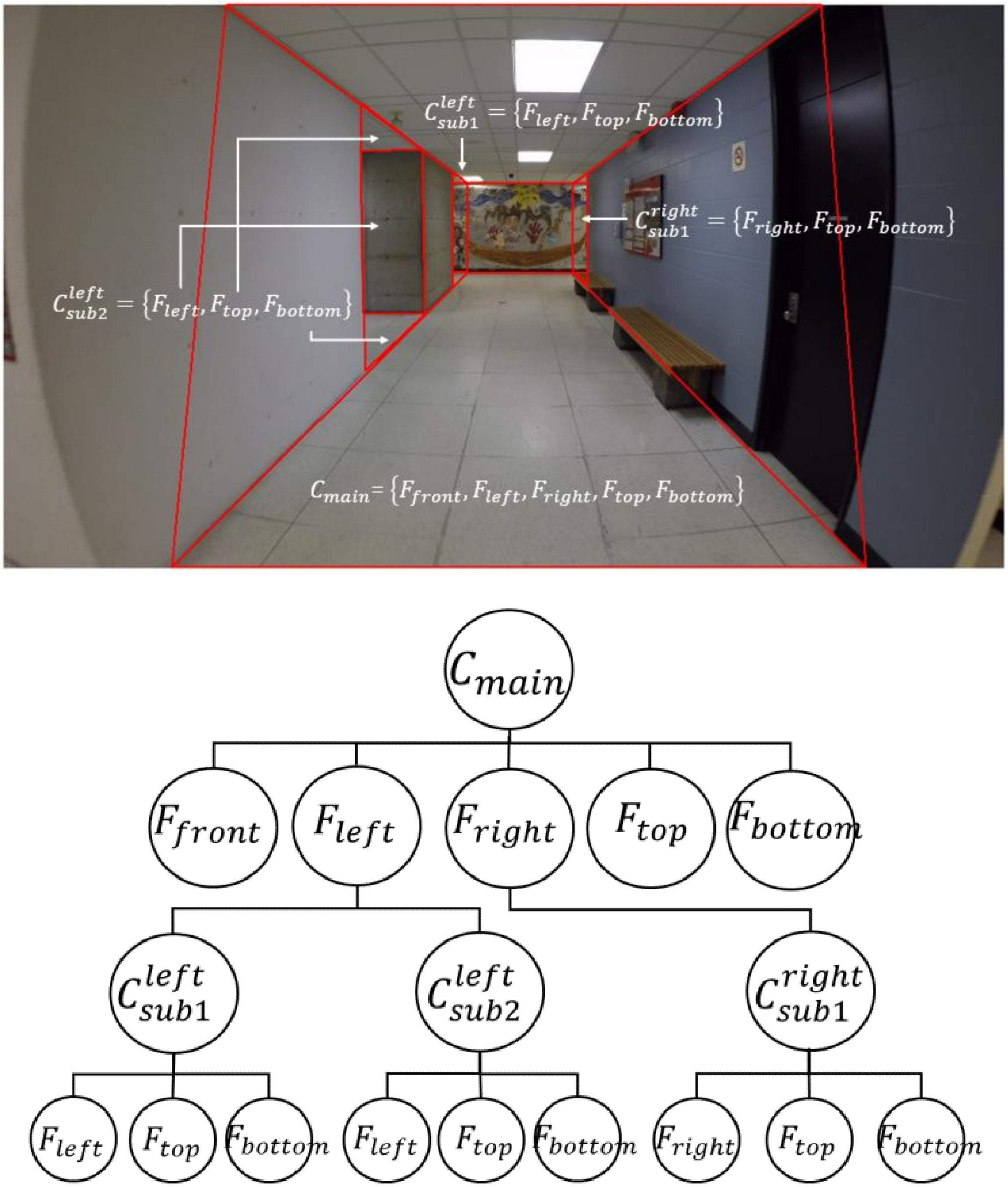

$C_{sub1}^{left} = \{F_{left}, F_{top}, F_{bottom}\}$
$C_{sub1}^{right} = \{F_{right}, F_{top}, F_{bottom}\}$
$C_{sub2}^{left} = \{F_{left}, F_{top}, F_{bottom}\}$
$C_{main} = \{F_{front}, F_{left}, F_{right}, F_{top}, F_{bottom}\}$
C_{main}
F_{front}
F_{left}
F_{right}
F_{top}
F_{bottom}
C_{sub1}^{left}
C_{sub2}^{left}
C_{sub1}^{right}
F_{left}
F_{top}
F_{bottom}
F_{left}
F_{top}
F_{bottom}
F_{right}
F_{top}
F_{bottom}

Figure 6.2 Top: The image of a test frame with all its specified corridors; Bottom: The respective corridor topological graph of the same test frame.

To examine the possibility of having a match between a test frame and the selected keyframe, the first step is to geometrically transform the test frame into the keyframe in the image space. Here, the 6 parameters affine transformation is applied as following:

$$X = a_0 + \ a_1 x + a_2 y \tag{6.3}$$

$$Y = b_0 + \ b_1 x + b_2 y \tag{6.4}$$

In the above equations, X and Y represent the image coordinates of the indoor corridor layout specified vertices on the key frame while x and y represent the same layout vertices coordinates on the test frame. Also, a_0, a_1, a_2, b_0, b_1 and b_2 are the affine transformation parameters. These parameters are calculated using the least square method. To identify the corresponding vertices between the test frame and the keyframe, we first compare two corridor topological graphs derived from those models. If faces of one corridor topological graph match ones of the other graph, the vertices belonging to the faces are considered as corresponding vertices. For example, C_{sub1}^{left} of the test frame is always corresponds to C_{sub1}^{left} of the keyframe and not to $C_{sub2,..,n}^{left}$. Therefore, the corresponding vertices are used to estimate the affine transformation parameters using the least square method.

After transforming the test frame indoor corridor layout into the selected keyframe through the affine transformation, a newly designed scoring function is used for finding the optimal match. Here, the proposed scoring function includes three terms which are measuring the resemblance of the two indoor corridor layouts by considering their topology, geometry, and radiometric similarities. The proposed scoring function is as following:

$$Score = (w_T \times S_T) + (w_G \times S_G) + (w_R \times S_R) \tag{6.5}$$

where $\mathrm{S_T}$, $\mathrm{S_G}$, and $\mathrm{S_R}$ represent topological similarity, geometry similarity and radiometric similarity, respectively. $\mathrm{w_T}$, $\mathrm{w_G}$, and $\mathrm{w_R}$ are weight parameters for $\mathrm{S_T}$, $\mathrm{S_G}$ and $\mathrm{S_R}$ respectively. These weight parameters are considered as equal in the experiments $\left(w_T = w_G = w_R = {}^{1}\!/_{3}\right)$. Based on the generated topological graphs, the topological similarity $S_T(t,k)$ is calculated by comparing the number of common faces $F_t \cap F_k$ between a test frame and a keyframe as follows:

$$S_T(t,k) = \frac{num(F_t \cap F_k)}{num(F_t)} \tag{6.6}$$

The geometric similarity $S_G(t,k)$ is calculated by measuring distances between the corresponding vertices belonging to common faces. If the measured distance d_{tk} between two corresponding vertices $V_t \cap V_k$ is less than a predefined threshold (T_1=100 pixel in experiments), indicator function δ_G for the geometric similarity is one, and zero otherwise.

$$S_G(t,k) = \frac{\sum_{V_t \cap V_k} \delta_G}{\sum_{V_t \cap V_k} 1} \ , \qquad \delta_G = \begin{cases} 1 & if \ \ d_{tk} \le T_1 \\ 0 & if \ \ d_{tk} > T_1 \end{cases} \tag{6.7}$$

The radiometric similarity $S_R(t,k)$ is calculated by comparing average colour values of corresponding faces $F_t \cap F_k$. For each individual layout face, the average values of pixels in three different bands (R, G, B) are calculated and assigned to the selected layout face. If the sum of colour differences in the three bands r_{tk} between F_t and F_k is less than a predefined threshold (T_2=50 in experiments), indicator function δ_R would be one and zero otherwise as follows:

$$S_R(t,k) = \frac{\sum_{F_t \cap F_k} \delta_R}{\sum_{F_t \cap F_k} 1} \ , \ \delta_R = \begin{cases} 1 & if \quad r_{tk} \leq T_2 \\ 0 & if \quad r_{tk} > T_2 \end{cases} \tag{6.8}$$

After scores for all keyframes are calculated, the optimal keyframe for the test frame is determined by selecting a keyframe which maximize the scoring function as following:

$$M^* = arg \max_{\forall M_k} Score(M_k) \tag{6.9}$$

If the maximum score is less than a user-defined threshold (T_3=0.9 in experiments), the test frame is considered not to be matched with keyframes. Note that in this chapter, thresholds values, weights and control parameters are chosen empirically, and how the algorithm will work in other conditions will be examined in future works.

6.3.3 Updating Layout Through Loop Closing

In the previous section, the matching of a test frame to a keyframe is explained which provides the base for associating the current measurements in the system with the previously built components of the map at an earlier time. Once the appropriate match is found, loop closure would be possible. Visual SLAM systems are commonly utilizing global pose-graph optimization or bundle adjustment to perform loop closure optimizations. Globally consistent trajectory could be obtained by applying these techniques that reduce the amount of drift in visual SLAM systems. Applying pose-graph optimization method, the whole environment would be represented as a graph that consists of camera poses as nodes which are connected through edges, representing camera motion. Note that additional transformations between images would be included as further edge constraints in this graph. Inferences about accumulated drift can be drawn by observing previously visited scenes. Consequently, the amount of drift can be calculated by considering all edge constraints.

