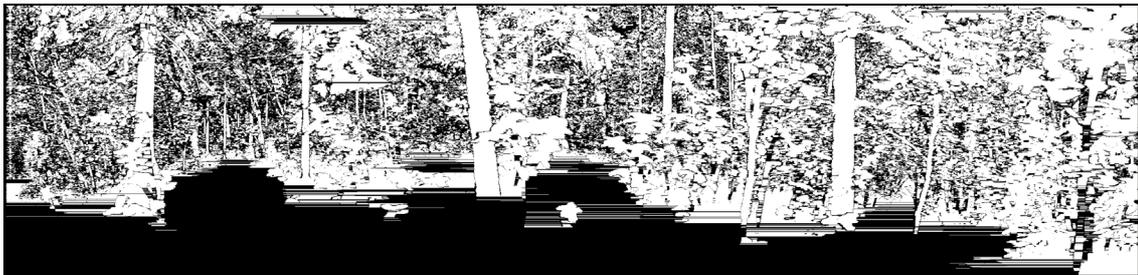


THREE-DIMENSIONAL LOCALIZATION AND MAPPING OF STATIC ENVIRONMENTS BY MEANS OF MOBILE PERCEPTION

Pekka Forsman



TEKNILLINEN KORKEAKOULU
TEKNISKA HÖGSKOLAN
HELSINKI UNIVERSITY OF TECHNOLOGY

THREE-DIMENSIONAL LOCALIZATION AND MAPPING OF STATIC ENVIRONMENTS BY MEANS OF MOBILE PERCEPTION

Pekka Forsman

Dissertation for the degree of Doctor of Technology to be presented with due permission for public examination and debate in Auditorium T2 at Helsinki University of Technology (Espoo, Finland) on the 23rd of November 2001, at 12 o'clock noon.

Helsinki University of Technology
Department of Automation and Systems Technology
Automation Technology Laboratory

Teknillinen korkeakoulu
Automaatio- ja systeemitekniikan osasto
Automaatiotekniikan laboratorio

Distribution:

Helsinki University of Technology
Automation Technology Laboratory
P.O. Box 5400
FIN-02015 HUT
FINLAND

e-mail pekka.forsman@hut.fi
Tel +358-9-451 3309
Fax +358-9-451 3308

© Pekka Forsman

ISBN 951-22-5740-8
ISSN 0783-5477

Pica-set Oy
Helsinki 2001

Abstract

Model-based task planning is one of the main capabilities of autonomous mobile robots. Especially for model-based localization and path planning, a large-scale description of the operation environment is required. Cognitive communication between man and his machine could be based on a common, three-dimensional understanding of the environment. In the case of a personal service robot, the operation environment may comprise both indoor and outdoor spaces. In this thesis, a method for the generation of a three-dimensional geometric model for large scale, structured and natural environments is presented.

The environment mapping method, which uses range images as measurement data, consists of three main phases: first, geometric features are extracted from each of the range images. Secondly, the relative coordinate transformations (i.e. registrations) between the sensor viewpoint locations, where the range data was measured, are computed. And, finally, an integrated map is formed by transforming the sub-map data into a common frame of reference.

Two types of geometric features are extracted from the range images: cylinder segments (or more generally truncated cone segments) and straight-line segments. With cylinder segments tree trunks and other elongated cylindrical objects can be modeled, whereas the straight line segments correspond to the upper corners of vertical walls. The features are utilized as natural landmarks for registration computation.

The presented method is tested by mapping three test sites representing structured, semi-structured and natural environments. The structured environment corresponds to a part of the premises of an office building, the semi-structured environment corresponds to the surroundings of a parking lot and the natural environment is a small forest area. The dimensions of the test sites are about 50 meters, 120 meters and 40 meters square, respectively. A simple incremental approach is used to build an integrated model for the parking lot and office corridor environments. For the principal mapping experiment, concerning the small forest area, a statistically more sound, optimal approach is applied. With respect to the feature extraction methods and the computation of the relative coordinate transformations between the viewpoints, robustness to outlier data and failure modes of the methods are discussed in more detail.

Keywords: Geometric feature extraction, viewpoint registration, 3D environment mapping

Preface

This study has been conducted at the Automation Technology Laboratory of Helsinki University of Technology during the years 1997-2001. The basic kinematics algorithms and data management programs for the laser scanner were developed and implemented when working on the Shico project, financed by the European Commission. Other parts of the study have been carried out within different projects for the development of the WorkPartner robot, financed by Technology Development Centre of Finland (TEKES) and Academy of Finland. The largest part of the financing has come from the general funds of the laboratory. All the sources of financial support are gratefully acknowledged.

I wish to express my sincere gratitude to Professor Aarne Halme, head of the Automation Technology Laboratory, for his encouragement, inspiration and expert guidance during this work. And, in particular, the freedom to concentrate on this work without having to worry too much about other duties in the laboratory.

Special thanks go to Mr. Janne Paanajärvi for offering his expertise and assistance in solving many algorithmic problems during the project. Also, Janne provided the c-language code for solving the linear system of equations, required for the computation of the optimal integrated map. Other colleagues, especially Professor Arto Visala and Mr. Jorma Selkänaho have helped me as well.

My thanks are further due to the whole personnel of the Automation Technology Laboratory for creating a friendly and inspiring atmosphere for the work.

I thank my preliminary examiners Docent Dr. Tapio Heikkilä and Professor Eduardo M. Nebot for their reviews and valuable suggestions on the thesis. Many thanks are also due to Kathleen Tipton for the English language revision of the manuscript. The remaining mistakes are my own.

Finally my warmest thanks go to my wife Briitta, and our daughters Minna and Anni for love, patience and support, which gave me the ultimate motivation to complete the work.

Espoo, November 2001

Pekka Forsman

Contents

| | |
|--|--|
| Abstract | |
| Preface | |
| Contents | |
| List of principal symbols, abbreviations and glossary of terms | |

1. Introduction 1

| | |
|--------------------------------------|---|
| 1.1 Description of the mapping task | 1 |
| 1.2 Motivation of the dissertation | 2 |
| 1.3 Contribution of the dissertation | 3 |
| 1.4 Outline of the study | 5 |

2. Three-dimensional environment mapping 7

| | |
|---|----|
| 2.1 Introduction | 7 |
| 2.1.1 Definition of an environment map | 7 |
| 2.1.2 Topological or metric map representation | 8 |
| 2.1.3 Level of abstraction of the model | 9 |
| 2.1.4 Action planning for efficient exploration | 9 |
| 2.1.5 2D versus 3D mapping | 9 |
| 2.2 Structures of metric 3D maps | 10 |
| 2.2.1 Grid maps | 10 |
| 2.2.2 Polygon mesh maps | 11 |
| 2.2.3 Geometric feature maps | 13 |

3. Probabilistic formulation of the simultaneous localization and mapping task 15

| | |
|---|----|
| 3.1 Introduction | 15 |
| 3.2 Stochastic, single hypothesis map representation | 21 |
| 3.2.1 Remote observation-global localization approach | 21 |
| 3.2.1.1 Single state space formulation | 21 |
| 3.2.1.2 Relocation-fusion strategy | 24 |
| 3.2.1.3 Decoupled stochastic mapping | 25 |
| 3.2.2 Relative observation-global localization map formulation | 25 |
| 3.3 Maximum likelihood, multiple hypothesis map estimation | 28 |
| 3.3.1 Probabilistic task models | 29 |
| 3.3.1.1 Prediction of sensor motion | 29 |
| 3.3.1.2 Perception model | 30 |
| 3.3.1.3 Approximation of multi-modal probability distributions | 30 |
| 3.3.2 Maximum likelihood map generation | 31 |
| 3.3.2.1 Globally optimal map from Gaussian approximation of uncertainties | 31 |
| 3.3.2.2 Expectation-maximization algorithm for map tuning | 31 |
| 3.3.3 Map topology | 32 |

4. Feature extraction from three-dimensional measurement data 33

| | |
|--|----|
| 4.1 Introduction | 33 |
| 4.2 Perception sensors for 3D data acquisition | 36 |
| 4.2.1 Triangulation-based measurement systems | 36 |
| 4.2.2 Single signal path-based range measurement | 38 |
| 4.2.3 Fusion of range data and camera image for 3D perception | 39 |
| 4.2.4 3D perception through motion | 40 |
| 4.3 Segmentation of range data | 41 |
| 4.3.1 Pixel-based region growing by means of variable order surface fitting | 42 |
| 4.3.2 Range image segmentation based on scan line approximation | 42 |
| 4.4 Shape extraction in structured environments | 43 |
| 4.5 Geometric models for natural outdoor environments | 44 |
| 4.5.1 Polynomial volumetric models | 44 |
| 4.5.2 Cylinder segment models | 45 |
| 4.6 Fitting cylinder segment models to narrow elongated objects | 46 |
| 4.6.1 Segmentation of cluttered range data | 46 |
| 4.6.2 Estimation of the circle model from segmented range data | 50 |
| 4.6.3 Construction of cylinder segment models from circle model data | 56 |
| 4.6.4 Experimental results | 57 |
| 4.7 Fitting straight line models in free space to unknown/occupied space borders | 65 |
| 4.7.1 Description of the method | 65 |
| 4.7.2 Implementation considerations | 68 |
| 4.8 Conclusions | 69 |

5. Registration of a pair of sub-maps 71

| | |
|--|----|
| 5.1 Introduction | 71 |
| 5.1.1 Problem description | 71 |
| 5.1.2 Raw measurement data versus feature-based registration | 73 |
| 5.1.3 Multiple hypothesis registration | 74 |
| 5.2 Methods for finding a locally optimal registration | 74 |
| 5.2.1 Iterative, point-based matching | 74 |
| 5.2.1.1 Iterative closest point | 74 |
| 5.2.1.2 Iterative dual correspondence | 75 |
| 5.2.2 Gradient ascent in search space | 76 |
| 5.3 Global registration methods | 76 |
| 5.3.1 An augmented local method | 76 |
| 5.3.1.1 Local method preceded by a coarse global registration step | 76 |
| 5.3.1.2 Combination with simulated annealing | 77 |
| 5.3.2 Maximum correlation registration | 77 |
| 5.3.3 Divide and conquer search based on Hausdorff metric | 78 |
| 5.3.3.1 Map similarity measure | 78 |
| 5.3.3.2 Global search strategy | 79 |

| | | |
|--|--|------------|
| 5.3.3.3 | Gaussian approximation of the registration uncertainty | 82 |
| 5.3.3.4 | Assessment of correctness of the registration result | 82 |
| 5.4 | Probabilistic registration of feature maps | 82 |
| 5.4.1 | Experimental results | 86 |
| 5.5 | Conclusions | 94 |
| 6. Environment modeling experiments | | 97 |
| 6.1 | Introduction | 97 |
| 6.1.1 | Description of the experimental system | 98 |
| 6.1.1.1 | The Riegl LMS-Z210 sensor | 98 |
| 6.1.2 | Description of the mapping method | 99 |
| 6.2 | Construction of an integrated map for the parking lot area | 102 |
| 6.3 | Modeling the interior of a building | 109 |
| 6.4 | Generating a globally optimal model for the forest terrain | 114 |
| 6.5 | Conclusions | 126 |
| 7. Key problems in automatic map construction | | 127 |
| 7.1 | Evaluation of the correctness of the map | 127 |
| 7.2 | Treatment of multiple map hypotheses | 129 |
| 8. Summary and conclusions | | 131 |
| References and literature | | 133 |
| Appendices | | 147 |
| Appendix A | Range image data | |
| Appendix B | Feature extraction from range images | |
| Appendix C | Registration of pairs of feature sub-maps | |
| Appendix D | Integrated environment model data | |
| Appendix E | State estimation for discrete-time, linear, stochastic systems | |
| Appendix F | State estimation for discrete-time, non-linear, stochastic systems | |
| Appendix G | Distribution approximation filter | |

List of principal symbols, abbreviations and glossary of terms

List of symbols

| | |
|---|--|
| $\mathbf{x}(k)$ | state vector corresponding to the unknown locations of the sensor platform (e.g. a mobile robot) and the N pieces of map objects (landmarks) at (time) instant/event k (REMO-GL) / state vector corresponding to the unknown locations of the N pieces of map objects (sub-maps) at (time) instant/event k (RELA-GL) |
| $\hat{\mathbf{x}}(k)$ | an estimate of the state vector at (time) instant/event k |
| $\hat{\mathbf{x}}_r(k)$ | an estimate of the robot location at (time) instant/event k |
| \mathbf{x}_i | map object location, i.e. the four dimensional (x, y, z, ϕ) -position vector, of the i^{th} map object with respect to the global frame of reference |
| $\hat{\mathbf{x}}_i(k)$ | an estimate of the i^{th} map object location at (time) instant/event k |
| $\boldsymbol{\varepsilon}_x$ | uncertainty associated to the estimate of the state vector |
| $\mathbf{P}(k)$ | covariance matrix of the state estimate uncertainty |
| $\mathbf{P}_{rr}(k)$ | covariance matrix of the robot location estimate uncertainty |
| $\mathbf{P}_{ir}(k)$ | covariance matrix between the uncertainty of the estimate of the i^{th} map object location and the uncertainty of the estimate of the sensor platform (e.g. robot) location |
| $\mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{v}, k)$ | (non-linear) function of system dynamics |
| $\mathbf{F}(k)$ | matrix of linear system dynamics |
| $\mathbf{u}(k)$ | known control input for the dynamic states in the state vector |
| $\mathbf{G}(k)$ | coefficient matrix of the control input |
| $\mathbf{v}(k)$ | zero-mean, white, random noise sequence of the system dynamics model |
| $\mathbf{Q}(k)$ | covariance matrix of the system noise model |
| $\mathbf{h}(\mathbf{x}, \mathbf{w}, k)$ | (non-linear) measurement model |
| $\mathbf{H}(k)$ | coefficient matrix of the linear measurement model |
| $\mathbf{w}(k)$ | zero-mean, white, random noise sequence of the measurement model |
| $\mathbf{R}(k)$ | covariance matrix of the measurement noise |
| $\mathbf{g}(\hat{\mathbf{x}}_r, \mathbf{z}_{new})$ | kinematic model for creating an initial estimate for the global location of a new model object |
| $\mathbf{G}_{x_r}, \mathbf{G}_{z_{new}}$ | jacobian matrices of $\mathbf{g}(\hat{\mathbf{x}}_r, \mathbf{z}_{new})$ |
| m_i | i^{th} sub-map/map object, $m_i = \{f_i, p_i, \mathbf{x}_i\}$ |
| M | integrated map, constructed by merging $N+1$ sub-maps $M = \{m_i\}_{i=0, \dots, N}$ |

| | |
|--|--|
| f_i | observed features within the i^{th} sub-map, $f_i = \{f_{ik}\}_{k=1,\dots,K_i}$, where K_i is the number of features |
| p_i | xyz-data points within the i^{th} sub-map, $p_i = \{\mathbf{p}_{il}\}_{l=1,\dots,O_i}$, where O_i is the number of data points; data points correspond to measured points or point coordinates extracted from the observed features |
| \mathbf{p}_{il} | 3D position vector of the x-, y- and z-coordinates of the l^{th} data point in the i^{th} sub-map |
| \mathbf{d}_{ij} | relative displacement of the i^{th} sub-map location with respect to the location of the j^{th} sub-map described by the four dimensional $(dx_{ij}, dy_{ij}, dz_{ij}, d\phi_{ij})$ -position vector. |
| $\bar{\mathbf{d}}_{ij}$ | an observation of the relative displacement |
| \mathbf{C}_{ij} | covariance matrix corresponding to the Gaussian approximation of the uncertainty related to the observation of the relative displacement between the locations of the i^{th} and j^{th} sub-map |
| $p(\mathbf{x}_{i+1} \mathbf{u}_i, \mathbf{x}_i)$ | motion model of the sensor platform |
| $p(p_i M, \mathbf{x}_i)$ | perception/sensor model, i.e. likelihood of the observation data points p_i (the data points p_i are modeled within the i^{th} sub-map with a bound precision) conditioned both on the global location of the i^{th} sub-map and on the known and fixed integrated map |
| $\alpha(i)$ | (probability) distribution of \mathbf{x}_i computed by means of measurement data collected before or while the sensor platform was/is at the i^{th} location. |
| $\beta(i)$ | (probability) distribution of \mathbf{x}_i computed by means of measurement data collected after the sensor platform was at the i^{th} location. |
| η | normalizer term ensuring that the estimate of the perception sensor location, \mathbf{x}_i , computed as the product of $\alpha(i)$ and $\beta(i)$, integrates to one i.e. is a valid probability distribution. |
| Δ | |
| $=$ | equal by definition |
| \approx | approximately equal |

Abbreviations

| | |
|---------|--|
| 2D | two-dimensional |
| 3D | three-dimensional |
| SLAM | simultaneous localization and mapping |
| REMO-GL | remote observation-global localization approach for SLAM |
| RELA-GL | relative observation-global localization approach for SLAM |
| dof | degree-of-freedom |
| SVD | singular value decomposition |
| ICP | iterative closest point |
| IDC | iterative dual correspondence |

Glossary of terms

location: A six dimensional vector of the translation lengths along and the rotation angles around the x, y and z axis of a Cartesian reference coordinate system.

position: Used along with **location** to describe the values of the three translation and three rotation degrees of freedom of an object with respect to a Cartesian reference coordinate system.

pose: Same as **location** or **position**.

optimal map: An integrated map computed by arranging the global locations of the sub-maps in proportion to the uncertainties of the observations of the relative alignment coordinates between pairs of sub-maps. In order to compute the optimal, least-squares estimates for the global sub-map locations a parallel network of observations is required.

Chapter 1

Introduction

1.1 Description of the mapping task

Environment mapping deals with the problem of building a topological and/or metric model of the scene visible to the mapping sensor. In many cases, only a part of the target environment can be modeled from a single viewpoint and multiple sets of measurement data must be collected to cover all corners of the environment. Therefore, integration of the local sub-maps into a unified model comprises one of the most central tasks of environment mapping.

A map built from an environment can contain different levels of qualitative and metric information. Topological maps describe characteristics, which are independent of environment geometry and sequential relationships within the environment. For example, the significant locations and travelable connections between them may be described with a topological map. If also information on the lengths of the connections is also contained in the map, the map structure can be called metric-topological. In Figure 1.1 an example of metric topological maps is depicted.

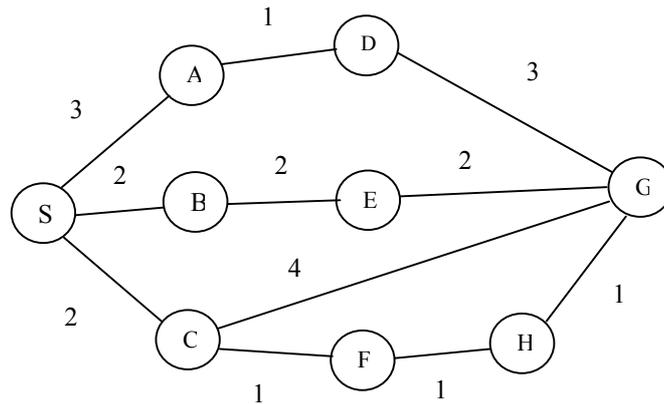


Figure 1.1 An example of the layout of metric-topological maps. The available paths from the starting location "S" to the goal location "G" are described within the map.

Metric maps represent the properties and coordinates of each location within the borders of the area of interest. A popular realization of a metric map, especially in the mobile robot research community, is the evidence/occupancy grid map. The mapped area is divided into regular sized cells, each of which contains the accumulated evidence of being intersected by an object or belonging to free space. Until the perception sensor probes a particular grid cell it will contain the 'unexplored' status.

An alternative to the metric grid maps is geometric feature maps. Instead of dividing the mapped space into regular sized grid cells, a more explicit representation of the geometry of the contours of the objects in the environment is used.

For example, straight-line segments and circles may be used to model the 2D projection of wall planes and major objects in an indoor office environment. The models discussed above suit well to a task such as the collision-free path planning of mobile robots, especially in structured indoor environments. However, 2D maps may not be rich enough for fully autonomous robots to reliably carry out complex operations, for long periods, in real environments. Navigation is clearly one of the key capabilities of any autonomous (service) robot. In structured environments, 2D navigation may be good enough to guarantee reliable operation of an autonomous mobile robot. However, manipulating the 3D world to perform some useful work is often a more challenging task, which requires 3D perception capabilities. When autonomous robots need to co-operate with human beings, a common 3D understanding of the operation environment in particular would be very beneficial.

1.2 Motivation of the dissertation

Future service robots should be able to co-operate with people seamlessly both in indoor and outdoor environments. Cognitive communication with the operator and autonomous capabilities to accomplish versatile tasks requires 3D modeling of the operation environment. In natural outdoor environments several characteristics make the 3D mapping task particularly challenging. Changes in foreground and

background illumination complicate the extraction of an object's contours from camera images. Occlusion in cluttered environments may change the visible scene contents drastically even due to a small displacement of the sensor. Acquiring a good initial estimate for the traveled distance between perception positions may be difficult due to rough outdoor terrain conditions.

Localization of a mobile robot carrying the perception sensor is the first step in the simultaneous localization and map building (SLAM) process. In the computer graphics community this is often referred to as the registration of viewpoints, where measurement data is acquired. For large scale mapping of indoor and outdoor environments, reliance on the availability of distinguishable landmarks may be problematic. Symmetric architectural features or natural, equal looking objects such as trees may complicate the composition of a rich enough set of unique feature pairs for registration computation. In that regard, the ideal registration method should possess the following characteristics:

- Robustness to outliers and missing data
- Ability to find globally optimal alignment within a relative large search space
- Support a multiple hypothesis registration result
- Give an estimate of the probability of correctness of the result (optimal alignment may not necessarily be the correct one)

Indeed, the registration techniques that are able to utilize indistinguishable feature data seem to offer the proper basis for the implementation of a generic method for computing the relative alignment between 3D feature maps.

1.3 Contribution of the dissertation

A method to construct a 3D model for static indoor and outdoor environments has been developed and tested with real measurement data, acquired with a high-end range-imaging sensor. The developed method consists of three main phases: first, for all range images, feature models are computed by means of the dense range image data. Then, sub-maps are created from the feature object models, by describing their coordinates with respect to the reference frame, which is attached to the location where the corresponding range image was originally acquired. Second, relative alignment coordinates among the partly overlapping sub-maps are computed with respect to the x-, y- and z- translation degrees of freedom as well as with respect to the rotation angle around the z-axis. As the third mapping step, an integrated large-scale map is formed from the feature object models, which were described with respect to the corresponding local sub-map, and from the relative alignment data computed for each of the overlapping sub-map pairs. For computing an integrated model, a straightforward, incremental approach was applied for the first two test environments whereas a statistically more sound, optimal approach was used to compute a large-scale model for the natural forest terrain.

The main contribution of the thesis is related to the development and experimental verification of a simultaneous localization and mapping (SLAM) method for static 3D environments. The method is based on existing estimation algorithms which have been studied, with the most proper one being selected for the method. This is

one of the first systems that exploit the new 3D range imaging technology for modeling large-scale environments. Moreover, building automatically an optimal feature-based map for a natural forest terrain, by using data acquired at several locations, is, to the author's knowledge, the first implementation of its kind.

The second contribution of this work is related to the extraction of geometric models from range image data for two different kinds of natural landmarks. The first landmark type corresponds to elongated cylindrical objects which can be used to model the geometry of supporting columns and light poles in structured environments and, in particular, to model the visible sections of tree trunks in natural environments. The method, for the modeling of curved elongated objects with a circular cross-section, is based on an iterative process. The generated models are compact cylinder segment approximations, ideal building blocks for an abstracted feature-based model of the natural (forest) terrain environment. The knowledge of knowing the geometric type of the observed objects can be utilized for high-level communication between human and autonomous robot. With a conventional polygon approximation of the object contours, the same information content of the model is not attained. The other landmark type corresponds to the upper edges of vertical wall planes, which augment the feature set of cylindrical objects in a structured environment to facilitate more reliable computation of the relative alignment coordinates between overlapping sub-map pairs. Here, the contribution is related to the technique of how the 3D point coordinates are generated from the 3D occupancy grid for straight-line approximation.

The third contribution of the work is the extension of an existing viewpoint registration algorithm to 3D environments. In particular, problems related to the approximation of the uncertainty of the registration result and assessment of the correctness of the result, are discussed. The individual contributions, in order of relative importance, can also be given in list form (starting with the most significant one):

- Development and experimental verification of a simultaneous localization and map building method for large scale, static environments. In this context, the residual information, computed from the integrated optimal map is demonstrated to provide means to assess the correctness of a new relative map alignment observation, which is to be added to the map building process. In this way a more error tolerant autonomous SLAM-method can be realized. (Chapters 6 and 7).

- Extraction of cylinder segment models, or more generally, truncated cone segment models from range image data for modeling elongated cylindrical objects. The contribution is related to the iterative algorithm for approximating curved elongated objects, with a circular cross-section, by a sequence of cylinder segment objects (or more generally, truncated cone segment objects). Moreover, the method can be configured to accept objects, or rather, measurement echoes from objects, which indicate different levels of circularity in the object cross-section. This is particularly important due to the wide laser beam pattern and significant divergence of the beam, which increases the noise in the measurement data as a function of the measured distance. Also, tree trunks, which are the principal category of objects considered in this work, do not have an absolutely circular cross section. (Chapter 4.6).
- Registration of a pair of feature maps in four dimensions. An alternative way to approximate the precision of the result, related to the implemented method, is suggested. Also, the possibility to assess the correctness of the result is discussed (Chapter 5.4). A method to classify the newly computed relative alignment hypothesis as being correct or faulty is then presented in Chapters 6.4 and 7.1.
- Approximation of the upper end corners of vertical wall planes by means of a straight-line segment model. The contribution is related to the extraction of the 3D point data from the 3D occupancy grid. The estimation of the 3D straight line equation parameters from the point data is done by means of a standard algorithm (Chapter 4.7).

1.4 Outline of the study

The dissertation consists of eight chapters. Two first chapters comprise an illustrative introduction to the subject of 3D model construction, especially for outdoor mobile robot applications. In Chapter 3, the state-of-the-art in simultaneous localization and map building (SLAM) is discussed. At the beginning of Chapters 4 and 5, previous research related to feature extraction methods and to the registration of overlapping feature maps is discussed, respectively. At the end of Chapters 4 and 5, our own contribution to the subjects is presented. In Chapter 6, construction of large-scale models for both structured and natural environments is discussed. In Chapter 7, a technique to automatically assess the correctness of the constructed optimal map is proposed. And finally, the developed methods are summarized and warrants for future research are presented in Chapter 8. A more detailed description of the contents of the work is given in the form of the following list:

- In Chapter 2, construction of 3D maps from the map contents point of view is discussed. Three different 3D model appearances are illustrated. The models were created by means of one of the range images, used later in the modeling experiments.

- In Chapter 3, the most significant, probabilistic map construction methods are summarized. The presented methods cover the majority of published SLAM implementations. The published methods, which often aim at achieving real time operation performance, usually produce a 2D model.
- In Chapter 4, different perception systems for acquiring 3D measurement data are first presented. Then, segmentation of range data as the first step in feature extraction is discussed. At the end of the chapter, our contribution to range image-based feature extraction will be presented.
- In Chapter 5, computation of the relative alignment coordinates between overlapping sets of perception data (i.e. sub-maps) for the construction of an integrated, large-scale map will be discussed. Again, published work within the subject is first presented. Then our own contribution, in the form of applying one of the published methods to the registration of 3D data sets, will be described.
- In Chapter 6, the main contribution of the thesis, which considers the construction of a large-scale 3D environment model, will be discussed. First the perception sensor used and the environment mapping method are described. Then experimental results with respect to three different test environments are presented.
- In Chapter 7, the possibilities to assess the correctness of the constructed environment model are discussed.
- And finally, a summary of the work and directions for future research are presented in Chapter 8.

Chapter 2

Three-dimensional environment mapping

2.1 Introduction

2.1.1 Definition of an environment map

The environment maps, considered in this work, can be defined as a description of a physical environment for some specific application. An environment model is usually formed by merging perception data acquired from a number of different positions within the environment. The locations of the coordinate frames, attached to the perception positions or to the locations of some important geometric features in the environment, constitute the skeleton of the environment model. The substance of the final model, designed for the end-user, will be built above the skeleton. For example, the structure or skeleton of the model may be estimated by means of natural landmarks or features extracted from the 3D perception data. However, the end user desires to have a detailed polygonal mesh approximation of the contours of the objects in the environment. The feature models, utilized to compute the coordinates of the local frames of reference forming the map structure, can now be replaced with the polygonal mesh approximations computed from the "raw" measurement data. The principle is illustrated in Figure 2.1.

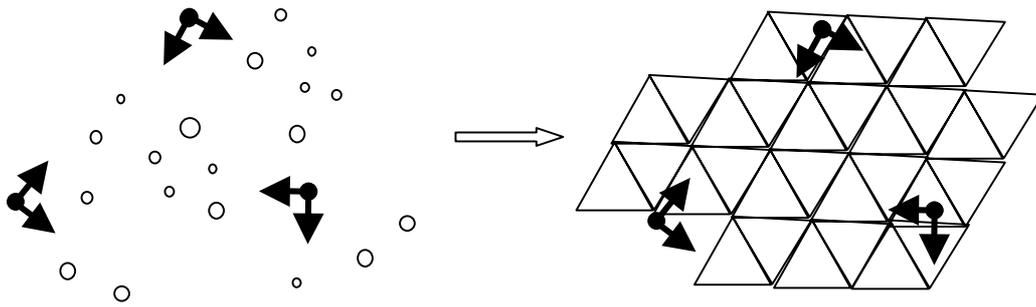


Figure 2.1 An illustration of the two phases in the construction of a geometric map. On the right, the map, desired by the end user.

In the left part of the figure, the relative alignment of the sub-maps (marked with the reference frames) has been estimated by means of the features corresponding to the dots in the image. In the right part of the figure, another representative form (a surface mesh) of the objects in the environment is laid out on the top of the reference frames (i.e. the skeleton of the map).

2.1.2 Topological or metric map representation

Topological maps can be described as a semantic layout of significant locations in the environment. The distribution of and connections between the significant locations are based on qualitative data. Topological maps are relatively compact and allow utilization of fast navigation and mapping algorithms. Their major drawback is the same as their benefit, which is the sparse and non-metric environment representation. Applications that require fine-grained information about a random location within the environment cannot rest solely on topological environment representations.

Metric environment models can be separated into two representation classes: numerical and geometric. In numeric maps the model space is divided in regions of constant numeric value of desired quantity. Grid-based models are the most popular representational types of numeric maps, in which the model space regions are rectangular cells.