Loop closing can also be performed through bundle adjustment over camera poses and point features (Triggs et al., 1999). In this thesis, bundle adjustment technique is adopted to perform loop closure optimization. Note that bundle adjustment can consider a local group of several images, instead of performing optimization over the whole camera poses. Bundle adjustment minimizes the reprojection errors through carrying out non-linear optimization of 3D point features Y and camera poses P. In other words, bundle adjustment minimizes the distance between 3D point features back projected into the image I_j and measured points y in the image space with the current camera pose estimate. Thus, the cost function which should be minimized can be defined as:

$$e = \sum_{i,j} || P_j Y_i - y_{i,j} || \tag{6.10}$$

Here, indexes j and i are related to the camera and features respectively. The Levenberg–Marquardt algorithm (LMA) which is suitable for solving non-linear least squares problems can be used as the optimization method. This algorithm interpolates between gradient descent method and the Gauss–Newton algorithm (GNA). Thus, it is more robust than the original GNA and in various cases can achieve global solution. Here, 6 parameters $(X_c, Y_c, Z_c, \omega, \varphi, k)$ per camera and 3 parameters (X_p, Y_p, Z_p) per point feature must be optimized. Initial values for camera pose and point features are provided through SLAM. Note that the detected loop through layout keyframe matching in previous section can impose a constraint to this optimization process. Hence, a set of point feature matches y_u and y_v with respect to camera positions P_u and P_v are available in 2D and 3D spaces respectively. The respective 3D coordinates of this set of point feature matches Y_u and Y_v are known as well. Due to the SLAM accumulated errors these 3D coordinates differ. Yet, the loop closing constraint dictates that the coordinates of these 3D points should be the same. Moreover, the reprojection of these 3D point features should be equal to y_u and y_v. Thus, Y_u needs to be back projected to both y_u and y_v. This information should be added to the bundle adjustment data structure. Having performed this task, the optimization can be accomplished in a way that both cameras poses and point features are corrected and the loop is closed consequently.

6.4 Experimental Results

To examine the performance of the proposed model-based loop closing method, the video datasets of Petrie Science and Ross buildings were used. More information on these two datasets is presented at previous chapter (Table 5.1). Here, these two datasets are chosen over the other available benchmarks because they are associated with ground truth indoor corridor models (introduced in chapter 3). This provides a unique opportunity to evaluate indoor layouts estimated by Layout SLAM.

6.4.1 Performance of Keyframe Selection Scheme

Earlier in this chapter, a procedure is introduced for selection of keyframes among all captured video frames to progressively benchmark Layout SLAM trajectories. The proposed cost function in this procedure (equation 6.1) examines the completeness and visibility of features where the side corridors are appeared in given video frames. These two factors together play an important role for identifying optimum keyframes. Table 6.1 reveals the average percentages of completeness and visibility of all features which are associated with the identified keyframes.

It can be seen in this table that the average visibility is not 100% even though the side corridors are fully appeared in the identified keyframes. This is due to that fact that when the visibility of a side corridor layout is at its maximum pick, some of its structural features are residing at the image borders. Thus, the density count of these features would be zero in the identified keyframe. Hence, reducing the amount of visibility to some extent will increase the count of density for a set of structural features. Note that in the proposed cost function equation (6.1), the weight parameter α intends to balance this relation. Yet, in experiments the value of this weight parameter is chosen empirically (α = 2).

Table 6.1 Percentage of completeness and visibility of features in identified keyframes.

Dataset	*Structural Features Visibility (average)*	*Structural Features Completeness (average)*	*Video Frames*	
			# of Keyframes	*Total # of Frames*
York University Petrie Science Building, one loop	*86%*	*83%*	*10*	*9,245*
York University Ross Building, one loop	*91%*	*94%*	*9*	*3,627*
York University Ross Building, two loops	*89%*	*78%*	*21*	*8,553*

As it can be seen in the above table, the number of keyframes compare to the total number of video frames is very small. Thus, the full Layout SLAM trajectory cannot be represented by this small number of keyframes. However, these keyframes together cover all changes of indoor layout on the run with respect to the initialized corridor layout. The proposed model-based loop closing technique can successfully perform regardless of having small number of identified keyframes. Thus, keyframes sparsity not only does not affect the loop closing scheme but also makes keyframe matching fast and robust.

6.4.2 Performance of Keyframe Matching Scheme

To detect loop closure instances through the proposed model-based method, a test frame (the latest keyframe) should be examined whether it matches any keyframe from the available keyframe set. If a test frame and a keyframe under the question match with high score, then the occurrence of a loop closure is confirmed. As mentioned previously, topology and geometry of the reconstructed layout models and radiometry of the original video frames are considered to address the occurrence of a matching instance. Note that the proposed method uses basic mathematics in detecting loop closure instances. Thus, it works swiftly while it performs keyframe matching scheme.

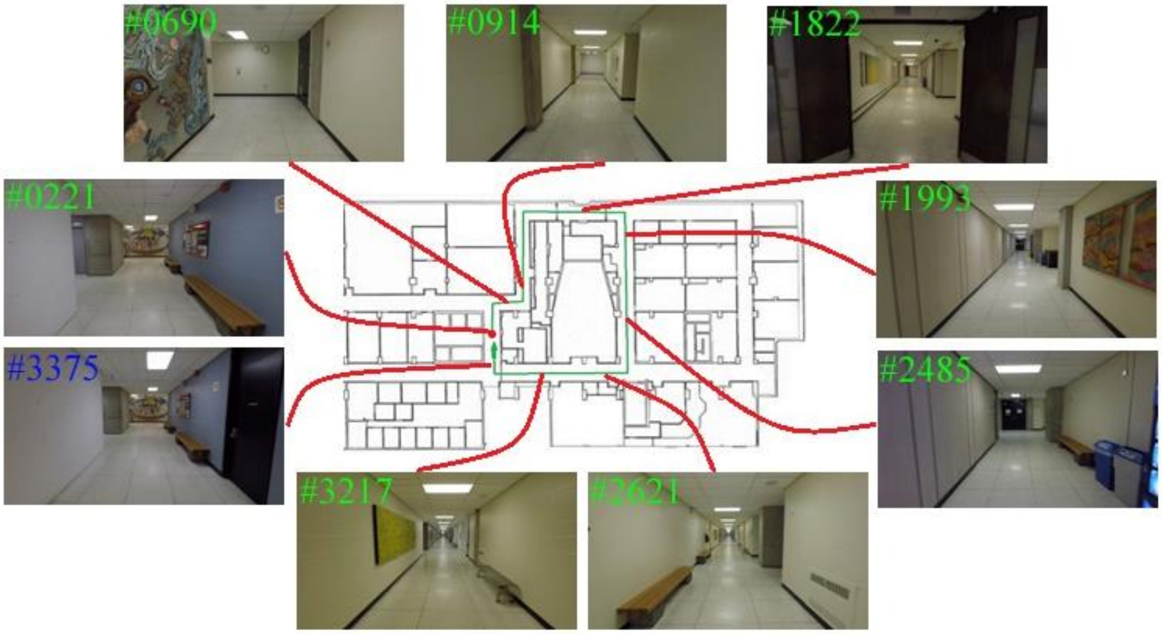
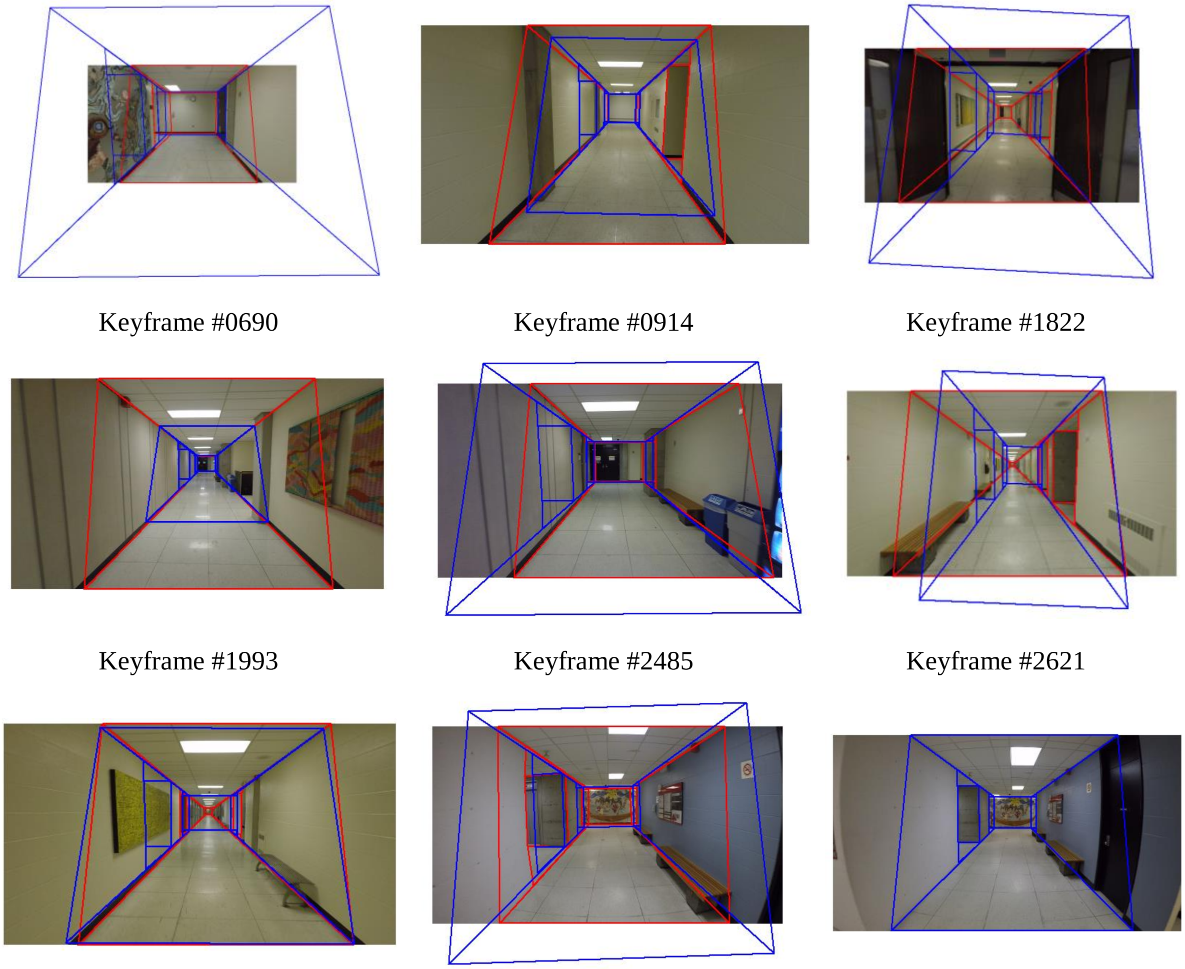


Figure 6.3 Schematic view of camera trajectory (Ross building data, first loop) accompanied with selected keyframes and a test frame in blue.

Figure 6.3 shows the camera path in Ross building, one loop dataset along with the selected keyframes. Since this dataset has the minimum number of identified keyframes, it is suitable for depicting the keyframe matching results. The first keyframe is chosen when the camera start observing the first corridor and the rest is added while it is crawling the other six corridors (total 8 keyframes and 1 test frame).



Keyframe #0690

Keyframe #0914

Keyframe #1822

Keyframe #1993

Keyframe #2485

Keyframe #2621

Keyframe #3217

Keyframe #0221

Test frame #3375

Figure 6.4 Test frame layout (in blue) is transformed into the other keyframes and compared to their layouts (in red).

The proposed keyframe matching technique applies the 2D affine transformation to project the layout of a test frame into the selected keyframe. Thus, matching individual keyframes would be more convenient to perform. Figure 6.4 shows the transformed test frame layout (frame #3375) in blue compared to different keyframe layouts in red for finding the best match.

Table 6.2 Quantitative assessment of matching a test frame (#3375) to other keyframes.

Dataset	*Keyframe*	*Test frame #3375*			
		S_G	S_T	S_R	*Total score*
York University Ross building, first loop	*#0221*	*0.941*	*1.000*	*1.000*	*0.980*
	#0690	*0.143*	*0.600*	*0.000*	*0.248*
	#0914	*0.000*	*0.500*	*1.000*	*0.500*
	#1822	*0.000*	*0.600*	*0.750*	*0.450*
	#1993	*0.769*	*0.317*	*0.409*	*0.232*
	#2485	*0.246*	*0.382*	*0.319*	*0.316*
	#2621	*0.000*	*0.833*	*0.000*	*0.278*
	#3217	*0.333*	*0.833*	*0.000*	*0.389*

Table 6.2 reveals the corresponding matching scores for this set of keyframes. As it can be seen in this table, keyframe #0221 is the best match for the test frame #3375 since achieving 98% of the matching scores. Details of the scores for topology, geometry, and radiometric similarities is given in this table as well. These individual scores together

evaluate the possibility of having a loop closure instance on the run. Note that threshold T_3 ($Score \geq T_3 = 0.9$) is applied for accepting two individual frames as correct match.

6.4.3 Corridor Models After Loop Closing

As stated at the beginning of this chapter, the prepared datasets of Ross and Petrie Science buildings were used to examine the performance of the proposed model-based loop closing method for Layout SLAM. The performance of Layout SLAM method with no loop closing is presented in chapter 5. The trajectory results were compared to the original Mono SLAM results (Civera et al., 2010) and RAWSEEDS dataset ground truth. Layout SLAM had encountered with corridor depth issues in Ross building dataset which was mostly due to the low number of features in images. Hence, in the new experiments the threshold which controls the minimum number of features in images is increased (T_f=50 in new experiments; T_f=20 in 5th chapter experiments). Note that increasing this threshold would increase the computational time for Layout SLAM. Yet, the loop closing experiments in this section were performed in off-line mode.

Figure 6.5 shows the top view of the estimated corridors by Layout SLAM performing on Ross building first loop dataset after model-based loop closing. Here, the trajectory is plotted up to the location where the last keyframe (test frame) is taken. The camera's trajectory starting and ending points are depicted with green and blue circles respectively. This figure makes the qualitative assessment possible and proves that Layout SLAM can successfully estimate correct orientation angles where the camera turns into the new corridor environment. Yet, estimation of the correct corridor length was still an issue on this dataset. Even though this problem has suppressed to some extent due to increasing the above-mentioned threshold, the ultimate solution is to close the loop.

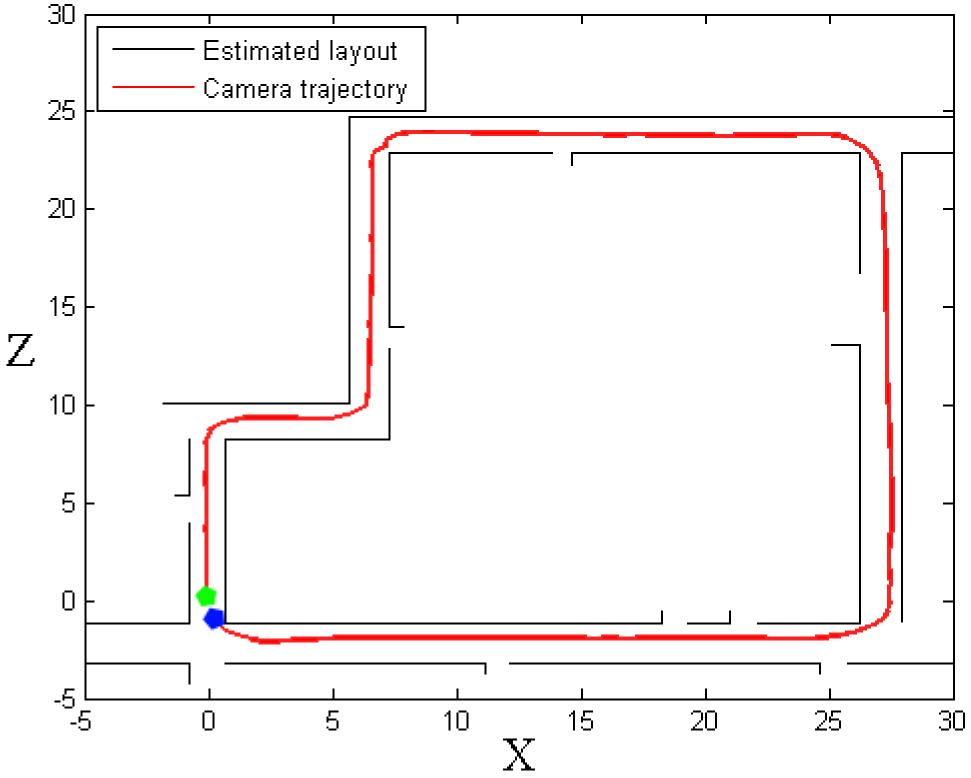


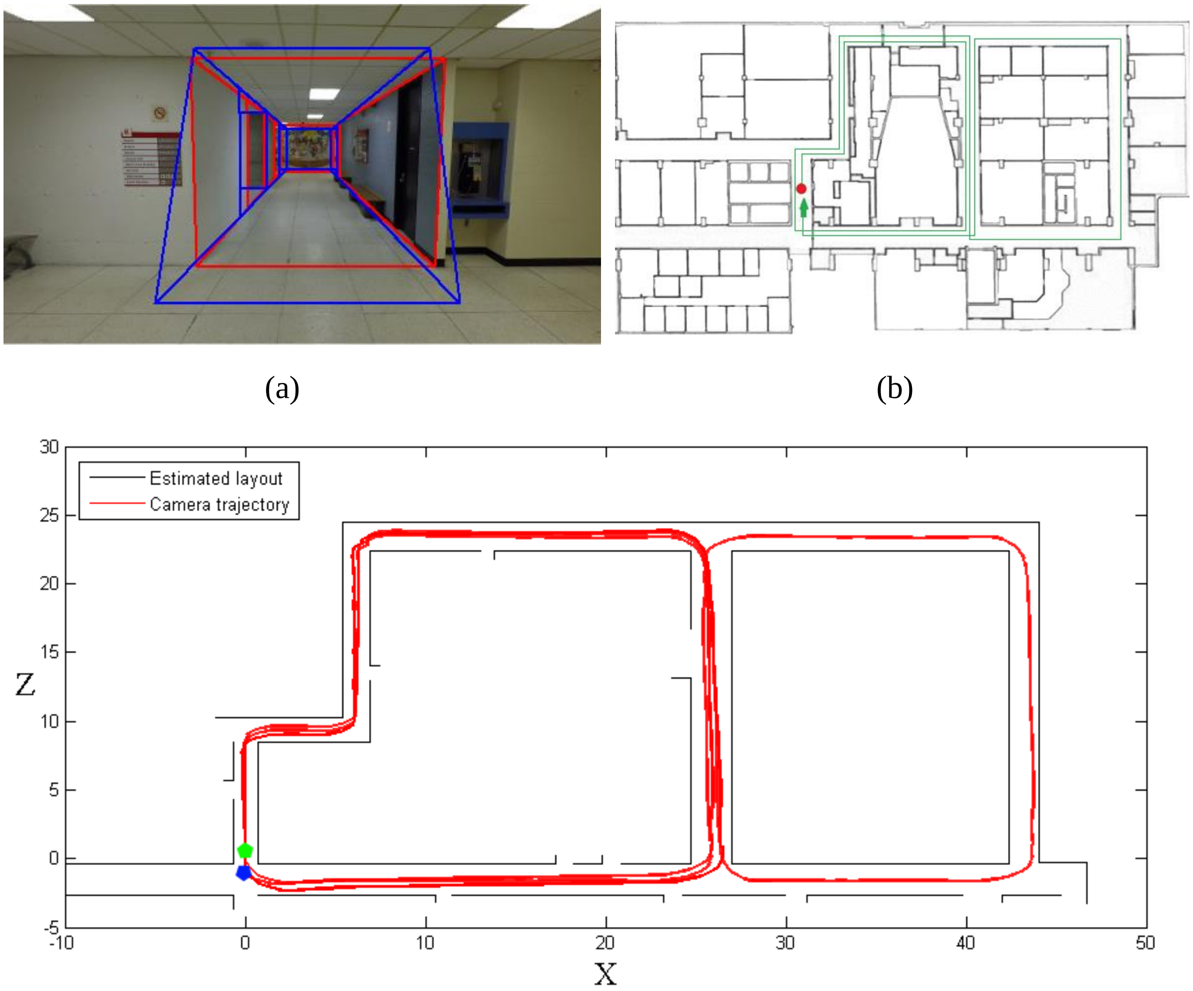
Figure 6.5 Generated corridor layouts for Ross building, first loop dataset; Adjusted corridor layouts after loop closing.