Geometric models explicitly approximate the objects of interest with a surface or volumetric primitive representation. The rest of the scene's contents remains undefined, with only the objects being modeled. An example of a generic representation tool for object surface geometry is the triangular mesh model. For example, the VRML-format is a realization of triangular mesh maps. Volumetric model primitives, on the other hand, describe the volume enclosed by the envelope defined by the model equation. A volumetric model primitive, utilized in this work, is the cylinder segment model, which defines the sub-space enclosed within its boundaries by means of six pose parameters and the radius and the length of the cylinder segment. A generalization of the cylinder segment model is the truncated cone segment model in which the radii of the two end points of the segment are defined separately. Realizations of the cylinder segment model can be seen in several pictures throughout the thesis.

2.1.3 Level of abstraction of the model

The representation modes for an environment map can be divided into different categories. Semantic and topological maps describe qualitative information about the environment whereas metric maps give numeric or geometric information. Qualitative information, such as the geometric primitive type of the object, can be given interactively by the operator [Forsman and Halme, 1989] and [Forsman and Halme, 1995]. However, perception sensors may also be programmed to extract semantic information on the environment. Quek et al. discussed a generic-to-specific refinement process for computing compound cylinder and planar object models from dense range data [Quek et al., 1993]. They explain that extracting feature models from the numeric measurement data corresponds to providing a semantic description of the perceived objects and the scene. Nevertheless, *a priori* knowledge, that the object types under consideration are a good interpretation of the scene's contents, is required. Indeed, many of the proposed scene modeling methods, for example [Högström and Wernersson, 1998] and [Betgé-Brezetz et al., 1994a], where the processing advances from a data-based description to a higher level representation, follow the generic-to-specific principle.

2.1.4 Action planning for efficient exploration

To make the exploration of unknown environments more efficient, planning of the perception positions can be applied. Active exploration methods, such as [Whaite and Ferrie, 1997] and [Fox et al., 1998], are capable of planning the route of the robot and perception positions in a way that takes into account, for example, the traversability of the terrain and the location of unmapped sections in the workspace.

2.1.5 2D versus 3D mapping

The majority of the work for simultaneous localization and mapping has only taken into consideration 2D environments. The restriction to operate on a 2D plane is justified when navigating in structured indoor environments. Outdoors, and especially in natural terrain, 3D models are required to describe the geometry of the environment. Maybe the simplest example of a 3D model is the elevation map, where the altitude value of the highest/lowest laser beam hit point within each of the small area patches (i.e. xy-grid cells) is recorded. The elevation map is one of the basic model types used in this work.

2.2 Structures of metric 3D maps

The realization of map geometry, built above the computed map structure, as depicted in Figure 2.1, depends on the end-user application. For example, in order to assess the traversability of an outdoor terrain, an elevation map may be desired. On the other hand, if the outer shape of a historical monument needs to be recovered, a polygonal mesh model might be preferred. For cognitive communication between a human operator and a service robot, a more abstracted feature model may be required. To illustrate the different model types, the scene shown in the photographs of Figure 2.2 will be mapped by using these three alternative representations of map geometry.



Figure 2.2 Two camera images of the scene visible from the location from where the range image "scanF" was measured.

In the images of Figure 2.2, the north side of the Computer Science Building of HUT and a group of trees are depicted. To demonstrate the different 3D model representations, the measurement data contained in the range image "scanE" is worked into an elevation map, triangular mesh map and geometric feature map representation. Note that the range image "scanE" was measured from a location near the trees on the right hand side of the driveway.

2.2.1 Grid maps

As an example of a metric grid map, an elevation map formed from the range image "scanE" is depicted in Figure 2.3. The dimensions of the grid are 600 elements in both vertical and horizontal directions.

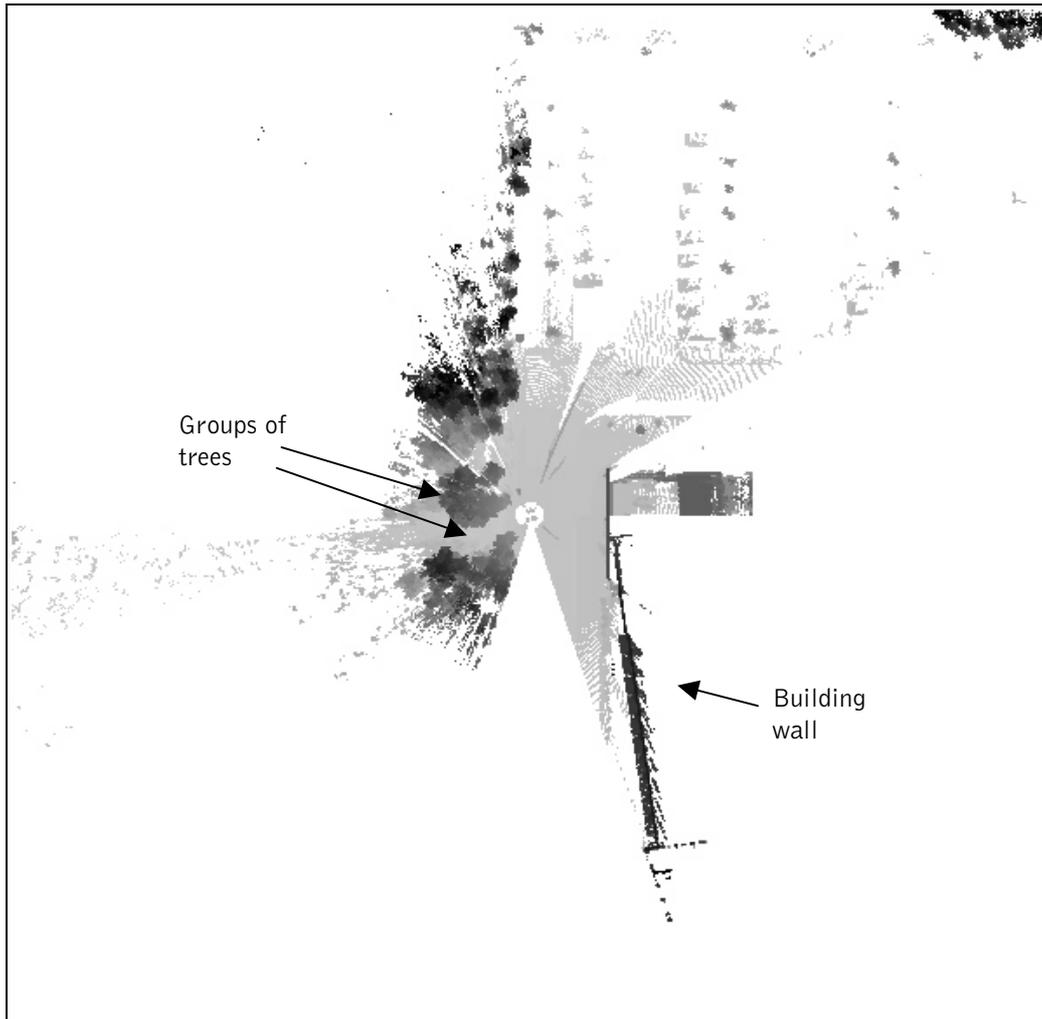


Figure 2.3 Elevation map computed by means of the range image "scanE". The size of the grid map is 120 meters square corresponding to the dimensions of 0.2 meters square for one grid cell.

An example of an environment, where elevation maps can be used is an open mine. Also, large scale elevation maps for open terrain can be generated from the data collected by an airborne sensor.

2.2.2 Polygon mesh maps

Polygon mesh maps approximate the original object geometry by triangular planar surface patches. A number of commercial software packages can generate a polygon representation from the original measurement data. Moreover, programs such as the PolyWorks utility program package can align and merge smaller models into a larger, unified representation. This is exactly the same problem discussed in this work. However, the PolyWorks program, studied in this work, seems to apply (actual methods are confidential) an iterative corresponding point or feature pair computation method. Such algorithms require a relative good initial approximation to converge to the globally optimal alignment solution. In our study, we apply a different alignment technique in order to be able to relax the accurate a

priori alignment requirement. In Figure 2.4 a polygonal mesh model of the scene, depicted in Figure 2.2, is shown.

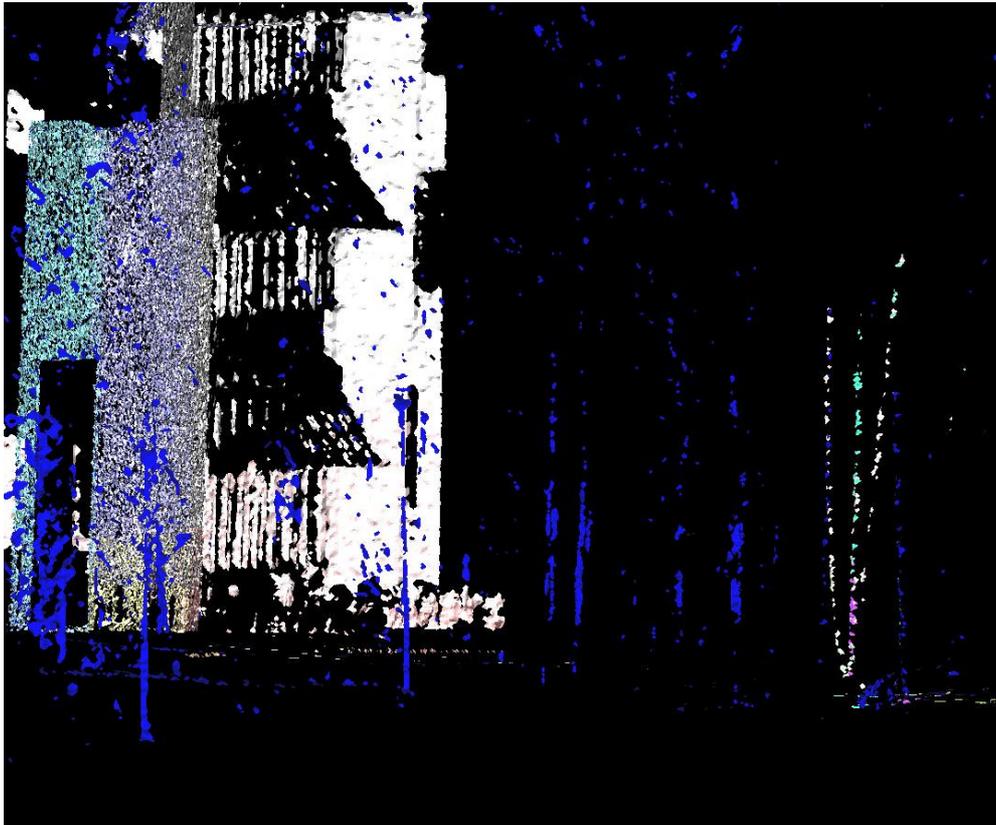


Figure 2.4 A polygonal mesh map generated from the range scanner data set "scanE". The PolyWorks program package of InnovMetric Software Inc. was used to generate the polygon representation.

The model, generated from a cluttered scene consists of a large number of polygons. Moreover, the qualitative information contained within a triangular mesh model is small. Actually, there is no abstracted information on the geometric type of the object from which the measurements originate. For a human being it is quite easy to figure out from the image its contents. To generate a corresponding algorithm for a machine may not be straightforward. Indeed, a more abstracted model, that would also yield the basic geometric type of the underlying physical object, might be a better basis for designing "intelligent" (or cognitive) communication interfaces for mobile robots.

2.2.3 Geometric feature maps

A geometric feature map consists of a set of geometric primitive objects. In this work, the geometric primitive models considered are cylinder segment (or truncated cone segment) models and straight-line models. The former object type is used to approximate the geometry of narrow elongated objects with a circular cross-section. Examples of physical objects in this category are pillars, telephone poles and tree trunks. Straight-line objects are used to describe the location of the upper corner of walls. The extracted feature objects are primarily used for computing the relative alignment between neighboring scanner positions. However, they can also be used to visualize the environment geometry as depicted in Figure 2.5. In the figure, the straight-line models corresponding to the upper end of the visible wall planes of the building, depicted in the left photograph of Figure 2.2, are extended up to the expected ground level (z-coordinate set to zero). The size of the model is only a few kilobytes.

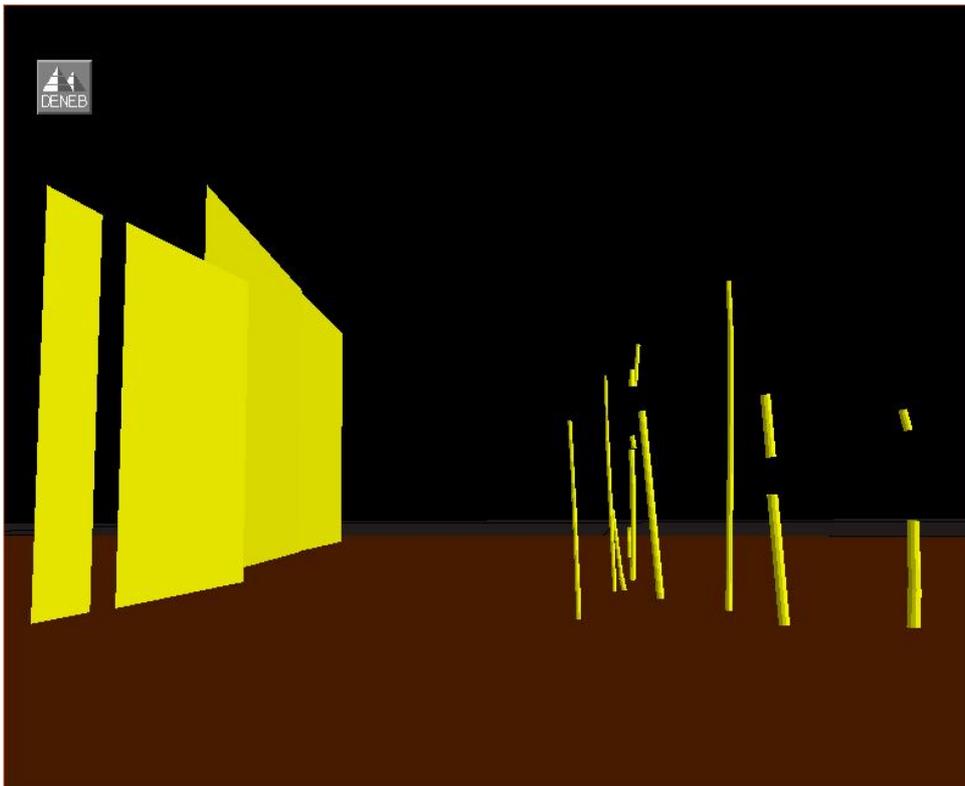


Figure 2.5 Feature object models generated from the data in the range image "scanE". The planar surfaces on the left correspond to the estimated wall planes of the building whereas the cylinder models to the right represent the visible sections of the tree trunks and the light pole.

The compactness and information content of the geometric primitive models make them attractive to real time applications, such as model based task planning and cognitive communication between human and robot. Moreover, the large-scale 3D description of the environment geometry can be utilized as the "ground truth" model for real-time navigation of the mobile robot. The robot can calibrate its global location parameters by utilizing the suitable 2D profile extracted from the large-scale 3D model. For example, building walls and the locations of the trees

can serve as beacons for resetting the accumulated error in the robot coordinates. The WorkPartner robot [Halme et al., 2001], which is the target application for the utilization of the feature-based 3D environment models developed in this work, is shown in Figure 2.6.



Figure 2.6 WorkPartner, the prototype of a mobile service robot designed to work interactively with humans in outdoor environments. The robot is equipped with a 2D laser scanner used for autonomous, model-based navigation.

The high-level communication interface of the WorkPartner robot can be based on a model such as the one depicted in Figure 2.5. In particular, if the virtual model is augmented with texture data corresponding to the natural, photograph-like appearance of the environment, a proper basis for cognitive communication between man and his machine can be established. Examples of reality-augmented virtual model views are given in Chapters 4.6.2 and 6.3.

It should be emphasized that the large-scale feature object models developed in this work comprise the basis of the final environment description. Dynamic and/or small sized objects can be augmented to the environment model by using, for example, an interactive method. With the quasi-coaxial arrangement of a laser pointer and a video camera, mounted on the head of WorkPartner, the human operator can easily augment the basic model appearance by using, for example, the method described in [Forsman, 1994].

Chapter 3

Probabilistic formulation of the simultaneous localization and mapping task

3.1 Introduction

Map building by means of a mobile perception sensor mounted, for instance, on an autonomous mobile robot can be described as a sequence of two alternating tasks: localization with respect to the existing map and an updating of the map based on the result of localization. The localization and map update steps are tightly coupled with each other and therefore we are faced with the classic chicken-and-egg problem.

Construction of a model for the environment concerns the fusion of perception data collected from a number of viewpoints. The perception data can be in the form of features extracted from the measurement data or sub-maps. Sub-maps correspond to a collection of entities such as measurement points or extracted features represented with respect to the reference frame of the sub-map. The uncertainty related to the location of the entities with respect to the reference frame of the sub-map is assumed to be bound. The error bounds can be determined from the accuracy of the perception sensor and/or the feature extraction method. The individual features or sub-maps correspond to the model objects, which are the building blocks to be welded together for a unified environment representation.

The main phases in simultaneous localization and map building process can be described as:

- Acquisition of external measurement data from several viewpoints.
- Registration of the measurement data with respect to each other or with respect to a common frame of reference.
- Globally optimal organization of the map items, based on their mutual registration dependencies.

The mutual synchronization of the phases may differ significantly. For example, the collection of measurements from the environment and the registration of measurement data can be carried out alternately, in real time, resulting in the popular simultaneous localization and mapping procedure. On the other hand, a major part of the perception data can be collected before the optimal map generation step is executed (off-line). Obviously, it is up to the laboriousness of the mapping method whether it can or cannot be carried out on-line, while exploring the environment.

A considerable amount of work concerning simultaneous localization and map building has been published during the past 10 to 15 years. The procedures for making observations and maintaining spatial relationships of most of the proposed methods can be classified into three categories. In the first category, the spatial locations of the map objects are first observed, remotely, with respect to the sensor frame. Based on the remote observations, relationships for the map objects with respect to a global reference frame are computed. We will call the principle here the *remote observation-global localization* (REMO-GL) approach. Technically, the global location of the map object (landmark) is updated as a function of the difference between the real and the predicted observations of the landmark. The predicted observation is computed as the difference between the estimates of the global location of the landmark and the global location of the sensor platform. The *REMO-GL* approach is clearly the most popular approach adopted by many research groups. Previous work related to the application of a *REMO-GL* type SLAM approach has reported, for example, in [Moutarlier and Chatila, 1989], [Rencken, 1993], [Feder et al., 1998a] and [Thrun et al., 1998a]. The key idea is that the perception sensor mounted on a mobile platform observes map objects, i.e. features or landmarks, remotely. The global location of the map objects and the global location of the perception sensor (or the equivalently global location of the mobile sensor platform) are updated concurrently, for example, within a Kalman filter mechanism. The *REMO-GL* principle is depicted in Figure 3.1.

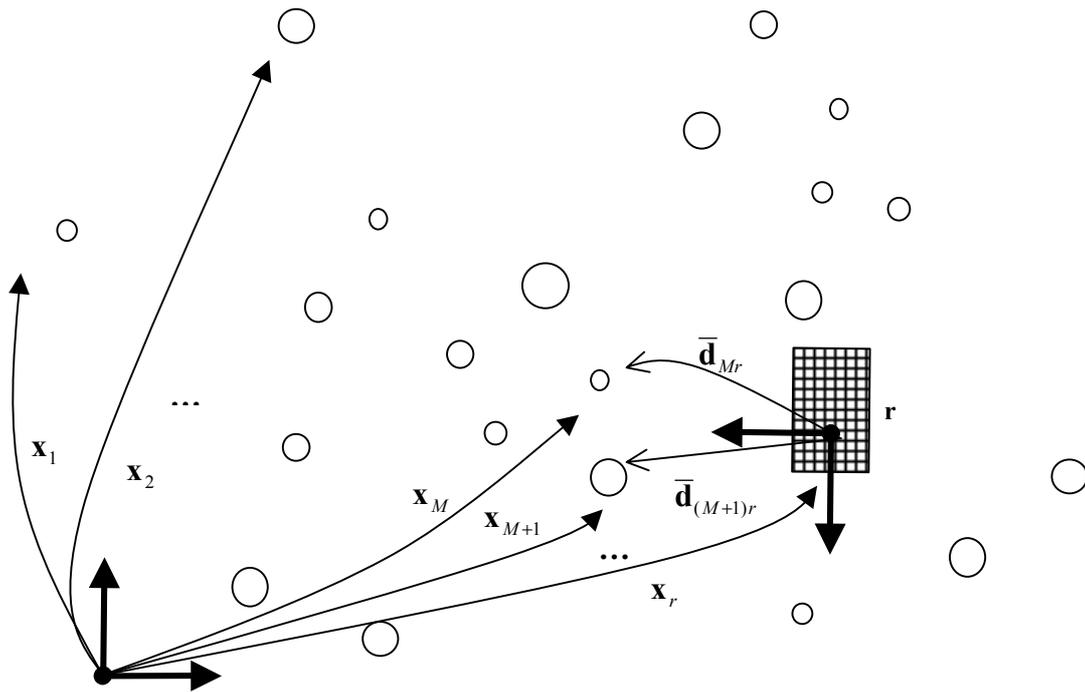


Figure 3.1 An illustration of the *remote observation-global localization* (REMO-GL) approach. The circles denote the landmarks, \mathbf{x}_M describes the estimate of the global location of the M^{th} landmark, \mathbf{x}_r is the estimate of the global robot location and $\bar{\mathbf{d}}_{Mr}$ is an observation of the M^{th} landmark with respect to the robot.

In the figure the symbols $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M, \dots, \mathbf{x}_N$ denote the true and unknown locations of the map objects (features or landmarks) to be estimated by making remote observations thereof with respect to the global location of the mobile sensor platform (such as the autonomous mobile robot). For example, the observation of the M^{th} landmark with respect to the reference frame of the robot, marked with the symbol $\bar{\mathbf{d}}_{Mr}$, is to be combined with the estimate $\hat{\mathbf{x}}_r$ of the current (unknown) location of the robot \mathbf{x}_r to yield the actual measurement of the global location of the landmark.

In the second category of methods, the global location of map objects is estimated through observations on relative relationships between map objects. Characteristic of the published implementations of this approach is that the map objects in the stochastic environment model correspond to past locations of the perception sensor augmented with the corresponding observation data [Lu and Milios, 1995] and [Thrun et al., 1998b]. Sensor locations combined with the corresponding measurement data compose sub-maps that are registered with each other in the maximum likelihood sense. The *relative observation-global localization* (RELA-GL) principle is illustrated in Figure 3.2.

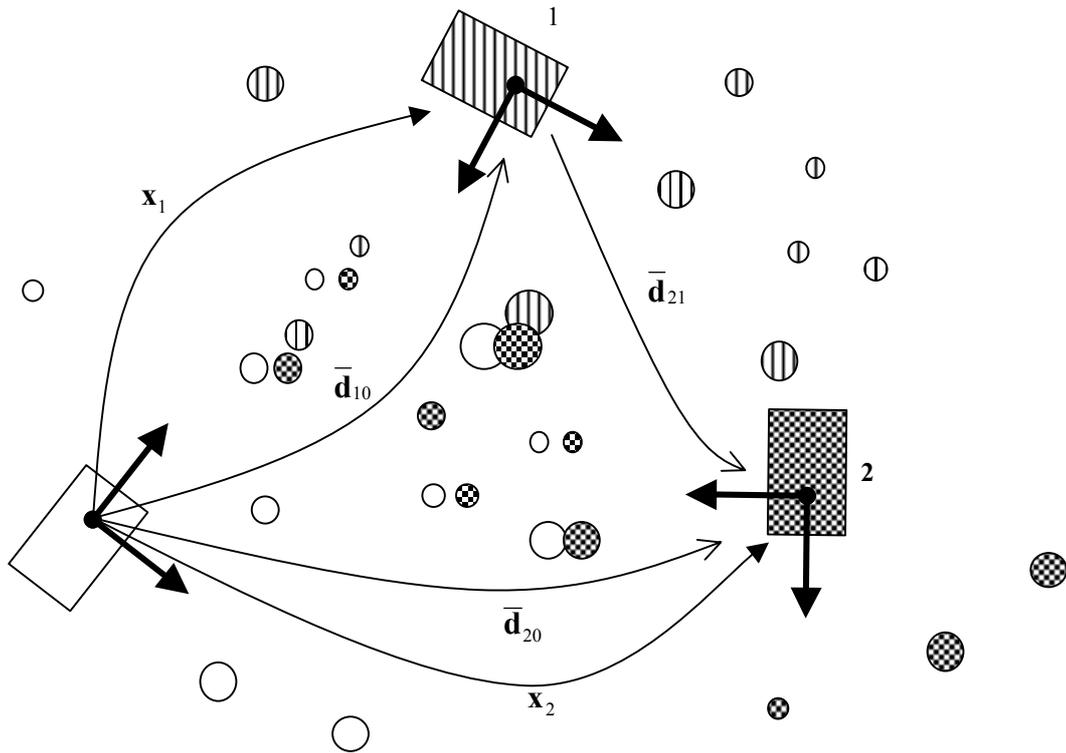


Figure 3.2 An illustration of the *relative observation-global localization* (RELA-GL) approach. The circles of a particular texture describe the feature objects extracted from the measurement data, acquired from the corresponding sensor location, symbolized with the box of the same texture. \mathbf{x}_1 is now the location of the sensor frame of reference of the perception position number one with respect to the global frame of reference, fixed to the first perception position indicated by the number zero. The computed relative alignment (i.e. registration) observations between the perception locations are marked with the symbol $\bar{\mathbf{d}}$.

In the figure, the global reference frame is selected to coincide with startup, the zeroth location of the sensor platform. The state variables to be estimated are the other locations of the mobile platform corresponding to the places where perception data was collected. They are marked with \mathbf{x}_1 and \mathbf{x}_2 in the figure. Measurements, marked by $\bar{\mathbf{d}}_{10}$, $\bar{\mathbf{d}}_{20}$ and $\bar{\mathbf{d}}_{21}$ in the figure, on the relative displacements between the state variables are made by optimally registering overlapping sub-maps. If no registration can be reliably computed between two consecutive measurement locations, an estimate of the relative displacement can also be formed by means of odometry information. Note that, in Figure 3.2, the multiple realizations of each single physical object are represented with the same texture as the corresponding sub-map frame of reference. For the final, unified model of the environment the different, partly overlapping geometric approximations should be merged to yield a similar model structure as depicted in Figure 3.1. Obviously, the *RELA-GL* approach best suits the implementations where the perception sensor remains motionless during the collection of external measurement data. However, it has been applied successfully in a situation, where

an autonomous robot, equipped with a 2D laser range scanner (and 24 sonar sensors), explored an unknown environment [Thrun et al., 1998b].

The connected network of past viewpoint locations form the backbone of the environment model. Within each sub-map the entities of interest, for example raw measurement data points or features extracted from the raw measurement data, are described with respect to the reference frame of the sub-map with known (or bound) uncertainty. A major difference of the *RELA-GL* approach with respect to the *REMO-GL* approach is that the current location of the perception sensor is not included in the map representation. Instead, the history of perception sensor locations is maintained and updated during maximum likelihood map estimation.

In the third category of methods, relative relationships between map objects are both observed and estimated during map building [Durrant-Whyte, 1988] and [Csorba et al., 1997]. A rigorous stochastic treatment of the approach is more challenging than that of the *REMO-GL* approach [Moutarlier and Chatila, 1989]. Moreover, it is not clear how this approach could be applied to yield stochastically consistent estimates of the global locations of map objects [Csorba et al., 1997] and [Csorba, 1997].

The simultaneous localization and map building methods can also be distinguished according to their capability to cope with different kinds of environments. In the stochastic mapping approaches, originally presented in [Smith et al., 1990] and [Moutarlier and Chatila, 1989], the statistics of the map objects are expressed by the approximated expected value (mean) and the approximated mean squared error (covariance). The main restriction, related to the approximation of a probability distribution by the first two moments, is related to the assumption that sufficient statistics for the uncertainty are concentrated around a single peak in the likelihood space. Exploration of environments that contain symmetric locations or featureless spaces requires accurate and reliable odometry information to be available, if only a single expected value together with the related covariance matrix should be robust enough to describe the location estimate of a map object. In real implementations of stochastic mapping, large displacements of the perception sensor between the measurement positions and imprecise odometry information about the displacement may result in the ambiguous association of measured features with corresponding features on the map. If these ambiguities are neglected and only one of the measured object-map object pairs is selected, risk for faulty association and consequent map divergence is considerable.

The Markov localization approach was proposed as a generic-form approximation of the likelihood function corresponding to the *belief* of where the robot might be [Burgard et al., 1996] and [Fox et al., 1999b]. In the original form of metric, Markov localization, the *belief* state of the robot is implemented by dividing the dimensions of the state space of the robot into discrete intervals forming a multidimensional grid. Each grid cell contains a scalar value corresponding to the probability (or belief) that the true and unknown location of the robot is within the envelope of the mobile robot configuration space described by the grid cell. By adjusting the size of the grid cells the desired level of accuracy of the probability density approximation can be achieved. In Figure 3.3, a likelihood distribution grid over the x- and y-coordinates of the relative alignment between two sub-maps is depicted.

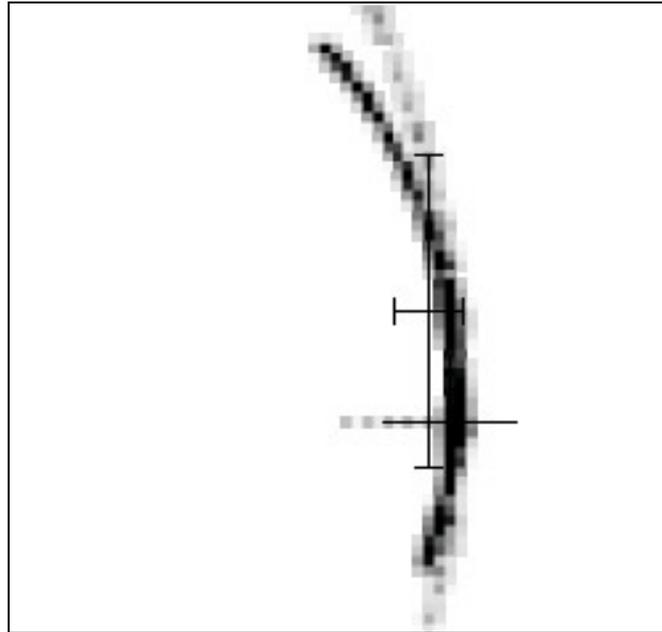


Figure 3.3 Ambiguous (multiple mode) distribution of the registration score density for the relative alignment between sub-maps "6" and "7" during the indoor mapping experiment. The image has been generated by projecting the connected set of four-dimensional registration search space cells onto the x- and y- search plane. The x- and y-coordinates of the discrete search space cell possessing the maximum score value is marked with a simple cross-hair. The x- and y- coordinates of the center of gravity of the score distribution "cloud" is marked with the ended cross-hair the lengths of which describe the sample standard deviation of the cloud. The dimensions of the search space in the figure are 10 meters square.