It should be noted that in these experiments the first estimated corridor always considered as fixed and the rest of the corridors are adjusted accordingly through loop closing. This is because Layout SLAM is always initialized by introducing the first true corridor layout through its structural corner point features which has been identified in the image space. Figure 6.5 proves that the proposed Layout SLAM method produces small orientation errors in estimation of the corridor layouts. Yet, the corridors depth scaling errors at long corridors are considerable where the count of features is low. This necessitates the implementation of the proposed model-based loop closing method in Layout SLAM architecture.

It should be noted that after the loop closing is performed on structural corner point features, the updated structural planes of corridor layouts may not be orthogonal. Thus, the layout orthogonality constraint is applied to the updated layouts considering the first corridor. Note that the orthogonality of the first corridor is preserved after loop closing since it was considered as fixed in the procedure.

Figure 6.6 shows the updated corridor layouts after model-based loop closing is performed on Ross building, two loops dataset. The camera's starting and ending points are depicted with green and blue circles respectively. Note that in this experiment after the camera completed its first loop and the loop closing has performed the updated corridor layouts will be considered as fixed for the rest of the run. Thus, adjusting the layout estimation errors in the second half (right side) corridors would not be troublesome.

After loop closing is performed on the estimated corridor layouts the outcoming results can be compared to the prepared ground truth layouts. As stated in chapter 3, three-dimensional ground truth corridor layouts are manually generated from laser point cloud. Also, 2D ground truth layouts are manually identified in the selected images. Hence, the comparison is possible both in 2D and 3D spaces. To achieve a better in-depth analysis the estimated layout for each individual keyframe image is compared to the ground truth layout. Since these two sets of layouts are prepared in different coordinate systems, the first task was to transform the estimated layouts into the same coordinate system as the laser data had. Here, side corridors and their connected walls are playing a great role in performing the 3D affine transformation.

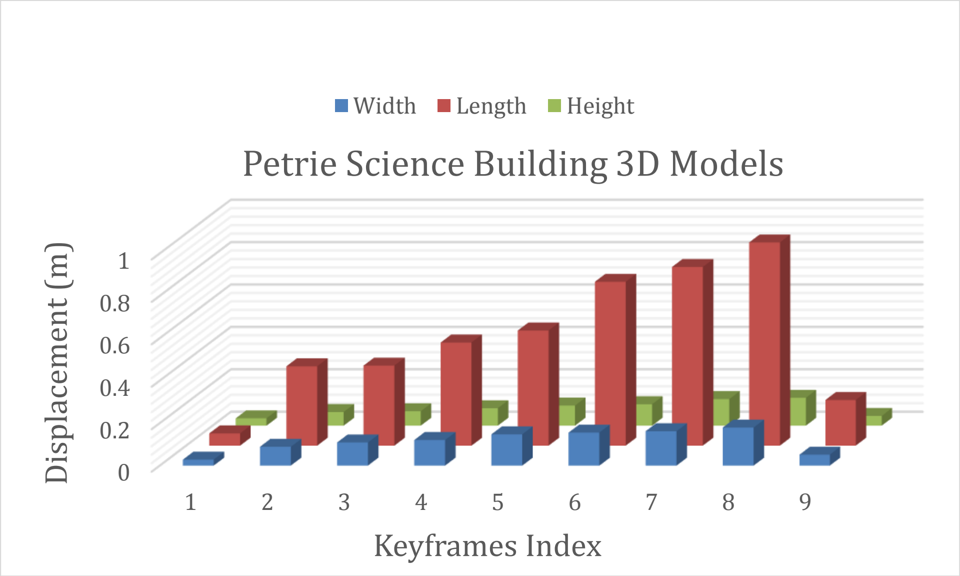


(a)

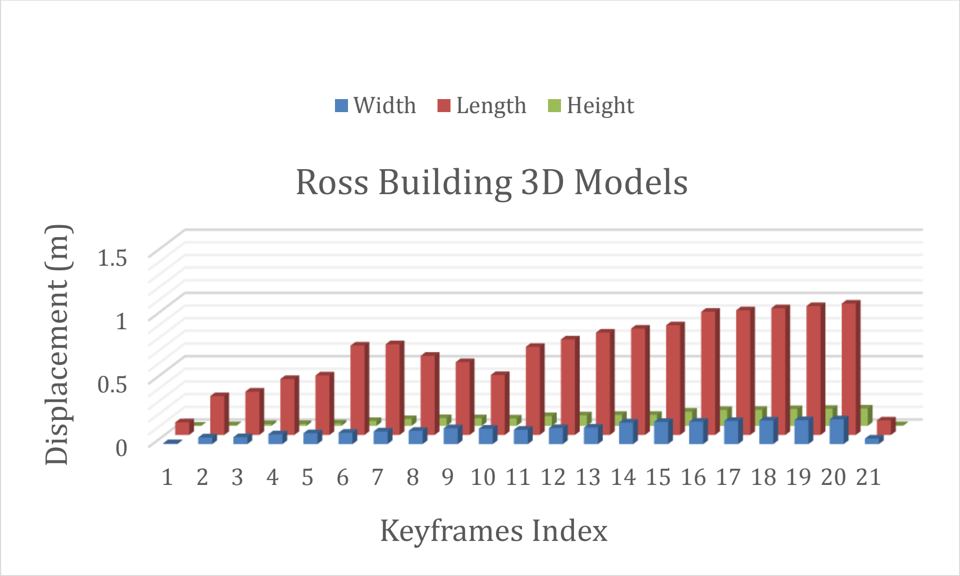
(b)

(c)

Figure 6.6 Estimated camera trajectory and corridor layouts, Ross building; (a) keyframe is matched to a test frame image (loop closing incident), (b) camera path schematic view, and (c) adjusted camera trajectory and corridor layouts after model-based loop closing.



(a)



(b)

Figure 6.7 Keyframe layouts comparison to the prepared ground truth data; (a) comparison of estimated keyframe corridor layouts for Petrie Science building dataset and (b) comparison of estimated keyframe corridor layouts for Ross building, two loops dataset.

Figure 6.7 shows the results of keyframe layouts comparison to the prepared ground truth data. Since the orthogonality of the generated layouts are preserved, the comparison could be performed in 3 major directions. As it can be seen in the above charts, the proposed Layout SLAM method could estimate the indoor corridor layouts with less than 20cm displacement errors in width and height and less than 1.05m in length. Table 6.3 reveals the mean and maximum trajectory errors for selected keyframes in the above experiments as well. Note that the ground truth trajectories were calculated for keyframes using the available laser point cloud. This table provides a better understanding over the Layout SLAM results. Here, the maximum absolute trajectory errors of 68cm in position and 2.84° in orientation (kappa angle) for approximately 315m path is obtained.

Table 6.3 Layout SLAM absolute trajectory errors on keyframes after loop closing.

Dataset	*Approximate path length*	*Position error (m)*			*Orientation error (degree)*		
		Average	*Maximum*	*Maximum error over trajectory*	*Omega Avg.*	*Phi Avg.*	*Kappa Avg.*
York University Petrie Science Building, one loop	*159m*	*0.239*	*0.546*	*0.34%*	*1.079*	*1.608*	*2.215*
York University Ross Building, two loops	*315m*	*0.301*	*0.673*	*0.21%*	*1.518*	*1.191*	*2.832*

6.5 Summary

In this chapter, we proposed a new method of loop closing detection for Layout SLAM. This model-based method can accurately identify the loop closing incidents when the camera visits the previously modeled corridor layouts in the seen. Since Layout SLAM algorithm is continuously challenged by long corridors and few numbers of features in the environments, applying a loop closing technique is essential for it. Hence, the proposed model-based loop closing method is specifically designed to address Layout SLAM drifting problems. The proposed method takes advantage of the estimated orthogonal layouts for various keyframes and identifies the loop closing incidents through applying a layout matching technique. Image and layout topology information is used to address keyframe matching. Considering image and layout topology information reduces matching ambiguity and increase the chance for global layout model consistency. Moreover, the mapping scale drifts would be suppressed while dealing with the same topological corridor layouts. Note that regardless of the textures in the scene the loop closing incident can be identified through examining layouts topology. This characteristic of the new method is more appealing when extracting enough features from texture-less environments is a challenging task. This method compensates the limitations of the existing loop closing algorithms at indoor corridor scenes. Yet, modeling errors including orientation error, shape deformation and boundary displacement can be addressed through model-based loop closing technique. The proposed method is evaluated through experiments conducted on York University dataset. The incoming result proves the robustness of this technique in detecting loop closing instances at indoor corridors.

Chapter 7

Conclusions and Future Directions

7.1 Conclusions

In this study, 3D reconstruction of indoor corridor layouts using a single image and visual SLAM is delivered. To achieve this goal, a new indoor modeling dataset is prepared, and major steps towards continuous indoor corridor modeling are presented. Thus, various 3D indoor modeling and reconstruction related topics are studied, and novel solutions are provided to identified problems. At the first step, a new method for reconstruction of geometrically accurate and regularly robust indoor corridor layout using an image is presented. At the second step, Layout SLAM is presented that automatically maps indoor corridors and simultaneously estimates camera's positions and orientations using set of images. At the third step, a model-based method for identifying SLAM loop closure incidents at indoor spaces is offered. Moreover, the quality of a reconstructed 3D indoor corridor layout is assessed using the prepared ground truth dataset. The following paragraphs will provide conclusions of each taken step towards automatically reconstruction of indoor corridor layouts in this thesis.

Chapter 3 introduced the newly generated indoor corridor modeling dataset. This dataset can be used for evaluating and assessing geometric qualities of reconstructed 3D indoor corridor layouts. The existing image-based modeling related datasets are mainly focused on single rooms and usually do not provide reference 3D models. Contrary to the existing datasets, the introduced dataset in this thesis is mainly focused on corridor scenes and provides ground truth indoor corridor layouts accompanied with camera trajectories for keyframes. Hence, the quality evaluation of both single image and visual SLAM reconstructed 3D indoor corridor layouts and the respective camera trajectories is feasible. Note that both TIMMS generated laser point clouds and manually reconstructed 2D and 3D corridor models are included in this dataset. Thus, measuring geometrical and topological accuracies of reconstructed indoor corridor models in both 2D and 3D spaces are possible. However, the limitation of the introduced dataset is that it does not reveal semantic related information of introduced indoor models. Therefore, addition of semantic-based information to this dataset should be investigated in future works.