In the figure, the likelihood of the relative alignment, determined by means of external perception data, is distributed in a complex manner. The ambiguous distribution of the likelihood is due to the characteristics of the geometric features used as reference for the computation of the "match score" for the different relative alignment coordinate hypotheses within the discrete alignment search space. For the sub-map pair "76", the only common features within the two maps were the straight-line segments corresponding to the upper end corner of vertical walls in the office corridor environment. Often, with such kind of features a large number of alignment hypotheses result in relatively good match score value. The widely distributed likelihood could be compressed by means of accurate odometry information. If an accurate initial estimate of the *a priori* displacement of the sensor would be available, it could be convolved with the perception-based registration distribution to yield a single strong candidate (peak) for the relative alignment between the two sub-maps.

Approximation of the entire likelihood score mass by its first two moments does not give a good estimate of the correct registration as indicated by the modified cross hair in Figure 3.3, which designates the center of gravity of the total score mass. On the other hand, selecting the alignment coordinates corresponding to the highest peak center, marked with the simple cross-hair in the figure, would neither necessarily yield the correct answer.

3.2 Stochastic, single hypothesis map representation

In the stochastic formulation of the mapping task, a system state vector with the position of the sensor platform as well as all the positions of the map objects, are maintained. The approach corresponds to the *REMO-GL* formalism depicted in Figure 3.1. Also, the uncertainty values of the position parameters and the correlation dependencies between the state variables are stored and updated within the (Kalman) filter mechanism. Generally, a Kalman filter-based simultaneous localization and map building algorithm (SLAM) works by tracking existing map features and by adding new features to the map as reliable measurements thereof have been acquired. In some cases, features may also be removed from the map, especially if they appear to correspond to dynamic objects.

Kalman filter-based estimation has been applied with great success in practical simultaneous localization and mapping applications [Moutarlier and Chatila, 1989] and [Feder et al., 1998b]. However, there are situations where the risk of divergence in the map increases. These situations are related especially to the ambiguities in data association. If a landmark, extracted from the measurement data, is associated with an existing landmark model corresponding to a different physical object, the computed corrections in the locations of the landmark models are made with erroneous information. This causes the true uncertainties in the state estimates to move outside the estimated error bounds determined by the system covariance matrix. The model becomes overconfident and the estimate diverges [Feder et al., 1998b].

3.2.1 Remote observation-global localization approach

3.2.1.1 Single state space formulation

In the first papers of stochastic, simultaneous localization and map building the position of all map objects including the robot were maintained in a single state vector [Smith et al., 1990] and [Moutarlier and Chatila, 1989]. Moutarlier and Chatila discussed two approaches to represent the spatial relationships among the reference frames of modeled features and the current location of the sensor platform [Moutarlier and Chatila, 1989]. In the first, the "Relation" approach, a state variable is related to another by an uncertain transformation between their reference frames [Durrant-Whyte, 1988]. The result is a network of links that has to be updated, as new data becomes available. In the second, the "Location" approach, the positions and related uncertainties of all map objects are described with respect to a common reference frame. They named the latter approach, which represents the feature and robot positions in a common world frame, as a preferable alternative. The formalism, which is called here the *REMO-GL* approach, was depicted in Figure 3.1.

The stochastic representation of the environment model includes an estimate $\hat{\mathbf{x}}(k)$ of the unknown state variables \mathbf{x} along with the covariance matrix $\mathbf{P}(k)$ modeling the uncertainty, $\boldsymbol{\varepsilon}_x$, related to the estimate,

$$\mathbf{x} = \hat{\mathbf{x}}(k) + \boldsymbol{\varepsilon}_x(k) \quad (3.1)$$

$$\mathbf{P}(k) = E\{\boldsymbol{\varepsilon}_x(k)\boldsymbol{\varepsilon}_x(k)^T\} \quad (3.2)$$

where k can be imagined to be, for example, a discrete time instant or the current "phase" of the mapping process. The states to be estimated include the location of the mobile robot (and possibly other terms such as the velocity of the robot) carrying the perception sensor, $\hat{\mathbf{x}}_r(k)$, and the locations of the geometric features corresponding to some physical objects in the environment, $\hat{\mathbf{x}}_1(k) \cdots \hat{\mathbf{x}}_N(k)$.

$$\hat{\mathbf{x}}(k) = \begin{bmatrix} \hat{\mathbf{x}}_r(k) \\ \hat{\mathbf{x}}_1(k) \\ \vdots \\ \hat{\mathbf{x}}_N(k) \end{bmatrix} \quad (3.3)$$

The covariance matrix, describing the uncertainty of the state estimates and the correlations among them can now be expanded:

$$\mathbf{P}(k) = \begin{bmatrix} \mathbf{P}_{rr}(k) & \mathbf{P}_{r1}(k) & \cdots & \mathbf{P}_{rN}(k) \\ \mathbf{P}_{1r}(k) & \mathbf{P}_{11}(k) & \cdots & \mathbf{P}_{1N}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{Nr}(k) & \mathbf{P}_{N1}(k) & \cdots & \mathbf{P}_{NN}(k) \end{bmatrix} \quad (3.4)$$

Moreover, let us assume that the inclination angles related to the location of the reference frames attached to the environment objects can be determined with a bound precision. Therefore, only the remaining four degrees of freedom of the 3D spatial relationships are to be included into the stochastic model of the environment. For example, the vector of uncertain spatial relationships of the first landmark in the map can be expanded into the following four-dimensional vector:

$$\hat{\mathbf{x}}_1(k) = \begin{bmatrix} \hat{x}_1(k) \\ \hat{y}_1(k) \\ \hat{z}_1(k) \\ \hat{\phi}_1(k) \end{bmatrix} \quad (3.5)$$

where $\hat{x}_1(k)$, $\hat{y}_1(k)$ and $\hat{z}_1(k)$ are the estimates of the unknown (constant) x-, y- and z-coordinates of the first landmark on the map and $\hat{\phi}_1(k)$ the estimate of the unknown (constant) rotation angle of the landmark frame around the z-axis of the global frame of reference. Each sub-matrix of the variance-covariance matrix $\mathbf{P}(k)$ can be expanded into a four times four matrix. For instance, the covariance matrix \mathbf{P}_{r1} has the form (time index (k) omitted for simplicity of presentation):

$$\mathbf{P}_{r1} = \begin{bmatrix} P_{x_r, x_1} & P_{x_r, y_1} & P_{x_r, z_1} & P_{x_r, \phi_1} \\ P_{y_r, x_1} & P_{y_r, y_1} & P_{y_r, z_1} & P_{y_r, \phi_1} \\ P_{z_r, x_1} & P_{z_r, y_1} & P_{z_r, z_1} & P_{z_r, \phi_1} \\ P_{\phi_r, x_1} & P_{\phi_r, y_1} & P_{\phi_r, z_1} & P_{\phi_r, \phi_1} \end{bmatrix} \quad (3.6)$$

where, for example, P_{x_r, x_1} corresponds to the product of uncertainties $\varepsilon_{x_r} \varepsilon_{x_1}$ of the x-coordinates of the robot and first feature locations, respectively. The dimensions of the system state vector and the associated variance-covariance matrix are $4(N+1)$ and $4(N+1) \times 4(N+1)$ respectively, where N is the number of features or map objects in the model. (Note that the velocity term of the sensor platform, if included, will increase the dimensions by four). In the problem of map building, discussed here, the feature objects are assumed to be static and only the location of the robot is assumed to be controllable. However, the state estimation method itself would equally well support dynamic environments, on the condition that models for feature dynamics would be available. So, let us assume that the sensor platform is the only dynamic object in the environment map. The dynamics of the discrete time state space representation of the sensor platform (e.g. a mobile robot) can be expressed with the (non-linear) equation,

$$\mathbf{x}_r(k+1) = \mathbf{f}(\mathbf{x}_r(k), \mathbf{u}(k), \mathbf{v}(k), k) \quad (3.7)$$

where $\mathbf{u}(k)$ corresponds to the control input for transporting the sensor platform to another location between the (time) instants k and $k+1$. The process noise $\mathbf{v}(k)$ is modeled as a zero-mean, white, random sequence with a known covariance matrix $\mathbf{Q}(k)$. It should be noted that $\mathbf{Q}(k)$ describes the "basic" noise level, which will be scaled up or down within the system dynamics model (Equation 3.7). For example, properties of the terrain and the curvature as well as the time constraints of the trajectory may have large effects on how uncertainty is accumulated into the sensor location during the transportation of the sensor from perception position A to perception position B.

Similarly, measurements of the state can be modeled as a (non-linear) function of the state as described by the measurement equation,

$$\mathbf{z}(k) = \mathbf{h}(\mathbf{x}(k), \mathbf{w}(k), k) \quad (3.8)$$

where $\mathbf{w}(k)$ is again assumed to be a zero-mean, independent, random noise sequence with a known covariance matrix, $\mathbf{R}(k)$.

The state space representation of the stochastic map is maintained and updated through the simultaneous localization and map construction mechanism. As new features are observed the state vector and the corresponding covariance matrix are augmented. In the opposite situation, existing map objects, which are not supported by the observation data, although they should be, may be deleted from the map. In order to add the location of the new feature into the state vector, its location with respect to the global frame of reference has to be computed from the relative observation,

$$\hat{\mathbf{x}}_{N+1} = \mathbf{g}(\hat{\mathbf{x}}_r, \mathbf{z}_{new}) \quad (3.9)$$

where the (non-linear) function \mathbf{g} corresponds to the measurement kinematics linking the current estimate of the robot location $\hat{\mathbf{x}}_r$, with the relative observation of the $(N+1)^{\text{th}}$ feature \mathbf{z}_{new} to yield an initial estimate of the location of the $(N+1)^{\text{th}}$ feature with respect to the global frame of reference. The new state vector is then added to the system state vector described by Equation 3.3. The covariance sub-matrices to be added to the system covariance matrix, Equation 3.4, can be calculated as [Hébert et al., 1995],

$$\mathbf{P}_{N+1N+1} = \mathbf{G}_{\mathbf{x}_r} \mathbf{P}_{rr} \mathbf{G}_{\mathbf{x}_r}^T + \mathbf{G}_{\mathbf{z}_{new}} \mathbf{R} \mathbf{G}_{\mathbf{z}_{new}}^T \quad (3.10)$$

$$\mathbf{P}_{N+1i} = \mathbf{P}_{iN+1}^T = \mathbf{G}_{\mathbf{x}_r} \mathbf{P}_{ri} \quad (3.11)$$

where $\mathbf{G}_{\mathbf{x}_r}$ and $\mathbf{G}_{\mathbf{z}_{new}}$ are the Jacobian of \mathbf{g} with respect to the current robot state and to the relative observation of the new feature, respectively.

In the *REMO-GL* map construction method, the robot location is included in the system state vector. This makes the system state correspond to a dynamic stochastic system, which evolves in time. This is the case even when assuming that the geometric features in the environment are static. Therefore, an estimation technique that also considers the change of state variables as a function of time is required. For the convenience of the reader, the principal estimation algorithms, utilized in the state-of-the-art SLAM implementations, are presented in Appendices E, F and G. First the basic Kalman filter algorithm is presented in Appendix E. Then, in Appendix F, the *extended Kalman filter* (EKF) is outlined. Linearization of the non-linear system equations, as part of the application of the extended Kalman filter, may be problematic in practice, especially if the system is composed of many states and is highly non-linear [Julier et al., 1995]. An alternative filtering approach, called the *distribution approximation filter*, will be presented in Appendix G.

3.2.1.2 Relocation-fusion strategy

In conventional stochastic mapping it is assumed that the correlations between the uncertainty in the location of the perception sensor platform and the uncertainty in the location of the landmarks are known. Often, due to the linearization of non-linear system equations and imperfect sensor models, the assumption is not justified and bias is introduced into the state estimates. This phenomenon decreases the quality of the resulting model as the mapping proceeds. To decrease the speed of deterioration of the environment model, Moutarlier and Chatila proposed the sub-optimal, relocation-fusion strategy [Moutarlier and Chatila, 1989]. In the method, the observations of existing map objects are first used to relocate the robot. Only after that the other (unobserved) map objects, that are correlated with the observed map objects and with the robot state, are updated. Similarly, after the relocation phase, new features can be integrated into the map.

3.2.1.3 Decoupled stochastic mapping

In addition to the relocation-fusion strategy presented in the previous chapter, another technique, in which the map objects are reorganized into local frames, can be used to alleviate the consequences of bias. In the method, neighboring map objects are grouped and their global transformation coordinates substituted with coordinates given with respect to a local frame of reference [Betgé-Brezetz et al., 1996]. The direct correlation link between the sensor location values and the landmark coordinates vanishes. However, the locations of the local frames are still modeled with respect to the global reference frame, similar to the location of the perception sensor, and unbound increase of the bias in the location estimates is then possible.

Another motivation for grouping the model data into sub-maps is the decreased computational complexity of stochastic map estimation [Leonard and Feder, 1999a]. For example, in sub-sea navigation applications the operation areas are large and it is difficult to supply the area with artificial beacons. The feature maps built for navigational purposes in such environments easily contain thousands of features. Maintaining all features in a single state vector becomes computationally heavy to update. As a solution to the problem of large scale mapping of unstructured environments Leonard and Feder propose decoupled stochastic mapping strategy where the operation area of the robot is, *a priori*, divided into overlapping regions. At any time, the robot location estimate together with the estimates on the locations of features in the region where the robot is situated, are updated. All the other feature states related to the other regions can be left intact. To traverse between the regions, the information on the state estimates is propagated in a way that aims at maintaining consistent estimates while at the same time maximizing the efficiency of state estimation.

3.2.2 Relative observation-global localization map formulation

The main difference of the *RELA-GL* map formulation with respect to the conventional, *REMO-GL* approach is that the current location of the perception sensor has not been, explicitly, included in the state space representation of the stochastic map. Instead, the history of perception sensor locations is maintained and updated during estimation. In fact, the past sensor locations form the structure of the stochastic map. The actual measurement data, i.e. the substance of the map, is brought to the overall map representation through their spatial relationships with respect to the corresponding (local) perception sensor frame. The *RELA-GL* map formulation was depicted in Figure 3.2.

Let us assume that the number of perception positions, where observation data has been collected is $(N+1)$. The state vector of the stochastic map can now be expressed as,

$$\hat{\mathbf{x}}(k) = \begin{bmatrix} \hat{\mathbf{x}}_1(k) \\ \vdots \\ \hat{\mathbf{x}}_N(k) \end{bmatrix} \quad (3.12)$$

where the state sub-vectors correspond to the past perception sensor locations at discrete times, indicated by the indexes 1 to N . The spatial location where the first set of perception data was measured is given the index zero. This location is chosen as the global frame of reference of the environment map. All the other perception positions are described with a spatial relationship with respect to this global frame (compare Figure 3.2). Time index N corresponds to the sensor location where the most recent perception data was collected. The corresponding covariance matrix can now be written as,

$$\mathbf{P}(k) = \begin{bmatrix} \mathbf{P}_{11}(k) & \mathbf{P}_{12}(k) & \cdots & \mathbf{P}_{1N}(k) \\ \mathbf{P}_{21}(k) & \mathbf{P}_{22}(k) & \cdots & \mathbf{P}_{2N}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{N1}(k) & \mathbf{P}_{N2}(k) & \cdots & \mathbf{P}_{NN}(k) \end{bmatrix} \quad (3.13)$$

As discussed already in Chapter 3.1, this kind of mode of operation is best suited for implementations where the sensor (or the robot carrying the sensor) stops to collect the perception data for the 3D model construction. A separate filter, using odometry information from the wheel encoders of the robot platform, carries out the estimation of the sensor motion in between the perception positions. This estimate of the relative displacement between two consecutive perception positions can be utilized as the initial guess for the map object-based registration of the perception positions. Or, if the correct registration cannot be determined with high probability, then the odometry-based estimate itself can be used as the relative observation of the spatial relationship between the two consecutive perception positions.

By definition, the states in the state vector, corresponding to the past perception positions, do not evolve as a function of time. Consequently, the dynamics equation, Equation 3.7, can be rewritten as,

$$\mathbf{x}(k+1) = \mathbf{x}(k) \quad (3.14)$$

where \mathbf{F} corresponds now to the identity matrix which is not shown in the equation above. The values that are observed are the relative displacements between the map objects. Map objects correspond here to the sub-maps composed of the position of the sub-map frame with respect to the global frame of reference and the geometric data defined with respect to the sub-map frame. The relative location of the sub-map frame i with respect to sub-map frame j is described as $\mathbf{d}_{ij} = \mathbf{x}_i - \mathbf{x}_j$. An observation of this relative relationship is denoted by $\bar{\mathbf{d}}_{ij}$ [Lu and Milios, 1995]. For the linear measurement equation, Equation 3.8, the coefficient matrix \mathbf{H} will contain 0, -1 or 1 on all entries. Let us take the three-location setup, depicted in Figure 3.2, as an example. The observations $\bar{\mathbf{d}}_{10}$, $\bar{\mathbf{d}}_{20}$ and $\bar{\mathbf{d}}_{21}$ can now be embedded into the measurement equation yielding

$$\mathbf{z}(k) = \begin{bmatrix} \bar{\mathbf{d}}_{10} \\ \bar{\mathbf{d}}_{20} \\ \bar{\mathbf{d}}_{21} \end{bmatrix} = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{w}(k) \quad (3.15)$$

which can be further expanded,

$$\begin{bmatrix} \bar{d}x_{10} \\ \bar{d}y_{10} \\ \bar{d}z_{10} \\ \bar{d}\phi_{10} \\ \bar{d}x_{20} \\ \bar{d}y_{20} \\ \bar{d}z_{20} \\ \bar{d}\phi_{20} \\ \bar{d}x_{21} \\ \bar{d}y_{21} \\ \bar{d}z_{21} \\ \bar{d}\phi_{21} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \end{bmatrix} [x_1 \ y_1 \ z_1 \ \phi_1 \ x_2 \ y_2 \ z_2 \ \phi_2]^T + \mathbf{w}(k) \quad (3.16)$$

In the equation, $\bar{d}\phi_{10}$ is an observation of the heading angle of the first sub-map frame with respect to the zeroth sub-map frame (i.e. with respect to the global frame of reference). In this case, $\bar{d}\phi_{10}$ itself is an observation of the unknown parameter ϕ_1 , which is to be estimated. On the other hand, $\bar{d}\phi_{21}$ is an observation of the difference in the unknown orientation angles ϕ_2 and ϕ_1 . In our case, the uncertainty related to the measurements can be determined separately for each observation. This is due to the fact that the observations, $\bar{\mathbf{d}}_{ij}$, are generated by applying a sub-map registration method that gives an estimate of the precision related to the computed relative alignment coordinates. The registration methods will be discussed in Chapter 5.

Now, let's assume that the uncertainty of the observation $\bar{\mathbf{d}}_{ij}$ can be approximated with a Gaussian distribution with covariance \mathbf{C}_{ij} and is independent of the uncertainties related to the other observations. The cost function over the entire set of observations, which is a summation of the squared observation errors weighted by the related observation uncertainty, can be written as the following Mahalanobis distance [Lu and Milios, 1995],

$$W = \sum_{(i,j)} (\bar{\mathbf{d}}_{ij} - \mathbf{d}_{ij})^T \mathbf{C}_{ij}^{-1} (\bar{\mathbf{d}}_{ij} - \mathbf{d}_{ij}) \quad (3.17)$$

which can be rewritten as,

$$W = \sum_{(i,j)} (\bar{\mathbf{d}}_{ij} - (\mathbf{x}_i - \mathbf{x}_j))^T \mathbf{C}_{ij}^{-1} (\bar{\mathbf{d}}_{ij} - (\mathbf{x}_i - \mathbf{x}_j)) \quad (3.18)$$

The cost function W can be expressed in matrix form,

$$W = (\bar{\mathbf{d}} - \mathbf{H}\mathbf{x})^T \mathbf{C}^{-1} (\bar{\mathbf{d}} - \mathbf{H}\mathbf{x}) \quad (3.19)$$

Minimizing the equation above gives the maximum likelihood or minimum variance estimate [Bar-Shalom and Fortmann, 1988]. The estimate for the global position of the sub-maps that minimizes the cost function W is given by,

$$\hat{\mathbf{x}} = (\mathbf{H}^T \mathbf{C}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{C}^{-1} \bar{\mathbf{d}} \quad (3.20)$$

And the associated covariance is computed as,

$$\mathbf{C}_x = (\mathbf{H}^T \mathbf{C}^{-1} \mathbf{H})^{-1} \quad (3.21)$$

The minimum variance estimation method presented above is a batch process, solving all the location variables at the same time. Every time the perception sensor is moved to a new location, a new sub-map is created. The new sub-map is then aligned with the existing neighboring sub-maps. The observations on relative alignments which are judged as being correct with high probability, are augmented to the $\bar{\mathbf{d}}$ vector and \mathbf{H} matrix while increasing their dimensions correspondingly. The equations above are then solved again to produce the new optimal estimates of the global sub-map locations.

3.3 Maximum likelihood, multiple hypothesis map estimation

The main drawback of the application of the stochastic filtering algorithms, such as the extended Kalman filter (EKF) or the *distribution approximation filter* (DAF) [Julier et al., 1995], into map construction is the fact that they assume that uncertain information can be modeled accurately with a single expected value together with the related covariance. Ambiguous operation environments and large prediction (odometry) errors may produce complex, multi-modal distributions of probabilities which cannot be easily handled by an estimation method that approximates the probability distributions by means of the first two moments. Numerical approximation methods, with which the distribution of information about the unknown parameters can be modeled up to the desired precision and extent, seem to be more appropriate to solve the problem.

In the base line SLAM implementations, new information is fused into the map once and is then forgotten. These implementations correspond to the REMO-GL approach presented in Chapter 3.1. While exploring an unknown environment, systematic error may accumulate to the robot location and consequently also to the map. When re-observing landmarks mapped at different stages of the exploration process, ambiguous data association situations may emerge, which cannot, algorithmically, be solved by a REMO-GL approach. The reason is that REMO-GL type methods are unable to revise measurements backwards in time [Thrun et al., 2000a].

The alternative, more generally applicable approach extends the SLAM methodology in these two respects [Thrun et al, 1998a], [Thrun et al, 1998b], [Thrun et al, 2000a]. The goal of the method is to find the most likely integrated map given the data,

$$M^* = \arg \max_M p(M|d) \quad (3.22)$$

where M is an integrated map built by porting the geometric data from the sub-maps into a common frame of reference. d corresponds to an alternated sequence of *a priori* alignment information and data from the external perception sensor,

$$d = \{p_0, \mathbf{u}_0, p_1, \mathbf{u}_1, \dots, p_{N-1}, \mathbf{u}_{N-1}, p_N\} \quad (3.23)$$

where p_i is a set of 3D measurement points used for computing the optimal alignment between a sub-map and an integrated map and \mathbf{u}_i is the *a priori* estimate of the displacement from the i^{th} perception position to the $(i+1)^{\text{th}}$ perception position. A set of viewpoint registration data points can be described as,

$$p_i = \{\mathbf{p}_{i1}, \dots, \mathbf{p}_{iR}\} \quad (3.24)$$

where R is the number of 3D point vectors in the i^{th} sub-map. The maximum likelihood function, Equation 3.22, can be re-written as,

$$M^* = \arg \max_M \int \prod_{i=0}^N p(p_i|M, \mathbf{x}_i) \prod_{i=0}^{N-1} p(\mathbf{x}_{i+1}|\mathbf{u}_i, \mathbf{x}_i) d\mathbf{x} \quad (3.25)$$

where the integral is over all possible viewpoint location combinations in the discrete transformation space. In the equation, $p(p_i|M, \mathbf{x}_i)$ is the perception model and $p(\mathbf{x}_{i+1}|\mathbf{u}_i, \mathbf{x}_i)$ describes the statistics of the prediction for the displacement between consecutive viewpoint locations.

3.3.1 Probabilistic task models

A probabilistic model is required to describe the statistics on the estimate of the predicted sensor location when moving from the current perception position to the next one. In order to correct the predicted location by means of measurement data from the external perception sensors, such as an ultrasonic or a laser range finder, a perception model is required.

3.3.1.1 Prediction of sensor motion

While moving between perception positions, uncertainty accumulates in the state estimate of the sensor location. The amount of uncertainty depends, for example, on terrain properties and traveled distance. Now, let us discuss the accumulation of uncertainty by means of an artificial example. Assume that a mobile robot is following the walls of a rectangular room. While the robot is going forward, uncertainty is accumulating in the estimate of the traveled distance in proportion to the distance. As a consequence, the uncertainty in the x- and y- coordinates increases correspondingly, depending on the direction of the straight line

movement. Similarly, when turning at the corners, uncertainty accumulates into the estimate of the heading angle of the robot. In Figure 3.4, the distribution of uncertainty with respect to x- and y-coordinates is depicted. The distributions have been realized by a set of 5000 samples propagated according to the robot motion commands and according to the predefined percentage of accumulated uncertainty per distance traveled or angle turned. The robot was commanded to return to the location from where it started.

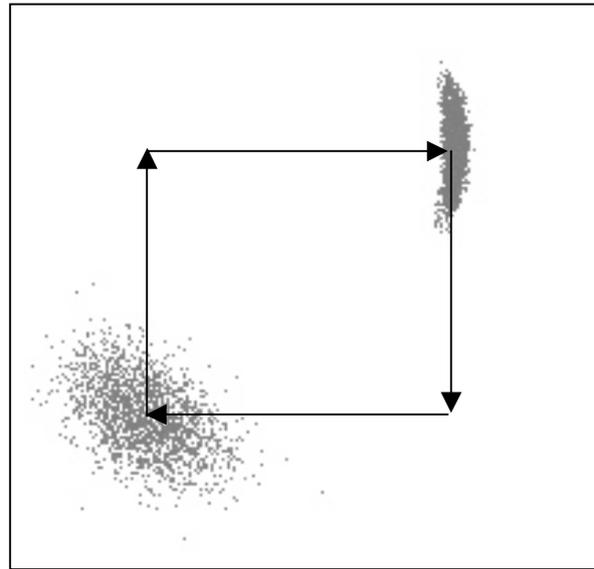


Figure 3.4 The evolution of uncertainty in the robot location while following the walls of a rectangular room. The distribution of uncertainty was approximated by means of 5000 samples. When the robot started at the lower left corner of the room its location was known (modeled by a point (Dirac delta) distribution).

In practice, it is often difficult to create an accurate model for the accumulating uncertainty. This is true especially while moving in natural outdoor environments.

3.3.1.2 Perception model

The perception model, marked as $p(p_i|M, \mathbf{x}_i)$, describes the likelihood of external measurement data with respect to a given sensor location hypothesis and existing environment map. An example of likelihood score distributions given by the perception model, utilized in this work, was illustrated in Figure 3.3.

3.3.1.3 Approximation of multi-modal probability distributions

As multiple, good candidates for a Gaussian location estimate are computed, it should be carefully considered whether the best among them should be selected and the other be discarded. A better alternative might be to preserve them all until new measurement data makes it easier to select the correct one among the good candidates. Austin and Jensfelt propose a tree structure to maintain the history of multiple hypotheses for the location of a mobile robot [Austin and Jensfelt, 2000].

Earlier, multiple Gaussian distributions have been utilized, for example, to track multiple target hypotheses [Bar-Shalom and Fortmann, 1988].

Another technique to approximate multi-modal probability distributions is probability grids. Burgard et al. used a position probability grid to estimate the absolute position of a mobile robot [Burgard et al., 1996]. The method was able to handle uncertain sensor information and multiple position hypotheses. An example of the grid approximation of a localization score distribution was depicted in Figure 3.3.

The third alternative to approximate a multi-modal probability distribution is to use a set of samples drawn from it. The method has been used to propagate conditional position estimate densities for contour tracking [Isard, 1998]. For mobile robot localization, sample based density approximation has been proposed in [Dellaert et al., 1999a] and [Fox et al., 1999a]. The application of sample based density approximation for SLAM has been discussed in [Thrun et al., 2000a]. An illustration for using samples to approximate the distribution of uncertainty in a location estimate was given in Figure 3.4.

3.3.2 Maximum likelihood map generation

3.3.2.1 Globally optimal map from Gaussian approximation of uncertainties

If the uncertainty, related to the perception and motion models in Equation 3.25, can be modeled as Gaussian distributions and if the observations of the relative alignment coordinates among viewpoint positions are generated from the external measurement data, the maximum likelihood map can be computed as a weighted least squares solution for the connected network of viewpoint (sub-map) registration transformations. The method, which was given the name *relative observation-global localization* (RELA-GL), was discussed in Chapter 3.2.2.

3.3.2.2 Expectation-maximization algorithm for map tuning

In a generic case, computation of the integral in Equation 3.25 for all possible maps would be computationally challenging. However, there exists an efficient technique for an iterative search of a local maximum within the space of possible map configurations. The algorithm, called Expectation-Maximization (EM), is a hill climbing routine in likelihood space, which alternates the two steps: the Expectation step and the Maximization step. In the context of SLAM the E-step corresponds to an estimation of the viewpoint locations by means of the *a priori* alignment information and the current map. Thereafter the M-step is executed, which re-computes the map by using the external perception data and the viewpoint locations, computed during the E-step.

The E-step computes estimates for the viewpoint locations by means of data collected both before and after the time when the robot, or equivalently, the perception sensor, was at the current location under consideration. Mathematically, this can be expressed as the normalized product of two terms [Thrun et al., 1998b],

$$p(\mathbf{x}_i|d, M) = \eta p(\mathbf{x}_i|p_0, \mathbf{u}_0, \dots, p_i, M) p(\mathbf{x}_i|\mathbf{u}_{i+1}, \dots, p_N, M) = \eta \alpha(i) \beta(i) \quad (3.26)$$

where η is a normalizer ensuring that the left hand side of the equation sums up to one. For each viewpoint location \mathbf{x}_i , the terms $\alpha(i)$ and $\beta(i)$ are computed separately. The former corresponds to a normal (Markov) localization step where an estimate for the location is computed by means of past measurements. The latter term is computed by means of measurement data that was collected at later points of time. Indeed, the $\beta(i)$ term adds in the extra capability to be able to build maps of environments where multiple, ambiguous links between sub-maps may emerge. It should be noted that the same principle of utilizing past and future measurements is realized also by the RELA-GL mapping method, presented in Chapter 3.2.2.

The M-step aims at computing the optimal integrated map given the perception data and the viewpoint locations, which corresponds to maximizing $p(M|\mathbf{x}_0, p_0, \dots, \mathbf{x}_N, p_N)$. During the next E-step, the estimates for the viewpoint locations are then computed by means of the new integrated map M . Note that in the RELA-GL approach, the neighboring sub-maps are always aligned with respect to each other, not with respect to an integrated map. The integrated map can be computed at the end of the mapping process above the "skeleton" consisting of the global locations of the sub-map frames yielded by the RELA-GL method.