Chapter 4 introduced a method for automatic 3D indoor corridor layout reconstruction using a single image. Indoor corridor modeling aided by extraction of low and middle level cues from a single image involves with critical problems. Reconstruction of a 3D indoor model from a single image is inherently an ill posed problem. Thus, to achieve a realistic 3D reconstruction of the layout, some prior knowledge must be inputted directly into the algorithm. Knowing camera’s height at the time of exposure and adopting Manhattan rule assumption to regularize indoor layout structures are considered in the proposed algorithm. The proposed method allows the estimated corridor layout to be comprised of multiple connected boxes to simply handle the presence of side corridors, contrary to existing methods, and solves the problem through hypothesizing-verifying multiple box primitives. Experiments reveal that using both physical line segments and virtual rays of vanishing points, the proposed method can generate corridor layout hypotheses even if clutters are occluding parts of the ceiling-wall or floor-wall boundaries. Both geometric and semantic information (orientation map and geometric context) of an image are extracted and incorporated in the suggested objective functions for finding the best fitting layout hypothesis to an image. The experimental results reveal that this incorporation enhances the performance of the proposed algorithm and reduces the occurrence of geometrical errors. The proposed method is one step forward towards image-based continuous indoor space modeling where the integration of individual indoor layouts is needed.

Chapter 5 presented Layout SLAM which is a new method for real-time simultaneous corridors layouts and camera poses estimation from a set of images. This chapter tackles an important problem in visual navigation and SLAM representation that deals with tracking camera's various poses in an unknown corridor environment. This method tracks corridors features in consecutive video frames and directly represents them in a 3D layout structure that has a known topological format. The proposed method solves the visual SLAM representation problem by directly providing layout topological information (eliminating the processing step to achieve better map representation). Experimental results depict that Layout SLAM can successfully compensate errors in camera orientation estimation through introducing the rotation compensation variable (matching consecutive vanishing directions on a unit sphere). Therefore, not only the problem of abrupt camera movements can be minimized, but also layout features tracking, and matching would be less cumbersome. Moreover, experimental results depict that providing line-based layout hypotheses for layout tracking can compensate the absence of textures and abundant geometrical features in the scene. Note that layout features matching cost function considers both global and local context information to deal with corridor scene related challenges. Experimental results reveal that Layout SLAM is robust against orientation error accumulations and produces very limited geometrical errors in estimation of the corridors layouts.

Chapter 6 presented the model-based loop closing technique which is designed for reducing layout and trajectory related errors in Layout SLAM system. The proposed method can identify previously visited layouts (related features) which is essential for loop closing. This method identifies keyframes and compares the topological graph of previously observed layouts to find the loop closing incident. Both semantical and layout topological information are incorporated to solve keyframe matching problem. Experimental results show that by considering layout topology graph, the proposed method can perform robustly in texture-less environments. Thus, mapping scale drifts could be successfully suppressed with less ambiguity. Hence, model-based loop closing can compensate the limitations of pure feature-based loop closing algorithms. Experimental results indicate that the proposed technique for loop closing detection, limits the search among the captured video frames and reduces the number of feature matching attempts which results in faster performance of the Layout SLAM.

7.2 Directions for Future Research

As stated in this thesis, the concept of continuous indoor space modeling involves with progressively reconstruct indoor models. This study provided a wide research platform for continuous indoor space modeling using a single source (image) data. This study's main goal is to automatically generate 3D indoor space models. However, beside images, laser scanning point clouds and other type of images (e.g. panorama and RGBD image) should be able to be incorporated in our continuous indoor space modeling framework; this subject can be studied in future researches. Following paragraphs will provide the future works for each chapter of this thesis:

- In terms of evaluating generated indoor space models, even though the prepared dataset provided valuable references of reconstructed indoor space models, this dataset only focused on providing geometric information of the models and the semantic information is missing. Thus, more works are needed to add semantic information to the prepared dataset. Moreover, in SLAM related reference dataset camera trajectories are only calculated for keyframe images which should be extended to the whole trajectory in the future.

- The proposed single image based indoor corridor modeling method provided promising results. However, the main limitation for this method was to accurately identify the end part of a corridor especially when the corridor is very lengthy (length is more than 50 meters). This is a main disadvantage of single image-based modeling approach that identifying layout’s front-face position is not possible if enough evidences are not present in an image. One possible solution would be the integration of other data sources in this process. Also, individual 3D indoor corridor models can be integrated by investigating their topological and semantical information. This approach would enable the progressive reconstruction of joint indoor corridor models while accommodating their scale differences. Hence, this cloud be a major future work in this research to integrate single models and step towards complete image-based indoor space modeling.

- The proposed Layout SLAM method is following the Manhattan rule assumption and only considers the respective structural format for output representation. In the future, Layout SLAM can be extended with less constrained geometric models. Also, Layout SLAM is applying the Extended Kalman Filtering (EKF) for inference. EKF handles the real-time camera localization in Layout SLAM architecture. However, EKF is not suitable for handling large number of features since its state vector grows by adding more features to the system that prolongates the computation time. Thus, applying other mathematical frameworks to handle back-end section of the Layout SLAM would be an interesting work for future.

- In relation to Layout SLAM development, a layout topology-based key frame selection and loop closing techniques are presented. The current version of this method is only considering the camera to move inside the main corridors and always facing the front-faces of them. Yet, the camera may reach a corridor from one of its sides. Thus, matching front-faces of side corridors to front-face of the main corridor in question would be an interesting future work in identifying the occurrence of a loop closing instance.

Bibliography

Ackerman, E., 2014. Dyson's robot vacuum has 360-degree camera, tank treads, cyclone suction. IEEE Spectrum.

Adán, A. and Huber, D., 2010. Reconstruction of wall surfaces under occlusion and clutter in 3D indoor environments. Robotics Institute, Carnegie Mellon University, Pittsburgh, PA CMU-RI-TR-10-12.

Anand, A., Koppula, H.S., Joachims, T. and Saxena, A., 2013. Contextually guided semantic labeling and search for three-dimensional point clouds. The International Journal of Robotics Research, 32(1), pp.19-34.

Antone, M. and Teller, S., 2000. Automatic recovery of relative camera rotations for urban scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 282–289.

Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M. and Savarese, S., 2016. 3d semantic parsing of large-scale indoor spaces. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1534-1543).

Azhar, S., 2011. Building information modeling (BIM): Trends, benefits, risks, and challenges for the AEC industry. Leadership and management in engineering, 11(3), pp.241-252.

Bajcsy, R., 1988. Active perception. Proceedings of the IEEE, 76(8), pp.966-1005.

Baligh Jahromi, A. and Sohn, G., 2015. Edge Based 3D Indoor Corridor Modeling Using a Single Image. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume II-3/W5, pp. 417–424.

Baligh Jahromi, A. and Sohn, G., 2016. Geometric Context and Orientation Map Combination for Indoor Corridor Modeling Using a Single Image. International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, 41. Volume XLI-B4, pp. 295–302.

Baligh Jahromi, A., Sohn, G., Shahbazi, M. and Kang, J. 2017. A PRELIMINARY WORK ON LAYOUT SLAM FOR RECONSTRUCTION OF INDOOR CORRIDOR ENVIRONMENTS. ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences, 4.

Baligh Jahromi, A., Sohn, G., Jung, J., Shahbazi, M. and Kang, J., 2018. LAYOUT SLAM WITH MODEL BASED LOOP CLOSURE FOR 3D INDOOR CORRIDOR

RECONSTRUCTION. ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences, 4(2).

Bazin, J.C., Seo, Y., Demonceaux, C., Vasseur, P., Ikeuchi, K., Kweon, I. and Pollefeys, M., 2012. Globally Optimal Line Clustering and Vanishing Point Estimation in Manhattan World. In: Proceedings of 25th IEEE Conference in Computer Vision and Pattern Recognition, pp. 638-645.

Bibby, C. and Reid, I., 2010. A hybrid SLAM representation for dynamic marine environments. In 2010 IEEE International Conference on Robotics and Automation (pp. 257-264).

Biber, P., Fleck, S., Busch, F., Wand, M., Duckett, T. and Strasser, W., 2005. 3D modeling of indoor environments by a mobile platform with a laser scanner and panoramic camera. In 2005 13th European Signal Processing Conference (pp. 1-4).

Bishop, C.M., 1995. Neural networks for pattern recognition. Oxford university press.

Bonarini, A., Burgard, W., Fontana, G., Matteucci, M., Sorrenti, D. and Tardos, J., 2006. RAWSEEDS: Robotics Advancement through Web-publishing of Sensorial and Elaborated Extensive Data Sets, IROS’06 Workshop on Benchmarks in Robotics Research, Beijing, China. pp. 16–23.

Bouguet, J. Y., 2004. Camera calibration toolbox for matlab.

Boulaassal, H., Landes, T., Grussenmeyer, P. and Tarsha-Kurdi, F., 2007. Automatic segmentation of building facades using terrestrial laser data. In ISPRS Workshop on Laser Scanning 2007 and SilviLaser 2007 (pp. 65-70).

Brenner, C., 2005. Building reconstruction from images and laser scanning. International Journal of Applied Earth Observation and Geoinformation, 6(3-4), pp.187-198.

Budroni, A. and Böhm, J., 2010. Automatic 3D modelling of indoor Manhattan-world scenes from laser data. Proceedings of the International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, pp.115-120.

Bueno, M., Bosché, F., González-Jorge, H., Martínez-Sánchez, J. and Arias, P., 2018. 4-Plane congruent sets for automatic registration of as-is 3D point clouds with 3D BIM models. Automation in Construction, 89, pp.120-134.

Burns, J. B., Hanson, A. R., Riseman, E. M., 1986. Extracting Straight Lines. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 4, pp. 425-455.

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I. and Leonard, J.J., 2016. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. IEEE Transactions on robotics, 32(6), pp.1309-1332.

Caprile, B., and Torre, V., 1990. Using Vanishing Points for Camera Calibration. In: International Journal of Computer Vision, vol. 4, no. 2, pp. 127-140.

Carrillo, H., Reid, I. and Castellanos, J.A., 2012. On the comparison of uncertainty criteria for active SLAM. In 2012 IEEE International Conference on Robotics and Automation (pp. 2080-2087).