3.3.1 Map topology

In real environments, multiple conflicting map hypotheses may be encountered. The modeling process should be able to weight the different hypotheses and select the best one or put them into an order of increasing likelihood. In practice, treatment of several map hypotheses can be considered a data administration problem. The situation occurs as multiple, good candidates for the relative alignment between the objects, currently extracted from the measurement data, and the existing map are identified. Each of the relative alignment hypotheses would correspond to a different location in the existing map whereto the new map data could be integrated. To handle the ambiguity with map construction, a new "map track" for each probable map hypothesis could be initiated. As new observations are acquired, map hypotheses that are supported by the data are maintained, while the others can be deleted.

Chapter 4

Feature extraction from three dimensional measurement data

4.1 Introduction

Segmentation of the measurement data is typically the first step in feature extraction. Data segmentation corresponds to the grouping of neighboring data points into homogeneous spans or regions. Homogeneity can be defined, for example, as (nearly) equal distance from the perception sensor or similar curvature of the surface computed locally by groups of neighboring data points.

In addition to data segmentation, feature extraction usually considers the “mining” of more abstracted information from the measurement data, grouped within the boundaries of a segment, as well. With environment mapping, the abstracted information is often the geometric shape of the object probed by the data points within the segment. The geometric shape or the realization of the parameters of a geometric model prototype (such as a polynomial equation) can be determined by fitting the model to the measurement data. A good fit will then speak for the assumption that the geometry of the visible object surface can be approximated with the given geometric model. In general, feature extraction can be described as the condensation of the original set of geometric signals into a compact and reliable geometric description of the objects’ visible shape [Hébert et al., 1994].

As an example, let us imagine that the task is to find out whether in some particular office environment cylindrical, vertical supporting columns exist. Further, assume that a scanning (with horizontal and vertical scanning mechanisms) laser range finder is used to collect 3D measurement data. Now, the task can be divided into two phases. First, the data of each overlapping, horizontal

line scan is segmented to sets of consecutive measurement points having a similar distance from the laser scanner. Thereafter, those segments having an approximated width, let's say, from 0.1 to 1 meters are selected for further analysis. Groups of similar size segments are then sought after in the vertical direction. Any overlapping set of segments might originate from a cylindrical shape, vertical supporting column. However, in an office environment, other architectural features, such as the doorframes, can generate similar range profiles and can, consequently, be confused with the supporting columns. Also, specular reflections can contribute to the (false) detection of a narrow elongated object within the range data. A range image taken from the corridor of the premises of the Automation Technology Laboratory is shown in Figure 4.1. The sector spanned by the image is 324 degrees horizontally and 77.4 degrees vertically. The resolution of the image is 3000x720, yielding altogether 2.16 million measurement points.

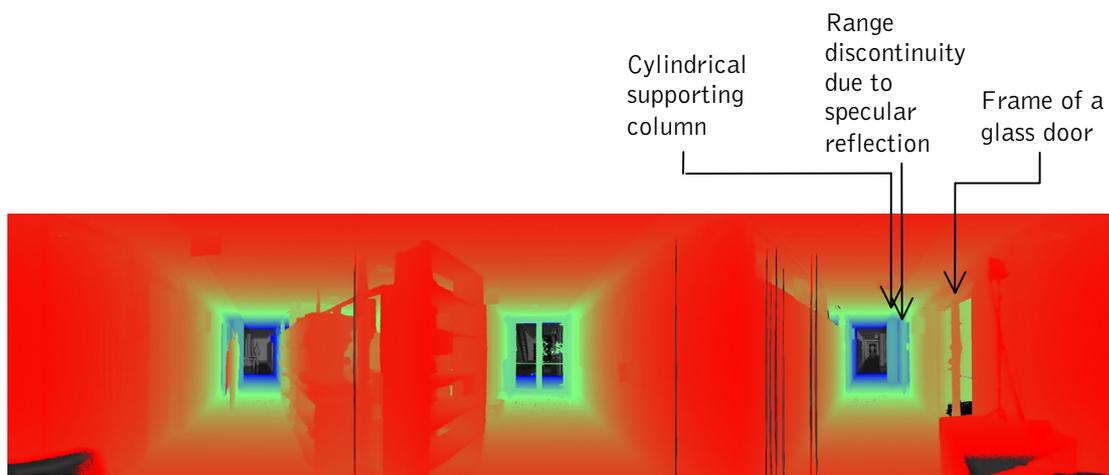


Figure 4.1 A wide angle, 324x77 degree range image acquired at the intersection of two corridors. In the image, the measured distances from one to 10 meters have been pseudo-colored from red to blue, respectively.

The black stripes in the image correspond to people passing by the range scanner at a distance less than one meter from the sensor. The acquisition of the image with the Riegl LMS-Z210 scanner took about 3 minutes and 40 seconds.

In order to find out whether the underlying object can be approximated (with relatively good precision) by a cylinder model, the data within each selected horizontal segment is fitted into a geometric circle model. By analyzing the distribution of the signs of the residual fit errors of the measurement points with respect to the estimated model, information on whether the true underlying object from which the data points originate is cylindrical can be gained. A similar procedure, called the region test, was used to determine the proper order of a polynomial surface model to be fitted to the segmented range data [Besl and Jain, 1988]. In Figure 4.2(a) an artificial example of a horizontal measurement profile for a cylindrical supporting column, such as the one marked in Figure 4.1, is illustrated together with the fitted circle model. In Figure 4.2(b) the distribution of the distance deviations between the measurement points and the circle contour are sketched. In Figure 4.2(c) and 4.2(d) the corresponding results are presented in the case where the underlying object is the doorframe, such as the one pointed out in Figure 4.1.

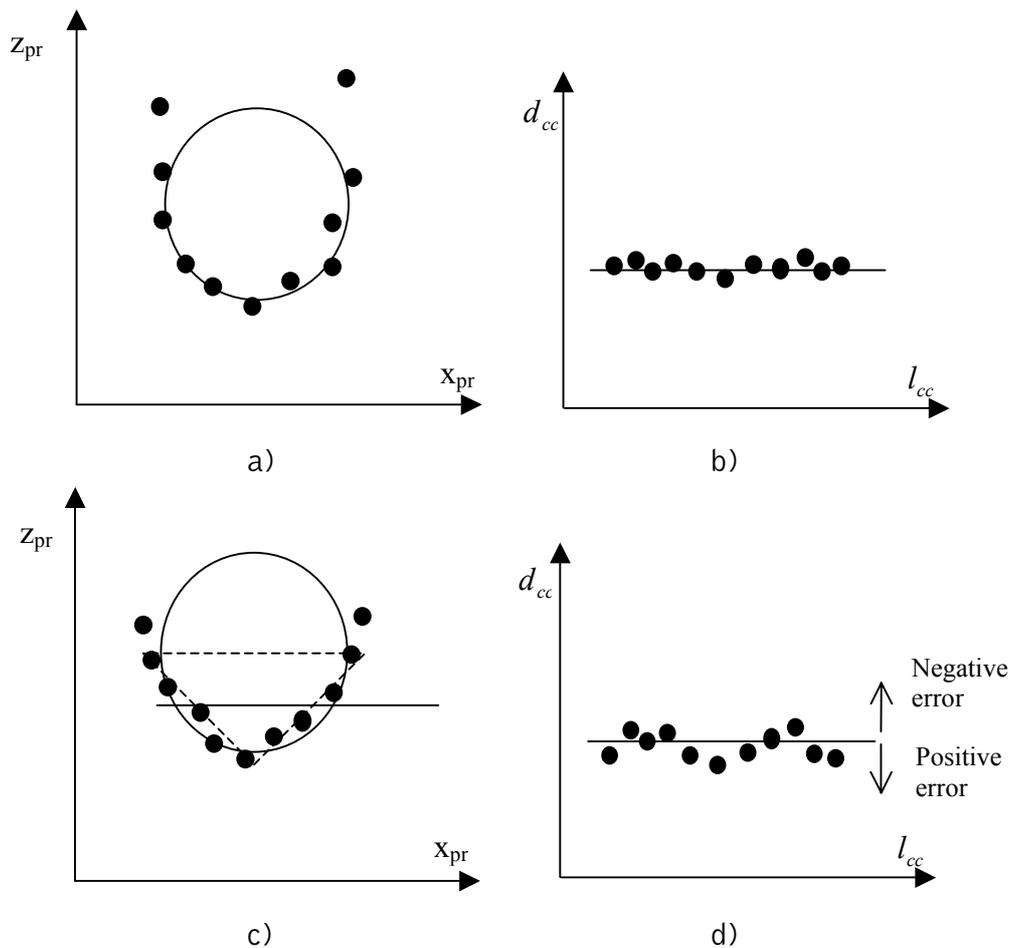


Figure 4.2 Analysis of the error between measurement points and the fitted model contour when fitting range data to a true cylindrically shaped object is demonstrated in images a) and b). The corresponding results for a target (marked with a dashed line) having a rectangular cross-section are presented in images c) and d). In images b) and d), d_{cc} corresponds to the distance from the circle center and l_{cc} corresponds to the distance along the half of the circle contour facing the sensor. The distance of the estimated circle contour from the circle center is indicated with the horizontal line.

The difference in the distribution of the signs of the residual fitting errors is substantial and can be utilized to determine the appropriateness of the geometric type hypothesis with respect to the underlying object image data.

Feature extraction, discussed in this work, emphasizes (optical) measurement techniques and related feature extraction methods, where the measured points are (randomly) scattered over the object surface. In practice, this means active sensing methods where energy is transmitted towards the target object, and the (range) measurements are computed as a function of the backscattered signal. Passive methods, such as stereovision, are only briefly outlined.

In what follows, different perception technologies for the extraction of three dimensional measurement data are discussed in Chapter 4.2. Then, the basic

approaches used for the segmentation of range data into homogeneous regions are discussed in Chapter 4.3. In the following two chapters, geometric models to describe objects' visible shape in structured indoor and natural outdoor environments, respectively, will be presented. In Chapter 4.6, our method for creating a cylinder segment model for narrow elongated objects, such as tree trunks, will be introduced. And finally, a method, which utilizes the 3D occupancy grid to extract straight line features corresponding to the upper corners of building walls, is presented.

4.2 Perception sensors for 3D data acquisition

When weighting up different perception systems, the spatial extent for acquiring 3D measurement data and the precision of the data are of primary importance. For example, the time-of-flight measuring principle utilized in many laser range finder systems allows a large measuring depth, which is a function of the power of the laser pulse. The range-measuring uncertainty is also rather insensitive to the measuring range. Another example is stereo camera systems, which utilize the triangular configuration of the sensors (i.e. the cameras) and the target to measure depth. With the measurement principle the uncertainty related to the measured distance increases proportionally to the distance.

The uncertainty of the "raw" measurement data is the starting point for determining the uncertainty for the extracted geometric features. In the limit, if the features are estimated with an unbiased estimator from an infinite set of measurements, whose uncertainty corresponds to a white, zero mean, random noise sequence, the uncertainty of the features will approach a very small number (zero). In such a case, the uncertainty value of the approximated feature parameters will, in practice, be well below the uncertainty to rise from other sources during the process of mapping a large-scale environment.

4.2.1 Triangulation-based measurement systems

Triangulation based measurement systems get their name from the geometry of the measurement setup, where the two passive receivers (dual camera stereo vision system) or the transmitter and the receiver (active stereoscopic system) and the measured location form a triangle. As an example, let us consider the active triangulation based measurement system shown in Figure 4.3.

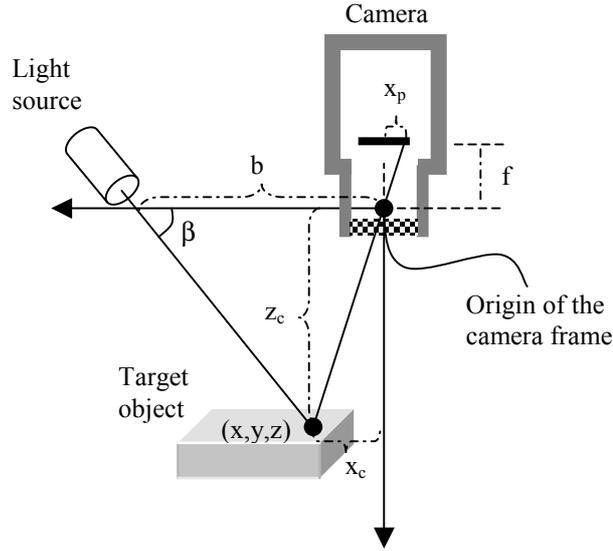


Figure 4.3 A simple, active triangulation-based measurement system.

In the figure, f is the focal length of the camera, x_p is the position of the light spot on the image plane of the camera, b is the distance of the light source from the camera and β is the angle of the light beam with respect to the x-axis of the camera frame. The coordinates of the measured point with respect to the camera frame of reference are given by x_c and z_c . The y-coordinate of the measured point, with respect to the camera frame, is assumed to be zero. Now, the equations for the x- and z- coordinates can be formed from the measurement triangle,

$$z_c = \frac{b}{\frac{x_p}{f} + \frac{1}{\tan \beta}} \quad (4.1)$$

and

$$x_c = z_c \frac{x_p}{f} = \frac{b}{1 + \frac{f}{x_p \tan \beta}} \quad (4.2)$$

The measurement "window", within which the target must be located in order to have the light spot hitting its surface and being visible in the camera image, is quite narrow. If the size of the measurement "window" is increased along the optical z-axis of the camera by increasing angle β , it will happen at a cost of decreased measurement resolution along the same axis. To find the sensitivity of the measurement resolution to the measurement geometry, the formula for the z-coordinate, Equation 4.1, can be differentiated with respect to x_p yielding,

$$\frac{\partial z_c}{\partial x_p} = \frac{\frac{b}{f}}{\left(\frac{x_p}{f} + \frac{1}{\tan \beta}\right)^2} \quad (4.3)$$

Now, if angle β is increased in order to increase the length and distance of the measurement window along the z-axis, the resolution will decrease proportionally to the second power of $\tan \beta$. From Equation 4.3 it can be also noted that the resolution (and uncertainty) of the measured distance is proportional to the square of the distance.

In general it is quite problematic to use such a triangulation based measurement system, especially if the precision and stretch requirements of the measurement task vary from case to case. Moreover, it may be difficult to plan the perception task such that both the light source and the camera catch sight of the target at the same time. This is the case especially in cluttered environments. With respect to the autonomous mobile robots, the most appropriate application fields for triangulation-based 3D perception include manipulation tasks and environment modeling within ranges of a few meters.

4.2.2 Single signal path-based range measurement

Single signal path based range measurement techniques do not suffer from the complications of the triangulation-based measurement systems related to the usable measurement range. If the target is visible from the current location of the sensor, its range can, in principle, be determined. The factors, which determine the upper limit for the measurable range, are the reflectance characteristics of the target surface and the power of the laser beam. The range measurement principle, based on quasi-single signal path, is illustrated in Figure 4.4.

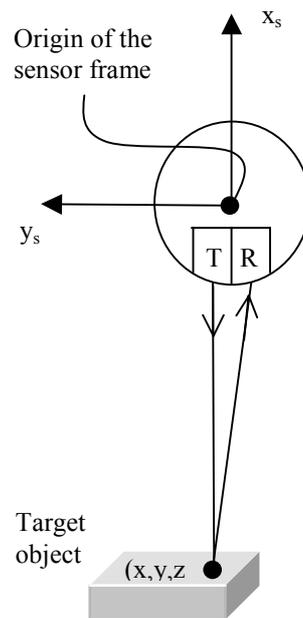


Figure 4.4 Range measurement geometry of the Riegl LMS-Z210 range imaging sensor.

In the Figure, the signal transmitter "T" and receiver "R" are two separate units mounted side by side. Range measurement techniques available for this category of devices include the time-of-flight of the laser pulse and phase shift between transmitted and received (continuous) signals [Everett, 1995].

4.2.3 Fusion of range data and camera image for 3D perception

The strength of single signal path-based range measurement devices is in the capability to accurately evaluate the range to the hit point of the laser beam. The drawbacks are related to the mechanical system rotating the beam around one or two axes. The angular resolution of such systems is relatively poor and acquisition times easily lengthen for dense coverage of the target environment. These difficulties related to the mechanical scanner systems can be alleviated by replacing the mechanical scanner with a camera for determining the object boundaries in angular direction [Juujärvi et al., 1998] and [Byrne and Singh, 1998]. Common to these methods is the sensor which is directed manually towards the target for distance measurement. The angular width is then determined automatically from the camera image. Both of the implementations were developed to determine the volumetric size of a tree, required for forest inventory. The principle for determining the circular cross-section of a tree from range and camera image data is depicted in Figure 4.5.

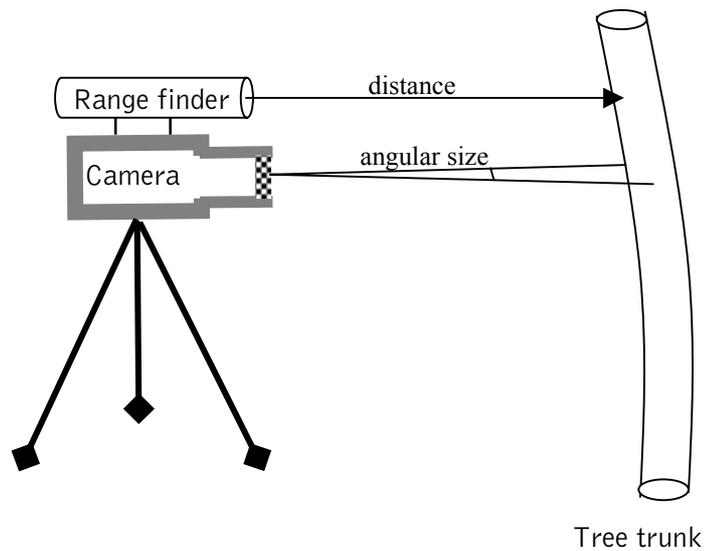


Figure 4.5 Manually manoeuvred tree size measuring system.

A more automated approach for the fusion of intensity and range images for 3D shape recognition was proposed in [Umeda et al., 1996]. The method is based on the assumption that a planar surface, which passes through the optical centre (i.e. the pin hole in the pin hole model of a camera) and the extracted straight line segment on the camera image, can be created. Moreover, the straight line segment is assumed to belong to the contour of a planar surface, for which a 3D plane model has already been estimated by means of range data. Now, the intersection of these two planes yields an estimate of the 3D straight line segment corresponding to the contour of the planar object surface. A similar, semiautomatic method for quasi-coaxial camera-range finder system was proposed in [Forsman, 1994].

4.2.4 3D perception through motion

Sensor movement can be utilized to construct a 3D representation from lower dimensional perception data. A three-dimensional model can be built by accumulating 2D perception data along the sensor path [Thrun et al., 2000a]. In the method, the robot is equipped with both a 2D laser range finder scanning forward, in a horizontal direction, and another 2D scanner tilted 90 degrees and scanning upwards. The system has been used to construct a 3D model for an indoor environment, by first estimating the maximum likelihood path for the robot and corresponding maximum likelihood map in two dimensions by using the forward-looking scanner. The 3D model was then built by integrating the measurement data from the upward-looking scanner into a multi-polygon surface model.

Another way to utilize motion to acquire 3D environment geometry is by the integration of motion between image acquisitions for the small displacement determination. The estimated displacement and the pair of images form a triangular configuration for which the same mathematics, as used in acquiring 3D geometry from a pair of stereo cameras, can be exploited. However, in this case,

the displacement is not rigid and is therefore known with limited precision, which has to be accounted for when estimating the uncertainty of the acquired geometry.

A limitation of both of the approaches outlined above is related to the requirement that the sensor must pass through the space that is to be mapped. This is due to the fact that the first approach uses the upward looking scanner to acquire 3D data at the current location of the robot. Similarly, the second approach, which utilizes optical flow for image acquisition, is most effective in mapping the parts of the space which are located perpendicular to the current trajectory of the robot.

4.3 Segmentation of range data

The problem of range image segmentation can be informally defined as “the process of labeling the pixels whose measurements are from the same surface with the same label” [Hoover et al., 1996]. Segmentation of a range image (2D case) or a range data sequence (1D/scan line case) is often the first step in the process of 3D environment perception. Therefore, the different characteristics of different segmentation methods with respect to “real world” segmentation problems may have different relative importance in different applications. For example, the tendency to fail to find an existing segment in a range image (false negative result) is not a desirable property of a range image based collision avoidance system. On the other hand, sensitivity to finding a landmark on a location where no landmark exists (false positive result) might confuse the functioning of a landmark-based localization system whereas, for collision avoidance, it is not, in general, too dangerous to see occasional phantom obstacles. So, the application framework dictates the criteria for selecting the most suitable segmentation method.

Most of the published range data segmentation methods expect outlier free data. Consequently, “abrupt” range discontinuities are always labeled as true segment borders. This may be a justified assumption in structured environments. In cluttered (outdoor) environments, however, filtering of non-surface data points is required. The subject will be discussed in Chapter 4.6.1.

Range image segmentation methods utilize different techniques for grouping the range measurements into segments. One of the main approaches is region growing, in which the segments are iteratively built up from the seed regions. Usually, the growing process is carried out pixel by pixel [Besl and Jain, 1988] and [Hoover et al., 1996], but another approach, which utilizes the scan line structure of the range image in the segment growing process has been presented as well [Jiang and Bunke, 1994]. These two region-growing approaches will be discussed in the following two chapters.

4.3.1 Pixel-based region growing by means of variable order surface fitting

The algorithm begins by computing the mean curvature and the Gaussian curvature across the range image. These curvature values can be combined to yield scalar surface features that are invariant to rotations, translations and changes in parameterization. The eight fundamental, viewpoint independent surface types that can be characterized by using only the mean curvature and the Gaussian curvature are [Besl and Jain, 1988]: Flat, Minimal Surface, Peak, Ridge, Saddle Ridge, Pit, Valley and Saddle Valley. This first step produces a coarse segmentation.

The second segmentation step starts by isolating the largest connecting region of any surface type in the *surface type label image*. Through a contraction operation, a small isolated sub-region is selected as the seed region for the iterative region-growing phase. First, a plane equation is least-squares fitted to the seed region. If the plane does not fit with the data points of the seed region better than the given error threshold, a higher order surface will be fitted. If all the polynomial surface equations from the order of one to the order of four are tested and the average fit error is never less than the threshold, the seed region is discarded. In the opposite case, if a surface equation passed the *error threshold test*, and the distribution of the signs of the fit errors is homogeneous enough (*region test*), the iterative region growing begins. Now, all pixels in all coarse regions of the image, currently outside the seed region, are tested for possible inclusion into the current region. To be joined to the seed region the depth of the pixel from the sensor must not differ more than a threshold from the corresponding depth to the fitted surface model. And, secondly, the estimated surface normal on the corresponding locations on the range image and the surface model must not differ from each other by more than another threshold [Besl and Jain, 1988]. The largest connected region, which is composed of range pixels in the seed region and range pixels that pass the compatibility tests, is chosen as the new seed region. Expansion continues until there is almost zero change in region size or the region does not pass the *error threshold test*. Now, the new region must pass a somewhat relaxed *error threshold test* (i.e. the error threshold used for the original seed region increased by 50 percent) in order to be accepted. If the region is accepted, all the corresponding range pixels in the *surface type label image* are marked off. In the event, it is not accepted, only the range pixels corresponding to the original seed region are discarded.

The results after applying the method for segmenting range images taken from scenes consisting of both planar and curved surfaces reveal large amount of over-segmentation and noise [Powell et al., 1998]. By considering the clutter and noise in the range images taken from the natural outdoor scenes, the tendency of the method to oversegment the data would be disadvantageous. In the next chapter, an alternative method for the segmentation of range images will be discussed.

4.3.2 Range image segmentation based on scan line approximation

The goal of scan line approximation is to detect and classify discontinuities in a sequence of range data points. The location and type of the discontinuities or edges can be utilized for segmenting the range data (and consequently the underlying object geometry). In range images, three different basic edge types can be distinguished [Jiang and Bunke, 1996]: jump edges, crease edges and smooth

edges. Jump edges are usually defined as discontinuities in the range data. Crease edges are formed where two surfaces meet, and are characterized by discontinuities in surface normals. And, finally, smooth edges correspond to object geometry where surface normals are continuous but surface curvature changes. Such an edge is formed, for example, at the intersection of a plane and a cylindrical surface. Examples of the three different edge types are given in Figure 4.6.

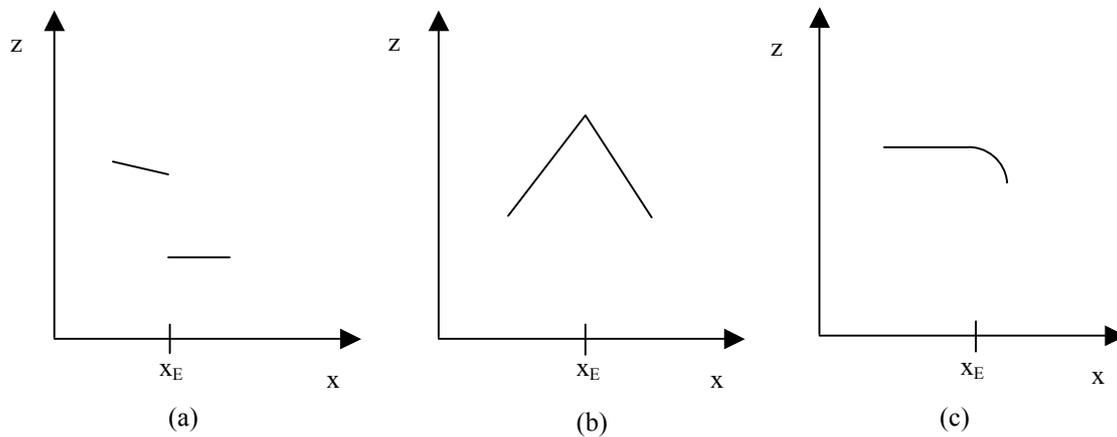


Figure 4.6 Examples of object geometries yielding a jump, crease and smooth edge respectively.

The underlying idea in the scan line-based edge detection is that each scan line corresponds to a curve in 3D space [Powell et al., 1998]. By detecting the edges within the scan line, a segmentation of the underlying object surface geometry can be achieved. If the perception data consists of parallel, overlapping scan lines, then, by grouping the detected edges, regions of homogeneous surface geometry can be formed. The parameters of the underlying geometric primitive models can then be estimated by means of the labelled measurement points within each region. The estimation of circle model parameters from segment data will be discussed in Chapter 4.6. Note that the range data used in our experiments is rather noisy compared to most of the test range images presented in the literature for the development and testing of the different range image segmentation methods. In the natural outdoor environments considered in this work, jump edges in particular are the most important type of range discontinuity. Actually, for the extraction of the cylindrical objects from the range images, to be discussed in Chapter 4.6, only jump edges will be considered for the initial segmentation of the range data.

4.4 Shape extraction in structured environments

In structured indoor environments most of the object surfaces are, usually, planes. In a typical 2D implementation, a mobile robot, equipped with a one-axis range scanner, is moving on a (planar) floor and is scanning the environment. The measurement points, acquired by the scanner, are distributed along straight line segments in 2D space. If the laser beam can also be tilted in the vertical direction, 2D range profiles of the 3D object surfaces can be acquired. In both cases, various range data segmentation methods can be used to group the measurement points belonging to the same surface. A comprehensive review, with experimental results, of the methods for segmenting range images in planar patches has been presented

in [Hoover et al. 1996] and is also available on the WWW at <http://marathon.csee.usf.edu/seg-comp/SegComp.html>.

After segmentation, the data within each segment can be fitted to a straight line model (2D-case) or to planar surface models (3D-case). The borders of the planar surfaces can be acquired by studying the range data within the segment or, if the range scanner is accompanied with a camera, from the camera image [Forsman, 1994]. In 2D operation space, other techniques, such as the Hough transform, can be utilized to extract the profiles of planar surfaces as well [Forsberg et al., 1995].

4.5 Geometric models for natural outdoor environments

Opposite to man-made structured environments, most major surfaces present in natural outdoor environments have a complex and irregular form. The main exception to this rule is tree trunks, the cross section of which can be approximated, in many cases, with a circle model up to a relatively good level of accuracy [Byrne and Singh, 1998]. Consequently, cylinder segment models, or more generally, truncated cone segment models, are one of the few volumetric primitive types of representation usable in natural outdoor environments. However, a bounding box type representation can be created for (any) object discover in an outdoor terrain environment.

4.5.1 Polynomial volumetric models

Betgé-Brezetz et al. compute, from a single range image, the center of mass of the measurement points (X_g, Y_g, Z_g) and the inertia matrix \mathbf{M} [Betgé-Brezetz et al., 1994a],

$$\mathbf{M} = \frac{1}{N} \sum_{i=1}^N [x_i - X_g, y_i - Y_g, z_i - Z_g]^T [x_i - X_g, y_i - Y_g, z_i - Z_g] \quad (4.4)$$

A coarse object model, which they call "3D blob", can then be constructed from the approximated object position, (X_g, Y_g, Z_g) ; object orientation, given by the eigen-vectors of \mathbf{M} ; and, an estimate of its size along each of the three inertia axes.

Hyperquadrics are volumetric shape models that can model a broad range of asymmetric shapes. A hyperquadric is a surface defined by a set of points (x, y, z) satisfying the following equation [Kumar et al., 1995],

$$\sum_i^N |A_i x + B_i y + C_i z + D_i|^{\gamma_i} = 1 \quad (4.5)$$

where N is any arbitrary number and $\gamma_i \geq 0 \quad \forall i$. For feature based mobile robot localization and viewpoint registration applications it is desirable that the model representations are as viewpoint invariant as possible. It would be even better, if the models were symmetric. With symmetric models, the requirement of having

visibility to the same part of the object surface from different viewpoints could be relaxed to the requirement that any side of the object surface is visible to the sensor. Superquadric, which is a special case of the hyperquadric, Equation 4.5, with

$$N = 3, A_1 = \frac{1}{a}, B_2 = \frac{1}{b}, C_3 = \frac{1}{c}, \text{ is given by,}$$

$$\left| \frac{x}{a} \right|^{\gamma_1} + \left| \frac{y}{b} \right|^{\gamma_2} + \left| \frac{z}{c} \right|^{\gamma_3} = 1 \quad (4.6)$$

In the equation, absolute values of the point coordinates are taken, therefore the superquadric is symmetric about the coordinate axes. Nevertheless, even points, sampled from a somewhat irregular object surface, could be fit to the superquadric model to give a coarse, symmetric approximation of the object shape. Indeed, Betgé-Brezetz et al., propose the superquadric as a more accurate alternative to the "3D blob" in the case that a more comprehensive distribution of measurement points on the object surface has been acquired [Betgé-Brezetz et al., 1994a].

4.5.2 Cylinder segment models

Högstöm and Wernersson proposed an automatic method for modeling forest scenes [Högstöm and Wernersson, 1998]. First, the range measurements were transformed into a 2D histogram by projecting them to the xy-plane. Peaks in the histogram were assumed to correspond to the tree trunks visible in the range image. The measurements around each of the significant peaks in the xy-histogram were labeled to belong to the particular tree trunk. Then, the horizontally projected data points labeled for a tree object hypothesis were fitted to a smoothed cubic spline for estimating the center line of the tree trunk. The radius of the tree trunk was determined by studying the range discontinuities around the center line approximation. The proposed method has the following drawbacks:

- The method relies on the high peaks of accumulated data points in the xy-histogram for identifying the tree trunk objects. This has the consequence that a major part of the tree trunk must be visible to the sensor to generate a clear peak in the histogram.
- The method cannot distinguish between cylindrical and other, nearly vertical, elongated objects. This can be a significant disadvantage, especially in structured environments.
- The method expects an object with a low degree of curvature.

The method, to be proposed in the next chapter, offers some improvements with respect to the aforementioned drawbacks.

4.6 Fitting cylinder segment models to narrow elongated objects

Basic properties of range data, measured from an unstructured, natural outdoor environment include:

- Large range discontinuities
- Cluttered data
- Bad viewpoint invariance of measurement point patterns due to occlusion
- Absence of specific types of features such as straight line segments and surface planes common in structured environments

These circumstances prevail in forested environments in particular. However, the possibility exists to utilize tree trunks as natural landmarks for viewpoint registration [Högström and Wernersson, 1998]. Tree trunks constitute a specific type of geometric object that can be reliably extracted from 3D range data. To perform segmentation and feature extraction one has to know that the corresponding model primitives are a reasonable abstraction of the underlying range data [Quek et al, 1993]. We apply the assumption that the visible parts of tree trunks (and possibly other visible objects with a circular cross-section) can be approximated as a connected sequence of cylinder segments or truncated cone segments. A truncated cone shaped model differs from a cylindrically shaped model in that the radius of the two end points can be different.

4.6.1 Segmentation of cluttered range data

In natural outdoor environments the external perception data can be cluttered, for example, due to bad weather conditions or due to vegetation blocking the view to the actual target objects. With respect to the segmentation of range scan data, discussed in Chapter 4.3, the clutter may cause "holes" to the range profiles formed by the laser beams hitting a smooth object surface on the background. These holes should not cause a break in the segment of (nearly) constant distance from the perception sensor to the target object. On the contrary, they should be filtered away from the measurement data belonging to the segment. The situation is illustrated in Figure 4.7 where a horizontal range scan is cast over a tree trunk, partly blocked by a branch of a tree.

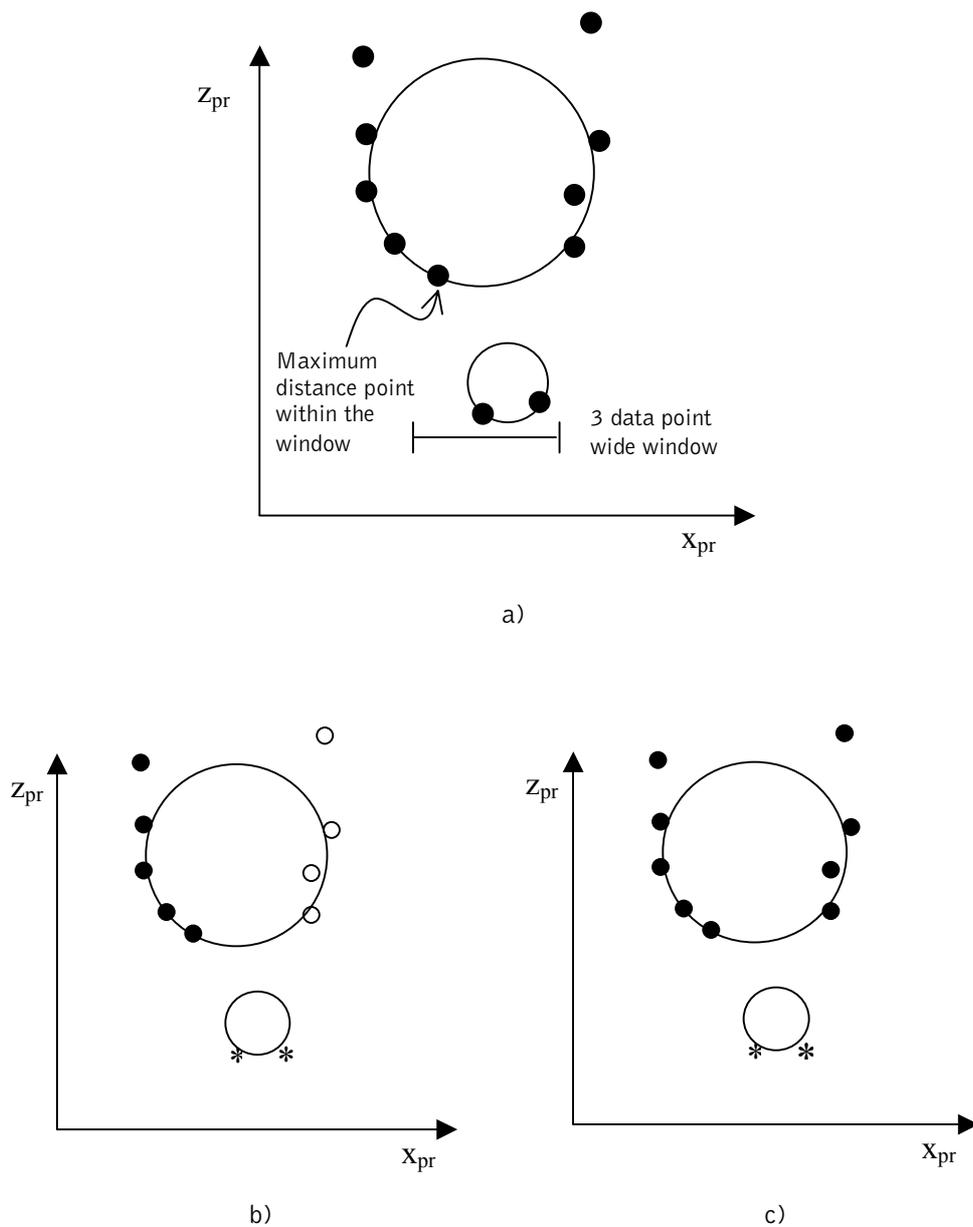


Figure 4.7 An artificial example of applying the outlier filtering technique for the initial segmentation of range profile data. In the upper image, the sliding window for computing a filtered distance to the contour of the object-of-interest is illustrated. In the lower left image, two separate segments (filled and non-filled circles) are extracted if the filtering technique is not applied. In the lower right image, a single segment corresponding to the size of the large object is extracted by means of the outlier filtering technique.

Random (outlier) noise can be smoothed out by going through the whole data set (i.e. range image or a range scan line) and by replacing the current data element with the median value computed by means of its (immediate) neighbors [Sinha and Schunck, 1992] and [Jiang and Bunke, 1994]. Instead of smoothing the data, the center range measurement of the $N \times N$ filtering window can be removed or be given the outlier label if it deviates more than the threshold with respect to the median value computed from the data points within the mask [Hoover et al., 1996].

However, in the situation, presented in the upper picture of Figure 4.7, the outliers do not appear randomly among inlier measurements, but form connected groups. The size of the groups is dependent on the size of the object blocking the view into the background surface. Conventional median filtering may not work properly with such outlier distribution. If the goal is to determine the borders of the object in the measurement data set, the maximum value instead of the median can be used to replace the center data element of the sliding filtering window. The size of the window can be determined by the maximum limit width of a branch, considered in outlier filtering. Objects, larger than the limit, will always cause the end of the segment to be declared. The "blocking object"-filtered distance values can then be used to determine the beginning and the end of the segment of nearly equal distance data points. After the borders of the segment have been determined within a horizontal range scan, the outlier-filtered distance to the middle of the segment is evaluated. All the measurements within the segment, that deviate more than a threshold from the middle point reference range value, are given the outlier status. When detecting narrow elongated objects, the threshold value can be determined by means of the above-evaluated width of the segment and the expected curvature of the underlying object. A good candidate for the outlier threshold, for an object with a circular (elliptical) cross-section, is given by

$$T_{out} = 0.5W_{seg} + 3\sigma_{range} \quad (4.7)$$

where W_{seg} is the evaluated width of the segment and σ_{range} is the standard deviation of the range measurement uncertainty. The formula for computing the value of the threshold was determined based on the experiments with real data. The value of T_{out} , computed according to Equation 4.7, means that measurement points, whose projected distances from the perception sensor are within the \pm approximate segment radius span with respect to the distance to the center point of the circular segment, are accepted as valid measurements. The additional component " $3\sigma_{range}$ " is added at both ends of the span to allow for the uncertainty in the range measurement. The results of applying the technique are shown in Figure 4.7(c), where the data points detected for a single segment are marked with filled dots whereas the other data points (outliers) are marked with a star symbol. The segmentation results, *without* applying the sliding window are shown in Figure 4.7(b) where two separate segments have been (incorrectly) extracted instead of one large segment. Note that data points hitting the blocking object, in Figure 4.7(a), form a group too small to be declared as a valid segment.

A range image of a natural area with tree trunks is presented in the following figure.

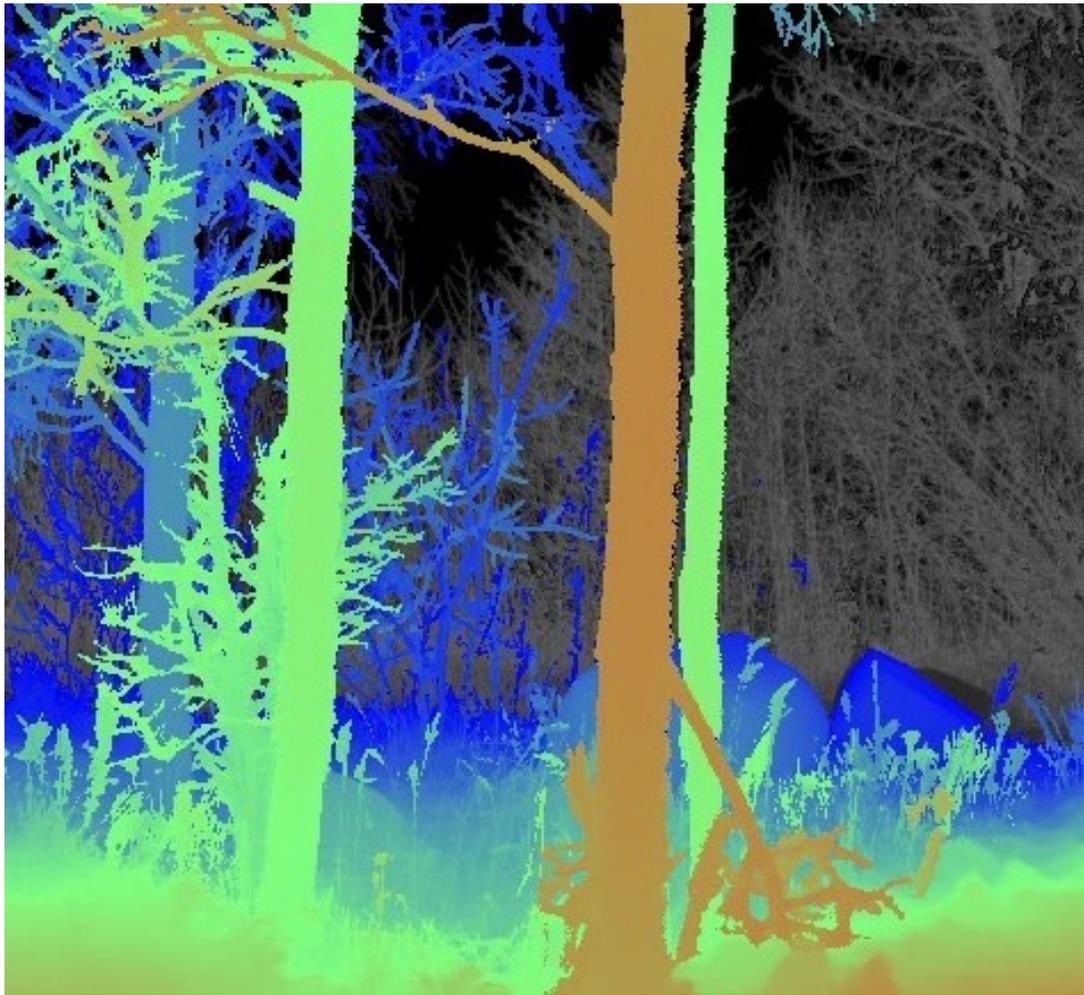


Figure 4.8 Range image from a group of trees. Ranges between 3.5 and 10 meters have been pseudo-coloured from brown to blue respectively.

The results of applying the segmentation method for the range image, presented in Figure 4.8, can be seen in Figure 4.9. During the segmentation, horizontal scan lines were processed in order to find connected sets of data points with a nearly equal distance from the range sensor. Another criterion was set for the segment width, which had to be between 0.06 and 1 meter.



Figure 4.9 Horizontal segments extracted from the range image, presented in Figure 4.8. The range segments, the length of which was found to be between 0.06 and 1.0 meters, have been painted black.

After applying the robust filtering technique, described above, during range image segmentation, further tolerance to outliers, in context of a least squares estimation method, can be achieved by analyzing the statistics in the measurement data. From the distribution of residual errors of the data points with respect to the estimated model, a threshold can be determined. The data points that are further from the model surface than the threshold are discarded. A new model and a threshold are calculated with the remaining data points and the process is repeated [Zhang, 1994]. A good candidate for the threshold to classify data points to inliers and outliers equals three times the standard deviation of the fitting errors. This technique has been applied for fitting a circle model to the data points within the range segments. The subject will be discussed in the following chapter.

4.6.2 Estimation of the circle model from segmented range data

Our method has adopted the idea of projecting the data points to a plane, perpendicular to the estimated cylinder center axis, to determine the radius of the cylinder [Umeda et al., 1996]. In the method of Umeda et al., the center axis orientation was determined from the Extended Gaussian Image (EGI). EGI represents the distribution of surface normal vectors computed from local groups of neighboring data points on the object surface. The method works fine if the

underlying object is a regular cylinder segment. In our case, the most important target group of objects, the tree trunks, has a more complex geometry with a smooth and curved surface in a longitudinal direction, which can be locally approximated with a circular cross-section [Byrne and Singh, 1998]. The objective is to approximate this curved, variable width object with cylinder segments (or truncated cone segments) extracted from the range data. To compute the circle cross-section at each location along the visible part of the tree trunk, the range data is first projected to a plane perpendicular to the estimated orientation of the tree trunk at that location. Another constraint, selected for the projection plane, is that it passes through the origin of the sensor frame. The principle of data projection and circle model approximation for the cross-section of a cylindrical target object (tree trunk), based on the projected data points, is illustrated in Figure 4.10.

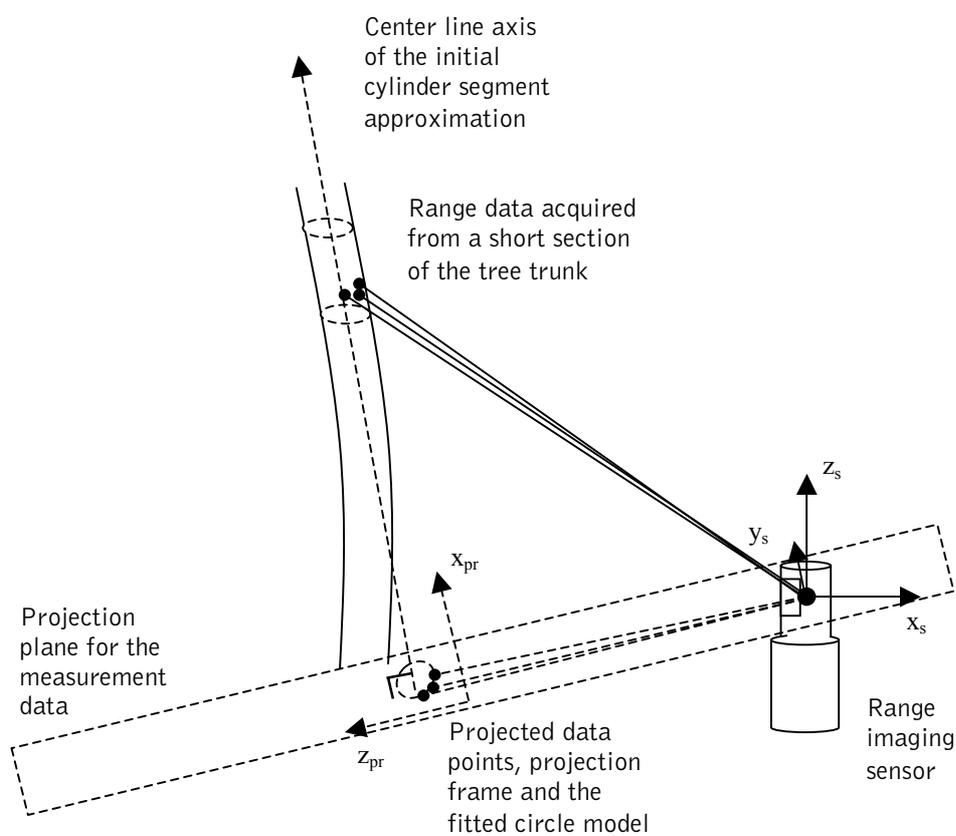


Figure 4.10 Illustration of the computation of a circle model approximation for the cross-section of a location on the tree trunk.

In our implementation, an alternative method to the Extended Gaussian Image is used to determine the orientation of the tree trunk at a particular location. The projection plane orientation is determined by means of the circle models estimated from the measurement data in the local neighborhood of the current measurement location during the previous iteration round. The projection plane normal can be estimated by computing the principal components of the 3D circle center point cloud corresponding to the overlapping circle models. If the Singular Value Decomposition method is used, the singular vector corresponding to the smallest singular value gives the direction vector of the longest principal axis of the fit data point cloud. Another way to compute the normal vector direction for a particular

circle model is to concatenate the individual direction vectors from the center point of the circle model to all the neighboring circle model center points. The unit vector computed from the concatenated direction vector can then be used as an estimate of the direction of the center axis of the tree trunk at that location. The direction vector will then be utilized to project the data points for the computation of the x- and z-position parameters of the center of the circle model on the projection plane and the circle radius as depicted in Figure 4.10. The latter approach has been used in our experiments. The computation of the orientation of the projection plane normal vector is carried out by using the point coordinates of the circle model center positions, computed in the previous cycle of the iterative circle model computation process. The computation of the projection plane normal direction is illustrated in Figure 4.11.

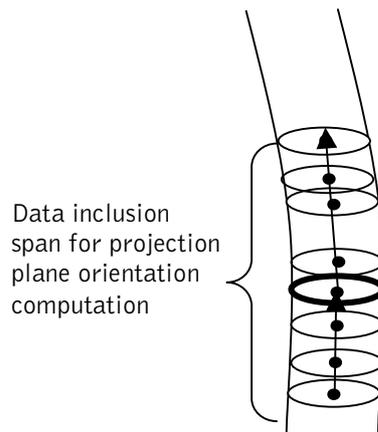


Figure 4.11 Computation of projection plane normal direction for a circle model.

In Figure 4.11, the configuration parameter for defining the available span for direction vector computation around the valid range segment under consideration (painted with thick line) was set at 0.5 to 0.7 meters in the experiments. The validity of a range segment is defined as the availability of a circle model computed for it at the previous iteration (or initialization) phase. Two orientation vectors, one above and one below, are computed if possible. Then the weighted average of the two vectors is used as the normal vector of the projection plane in re-computing the circle model parameters for the current range segment during the next iteration. The iterative process is discussed in more detail in Chapter 4.6.3. It should be noted that the computed circle models are projected back to the true spatial 3D location of the measured object surface.

At the beginning of the cylinder model approximation process, no circle segment models exist and we have to start with the original range image. During the segmentation step, groups of measurement points of nearly equal distance from the range sensor along desired (true or artificial) scan lines are extracted. For the generation of a geometric model for the tree trunks, horizontal rows of the range image are used as the scan lines. All horizontal scan lines are segmented and sets of approximately overlapping segments are searched for further processing. Sequences of overlapping 2D segments can now be grouped together as potentially belonging to the same elongated, approximately vertical object. The initial orientation approximation of the projection plane (compare with Figure 4.10) can be attained from the overlapping segment data or, as used in the context of tree

trunk/supporting column modelling, from the assumption that the objects are roughly vertical (within ± 30 degree bounds).

Now, a number of data points collected from the close neighbourhood of overlapping 2D segments around the current segment under study have been projected onto the projection (fitting) plane. The projected data points, on the 2D fitting plane, are next fit into a circle model. A circle model for approximating the projected data points is given by,

$$\begin{aligned} (z_{pr} - z_0)^2 + (x_{pr} - x_0)^2 &= r^2 \\ \Leftrightarrow z_{pr}^2 + x_{pr}^2 - 2z_0z_{pr} - 2x_0x_{pr} &= r^2 - z_0^2 - x_0^2 \\ \Leftrightarrow \left(\frac{1.0}{v}\right)(z_{pr}^2 + x_{pr}^2) - \frac{2z_0}{v}z_{pr} - \frac{2x_0}{v}x_{pr} &= 1.0, \text{ where } v = r^2 - z_0^2 - x_0^2 \end{aligned} \quad (4.8)$$

where x_{pr} and z_{pr} correspond to the coordinates of the projected measurement points, x_0 and z_0 are the unknown coordinates of the circle center with respect to the projection frame and r is the unknown circle radius to be estimated. A linear equation with respect to the unknown parameters can be formed with the following substitutions,

$$\begin{aligned} a &= \frac{1.0}{v} \\ b &= -\frac{2z_0}{v} \\ c &= -\frac{2x_0}{v} \end{aligned} \quad (4.9)$$

yielding,

$$a(z_{pr}^2 + x_{pr}^2) + bz_{pr} + cx_{pr} = 1.0 \quad (4.10)$$

The circle model parameters, i.e. the coordinates of the circle centre point, x_0 and z_0 , and the radius of the circle r , can now be computed from the least squares estimates of the parameters as follows,

$$x_0 = -\frac{c}{2a} \quad (4.11a)$$

$$z_0 = -\frac{b}{2a} \quad (4.11b)$$

$$r = \sqrt{\frac{1.0}{a} + x_0^2 + z_0^2} = \sqrt{\frac{4a + c^2 + b^2}{4a^2}} \quad (4.11c)$$

If required, the covariance matrix of the estimated parameters can be readily computed from the results of the singular value decomposition (SVD) method [Press et al., 1992]. Let us mark the 3x3 covariance matrix of the estimated

parameters with \mathbf{C}_{abc} . The diagonal of the covariance matrix, i.e. the variances of the fitted parameters, are of primary interest. They should be propagated through the non-linear model linking the estimated parameters with the actual circle model parameters, Equation 4.11. By assuming that the non-linear equation can be approximated (in the close neighbourhood of the estimated parameters) with a linear equation composed of the first two terms of the Taylor series expansion of the non-linear equation, the variances of the circle parameters can be computed from the following equation,

$$C_{x_0x_0} \approx (\nabla(x_0)_a)^2 C_{aa} + (\nabla(x_0)_c)^2 C_{cc} \quad (4.12a)$$

$$C_{z_0z_0} \approx (\nabla(z_0)_a)^2 C_{aa} + (\nabla(z_0)_b)^2 C_{bb} \quad (4.12b)$$

$$C_{rr} \approx (\nabla(r)_a)^2 C_{aa} + (\nabla(r)_b)^2 C_{bb} + (\nabla(r)_c)^2 C_{cc} \quad (4.12c)$$

where $C_{x_0x_0}$ is the approximated variance of the x_0 coordinate of the circle model with respect to the projection plane (compare with Figure 4.10), $\nabla(x_0)_a$ is the derivate of the right hand side of Equation 4.11a with respect to a and C_{aa} is the variance of a given by the SVD method. The range segments in Figure 4.9, for which a valid circle model could be computed, are shown in Figure 4.12.

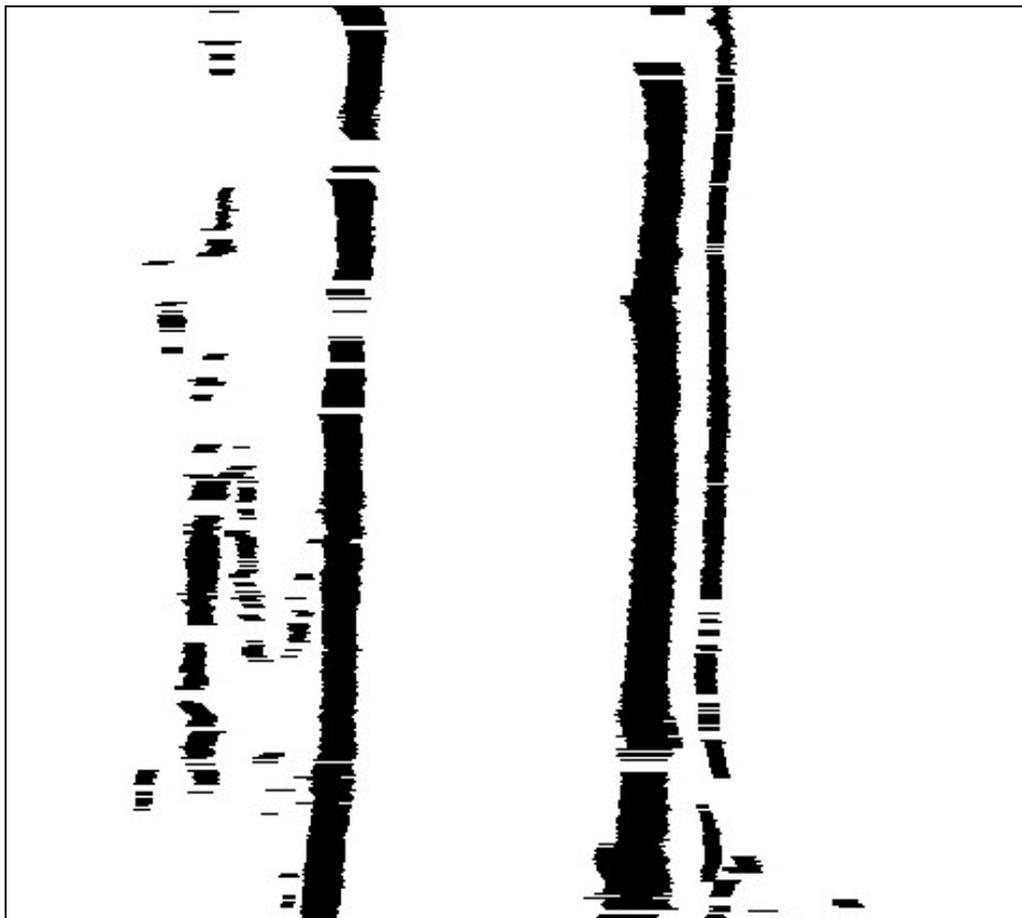


Figure 4.12 The valid circle models computed from the measurement points stored within the range segments, shown in Figure 4.9.

In Figure 4.12, the width of the estimated circle models is clearly smaller than the width of the corresponding range segments, shown in Figure 4.9. This is due to the fact that the Riegl LMS-Z210 range scanner (refer to Chapter 6.1.1 for more detail) is designed for measurement distances up to 350 meters. Therefore, even a minor share of the emitted laser pulse, hitting the surface of a close target, yields a valid range measurement. Overall, the width of the range segments for cylinder model computation, extracted in this work, overestimates the true object cross-section. Interestingly, in another reported forest mapping research work, the width of the range segments corresponding to the cross section of tree trunks, was found to be systematically smaller than the true tree width [Högström and Wernersson, 1998]. This is, most probably, the consequence of a less powerful laser range measuring system, which requires that more than the half of the laser beam hits the object surface for a valid range measurement.

Due to vegetation and branches, blocking the view to the tree trunks, the sets of overlapping valid range segments are rather fragmented in Figure 4.12. Nevertheless, cylinder segment models, yielding an approximation of a major part of the tree trunks visible in the range image, could be assembled from the computed circle models. In Figure 4.13, a reality augmented, virtual camera view into the generated cylinder segment models estimated from the range image data of Figure 4.8, is presented.



Figure 4.13 Camera view of an outdoor scene overlaid with the 3D cylinder model representations, computed from the range image data, shown in Figure 4.8.

The method for constructing the cylinder segment models will be discussed in the next chapter.

4.6.3 Construction of cylinder segment models from circle model data

The iterative process for extracting cylinder segment models from range images is illustrated in Figure 4.14.

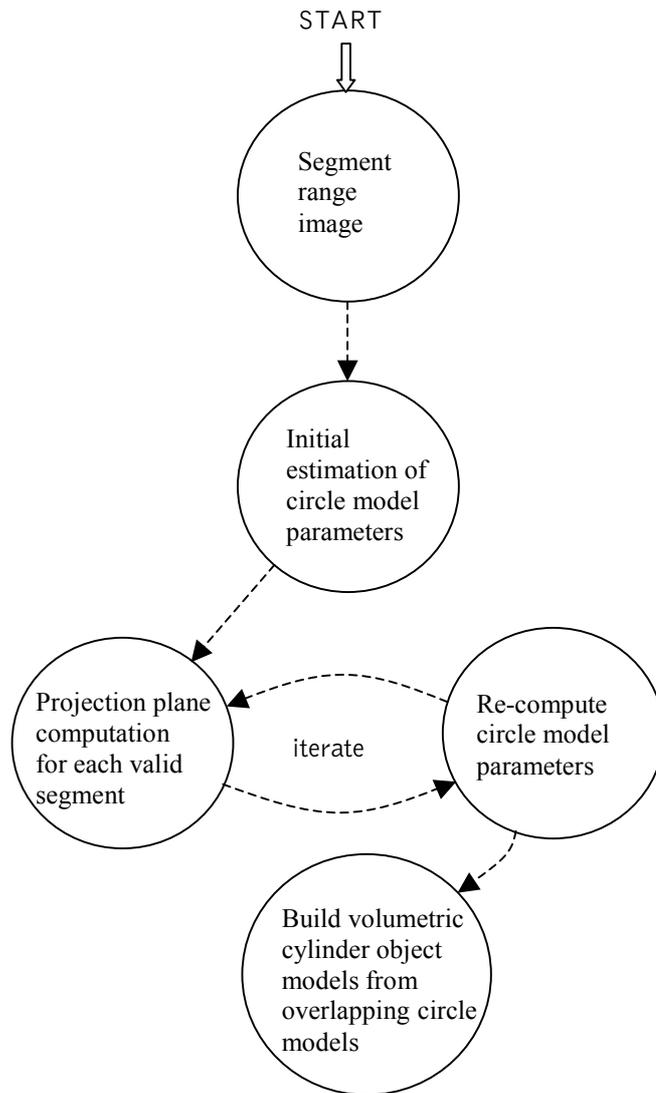


Figure 4.14 A schematic diagram of the iterative process for modeling the narrow, elongated objects, with circular cross-sections, visible in the range image.