Chao, Y.W., Choi, W., Pantofaru, C., and Savarese, S., 2013. Layout estimation of highly cluttered indoor scenes using geometric and semantic cues. In: Proceedings of the International Conference on Image Analysis and Processing, pp. 489-499.

Chen, J. and Cho, Y.K., 2016. Real-time 3D mobile mapping for the built environment. In ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction (Vol. 33, p. 1). Vilnius Gediminas Technical University, Department of Construction Economics & Property.

Chen, K., Lai, Y.K. and Hu, S.M., 2015. 3D indoor scene modeling from RGB-D data: a survey. Computational Visual Media, 1(4), pp. 267-278.

Chen, K., Lai, Y., Wu, Y.X., Martin, R.R. and Hu, S.M., 2014. Automatic semantic modeling of indoor scenes from low-quality RGB-D data using contextual information. ACM Transactions on Graphics, 33(6).

Cieslewski, T., Lynen, S., Dymczyk, M., Magnenat, S. and Siegwart, R., 2015. Map api-scalable decentralized map building for robots. In 2015 IEEE International Conference on Robotics and Automation (ICRA) (pp. 6241-6247).

Cipolla, R., Drummond, T., Robertson, D., 1999. Camera calibration from vanishing points in images of architectural scenes. In: Proceedings of British Machine Vision Conference, 13–16 September, Nottingham, UK, pp. 382–391.

Civera, J., Grasa, O. G., Davison, A. J., and Montiel, J. M. M., 2010. 1-Point RANSAC for extended Kalman filtering: Application to real-time structure from motion and visual odometry. Journal of Field Robotics, 27(5), pp. 609-631.

Clemente, L.A., Davison, A.J., Reid, I.D., Neira, J. and Tardós, J.D., 2007, June. Mapping Large Loops with a Single Hand-Held Camera. In Robotics: Science and Systems (Vol. 2, No. 2).

Clowes, M.B., 1971. On seeing things. Artificial intelligence, 2(1), pp.79-116.

Coughlan, J.M. and Yuille, A.L., 1999. Manhattan world: Compass direction from a single image by bayesian inference. In Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on (Vol. 2, pp. 941-947).

Criminisi, A., Reid, I., and Zisserman, A., 2000. Single view metrology. International Journal of Computer Vision, vol. 40, no. 2, pp. 123–148.

Cummins, M. and Newman, P., 2008. FAB-MAP: Probabilistic localization and mapping in the space of appearance. The International Journal of Robotics Research, 27(6), pp.647-665.

Curless, B. and Levoy, M., 1996. A volumetric method for building complex models from range images.

Davison, A. J., 2003. Real-Time Simultaneous Localisation and Mapping with a Single Camera. In: International Conference on Computer Vision. Vol. 3, pp. 1403-1410.

Davison, A. J., Reid, I. D., Molton, N. D., & Stasse, O., 2007. MonoSLAM: Real-time single camera SLAM. IEEE transactions on pattern analysis and machine intelligence, 29(6), pp. 1052–1067.

Delage, E., Lee, H., and Ng, A. Y., 2006. A dynamic Bayesian network model for autonomous 3D reconstruction from a single indoor image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2418-2428.

Del Pero, L., Bowdish, J., Fried, D., Kermgard, B., Hartley, E. and Barnard, K., 2012. Bayesian geometric modeling of indoor scenes. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp. 2719-2726.

Denis, P., Elder, J.H., and Estrada, F.J., 2008. Efficient edge-based methods for estimating Manhattan frames in urban imagery. In: Proceedings of the 10th European Conference on Computer Vision, Part II, pp. 197–210.

Desolneux, A., Moisan, L., Morel, J. M., 2000. Meaningful Alignments. International Journal of Computer Vision, vol. 40, no. 1, pp. 7-23.

Desolneux, A., Moisan, L., Morel, J.M., 2008. From Gestalt Theory to Image Analysis. Interdisciplinary Applied Mathematics, vol. 35. no.2, pp. 183-206.

Díaz-Vilariño, L., Khoshelham, K., Martínez-Sánchez, J. and Arias, P., 2015. 3D modeling of building indoor spaces and closed doors from imagery and point clouds. Sensors, 15(2), pp.3491-3512.

Eade, E. and Drummond, T., 2009. Edge landmarks in monocular slam. Image and Vision Computing, 27(5), pp. 588–596.

El-Hakim, S.F., 2000, December. Three-dimensional modeling of complex environments. In Video metrics and Optical Methods for 3D Shape Measurement (Vol. 4309, pp. 162-174).

Endres, F., Hess, J., Engelhard, N., Sturm, J., Cremers, D. and Burgard, W., 2012. An evaluation of the RGB-D SLAM system. In Robotics and Automation (ICRA), 2012 IEEE International Conference on (pp. 1691-1696).

Engel, J., Sturm, J. and Cremers, D. 2014. Scale-aware navigation of a low-cost quadrocopter with a monocular camera. Robotics and Autonomous Systems, 62(11), 1646-1656.

Estrada, C., Neira, J. and Tardós, J.D., 2005. Hierarchical SLAM: Real-time accurate mapping of large environments. IEEE Transactions on Robotics, 21(4), pp.588-596.

Everingham, M., Van Gool, L., Williams, C.K., Winn, J. and Zisserman, A., 2010. The pascal visual object classes (voc) challenge. International journal of computer vision, 88(2), pp.303-338.

Fidler, S., Dickinson, S., and Urtasun, R., 2012. 3D object detection and viewpoint estimation with a deformable 3D cuboid model. In: Advances in Neural Information Processing Systems, pp. 611–619.

Fischler, M.A. and Bolles, R.C., 1981. A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography (reprinted in Readings in Computer Vision, ed. MA Fischler,". Comm. ACM, 24(6), pp.381-395.

Flint, A., Murray, D. and Reid, I., 2011. Manhattan scene understanding using monocular, stereo, and 3d features. In 2011 International Conference on Computer Vision (pp. 2228-2235).

Frueh, C., Jain, S. and Zakhor, A., 2005. Data processing algorithms for generating textured 3D building facade meshes from laser scans and camera images. International Journal of Computer Vision, 61(2), pp.159-184.

Fuchs, C., Förstner, W., Gülch, E., Heipke, C., Eder, K., 1998. OEEPE survey on 3D-city models. Bundesamt für Kartographie und Geodäsie.

Furukawa, Y., Curless, B., Seitz, S.M. and Szeliski, R., 2009, June. Manhattan-world stereo. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on (pp. 1422-1429).

Furukawa, Y. and Hernández, C., 2015. Multi-view stereo: A tutorial. Foundations and Trends® in Computer Graphics and Vision, 9(1-2), pp.1-148.

Gimenez, L., Robert, S., Suard, F. and Zreik, K., 2016. Automatic reconstruction of 3D building models from scanned 2D floor plans. Automation in Construction, 63, pp.48-56.

Gioi, R. G., Jakubowicz, J., Morel, J. M., Randall, G., 2010. LSD: A Fast Line Segment Detector with a False Detection Control. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 4, pp. 722-732.

Grisetti G, Kümmerle R, Stachniss C, and Burgard W. 2010. A tutorial on graph-based slam. Intell Transp Syst Mag IEEE 2(4):31–43.

Grün, A., Baltsavias, E., Henricsson, O. (Eds.), 1997. Automatic extraction of man-made objects from aerial and space images (II). Birkhäuser, Basel.

Grün, A., Kübler, O., Agouris, P. (Eds.), 1995. Automatic extraction of man-made objects from aerial and space images. Birkhäuser, Basel.

Guzmán, A., 1968. Decomposition of a visual scene into three-dimensional bodies. In Proceedings of the December 9-11, 1968, fall joint computer conference, part I (pp. 291-304).

Hähnel, D., Burgard, W. and Thrun, S., 2003. Learning compact 3D models of indoor and outdoor environments with a mobile robot. Robotics and Autonomous Systems, 44(1), pp.15-27.

Hahnel, D., Triebel, R., Burgard, W. and Thrun, S., 2003. Map building with mobile robots in dynamic environments. In Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on (Vol. 2, pp. 1557-1563).

Han, F., and Zhu, S.C., 2005. Bottom-up/top-down image parsing by attribute graph grammar. In: Proceedings of the IEEE International Conference on Computer Vision, vol. 2, pp. 1778–1785.

Han, F. and Zhu, S.C., 2009. Bottom-up/top-down image parsing with attribute grammar. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(1), pp.59-73.

Harris, C. and Stephens, M., 1988. A Combined Corner and Edge Detector. In Proceeding of the Alvey Vision Conference, pp. 147–152.

Hedau, V., Hoiem, D., and Forsyth, D., 2009. Recovering the spatial layout of cluttered rooms. In: Proceedings of the 12th IEEE International Conference on Computer Vision, pp. 1849–1856.

Hedau, V., and Hoiem, D., 2010. Thinking inside the box: using appearance models and context based on room geometry. In: Proceedings of the European Conference on Computer Vision, pp. 1–14.

Hedau, V., Hoiem, D., and Forsyth, D., 2012. Recovering free space of indoor scenes from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2807-2814.

Henry, P., Krainin, M., Herbst, E., Ren, X. and Fox, D., 2014. RGB-D mapping: Using depth cameras for dense 3D modeling of indoor environments. In Experimental robotics, (pp. 477-491).

Heredia, F. and Favier, R., 2012. Kinect Fusion extensions to large scale environments. The Point Cloud Library.

Ho, K.L. and Newman, P., 2006. Loop closure detection in SLAM by combining visual and spatial appearance. Robotics and Autonomous Systems, 54(9), pp.740-749.

Hoiem, D., Efros, A., and Hebert, M., 2005. Geometric context from a single image. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 654–661.

Hoiem, D., Efros, A. A., & Hebert, M., 2007. Recovering surface layout from an image. International Journal of Computer Vision, vol. 75, no. 1, pp. 151–172.

Huang, A.S., Bachrach, A., Henry, P., Krainin, M., Maturana, D., Fox, D. and Roy, N., 2017. Visual odometry and mapping for autonomous flight using an RGB-D camera. In Robotics Research (pp. 235-252).