In Figure 4.14, for the initial estimation of circle model parameters, the default projection plane orientation is applied. In the experiments, it corresponds to the vertical direction. For the initial estimation, the control parameters, which are used for discarding circle models corresponding to non-circular objects, are relaxed somewhat. In this way, a valid circle model can also be computed for the cross-sections of narrow objects, which are not orientated strictly vertically. After the initial phase, the computation of projection plane parameters and the subsequent

estimation of circle models are repeated until the model parameters stabilize. Usually, two iterations (in addition to the initial estimation phase) are required. And finally, volumetric cylinder object (or truncated cone segment object) models are assembled from the overlapping circle models. In all the experiments presented in this work, cylinder segments are used as volumetric models. The radius of such a cylinder model is approximated as the average of the corresponding circle object models, allocated for the cylinder object. The utilization of the more general, truncated cone segment models was omitted due to the relatively large variation in the radius values of the circle models, especially in outdoor environments. The variation was due to noisy data, and also due to the, strictly speaking, non-circular shape of the tree trunk cross-sections. Nevertheless, this has no effect on the relative alignment computation between the overlapping sub-maps, because the radius information is not utilized to generate the data point sets for alignment computation. More details concerning the computation of relative alignments among pairs of sub-maps can be found in Chapters 5.3 and 5.4.

4.6.4 Experimental results

To extract cylinder segment (or truncated cone segment) feature models from the range image, several computation configuration parameters are required. Those most central to the initial range image segmentation are listed in Table 4.1.

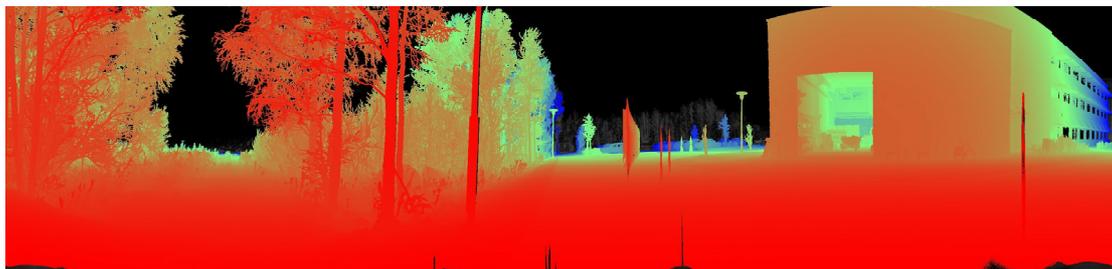
Table 4.1 Typical values of the key configuration parameters for range data segmentation. The indoor test case is presented on the left side of the table and the outdoor/forest test case is on the right.

| Parameter name | Value | Parameter name | Value |
|---|-------------|---|-------------|
| Low limit for segment/circle size | 0.2 meters | Low limit for segment/circle size | 0.06 meters |
| High limit for segment/circle size | 1.0 meter | High limit for segment/circle size | 1.0 meter |
| Maximum occluding object size | 0.04 meters | Maximum occluding object size | 0.04 meters |
| Standard deviation of a single range measurement | 0.025 | Standard deviation of a single range measurement | 0.025 |
| Range step for segment end declaration | 0.11 meters | Range step for segment end declaration | 0.11 meters |
| Minimum segment size in terms of measurement number | 4 | Minimum segment size in terms of measurement number | 4 |

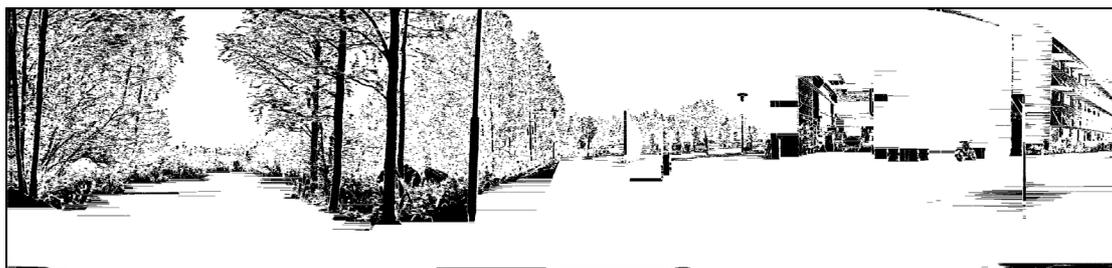
In the table, the “maximum occluding object size” parameter corresponds to the maximum size of an occluding object, such as the branch of a tree, depicted in Figure 4.7. The “standard deviation of a single range measurement” is also used to model the precision of a single measurement point for the circle model computation phase. The effective uncertainty due to the angular precision of the laser beam pointing system is not considered separately, but is assumed to be covered by the aforementioned value. This can be justified by the fact that the effective metric

uncertainty due to the uncertainty in the laser beam pointing angles is relative small (for more detail, refer to Chapter 6.1.1). The "range step for segment end declaration" parameter should be set to be larger than three times the standard deviation of the uncertainty of a single range measurement.

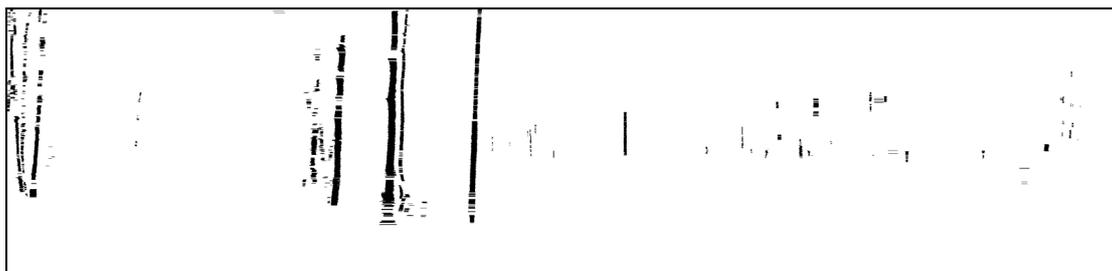
As an example of the overall performance of the method, the range image, labeled "scanE", was segmented and circle models were estimated from the data belonging to the valid segments. The output of range image segmentation is illustrated by painting black the locations on a white binary image of the same size as the original range image (3000x720 pixels) where a valid horizontal segment of smoothly varying range values was found. Locations on the image, related to segments shorter than 0.06 meters or longer than 1.0 meter are left white. The segmentation results are depicted in Figure 4.15b).



a)



b)



c)

Figure 4.15 In the middle image, the results of the initial segmentation of the range image are presented. In the lower image the valid circle models computed from the segments in the middle image are shown. The original range image "scanE" is given in the upper image for reference.

In Figure 4.15b) it can be seen how long horizontal segments, corresponding to the ground or large wall planes, are left white, i.e. are discarded from cylinder object modeling. In Figure 4.15c) the segments (or rather small sets of overlapping segments) yielding a valid circle model are painted black.

The key parameters in estimating the cylinder (or truncated cone segment) model from the segment data are listed in Table 4.2. In the table the first parameter describes the average *signed* error distribution of the measurement points from the circle contour. The interpretation of the *signed* error was illustrated in Figure 4.2. This parameter has turned out to be very important in discriminating the circularity of an object cross-section and hence assigning to the circle model hypothesis the "valid" or "outlier" status.

Table 4.2 Typical values of the key configuration parameters for circle/cylinder segment model parameter computation. The indoor test case is presented on the left side of the table and outdoor/forest test case in on the right part.

| Parameter name | Value | Parameter name | Value |
|---|-------------------|---|--------------------|
| Measurement point inlier/outlier error symmetry threshold | 0.7 millimeters | Measurement point inlier/outlier error symmetry threshold | 4.1 millimeters |
| Distance above and below the current segment for collecting data points | ± 0.15 meters | Distance above and below the current segment for collecting data points | ± 0.15 meters |
| Maximum number of segments above and below the current segment for collecting data points | 14 | Maximum number of segments above and below the current segment for collecting data points | 14 |
| Minimum length of cylinder segment object | 1.0 meter | Minimum length of cylinder segment object | 0.3/ 0.5 meters |
| Curvature threshold to start a new cylinder segment | 12.0 degrees | Curvature threshold to start a new cylinder segment | 10.0/ 12.0 degrees |

As the circle model parameters have been stabilized for the final values, it is time to compute the volumetric cylinder segment models from the circle model data. In natural environments, the principal objects of interest are tree trunks, the orientation of which is smoothly varying. To partition the model of the tree trunk into cylinder segments, the information of the projection plane normal orientation, stored into the data structure of each of the circle models, is utilized. The situation is depicted in Figure 4.16.

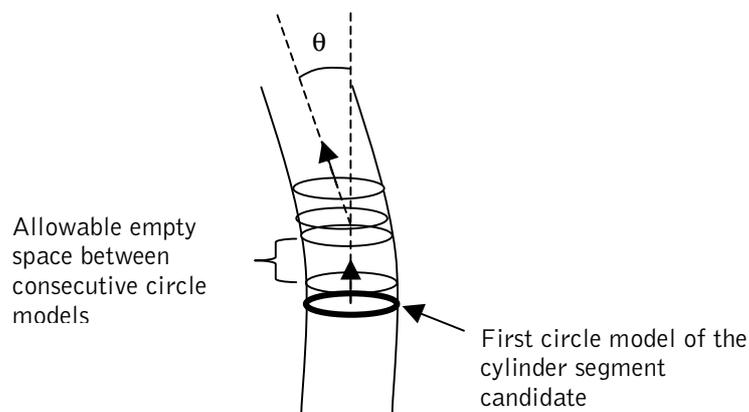


Figure 4.16 Illustration of the parameters involved in cylinder segment computation.

A cylinder segment model is formed by starting from the next valid circle model, which has not yet been considered by the cylinder object model approximation process. The circle model has been drawn with the thick line in Figure 4.16. After that, the orientations of the overlapping circle models are studied to determine whether they deviate from the orientation of the first circle model more than the given threshold. For example, in the outdoor (i.e. parking lot) mapping experiment, the threshold was set at 10 degrees. The search is continued until either the θ angle exceeds the threshold or overlapping circle models are no longer available. As indicated in Figure 4.16, empty spaces are allowed to exist between two consecutive circle models. The value of the maximum allowable size of an empty space, applied in most of the experiments, equals at 0.24 meters. The minimum length for a valid cylinder segment was set at 0.3, 1.0 and 0.5 meters in the parking lot, indoor environment and forest terrain mapping experiments, respectively.

Next, the accuracy properties of the circle model computation method are studied by modeling the cylindrical pillar objects in the office corridor environment. The manually measured ground truth radius of the pillars is 0.192 meters. In the indoor mapping experiment, a pillar object was modeled 17 times across its full length, from the floor level up to the ceiling at a height of about 2.3 meters. Moreover, the cylinder segment models were generated from a distance with respect to the perception sensor that varied from 1.2 to 18.4 meters. The radius of the cylinder segment model for a pillar object was computed as the average of the radii of the individual circle models belonging to the object. In Figure 4.17 the correlation between the computed cylinder radii with respect to the distance from the perception sensor is illustrated.

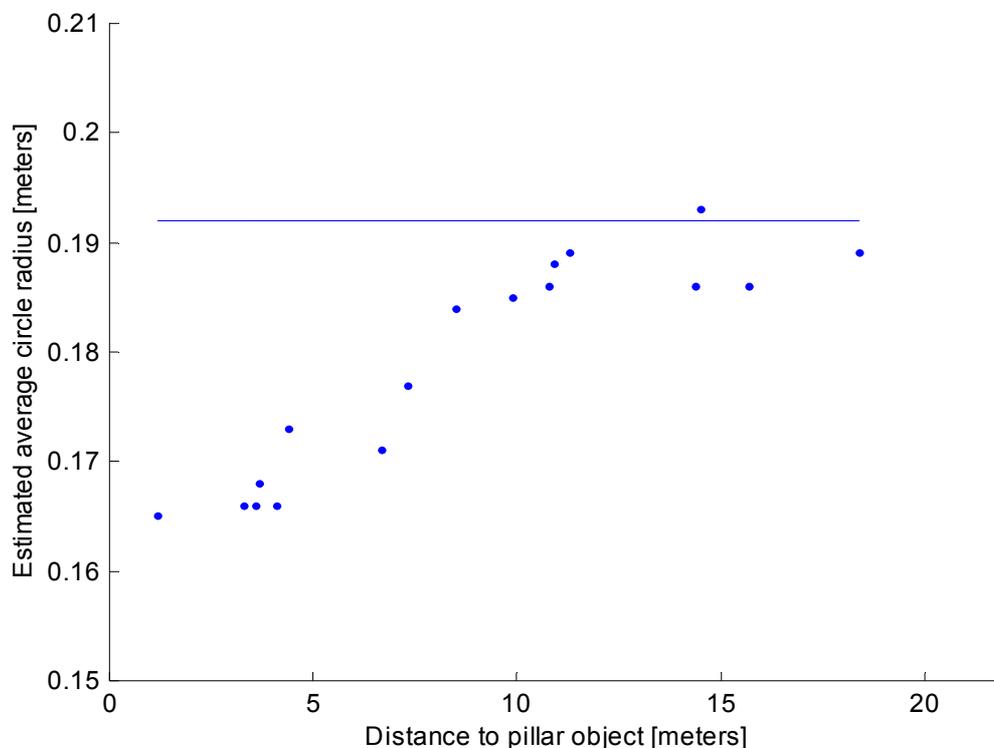


Figure 4.17 In the figure, the correlation between the measurement distance and the computed radius of the pillar object is shown. The manually measured radius is indicated with a solid horizontal line.

As can be seen in Figure 4.17, a relatively accurate estimate for the pillar radius can be computed when the pillar object is located at a distance from the sensor. With close measurement distances, a radius value, which is too small, is obtained. A possible cause for the phenomenon is the wide laser beam pattern and consequent folding of the beam along the object surface. The folding may produce a non-symmetric error component to the measured beam hit point coordinates, which distorts the circle model parameters. However, care should be taken, if the results of Figure 4.17 were to be generalized to other measurement environments. The characteristics of the non-symmetric error component probably depend on the properties of the cylindrical target object, especially on its size. The measurement set-up for acquiring range data from a pillar object is depicted in Figure 4.18.

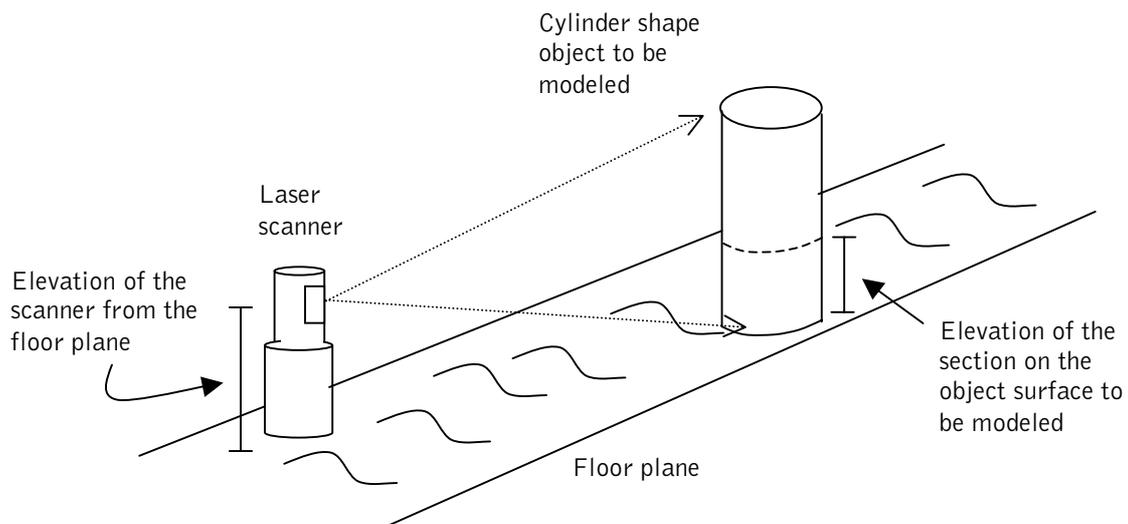


Figure 4.18 Measurement set-up for computing a cylinder model for the pillar object.

Next, the precision of the circle model approximation, computed for an object cross-section, is studied in more detail. In Figure 4.19, the estimated radius of a cylindrical pillar object as a function of the elevation of the circle model from the floor plane is depicted. The distance of the pillar object from the range scanner was 14.5 meters. In the experiment, the elevation of the range sensor from the floor plane was about 1.3 meters. In the figure, curves corresponding to the initial estimation phase and the second (final) iteration phase of the feature extraction process are presented. (The iterative process was depicted in Figure 4.14). The curve drawn with the dash dotted line corresponds to the initial estimation of the circle parameters. For the initial estimation phase, the projection plane normal orientation, tilted 30 degrees sideways with respect to the measurement direction from the correct vertical orientation, was applied. The curve drawn with the solid line corresponds to the circle radii computed during the final iteration phase. Corresponding results for the situation, where the initial orientation of the projection plane normal was tilted 30 degrees along the measurement direction are illustrated in Appendix B, Figure B.10.

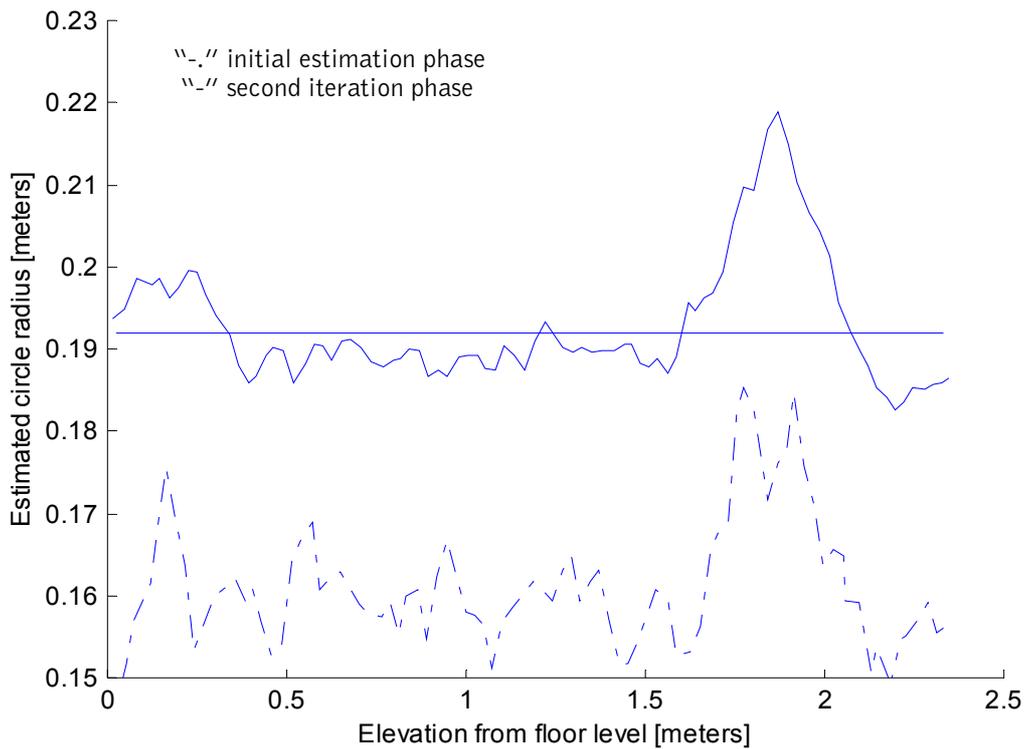


Figure 4.19 The computed circle model radius as the function of its elevation from floor level. The distance of the pillar object from the range scanner was about 14.5 meters. The projection plane for the initial phase was tilted 30 degrees sideways with respect to the measurement direction. The manually measured radius (0.192 meters) is indicated with the horizontal line.

The computed radius estimate is slightly below the manually measured value within the elevation span 0.3 to 1.6 meters. The excessively wide radius, computed for the top part of the pillar object as well as for a short section of the pillar close to the floor level, was due to an "extra" range beam captured by the object surface. A closer view of the range data, corresponding to the pillar object, is shown in Figure 4.20.

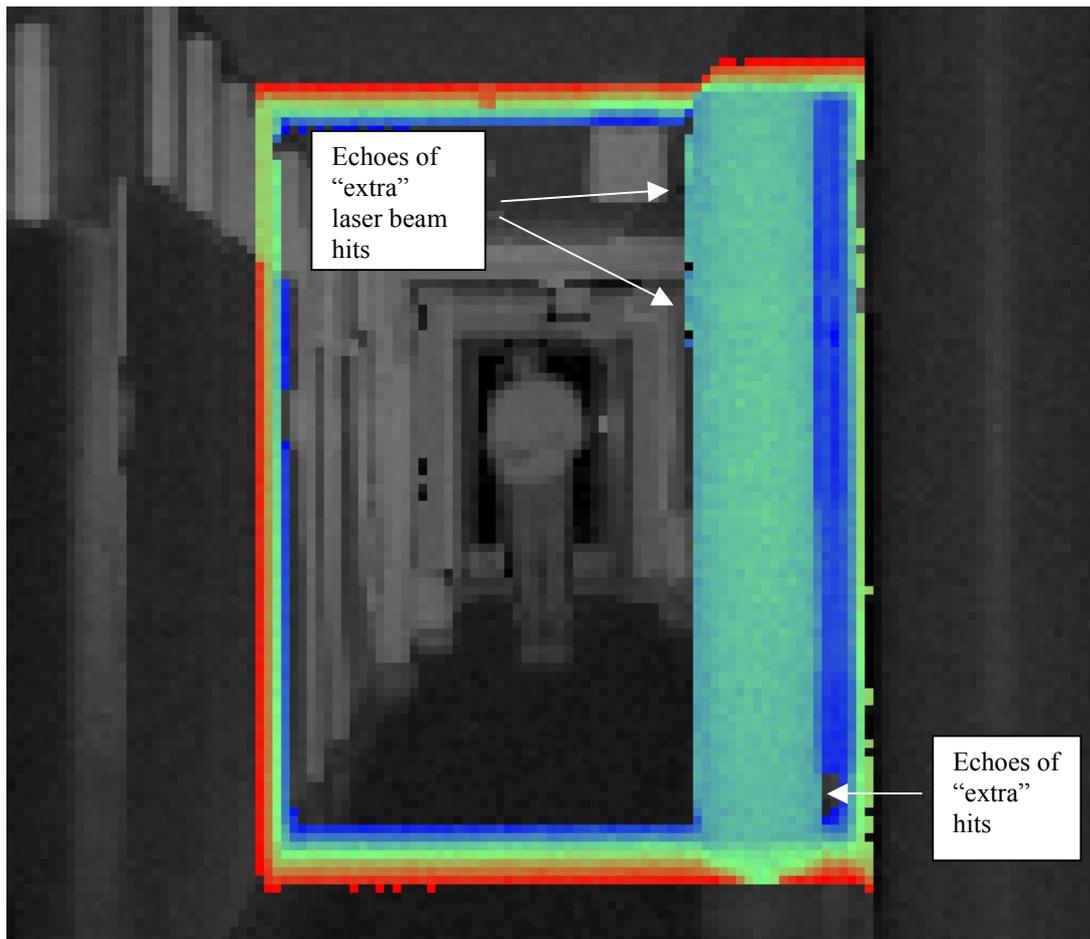


Figure 4.20 A closer look of the pillar object at about 14.5 meters' distance. The echoes of the "extra" laser beams, captured by the pillar surface can be distinguished at the top part of the pillar and also near by the floor plane. The range image patch was taken from the range scan data set, labeled "scan1", shown in Figure 4.1.

The average radius value, computed from the radius estimates shown in Figure 4.19, equals 0.193 meters, which is (by chance) almost the same as the ground truth value. Now, the corresponding results for a pillar object at 3.7 meters' distance from the range sensor are presented in Figure 4.21 and Figure B.11, in Appendix B. A closer view of the range data, corresponding to the pillar object at 3.7 meters' distance, is depicted in Appendix B, Figure B.9.

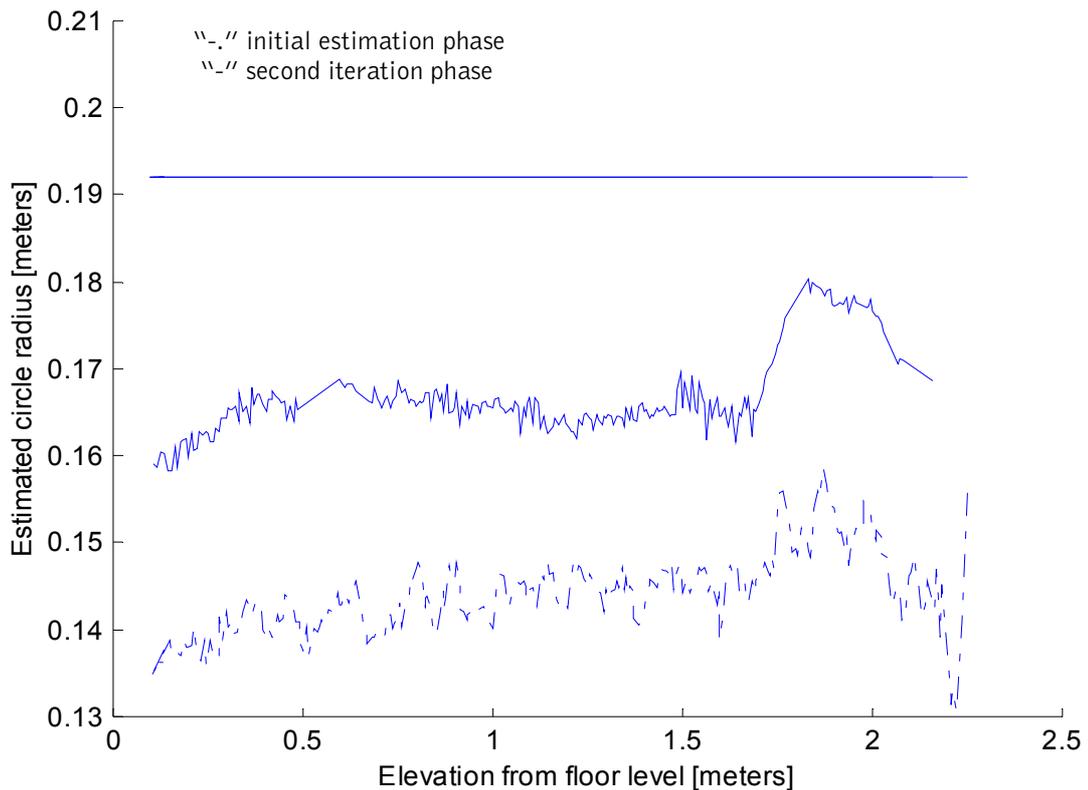


Figure 4.21 The computed circle model radius as the function of its elevation from floor level. The distance of the pillar object from the range scanner was 3.7 meters. The manually measured radius (0.192 meters) is indicated with the horizontal line.

In Figure 4.21, the decrease in the estimated radius during the initial estimation phase, when the data points are projected to a plane that has been tilted sideways with respect to the measurement direction, is clearly visible. An opposite phenomenon, i.e. increase in the estimated radius during the initial estimation phase, can be observed in the results, depicted in Figure B.11, in Appendix B. For the initial estimation phase curve, shown in Figure B.11, the projection plane was tilted 30 degrees along the measurement direction. Both phenomena can be understood by considering the spread of the measurement points across the projection plane in the two cases. As already indicated in Figure 4.17, the final radius estimates, computed for the cross section of a closely lying pillar object, are consistently below the correct value.

To demonstrate the quality of the measurement data, examples of the estimation of a circle model from the projected measurement data are presented in Appendix B. In Figures B.12, B.13 and B.14, the results for the circle cross-section approximation of the pillar object at about 14.5 meters' distance are presented. The circle model was computed for a cross-section cut of the pillar at about 1.4 meters' elevation from the floor plane. Circle model estimates, corresponding to two different initial iteration phases and a final estimation phase, are presented. In Figures B.15 and B.16, the results for the circle cross-section approximation of the right most, relative thin tree (0.055 meters radius) in Figure 4.8, at about 1.5 meters' elevation from the ground, are presented. In this experiment, the default projection plane orientation was set equal to the vertical direction.

4.7 Fitting straight line models in free space to unknown/occupied space borders

The cylinder segment objects, considered in this work, are oriented, more or less, vertically. The computation of the relative coordinate transformation between overlapping sub-maps, containing only vertical objects, would be difficult, especially in indoor environments. Outdoors, tree trunks are normally somewhat tilted with respect to the vertical direction such that a reliable registration with respect to the z-coordinate direction can be achieved as well. To augment the set of vertical features in structured environments, horizontal straight line segments, extracted from the free space to unknown/occupied space borders of the 3D occupancy grid map, were selected as the second category of natural landmarks.

4.7.1 Description of the method

A schematic diagram of the method for extracting the wall/ceiling corner line segments from the range image data is depicted in Figure 4.22.

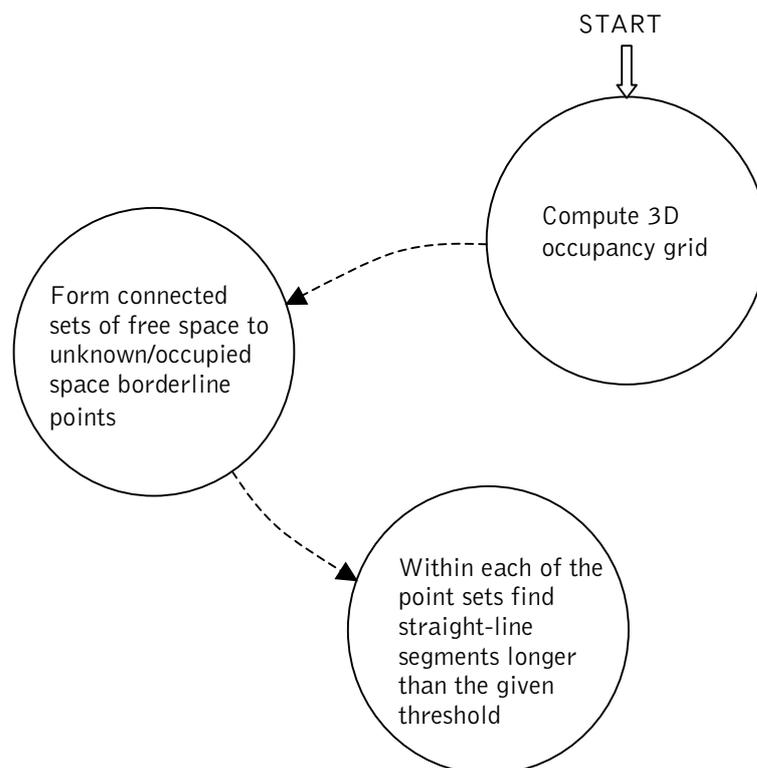


Figure 4.22 A schematic diagram of the wall/ceiling corner line extraction process.