Ikehata, S., Yang, H. and Furukawa, Y., 2015. Structured indoor modeling. In Proceedings of the IEEE International Conference on Computer Vision, pp. 1323-1331.

International Herald Tribune, 2008. UN says half the world's population will live in urban areas by end of 2008. https://web.archive.org/web/20090209221745/http://www.iht.com/articles/ap/2008/02/26/news/UN-GEN-UN-Growing-Cities.php

Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A. and Fitzgibbon, A., 2011. KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera. In Proceedings of the 24th annual ACM symposium on User interface software and technology (pp. 559-568).

Jung, J. and Sohn, G., 2019. A line-based progressive refinement of 3D rooftop models using airborne LiDAR data with single view imagery. ISPRS Journal of Photogrammetry and Remote Sensing, 149, pp.157-175.

Kada, M., Wichmann, A., 2012. Sub-surface growing and boundary generalization for 3D building reconstruction. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, I (3), pp. 223-238.

Kaess, M., 2015. Simultaneous localization and mapping with infinite planes. In 2015 IEEE International Conference on Robotics and Automation (ICRA) (pp. 4605-4611).

Kanade, T., 1980. A theory of origami world. Artificial intelligence, 13(3), pp.279-311.

Klein, G. and Murray, D., 2008. Improving the Agility of Keyframe-Based SLAM. In: Proceedings of the European Conference on Computer Vision, volume 2, pp. 802–815.

Konolige, K. and Agrawal, M., 2008. FrameSLAM: From Bundle Adjustment to Real-Time Visual Mapping. IEEE Transactions on Robotics, 24(5), pp. 1066–1077.

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

Košecká, J. and Zhang, W., 2002. Video compass. In European conference on computer vision (pp. 476-490).

Košecká, J. and Zhang, W., 2005. Extraction, matching, and pose recovery based on dominant rectangular structures. Computer Vision and Image Understanding, 100(3), pp.274-293.

Kress, G.R. and Van Leeuwen, T., 1996. Reading images: The grammar of visual design. Psychology Press.

Kroeger, T., Dai, D., and Van Gool, L., 2015. Joint vanishing point extraction and tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2449-2457.

Lai, K., Bo, L. and Fox, D., 2014. Unsupervised feature learning for 3d scene labeling. In Robotics and Automation (ICRA), 2014 IEEE International Conference on (pp. 3050-3057).

Langguth, F., Sunkavalli, K., Hadap, S. and Goesele, M., 2016. Shading-aware multi-view stereo. In European Conference on Computer Vision (pp. 469-485).

Lee, D.C., Hebert, M., Kanade, T., 2009. Geometric reasoning for single image structure recovery. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2136–2143.

Lee, D.C., Gupta, A., Hebert, M., Kanade, T., 2010. Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In: Advances in Neural Information Processing Systems, pp. 1288-1296.

Lehtola, V.V., Kaartinen, H., Nüchter, A., Kaijaluoto, R., Kukko, A., Litkey, P., Honkavaara, E., Rosnell, T., Vaaja, M.T., Virtanen, J.P. and Kurkela, M., 2017. Comparison of the selected state-of-the-art 3D indoor scanning and point cloud generation methods. Remote Sensing, 9(8), p.796.

Li, M., Lin, R., Wang, H. and Xu, H., 2013. An efficient SLAM system only using RGBD sensors. In Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on (pp. 1653-1658).

Li, Y., Wu, X., Chrysathou, Y., Sharf, A., Cohen-Or, D. and Mitra, N.J., 2011. Globfit: Consistently fitting primitives by discovering global relations. In ACM Transactions on Graphics (TOG) (Vol. 30, No. 4, p. 52).

Liu, C., Schwing, A.G., Kundu, K., Urtasun, R. and Fidler, S., 2015. Rent3D: Floor-plan priors for monocular layout estimation. In: Computer Vision and Pattern Recognition (CVPR), pp. 3413-3421.

Liu, M., Guo, Y. and Wang, J., 2017. Indoor scene modeling from a single image using normal inference and edge features. The Visual Computer, 33(10), pp.1227-1240.

Lowe, D. G., 2004. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2), pp. 91–110.

Lowry, S., Sünderhauf, N., Newman, P., Leonard, J.J., Cox, D., Corke, P. and Milford, M.J., 2016. Visual place recognition: A survey. IEEE Transactions on Robotics, 32(1), pp.1-19.

Lu, Y. and Song, D., 2015. Visual navigation using heterogeneous landmarks and unsupervised geometric constraints. IEEE Transactions on Robotics, 31(3), pp.736-749.

Lutz, W., Butz, W.P. and Samir, K.E. eds., 2017. World population & human capital in the twenty-first century: an overview. Oxford University Press.

Maas, H.G. and Vosselman, G., 1999. Two algorithms for extracting building models from raw laser altimetry data. ISPRS Journal of photogrammetry and remote sensing, 54(2-3), pp.153-163.

Macher, H., Landes, T. and Grussenmeyer, P., 2017. From Point Clouds to Building Information Models: 3D Semi-Automatic Reconstruction of Indoors of Existing Buildings. Applied Sciences, 7(10), p.1030.

Macworth, A.K., 1973. Interpreting pictures of polyhedral scenes. Artificial intelligence, 4(2), pp.121-137.

Mair, E., Hager, G. D., Burschka, D., Suppa, M. and Hirzinger, G., 2010. Adaptive and Generic Corner Detection Based on the Accelerated Segment Test. In: Proceedings of the European Conference on Computer Vision, pp. 183–196.

Mattausch, O., Panozzo, D., Mura, C., Sorkine-Hornung, O. and Pajarola, R., 2014. Object detection and classification from large-scale cluttered indoor scans. In Computer Graphics Forum (Vol. 33, No. 2, pp. 11-21).

Meltzer, J. and Soatto, S., 2008. Edge Descriptors for Robust Wide-Baseline Correspondence. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8.

Micusik, B., Wildenauer, H., Kosecka, J., 2008. Detection and matching of rectilinear structures. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-7.

Murali, S., Speciale, P., Oswald, M.R. and Pollefeys, M., 2017. Indoor Scan2BIM: Building information models of house interiors. In Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on (pp. 6126-6133).

Mur-Artal, R., Montiel, J.M.M. and Tardos, J.D., 2015. ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE transactions on robotics, 31(5), pp.1147-1163.

Nedovic, V., Smeulders, A.W., Redert, A. and Geusebroek, J.M., 2007. Depth information by stage classification. In 2007 IEEE 11th International Conference on Computer Vision (pp. 1-8).

Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A.J., Kohi, P., Shotton, J., Hodges, S. and Fitzgibbon, A., 2011. KinectFusion: Real-time dense surface mapping and tracking. In Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on (pp. 127-136).

Newman, P. and Ho, K., 2005. SLAM-loop closing with visually salient features. In Robotics and Automation. ICRA 2005. Proceedings of the 2005 IEEE International Conference on (pp. 635-642).

Nießner, M., Zollhöfer, M., Izadi, S. and Stamminger, M., 2013. Real-time 3D reconstruction at scale using voxel hashing. ACM Transactions on Graphics (ToG), 32(6), p.169.

Nist´er, D., Naroditsky, O. and Bergen, J., 2004. Visual odometry. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 1, pp. 1–8.

Nüchter, A., 2008. 3D robotic mapping: the simultaneous localization and mapping problem with six degrees of freedom (Vol. 52).

Nüchter, A., Elseberg, J., Schneider, P. and Paulus, D., 2010. Study of parameterizations for the rigid body transformations of the scan registration problem. Computer Vision and Image Understanding, 114(8), pp.963-980.

Nüchter, A. and Hertzberg, J., 2008. Towards semantic maps for mobile robots. Robotics and Autonomous Systems, 56(11), pp.915-926.

Nüchter, A., Surmann, H., Lingemann, K. and Hertzberg, J., 2003. Semantic Scene Analysis of Scanned 3D Indoor Environments. In VMV (pp. 215-221).

Ochmann, S., Vock, R., Wessel, R. and Klein, R., 2016. Automatic reconstruction of parametric building models from indoor point clouds. Computers & Graphics, 54, pp.94-103.

Parodi, P., Piccioli, G., 1996. 3D shape reconstruction by using vanishing points. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, pp. 211–217.

Pero, L., Bowdish, J., Fried, D., Kermgard, B., Hartley, E., and Barnard, K., 2012. Bayesian geometric modeling of indoor scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2719-2726.

Philbin, J., Chum, O., Isard, M., Sivic, J. and Zisserman, A., 2007. Object retrieval with large vocabularies and fast spatial matching. In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on (pp. 1-8).

Phillips, C.J., Lecce, M., Davis, C. and Daniilidis, K., 2015. Grasping surfaces of revolution: Simultaneous pose and shape recovery from two views. In 2015 IEEE International Conference on Robotics and Automation (ICRA) (pp. 1352-1359).

Pillai, S. and Leonard, J., 2015. Monocular slam supported object recognition. arXiv preprint arXiv:1506.01732.

Pronobis, A. and Jensfelt, P., 2012. Large-scale semantic mapping and reasoning with heterogeneous modalities. In 2012 IEEE International Conference on Robotics and Automation (pp. 3515-3522).

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(6), pp.575-584.

Quattoni, A. and Torralba, A., 2009. Recognizing indoor scenes. In 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 413-420.

Rassia, S. T., 2017. Workplace Environmental Design in Architecture for Public Health: Impacts on Occupant Space Use and Physical Activity. Springer.

Requicha, A.G., 1980. Representations for rigid solids: Theory, methods, and systems. ACM Computing Surveys (CSUR), 12(4), pp.437-464.

Ripperda, N. and Brenner, C., 2009. Application of a formal grammar to facade reconstruction in semiautomatic and automatic environments. In Proc. of the 12th AGILE Conference on GIScience (pp. 1-12).

Rosten, E. and Drummond, T., 2006. Machine learning for high-speed corner detection. In: Proceedings of the European Conference on Computer Vision, pp. 430–443.

Rother, C., 2000. A new approach for vanishing point detection in architectural environments. In: Proceedings of 11th British Machine Vision Conference, pp. 382–391.

Rottensteiner, F., Trinder, J., Clode, S., Kubik, K., 2005. Automated delineation of roof planes from LiDAR data. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 36 (Part 3/W4), pp. 221–226.