Range imaging devices with a high angular resolution, such as the Riegl LMS-Z210 scanner, can be used to acquire an exhaustive scan throughout the 3D workspace around the sensor. Further, the dense range image can be converted into a 3D occupancy grid, with clear borders between free space and unknown/occupied space. A 2D projection of the free space to unknown/occupied space border is shown in Figure 4.23.

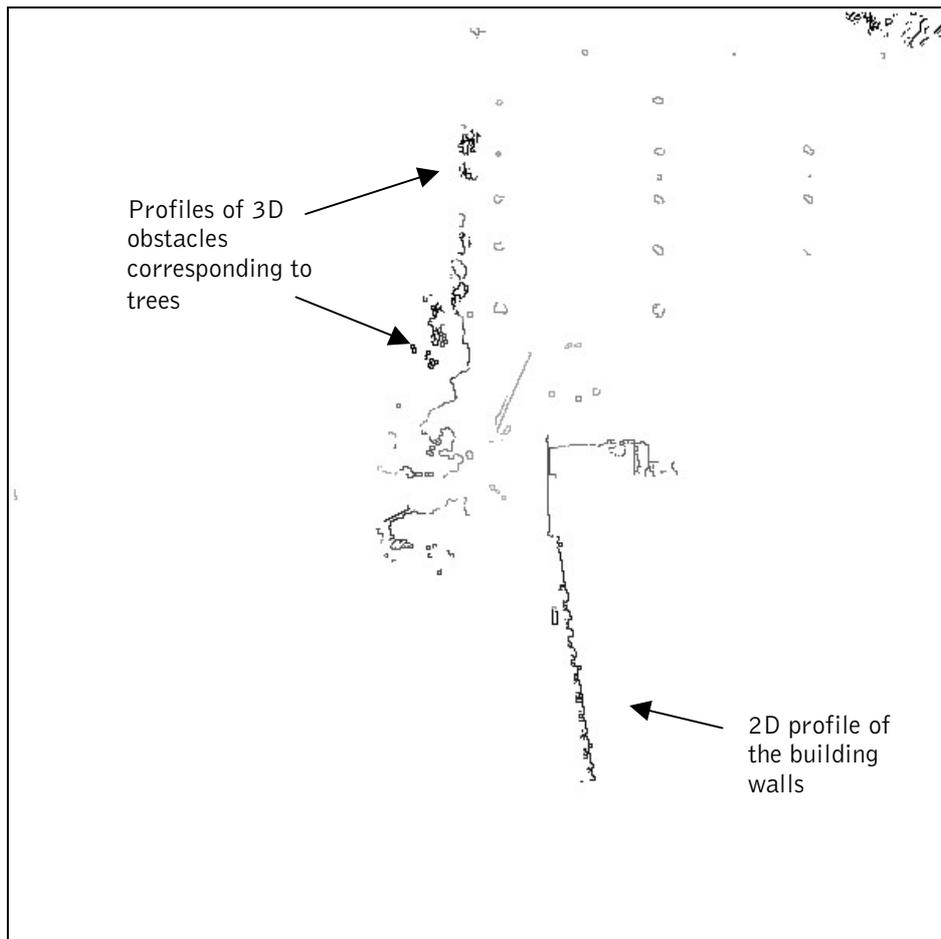


Figure 4.23 A 2D projection of the free space to unknown/occupied space border computed from the 3D occupancy grid and corresponding to the range image "scanE" presented in Figure 4.15a). The darker the colour of the border line, the higher the elevation.

In the image, which has been formed from the scan depicted in Figure 4.15a), the gray scale value at the borderlines corresponds to the height of the free space to unknown/occupied space border with respect to the perception sensor frame of reference. Consequently, the x-, y- and z-point coordinates can be extracted from the height information combined with the corresponding location of the border point in the xy-plane. 3D straight line segments corresponding to the consecutive sets of these points can now be computed.

The computation of the 3D line segments is separated into two phases. First the direction vector of the 3D line segment is computed by passing the 3D point coordinates into a singular value decomposition (SVD) routine [Press et al., 1992], which yields the orthogonal main directions and the relative lengths of the 3D point cloud along the coordinate axes of the reference frame. The direction (unit) vector corresponding to the smallest singular value, or equivalently the largest variance, gives the direction of the 3D straight line segment. This is due to the fact that it is assumed that the spread of the data points along the free space boundary is the largest. A 3D point, through which the line is passing, can be computed as the center of gravity of the 3D point cloud. A similar method has been applied in the context of surface classification for estimating the coefficients of a

plane equation [Flynn and Jain, 1988]. Now, the formula for the 3D straight-line can be given by means of the unit direction vector and the line via point. Taking the furthest projection points of the 3D point cloud with respect to the straight-line model, the two end points of the straight line segment can be approximated. The method is illustrated in Figure 4.24.

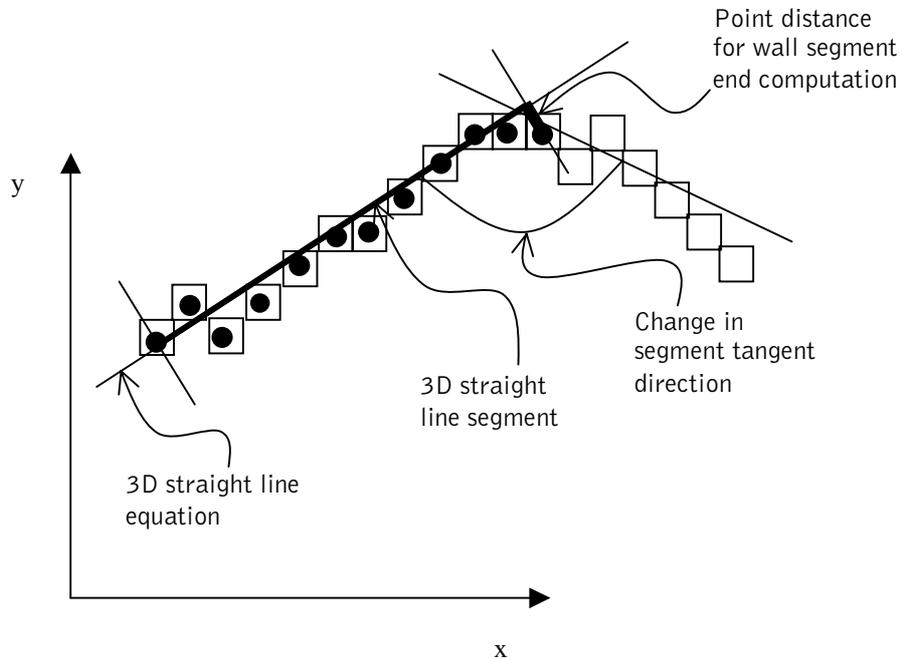


Figure 4.24 In the drawing, the grid cells corresponding to the free space to unknown/occupied space border, are marked with the squares. The 3D border grid cells, which were found to correspond to a nearly straight line segment are marked with the dots.

The straight line segments, which were computed based on the data shown in Figure 4.23, can be seen in Figure 4.25.

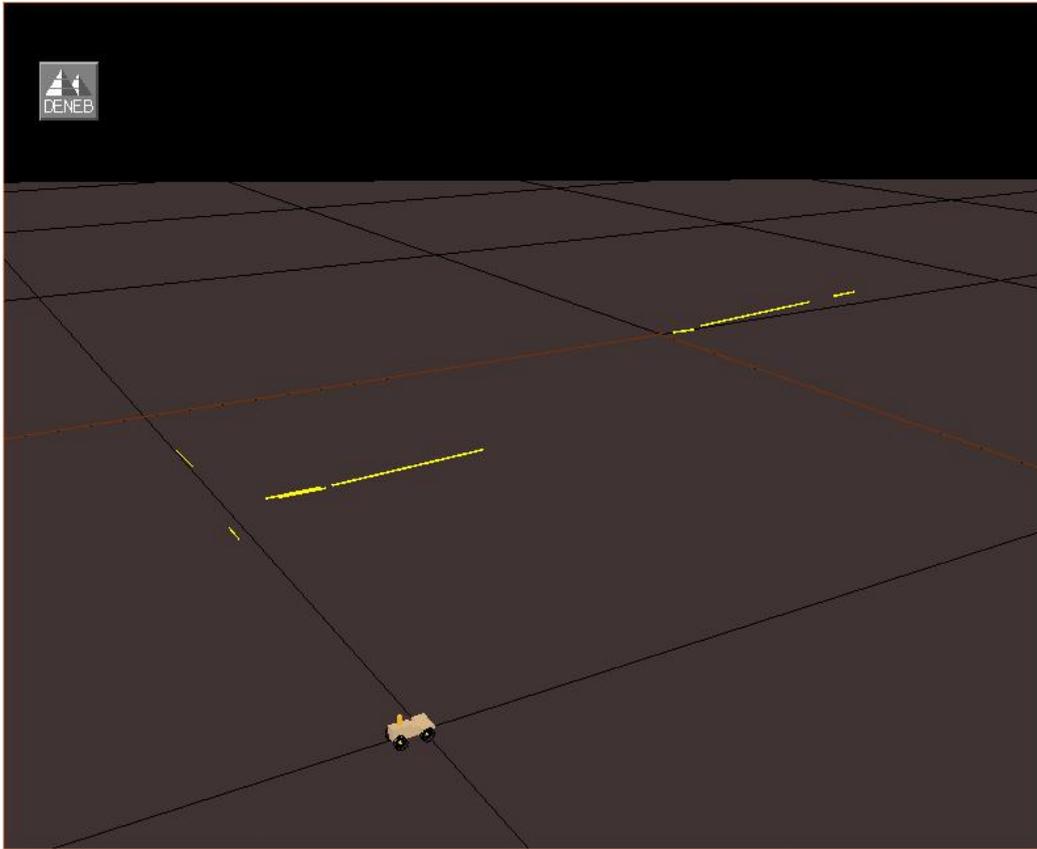


Figure 4.25 A 3D view of the straight-line features corresponding to the wall-ceiling corners extracted from the 3D point data displayed in Figure 4.23. A CAD model of a mobile robot is shown at the location where the range image "scanE" was measured.

4.7.2 Implementation considerations

There are a number of parameters for controlling the process of wall/ceiling border straight line segment estimation. For example, the allowed orientation of acceptable wall/ceiling corner lines with respect to the horizontal plane can be adjusted. In the experiments, presented in this work, this threshold was set at ± 5 degrees, i.e. only horizontal wall/ceiling borderlines were used as landmarks in view-point registration.

As can be seen in Figure 4.25, in some cases, only a part of the, in a large scale straight, wall/ceiling borderlines are successfully transformed into straight line segments. This is often due to local curvature variations in the point data queues generated from the free space to unknown/occupied space borderline data such as the one depicted in Figure 4.23. The local curvature variation can cause a break in the straight line segment, resulting in a large number of short segments that do not pass the "minimum acceptable wall segment length" threshold given in Table 4.3.

Table 4.3 Typical values for the key configuration parameters for wall/ceiling edge segment model parameter computation. Indoor test case is presented on the left side of the table, and outdoor test case is on the right. For the interpretation of the parameters, refer to Figure 4.24.

| Parameter name | Value | Parameter name | Value |
|---|--------------|---|--------------|
| Minimum acceptable wall segment length | 2.0 meters | Minimum acceptable wall segment length | 1.5 meters |
| Segment tangent direction difference threshold | 80.0 degrees | Segment tangent direction difference threshold | 90.0 degrees |
| Interval length for segment tangent computation | 1.0 meters | Interval length for segment tangent computation | 1.4 meters |
| Point distance for wall segment end computation | 0.15 meters | Point distance for wall segment end computation | 0.4 meters |
| Grid cell size along x-, y- and z-dimensions | 0.1 meters | Grid cell size along x-, y- and z-dimensions | 0.2 meters |

The problem can be alleviated by increasing the “segment tangent direction difference” angle threshold parameter (compare with Figure 4.24). However, if the threshold is increased too much, the risk that major changes in the wall direction are not detected increases. Values for the parameters in Table 4.3 were determined empirically, based on trial and error. More results on extracting straight line models from the range images can be found in appendices B and D.

4.8 Conclusions

Two types of feature objects were modeled from range image data. The first object type corresponds to the narrow elongated objects with a circular cross section. An iterative method was developed to form the cylinder segment models (or, more generally, truncated cone segment models) to approximate the volume of the objects. The method is designed in particular for modeling objects that may not be strictly rectilinear or have a perfectly circular cross-section. In the work, only nearly vertical objects (tilted at a maximum of about ± 30 degrees with respect to the vertical direction) were considered. However, the method can be easily adapted to model, for example, horizontally aligned cylinder shape objects.

The other type of landmark objects to be modeled from dense range data were the upper corner lines of vertical walls. These features were selected due to the fact that they usually provide the maximum achievable visibility to the perception sensor. Also, they are horizontally oriented and therefore nicely augment the feature set consisting of the vertical cylindrical objects. The new idea for modeling the wall/ceiling corner lines was related to the utilization of the 3D occupancy grid for the generation of the 3D data points for the computation of the parameters of the straight line equation.

A major problem, related to the extraction of the two feature object types, is related to the determination of the computation configuration parameters. These

parameters contribute to the balance between false negative and false positive modeling results. It is up to the particular application to set preferences between the sensitivity of the modeling system to ignore an existing object or create a model for a phantom object. For the experiments, presented in this work, the parameter values were determined through trial and error. However, an automatic self-calibration function would be a considerable improvement to the presented methods.

The primary motivation to extract the feature object models from the range images was the generation of the 3D data points for the computation of the relative alignment coordinates between a pair of (neighboring) viewpoint locations where range images were acquired. The subject is discussed in Chapter 5.

Chapter 5

Registration of a pair of sub-maps

5.1 Introduction

Within this work, a large-scale environment model is built from local sub-maps by importing the data, contained within the local maps, into the common frame of reference of the integrated map. The first step in this process is the determination of the most likely match of the local sub-maps with respect to the other (overlapping) sub-maps. The most likely match is described as the coordinate transformation that, for example, minimizes the sum of the shortest distances between each of the data points in the first map with respect to all the data points in the second map.

5.1.1 Problem description

Viewpoint registration is the problem of finding an optimal estimate of the translation and rotation that would align the Cartesian coordinate frames located at the origins of the viewpoints. In the probabilistic framework the optimality criterion usually refers to the maximum likelihood estimate. The output of probabilistic viewpoint registration methods is a probability distribution over the translation and rotation degrees of freedom of the relative viewpoint pose parameters. To solve the problem it is assumed that the 3D measurement data from external perception sensors and, optionally, odometry data from the internal sensors of the platform, carrying the perception sensors, is available. The perception data can be in the form of measurement data points (x-, y- and z-coordinates together with corresponding measurement uncertainty) or geometric features extracted from the measurement data. The probabilistic perception model (also called sensor model) can be described with the following conditional probability density (likelihood) function,

$$p(d|\mathbf{x}, m) \tag{5.1}$$

where d corresponds to the sensor data, \mathbf{x} is the *a priori* estimate of the sensor alignment and m symbolises an existing map. The *a priori* estimate of the sensor alignment can be obtained by means of the motion model of the sensor platform. The probability density value, after acquisition of perception data d , that the perception sensor location corresponds to \mathbf{x} , can now be computed as the product of the likelihood of observing d when being in \mathbf{x} with the prior probability that the perception sensor was moved to \mathbf{x} from the previous perception position,

$$p(\mathbf{x}|d, m) = \eta p(d|\mathbf{x}, m)p(\mathbf{x}|m) \tag{5.2}$$

where $p(\mathbf{x}|m)$ corresponds to the prior probability (computed, for example, by means of the odometry information) that \mathbf{x} is the correct sensor location. With η it is ensured that the integral of the probability $p(\mathbf{x}|d, m)$ with respect to \mathbf{x} equals one.

The limits on the acceptable uncertainty of the initial estimate depend on the uniqueness of different parts of the environment to be mapped as well as on the characteristics of the measurement update model. If the environment contains long featureless sections or symmetric (ambiguous) features, good initial prediction of the relative viewpoint position is required. This is the case also if a greedy search method, which uses the predicted position as a starting value, is used to find the optimal alignment. In contrast, an environment with unique features and utilisation of a global search method to find the optimal alignment may permit viewpoint registration without an *a priori* estimate. It should be noted that viewpoint registration problem as described above is analogous to perception based mobile robot localization. In place of the robot vehicle, other means of carrying the sensor can be imagined. For example, the sensor can be moved manually to each viewpoint to collect the perception data. In such a case, odometry information can be replaced with a subjective assessment of the relative transformations between the viewpoints or it can be omitted entirely. Without a good initial estimate, however, it may not be possible to find the globally optimal (and correct) registration.

Viewpoint registration methods found in the literature can be grouped into two categories based on the need to solve explicitly the correspondence problem before the optimal alignment can be computed. In solving the correspondence problem the pairs of corresponding data entities in the current measurement data set, called the local map, and the other data set, called the global map, are found. The distinguishable characteristics of the map objects can be used as the correspondence criteria for finding a pair of data entities from the two maps that belong to the same physical object in the environment. The least squares estimation techniques and the (Extended) Kalman filter algorithm are examples of methods which require that the data elements in the current measurement set and in the reference map must first be associated together. Betgé-Brezetz et al., extracted two types of object-based representations from range data: coarse ellipsoid models and more accurate superquadrics models [Betgé-Brezetz et al., 1995]. Depending on the accuracy requirement of the registration between the local and global maps,

pairs of xyz-points from ellipsoid or superquadrics models are used to compute the least-squares solution for the alignment. Huber and Hebert compute local shape signatures, called spin-images, for a fraction of the 3D points of a high-resolution triangular mesh formed from dense measurement data [Huber and Hebert, 1999; Huber et al., 2000]. After finding a set of pairs of corresponding spin images, an estimate of the transformation that aligns the model surfaces is computed. Thereafter the ICP (Iterative Closest Point) algorithm is applied to find a more accurate alignment.

In the second class of registration methods, in which we do not have to solve the correspondence problem explicitly, we find map correlation computation techniques. Konolige and Chou project the sensor data onto a local discretized occupancy map (called sensor patch), which is correlated with a discretized global map [Konolige and Chou, 1999]. Olson proposed a viewpoint localisation technique that is able to utilise indistinguishable landmarks and is able to find the globally optimal alignment between the local and global maps [Olson, 2000]. In the method the space for possible robot positions is examined by means of a multi-resolution, hierarchical divide-and-conquer strategy. The match score or similarity measure between the transformed local map and the global map is computed as a function of the distances from each data element in the local map to the closest element in the global map. The match score models measurement uncertainty and the possibility of having no counterpart for a data element from one map to another in the second map. This makes the map similarity measure robust to outliers and missing data

In the large scale mapping of indoor and outdoor environments reliance on the availability of distinguishable landmarks may be problematic. Symmetric architectural features or natural, equally looking objects such as trees may complicate the composition of a large enough set of unique feature pairs for registration computation. In that respect, the ideal registration method should possess the following characteristics:

- Robustness to outliers and missing data
- Ability to find globally optimal alignment within the given search space
- Support multiple hypothesis registration result
- Give an estimate on the probability of correctness of the result (optimal alignment may not necessarily be the correct one)

Indeed, the registration techniques that are able to utilise indistinguishable feature data seem to offer a proper basis for the implementation of a generic method for computing the relative alignment between maps. Therefore, in the following chapters, the methods that work directly over the 3D map data sets, without requiring the determination of *a priori* correspondence, are discussed in more detail.

5.1.2 Raw measurement data versus feature-based registration

In order to be able to align two sets of 3D perception data, it is required that overlap between the data sets exists. The required percentage of overlap depends on the registration method. For example, the least-median-of-squares (LMS) method requires an overlap of at least 50 percent between the data sets. If the

“raw” data points, delivered by the measurement device, are used to compute the optimal alignment, a relatively restricted motion between the viewpoints can be allowed. In cluttered environments especially, the contents of the scene visible to the sensor changes rapidly as a function of the perception position. However, if symmetric features are available in the scene they could be used as the source of data points to compute the relative motion between the viewpoints. In this way the requirement of having an overlap over object surfaces would be relaxed to having any side of the same symmetric object within view of the two measurement locations.

5.1.3 Multiple hypothesis registration

In real world applications locations may not always be unambiguously resolvable. Symmetric environment geometry or a lack of features may result in multiple likely hypotheses for the relative alignment between maps. Premature pruning of the somewhat less likely hypotheses at an early mapping phase may prevent us from finding the topologically correct map. Instead, if both of the potentially correct map alignment candidates are saved, new data acquired at a later mapping phase can, retrospectively, help to resolve the ambiguity.

5.2 Methods for finding a locally optimal registration

5.2.1 Iterative, point-based matching

5.2.1.1 Iterative closest point

The iterative closest point (ICP) algorithm is an iterative procedure that converges monotonically to the nearest local minimum [Besl and McKay, 1992]. At each iteration, the closest 3D point in the second map is found for each point in the first map. The rigid motion between the maps is then computed as a function of the point pairs. The rigid motion for example of a 3D point corresponding to the current viewpoint, \mathbf{v}_i , is given by a linear transformation

$$T(\mathbf{v}_i) = \mathbf{R}\mathbf{v}_i + \mathbf{t} \quad (5.3)$$

The 3x3 rotation matrix, \mathbf{R} , and the translation vector, \mathbf{t} , corresponding to the rigid motion can be computed by using a quaternion-based algorithm [Besl and McKay, 1992] or by the singular value decomposition (SVD) method [Haralick et al., 1989]. These methods assume that pairs of data points in the maps to be aligned can be found for computing the least squares solution for the alignment. Unfortunately, even one false outlier pair of data points can cause the result to deteriorate. Transition to a combination of random sampling and least-median-of-squares (LMS) estimation techniques is one way to alleviate the problem [Masuda and Yokoya, 1994]. However, the number of subsets of pairs to be sampled from the original set of feature pairs to find an outlier free sample with high probability may not be computationally affordable.

A drawback of the ICP algorithm is that it does not give an estimate of the uncertainty, related to the computed rotation and translation parameter values, nor

can the possibly different error values of the individual data points be taken into account. A solution to circumvent the problem is to use ICP to solve the correspondence problem and then apply the Extended Kalman Filter to yield the final estimate of the registration parameters [Madhavan et al., 1998].

5.2.1.2 Iterative dual correspondence

The convergence properties of the ICP algorithm have been found to be rather slow, especially if the object's surfaces, from which the data points are extracted, are curved. This is due to the fact that the *closest-point* correspondence rule seems to provide little information about the rotation between the two observation locations [Lu and Milios, 1994]. Lu and Milios propose a different correspondence rule, which represents the situation when the translation movement between the two observation locations is negligible. In this case, a pair of points, whose distance from the corresponding measurement frames is the same, have a difference in their polar angles that corresponds (with high probability) to the unknown rotation displacement between the measurement frames. The situation is illustrated in Figure 5.1.

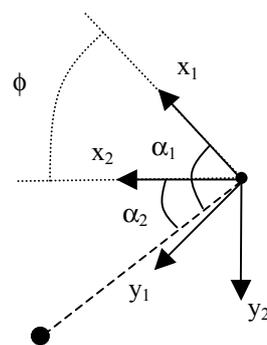


Figure 5.1 Two observation frames with only a mutual rotation displacement. The measured angle to the observed target point changes from α_1 to α_2 .

In the figure, ϕ is the unknown rotation between the measurement frames and α_1 to α_2 are the polar angles to the same physical position on the object's surface with respect to the first and second observation frames respectively. Now, for every point in scan i a pair is found in scan j by using both the closest point rule and the matching (nearly equal) range rule. Then, a least squares solution for the displacement between the observation locations is found for both sets of point pairs. And, finally, the rotation component of the matching range rule solution and the translation component of the closest point rule solution are selected as the relative location update to iteratively search for the optimal estimate of the rigid motion between observation locations. The method is called the iterative dual correspondence (IDC) algorithm [Lu and Milios, 1994].

5.2.2 Gradient ascent in search space

In this case it is assumed that the likelihood score, computed over the registration search space, can be described with a monotonically increasing function and gradient ascent techniques can be applied to find a locally optimal registration. In the method, the gradients of the registration score function are evaluated at the current relative map location with respect to each of the degrees of freedom of the search space. Then, the current relative location coordinates are updated in the direction given by the gradient vector. The step size of the update can be determined, for example, as a function of the local curvature of the likelihood function.

5.3 Global registration methods

The global registration methods are guaranteed to find the global maximum of the registration score within the given search space. The uncertainty related to the *a priori* alignment between the maps can be used to determine the size of the required search space. In real-time applications and in patch type map building processes the speed/precision requirements of the methods may differ significantly.

A local registration method, such as iterative closest point matching, converges on the nearest local maximum of the cost function. To find the global maximum the local method can be combined with a preprocessing step which yields a starting position close to the globally optimal alignment. Alternatively, the local search method may be started at several random locations within the search space and the registration corresponding to the best of the local maximums is selected as the globally optimal registration. However, the required number of random locations may be difficult to determine in practice, especially if the cost function has many closely lying maximums. In such a case the number of random starting locations for the local search method, which would guarantee that the global optimum is found with a high probability, may become unacceptably large.

5.3.1 An augmented local method

In some cases, the most effective search strategy can be designed as the combination of a local, greedy search method, preceded by a coarse registration step.

5.3.1.1 Local method preceded by a coarse global registration step

A local registration method, such as the gradient descent search or the iterative closest point algorithm, can be combined with an exhaustive, global search phase. The purpose of the global search method is to restrict the search space to a small neighborhood around the global maximum of the registration score function.

5.3.1.2 Combination with simulated annealing

As it was stated before, iterative closest point algorithm (ICP) converges to the closest minimum of the objective function. If the algorithm could be started from a number of locations within the registration state space that would cover all the regions of attraction, the best of the local minimums would yield the globally optimal alignment [Besl and McKay, 1992]. Usually the error surface is not analyzed but the algorithm is started at random locations and the best solution is used. A better alternative is to use simulated annealing (SA) to escape a local minimum that has been found by ICP [Luck et al., 2000]. In this way, SA produces guided restarts for ICP. This process is repeated until the error score is driven below a given threshold.

5.3.2 Maximum correlation registration

In a correlation based registration method, the similarity between two data sets is computed over the given registration space. The alignment corresponding to the maximum similarity then yields the most likely registration. The cross-correlation between two stochastic functions can be given by [Weiss et al., 1994],

$$c(y) = \lim_{X \rightarrow \infty} \frac{1}{2X} \int_{-X}^X f(x)g(x+y)dx \quad (5.4)$$

where $c(y)$ corresponds to the similarity measure computed as the function of the unknown alignment y . The functions $f(x)$ and $g(x+y)$ may correspond, for example, to distances to the environment objects as a function of the polar scan angle of the data points in the first data set and the sum of the polar scan angle and the rotation shift between the observation frames corresponding to the second data set, respectively. As already noted in Chapter 5.2.1.2, the distances of the measurement points from the measurement frames can be, ideally, used to resolve the unknown displacement, which (practically) only consist of the rotation component. For the situation where the relative displacement to be estimated consists of both the rotation and translation components, Weiss et al. propose the angle histogram [Weiss et al., 1994]. The computation of the angle histogram is most conveniently described if each measurement in the measurement set is considered a position vector pointing from the origin of the measurement frame to the measured position. The angle histogram is then formed by recording the angles of the differences of each pair of consecutive measurement vectors with respect to the measurement frame. The maximum of the angle histogram gives the rotation component of the relative displacement between the observation locations. Maximums in the angle histogram are due to linear surface segments visible to the perception sensor. In structured environments the peaks in the angle histogram usually correspond to building walls. After applying the rotation transformation to the data points of the second data set, the alignment along the translation degrees of freedom can be computed, for example, by using an occupancy grid based technique. Before that, the range measurements have to be projected onto an occupancy/evidence grid map.

In the occupancy grid framework, the correlation score for a particular alignment can be computed by summing up the results of occupancy values of each grid

element of the transformed grid map multiplied with the occupancy value of the corresponding grid element in the other map. Schultz et al., compared two different map matching score computation techniques for evidence grids [Schultz et al., 1999]. In the binary match technique score "1" was returned if the grid values in both maps indicated evidence of an empty grid cell, i.e. the value of the element in the range $[-1, 0)$. "1" was also returned if the grid values in both maps indicated evidence on an occupied grid cell, i.e. value of the element in the range $(0, 1]$. In case of a disagreement between the grid elements, "0" was returned. In the alternative, product match technique, all the corresponding cell values in the current measurement data-based map and the existing map were multiplied and added to the total score. In the study as well, two different search techniques were compared to search for a more accurate robot alignment with respect to the existing map in the close neighborhood of the robot's *a priori* pose estimate. In both of the search methods the search space was divided into discrete steps. The combination of product match score with a search technique that takes pose samples for score computation throughout the given search space and then computes the center of the mass of the cell scores as the optimal registration was selected for future work.

5.3.3 Divide and conquer search based on Hausdorff metric

5.3.3.1 Map similarity measure

By following the formulation in [Olson, 2000], let us assume that the local map, \mathcal{B} , corresponding to the current viewpoint for data acquisition, contains n data elements $\{\mathbf{b}_1, \dots, \mathbf{b}_n\}$. The other map, \mathcal{A} , which is sometimes called the "global map", may correspond to the data collected from another viewpoint or a more complete integrated map, contains m data elements $\{\mathbf{a}_1, \dots, \mathbf{a}_m\}$. Let us mark a Cartesian coordinate transformation, which maps the data elements in the local map (map \mathcal{B}) to the global map (map \mathcal{A}) with T_k . Now, the (Euclidean) distance between a data element \mathbf{b}_i of the transformed local map and a data element \mathbf{a}_j in the global map is given by

$$d_{ij}(T_k) = \text{dist}(T_k(\mathbf{b}_i), \mathbf{a}_j) \quad (5.5)$$

The distance from a map data element in the local map to the closest data element in the global map is called $D_i(T_k)$.

$$D_i(T_k) = \min_{1 \leq j \leq m} d_{ij}(T_k) \quad (5.6)$$

The assumption that these distances are not correlated is not strictly speaking true, but modeling them as such has been found to yield good results [Olson, 2000]. Now, we can formulate a likelihood function for the perception model.

$$L(T_k) = \prod_1^n p(D_i(T_k)) \quad (5.7)$$

A good approximation for the probability density function that models both the precision of data elements and the presence of outliers in the local map is given by the sum of a normal distribution and a constant factor.

$$p(D_i(T_k)) = k_1 + \frac{k_2}{\sigma\sqrt{2\pi}} e^{-(D_i(T_k))^2/2\sigma^2} \quad (5.8)$$

where the constant factor k_1 makes a lower bound for the probabilities between unmatched pairs of data elements. For the lower bound factor the average distance of a feature point in map A with respect to a feature point in map B is computed over all possible alignments within the registration search space. It will correspond to the average mutual distance of a pair of outlier points. Now, constant factor k_1 is given as the product of the assessed share of outlier points in data set B and the probability density value corresponding to the average outlier distance. The constant factor k_2 corresponds to the share of inliers with respect to the total number of data elements in the local map. The standard deviation of measurement uncertainty of a data element, which is an inlier, is marked by σ . In general, the map registration task is not sensitive to the values of these constants. It should be emphasized that $p(D_i(T_k))$ is not a probability distribution, because it does not integrate into one. However, the use of a score function such as Equation 5.8 is unavoidable if we want to have a lower bound for the value given by the score function for large distances $D_i(T_k)$ [Olson, 2000]. For computational convenience, the likelihood function, Equation 5.7, which was formed as the product of the likelihood score values, can be replaced with the sum of the natural logarithms of the score values [Olson, 1998] yielding,

$$\ln L(T_k) = \sum_{i=1}^n \ln p(D_i(T_k)) \quad (5.9)$$

Because the logarithm is a monotonic function, the ordering of the hypotheses of the relative map positions is preserved. If an estimate of the *a priori* alignment between the perception positions is available, it can be incorporated into the likelihood score function, Equation 5.9, yielding

$$\ln L(T_k) = \ln p(T_k) + \sum_{i=1}^n \ln p(D_i(T_k)) \quad (5.10)$$

where $p(T_k)$ models the *a priori* distribution of the likelihood for the relative alignment of map B with respect to map A .