Rusu, R.B., Blodow, N., Marton, Z., Soos, A. and Beetz, M., 2007. Towards 3D object maps for autonomous household robots. In Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on (pp. 3191-3198).

Salas-Moreno, R.F., Newcombe, R.A., Strasdat, H., Kelly, P.H. and Davison, A.J., 2013. Slam++: Simultaneous localisation and mapping at the level of objects. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1352-1359).

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

Sattler, T., Havlena, M., Schindler, K. and Pollefeys, M., 2016. Large-scale location recognition and the geometric burstiness problem. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1582-1590).

Saxena, A., Chung, S.H. and Ng, A.Y., 2006. Learning depth from single monocular images. In Advances in neural information processing systems (pp. 1161-1168).

Saxena, A., Sun, M. and Ng, A.Y., 2009. Make3d: Learning 3d scene structure from a single still image. IEEE transactions on pattern analysis and machine intelligence, 31(5), pp.824-840.

Schneider, J., Läbe, T. and Förstner, W., 2013. Incremental real-time bundle adjustment for multi-camera systems with points at infinity. ISPRS Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-1/W2: 355-360.

Schöning, J. and Heidemann, G., 2015, September. Evaluation of multi-view 3D reconstruction software. In International conference on computer analysis of images and patterns (pp. 450-461).

Schwing, A. G., Hazan, T., Pollefeys, M., Urtasun, R., 2012. Efficient Structured Prediction for 3D Indoor Scene Understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2815-2822.

Schwing, A. G., Urtasun, R., 2012. Efficient Exact Inference for 3D Indoor Scene Understanding. In: Proceedings of the European Conference on Computer Vision, pp. 299-313.

Schwing, A. G., Fidler, S., Pollefeys, M., Urtasun, R., 2013. Box in the box: Joint 3d layout and object reasoning from single images. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 353-360.

Seitz, S.M., Curless, B., Diebel, J., Scharstein, D. and Szeliski, R., 2006. A comparison and evaluation of multi-view stereo reconstruction algorithms. In null (pp. 519-528).

Shao, T., Xu, W., Zhou, K., Wang, J., Li, D. and Guo, B., 2012. An interactive approach to semantic modeling of indoor scenes with a RGBD camera. ACM Transactions on Graphics (TOG), 31(6), p.136.

Shapiro, V., 2002. Solid modeling. Handbook of computer aided geometric design, 20, pp.473-518.

Shukor, S.A., Young, K.W. and Rushforth, E.J., 2011. 3d modeling of indoor surfaces with occlusion and clutter. In Mechatronics (ICM), 2011 IEEE International Conference on (pp. 282-287).

Sibley, G., Mei, C., Reid, I. and Newman, P., 2010. Vast-scale Outdoor Navigation Using Adaptive Relative Bundle Adjustment. International Journal of Robotics Research, 29(8), pp. 958–980.

Silberman, N. and Fergus, R., 2011. Indoor scene segmentation using a structured light sensor. In Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on (pp. 601-608).

Silberman, N., Hoiem, D., Kohli, P. and Fergus, R., 2012. Indoor segmentation and support inference from rgbd images. In European Conference on Computer Vision (pp. 746-760).

Sohn, G., and Dowman, I., 2007. Data fusion of high-resolution satellite imagery and lidar data for automatic building extraction. ISPRS Journal of Photogrametry and Remote Sensing, 62(1), pp. 43-63.

Sugihara, K., 1984. A necessary and sufficient condition for a picture to represent a polyhedral scene. IEEE Transactions on Pattern Analysis and Machine Intelligence, (5), pp.578-586.

Sunderhauf, N., Shirazi, S., Jacobson, A., Dayoub, F., Pepperell, E., Upcroft, B. and Milford, M., 2015. Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free. Proceedings of Robotics: Science and Systems XII.

Tang, S., Zhu, Q., Chen, W., Darwish, W., Wu, B., Hu, H. and Chen, M., 2016. Enhanced RGB-D mapping method for detailed 3D indoor and outdoor modeling. Sensors, 16(10), p.1589.

Tardif, J. P., 2009. Non-iterative approach for fast and accurate vanishing point detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1250–1257.

Tarsha-Kurdi, F., Landes, T., Grussenmeyer, P., 2008. Extended RANSAC algorithm for automatic detection of building roof planes from lidar data. The photogrammetric journal of Finland, 21(1), pp. 97–109.

Tomljenovic, I., Tiede, D. and Blaschke, T., 2016. A building extraction approach for Airborne Laser Scanner data utilizing the Object Based Image Analysis paradigm. International Journal of Applied Earth Observation and Geoinformation, 52, pp.137-148.

Tóth, Z., Magnucz, P., Németh, R. and Tamás, J., 2015. Data model for hybrid indoor positioning systems. Production Systems and Information Engineering, 7(1), pp.67-80.

Triggs, B., McLauchlan, P.F., Hartley, R.I. and Fitzgibbon, A.W., 1999. Bundle adjustment—a modern synthesis. In International workshop on vision algorithms (pp. 298-372).

Ulusoy, A.O., Black, M.J. and Geiger, A., 2017. Semantic multi-view stereo: Jointly estimating objects and voxels. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 4531-4540).

Valentin, J.P., Sengupta, S., Warrell, J., Shahrokni, A. and Torr, P.H., 2013. Mesh based semantic modelling for indoor and outdoor scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2067-2074.

Valero, E., Adán, A. and Cerrada, C., 2012. Automatic method for building indoor boundary models from dense point clouds collected by laser scanners. Sensors, 12(12), pp.16099-16115.

Vanegas, C.A., Aliaga, D.G. and Beneš, B., 2010. Building reconstruction using manhattan-world grammars. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (pp. 358-365).

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

Waltz, D., 1972. Generating semantic descriptions from line drawings of scenes with shadows. Tech. Rep. AI-TR-271.

Wang, C., Hou, S., Wen, C., Gong, Z., Li, Q., Sun, X. and Li, J., 2018. Semantic line framework-based indoor building modeling using backpacked laser scanning point cloud. ISPRS journal of photogrammetry and remote sensing, 143, pp.150-166.

Wang, H., Gould, S., Koller, D., 2010. Discriminative learning with latent variables for cluttered indoor scene understanding. In: Proceedings of the European Conference on Computer Vision, pp. 435-449.

Wang, H., Gould, S. and Roller, D., 2013. Discriminative learning with latent variables for cluttered indoor scene understanding. Communications of the ACM, 56(4), pp.92-99.

Wang, X. and Gupta, A., 2016. Generative image modeling using style and structure adversarial networks. In European Conference on Computer Vision (pp. 318-335).

Whelan, T., Kaess, M., Fallon, M., Johannsson, H., Leonard, J. and McDonald, J., 2012. Kintinuous: Spatially extended kinectfusion.

Whelan, T., Kaess, M., Johannsson, H., Fallon, M., Leonard, J.J. and McDonald, J., 2015. Real-time large-scale dense RGB-D SLAM with volumetric fusion. The International Journal of Robotics Research, 34(4-5), pp.598-626.

Whelan, T., Leutenegger, S., Salas-Moreno, R., Glocker, B. and Davison, A., 2015. ElasticFusion: Dense SLAM without a pose graph. Robotics: Science and Systems.

Williams, B., Cummins, M., Neira, J., Newman, P., Reid, I. and Tardós, J., 2008. An image-to-map loop closing method for monocular SLAM. In Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on (pp. 2053-2059).

Williams, B., Cummins, M., Neira, J., Newman, P., Reid, I., and Tardós, J. 2009. A comparison of loop closing techniques in monocular SLAM. Robotics and Autonomous Systems, 57(12), 1188-1197.

Xiao, J. and Furukawa, Y., 2014. Reconstructing the world’s museums. International journal of computer vision, 110(3), pp.243-258.

Xiao, J., Owens, A. and Torralba, A., 2013. Sun3d: A database of big spaces reconstructed using sfm and object labels. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1625-1632).

Xiao, W., Vallet, B., Brédif, M. and Paparoditis, N., 2015. Street environment change detection from mobile laser scanning point clouds. ISPRS Journal of Photogrammetry and Remote Sensing, 107, pp.38-49.

Xie, L., Zhu, Q., Hu, H., Wu, B., Li, Y., Zhang, Y. and Zhong, R., 2018. Hierarchical Regularization of Building Boundaries in Noisy Aerial Laser Scanning and Photogrammetric Point Clouds. Remote Sensing, 10(12), p.1996.

Xiong, X., Adan, A., Akinci, B. and Huber, D., 2013. Automatic creation of semantically rich 3D building models from laser scanner data. Automation in Construction, 31, pp.325-337.

Xiong, X. and Huber, D., 2010. Using Context to Create Semantic 3D Models of Indoor Environments. In BMVC, pp. 1-11.

Yang, Y., Jin, S., Liu, R., Bing Kang, S. and Yu, J., 2018. Automatic 3D Indoor Scene Modeling from Single Panorama. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 3926-3934).

Yang, S., Maturana, D., and Scherer, S., 2016. Real-time 3D scene layout from a single image using convolutional neural networks. In Robotics and Automation (ICRA), IEEE International Conference, pp. 2183-2189.

Yin, X., Wonka, P. and Razdan, A., 2009. Generating 3d building models from architectural drawings: A survey. IEEE computer graphics and applications, 29(1).

Yu, S., Zhang, H., Malik, J., 2008. Inferring spatial layout from a single image via depth-ordered grouping. In: Proceedings of the IEEE Workshop on Perceptual Organization in Computer Vision, pp. 1-7.

Zhang, Y., Song, S., Tan, P., & Xiao, J., 2014. PanoContext: A whole-room 3D context model for panoramic scene understanding. In: Proceedings of the European Conference on Computer Vision, pp. 668-686.

Zhou, H., Zou, D., Pei, L., Ying, R., Liu, P., and Yu, W., 2015. StructSLAM: Visual SLAM with building structure lines. IEEE Transactions on Vehicular Technology, 64(4), pp. 1364-1375.

Zhu, H., Weibel, J.B. and Lu, S., 2016. Discriminative multi-modal feature fusion for RGBD indoor scene recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2969-2976.