5.3.3.2 Global search strategy

To find the globally optimal alignment between the local and global maps the likelihood score, Equation 5.9, is first computed for the predicted position of the local map with respect to the global map. Then the transformation space is divided into rectilinear cells (at a coarse resolution) and each cell is tested whether it could contain a position that is better than the best position found so far. The cells, that

cannot contain a position that could contribute to a higher score value, are removed from the search space and the other cells (i.e. stretches in the transformation space) are divided into two parts along the longest search direction and are tested in the same way. The progress of the search, in a two-dimensional registration example, is illustrated in Figure 5.2.

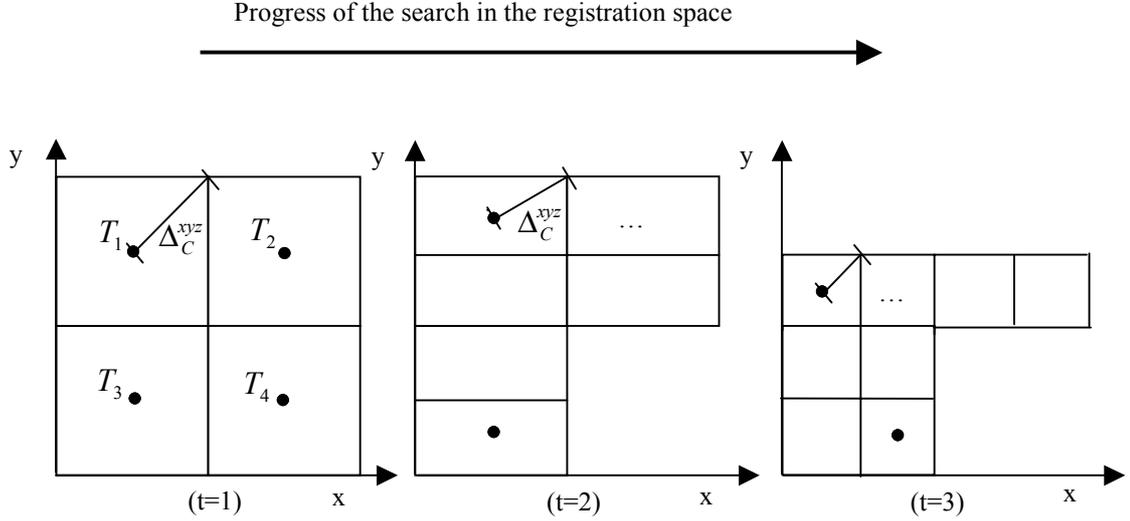


Figure 5.2 Multi-resolution, divide and conquer search, for the maximum likelihood registration with respect to the x- and y-axis. T_k indicates the center of the cell and Δ_C^{xyz} is the maximum distance of the center from the border of the cell. The consecutive search phases are indicated by the index t .

To test cell C_k , in a conservative way, the distances of the data elements in the map B with respect to the closest data elements in the map A , $D_i(T)$, are computed for the transformation T_k corresponding to the center of the cell. Then value Δ_C is subtracted from each data element distance to compute an upper bound for the best overall likelihood score that can be found within the transformation subspace enclosed within the boundaries of cell C_k ,

$$\text{MAX}_{X \in C_k}(L(X)) = \sum_{i=1}^n \ln p(D_i^C(X)) \quad (5.11)$$

where

$$D_i^C = \max(D_i - \Delta_C, 0) \quad (5.12)$$

It is easy to compute Δ_C for the translation degrees of freedom of a search space stretch as illustrated in Figure 5.2. Δ_C^{xyz} is computed simply as the distance of a corner from the center of the 3D rectilinear cell. This value is independent of the locations of the data elements within the maps to be aligned. A somewhat more complicated task is the computation of the upper bound for the change in the

distance between data elements in the local and global maps due to a span in the heading direction Δ_C^ϕ . This value is a function of the distance of a data element from its origin on the local map. For a particular rotation movement corresponding to the half of a cell in the discretized heading direction search space, the distance change that a data element undergoes due to the angular change of the (heading) orientation of the local map with respect to the global map, can be computed as

$$\Delta_C^\phi = 2d_e \sin\left(\frac{\phi}{2}\right) \quad (5.13)$$

where d_e corresponds to the distance of a feature from the origin of the local map and ϕ is half of the rotation angle of the search space span under consideration. The computation of Δ_C^ϕ is depicted in Figure 5.3.

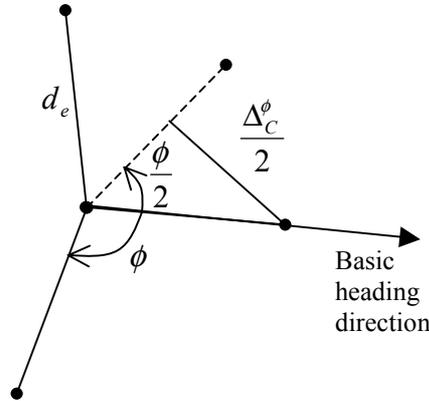


Figure 5.3 Computation of the relative distance of a data point, Δ_C^ϕ , due to a rotational movement corresponding to ϕ , which is half of the z-rotation dimension of the discretized search space span. The distance of the data point from the origin of the local map frame is marked by d_e .

Values for Δ_C^{xyz} and Δ_C^ϕ can be computed off-line and stored into a lookup table. The maximum distance change due to the dimensions of the cell in the search space to determine an upper bound for the likelihood value corresponding to a data element in the local map can be computed as the sum of the above values.

$$\Delta_C = \Delta_C^{xyz} + \Delta_C^\phi \quad (5.14)$$

Actually, the sum gives a rather conservative measure for the upper bound. This has the drawback that more cells will survive the pruning phase, which slows down the registration computation. A more precise value for the upper bound distance due to the dimensions of the pose cell in the transformation space, Δ_C , can be computed by projecting the distance change component values Δ_C^{xyz} and Δ_C^ϕ along each of the x-, y- and z-coordinate directions, summing them up and then computing Δ_C as the square root of the squared sums of axis-wise upper bound

distances. The search for a more precise alignment can be continued by always dividing the cells in the search space along its longest dimension until a desired level of precision is achieved.

5.3.3.3 Gaussian approximation of the registration uncertainty

After having reached the desired level of accuracy within the discretized search space a sub grid cell size solution for the localization problem can be computed. Olson proposed that the log-likelihood values at the close neighborhood of the maximum value could be fit to the parabolic equation to yield the parameters of a Gaussian distribution [Olson, 1998]. In the experiments, we found this kind of uncertainty approximation unsatisfactory, when the original distribution has a rough, relatively non-symmetric form. Instead, we compute the center-of-gravity of the connected score cell cloud around the corresponding maximum score cell to serve as the registration parameter estimate. The uncertainty of the estimate can then be computed as the normalized, weighed sample covariance of the connected score cloud. The Gaussian approximation of the registration uncertainty will be utilized to compute an optimal integrated map from the network of relative alignment dependencies between the sub-maps as will be described in Chapter 6.4. The optimal, *relative observation-global localization* map building method was presented in Chapter 3.2.2.

5.3.3.4 Assessment of correctness of the registration result

As the computed relative alignment coordinates that correspond to the maximum overlap between the two feature maps, are not necessarily correct, Olson proposed a measure to evaluate the correctness of the result [Olson, 1999]. The correctness measure is based on the assumption that the majority of the registration score mass is concentrated within the immediate neighborhood of the registration peak in the case that the found registration is the correct one. However, the percentage of score mass required to be concentrated around the peak alignment result seems to depend on the distribution, overlap and richness of the features sets on the two maps to be registered. It makes the applicability of the correctness evaluation technique difficult.

Even though the correctness of a relative location estimate of a sub-map with respect to another map may be difficult to resolve separately, the characteristics of an integrated map built from several local maps may provide information about the existence of a false registration result. In Chapters 6.4 and 7, the possibilities to assess the correctness of the constructed integrated map will be discussed in more detail.

5.4 Probabilistic registration of feature maps

The map based viewpoint registration method to be presented is based on the work of Olson and colleagues [Olson, 2000], discussed in Chapter 5.3.3. In their implementation, the relative alignment of the new robot location (or equivalently the viewpoint location) with respect to the previous location was sought for along

the x- and y-translation coordinate directions. They also briefly discussed how the method could be extended to consider rotation degrees of freedom.

Our contribution is related to the implementation and experimental verification of the method for finding an optimal alignment between the viewpoints in three-dimensional translation and one-dimensional rotation space. The remaining two rotation degrees of freedom, i.e. the inclination angles of the sensor platform, are measured explicitly. The data points, among which the maximum correlation (overlap) is sought, are formed by sampling the straight line features at desired intervals, as depicted in Figure 5.4.

Let us briefly explain how the environment mapping method, discussed in this work, proceeds. First, 3D perception data is collected at N locations in the environment. Second, geometric features are extracted from each of the N sets of perception data (i.e. sub-maps) with relatively good precision described by the Euclidean distance ε_{if} , where i is the index of the sub-map and f identifies the feature within the sub-map. Each of the N sets of extracted features, annotated with a Cartesian frame of reference, constitute a sub-map. In the third mapping step, the mutual relative alignment of pairs of overlapping sub-maps is computed. For the computation of the alignment, sets of 3D data points are formed from the geometric feature models within the sub-maps. For example, the data points may be generated by sampling from the geometric feature model, such as the center line of a cylinder segment model. In that case, from the sub-map registration point-of-view, a major part of the uncertainty of the sampled data points will be determined as a function of the spacing between the sampled points on the line model. The dependence is due to the fact that the locations from where the point samples are taken do not coincide in practice. Ideally, the sampled locations on the centerline segments of the corresponding cylinder models in the two maps should correspond to the same locations with respect to the physical object (such as a section of a tree trunk). The situation where the physical object to be modeled corresponds to a section of a tree trunk is illustrated in Figure 5.4.

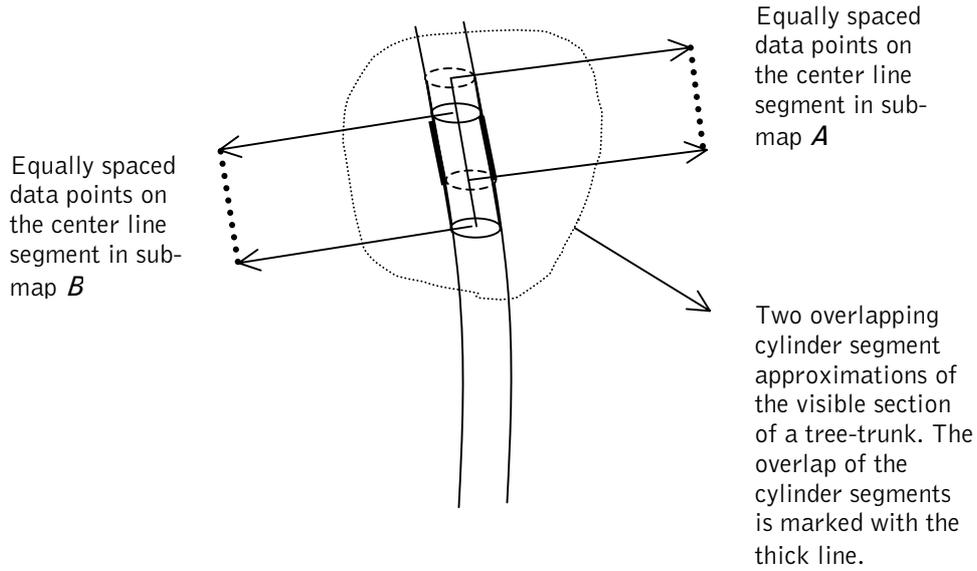


Figure 5.4 Two sequences of xyz-data points sampled from the cylinder segment models corresponding to the visible sections of the same tree trunk observed from the origins of the sub-map frames **A** and **B**.

In the figure, the point sets, for the overlapping part, will be sampled from slightly different physical spots of the tree trunk. The uncertainty, due to the mismatch of the closest points in data sets **A** and **B**, can be evaluated as the function of the spacing, Δ_d , between the two consecutive point samples,

$$\varepsilon_d = \frac{\Delta_d}{\sqrt{12}} \quad (5.15)$$

where ε_d is the standard deviation of the uncertainty of the sampled data points and Δ_d is the spacing between two consecutive sampled data points. Note, that Equation 5.15 corresponds to the standard deviation of a rectangular distribution within the interval $\pm \frac{1}{2} \Delta_d$. Now, if the sampling interval is 50 millimeters, the uncertainty, which is related to the assumption that the nearest data point in set **B** with respect to a data point in set **A** actually corresponds to the same physical location on the centerline axis of the tree trunk, becomes 14.4 millimeters. Note that the analysis is relevant to the approximation of the uncertainty of the data points extracted from the overlapping parts of the geometric feature models. The uncertainty of the data points in set **A** which have no counterpart in set **B** has to be considered separately. More information on approximating the uncertainty of unmatched points can be found in Chapter 5.3.3.1. Now, if ε_d can be proven to be considerably larger than the Euclidean distance values corresponding to the uncertainty of the geometric feature models from which the data points were sampled, i.e.,

$$\varepsilon_d \gg \varepsilon_{if} \in (i, f) \quad (5.16)$$

the explicit uncertainty value, ε_{if} , for the particular geometric feature model can be omitted. However, the number of measurement points to compute an estimate for the unknown parameters of a geometric feature model is limited in practice and an explicit estimate for the uncertainty of the feature parameters may become important to consider at the later stages of a mapping process.

A third source of uncertainty is related to the assumption that the circle model is a good approximation of the cross section of a tree trunk. This is not exactly true and an additional uncertainty component is created due to this fact. Altogether, all these sources of uncertainty should be summed up to approximate the uncertainty of an inlier data point.

The search for the location, in the discrete search space, that yields the maximum "match" score value between the two 3D point sets is illustrated in Figure 5.5.

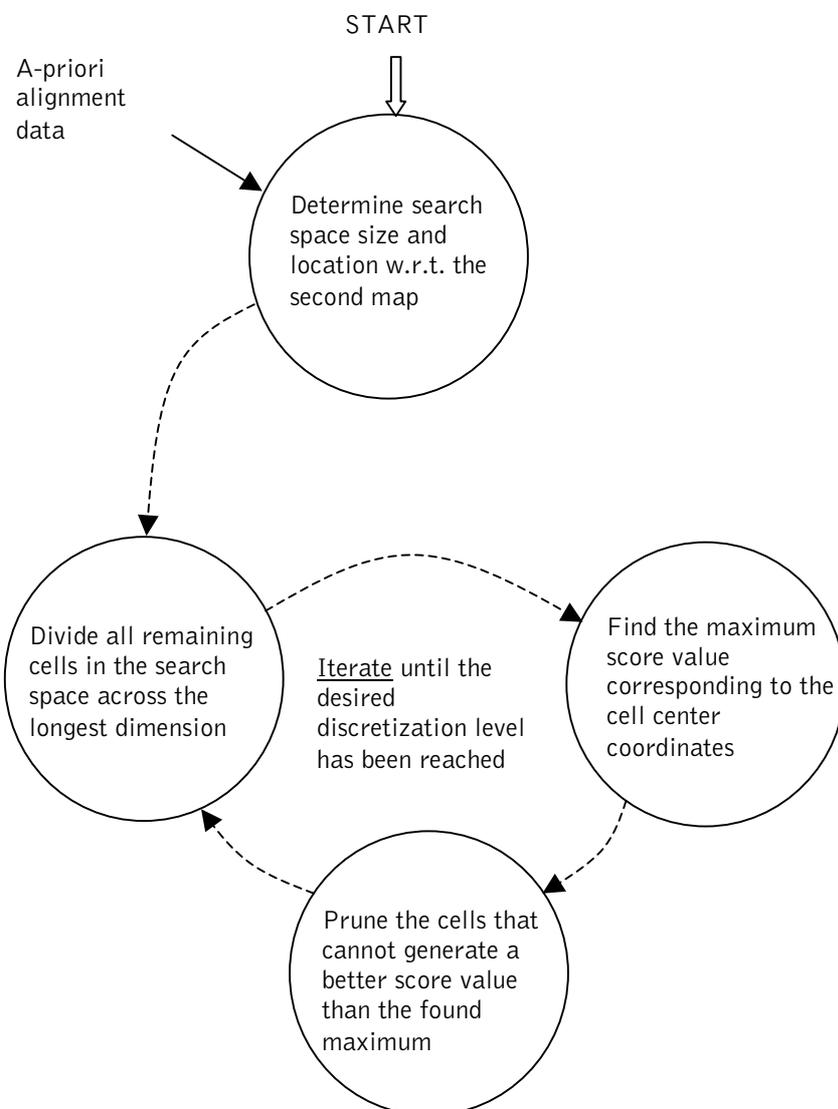


Figure 5.5 A schematic diagram of the divide and conquer alignment search method. The method is able to find the location in the multidimensional (here four dimensional) discrete search space, that corresponds to the maximum score value.

5.4.1 Experimental results

In order to implement the method, described in Chapter 5.3.3, the parameters for the map similarity measure have to be determined. The parameters, assigned to Equation 5.8, were given the following values: the standard deviation of a single data point, σ , extracted from the feature models was given the value 0.035 meters. This uncertainty value corresponds to an inlier feature point, for which there is assumed to be a counterpart on the other map. Its value was determined to reflect the uncertainty from two main sources: The first uncertainty component comes from the discrete sampling interval of the data points. This component, computed by means of Equation 5.15 yields an uncertainty of 1.44 centimeters for the sampling interval of five centimeters. The other source of uncertainty is related to the deviation of the cylinder approximation model from the actual tree trunk object. This has been assumed, based on feature model construction experiments, to be in the range from two to five centimeters. The share of inliers parameter with respect to the total number data points, k_2 , was set at 0.3 (i.e. 30 percent). Parameter k_1 , in Equation 5.8, was computed by using the distance threshold of 0.3 meters to classify data points as inliers or outliers. This means that the lower bound score value for any pair of closest data points in the two maps was computed by assuming the distance between the data points newer to be greater than 0.3 meters. (For more detail refer to Chapter 5.3.3.1).

For the computation of the relative alignment between two sub-maps, the initial size of the search space has to be defined. The size values with respect to x-, y- and z-translation degrees-of-freedom (dof) and with respect to the heading rotation can, for example, correspond to the uncertainty of the *a priori* estimate of the relative alignment. Also, the desired accuracy of the divide and conquer search in the form of the final size of the pose cell has to be determined. The search will stop as the final size of the pose cell, for example 0.04 meters, along each of the four search dimensions is reached. For the rotation dof of the 4D pose cell, an angular value corresponding to the size value, given in meters, can be determined by assigning the average horizontal distance of the 3D point features from the origin of sub-map B into the Equation 5.13.

Now, let us compute the relative alignment of the sub-map "F" with respect to the sub-map "E" (*note: real sub-map data sets are referenced with a label enclosed within the quotes. For example, the label "X" refers to the corresponding range image "scanX"*). The corresponding range images as well as the cylinder segment features and the straight line features, extracted from the range images have been illustrated in Figures A.1, A.2, D.7 and D.8 respectively. For the experiment the following results, presented in Table 5.1, were acquired.

Table 5.1 Results for the computation of the relative alignment between sub-maps "F" and "E". Units are meters for the x-, y- and z-coordinates and degrees for the z-rotation angle. The alignment estimate corresponds to the discrete registration space cell size of 0.04 meters.

| Search space degree of freedom | Initial search space dimensions along both positive and negative directions of the given dof | <i>A priori</i> alignment | Computed alignment estimate | Standard deviation of the estimate |
|--------------------------------|--|---------------------------|-----------------------------|------------------------------------|
| x | 5.0 | 11.0 | 12.37 | 0.04 |
| y | 5.0 | 0.0 | 0.03 | 0.08 |
| z | 1.25 | 0.0 | 0.15 | 0.37 |
| ϕ | 180.0 | -90.0 | -87.75 | 0.43 |

In the table, the size of the search space, along with both negative and positive directions of the given dof, are presented. The total transformation space covered was twice the given values. Thus, for the z-rotation, the entire search space corresponds to 360 degrees. The discretization threshold for stopping the search was set at 0.02 meters along both negative and positive search directions corresponding to a length of the pose cell of 0.04 meters along each of the translation dofs and 0.35 degrees around the heading angle search direction. The *a priori* alignment was determined by subjectively assessing the traveled distance between the measurement locations. (The laser scanner was mounted on a manually maneuvered cart). The estimate, given in Table 5.1, was computed from the connected score cloud around the score cell corresponding to the maximum score value. The estimate corresponds to the center of gravity of the connected score cloud and the standard deviation of the estimate was given as the squared root of the sample variance parameters. The predicted estimate is rather accurate compared to the computed alignment estimate. However, it is not necessary for the method to find the alignment corresponding to the maximum registration score value, as long as the optimal alignment is located within the search space borders around the predicted alignment coordinates [Forsman, 2001].

If one of the feature sets consists of only a small number of features or the distribution of the features in the environment is not unique, multiple good registration peaks within the search space may be found. Such a situation can appear in structured environments if, for example, wall planes are used as natural landmarks for viewpoint registration. In that case, a good initial alignment estimate is desired.

The convergence properties of the divide and conquer registration search algorithm can be studied by means of the following example, illustrated in Figure 5.6. In the figure, the peak alignment, corresponding to the pose cell with the highest score value, against five different alignment discretization search thresholds are plotted with the solid line. In the same images, the location of the estimate, corresponding to the center of gravity of the score mass around the peak alignment, is plotted with a dash-dotted line for reference. Theoretically, the center of gravity alignment

estimate should yield a more accurate, sub-pixel estimate for the alignment coordinates. In the figures, the corresponding final center of gravity alignment estimate is shown for reference with the horizontal dashed line. The size of the pose cell (along each of the four search directions) in the discrete search space corresponding to 0.64, 0.32, 0.16, 0.08 and 0.04 meters are indicated with the numbers 1,2,3,4 and 5 on the horizontal axis.

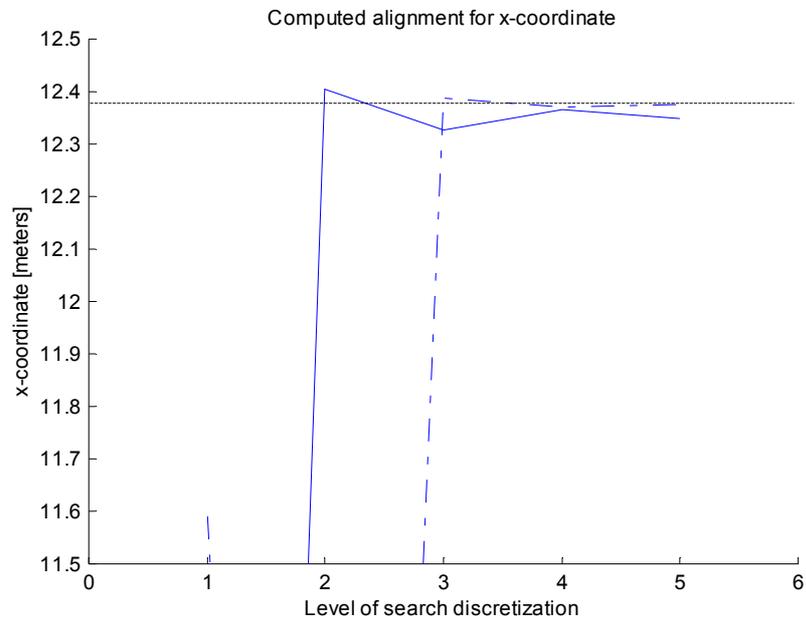


Figure 5.6 a) Convergence of the alignment search with respect to the x- search direction.

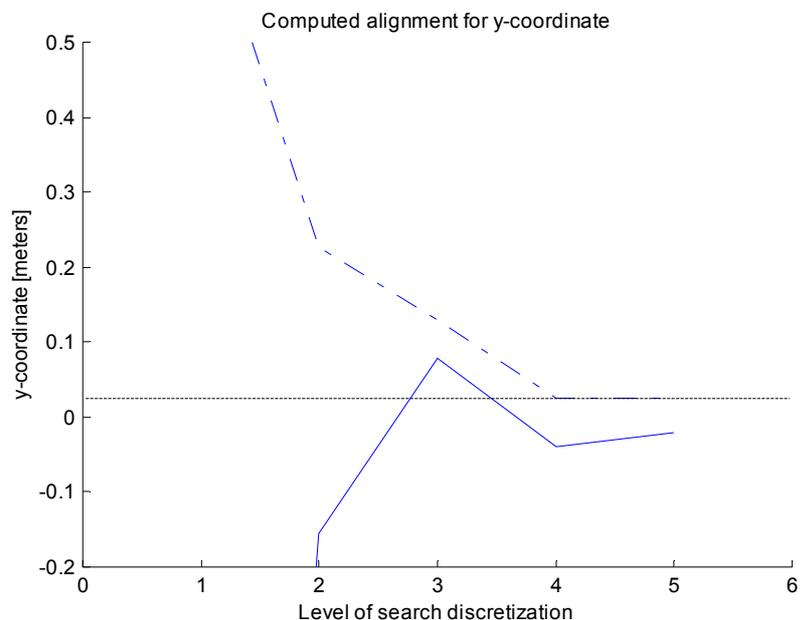


Figure 5.6 b) Convergence of the alignment search with respect to the y-axis search direction.

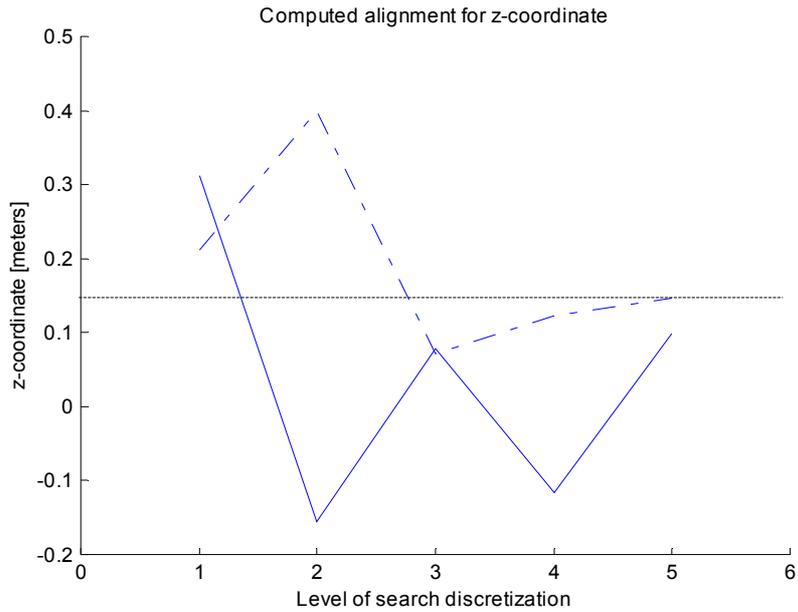


Figure 5.6 c) Convergence of the alignment search with respect to the z-axis search direction.

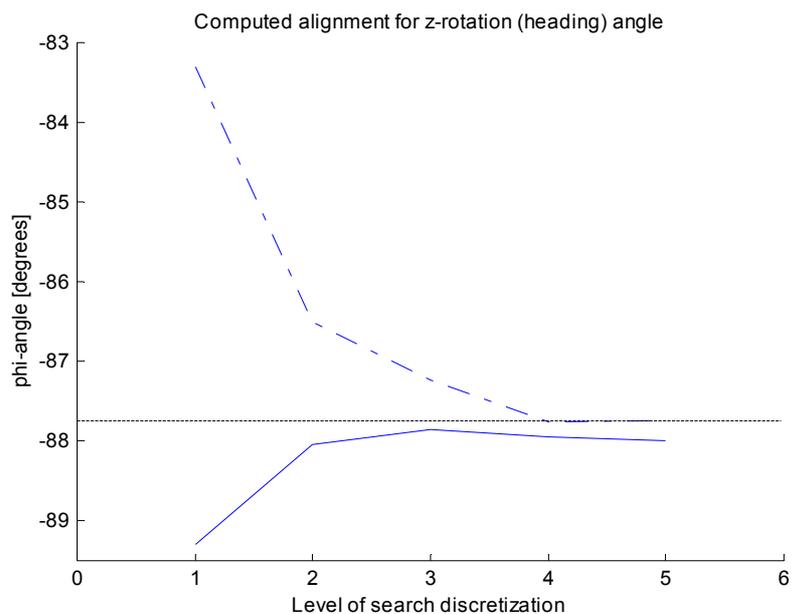


Figure 5.6 d) Convergence of the alignment search with respect to the (z-rotation) phi angle search direction.

From the results of Figure 5.6 it can be observed that the center of the score cloud-based estimate, marked with a dash-dotted line, yields a more accurate alignment result than the alignment corresponding to the highest registration computation score value only at the end of the search. This is due to the fact that with larger score cell sizes the score (probability) mass is distributed, asymmetrically, over a much larger area around the highest peak.

As presented in Figure 5.6, two different alignment values are computed in the registration experiments. First, the alignment corresponding to the center of the search space cell of the highest overall score value is recorded. And second, the

connected set of search space cells around the highest score alignment cell are grouped into a score cell cloud. Then the center of gravity location and the sample standard deviation dimensions of the cell cloud are computed. These values are recorded in Tables 5.2 and 5.3 for the indoor test environment when using both the actual and corrupted *a priori* alignment values as the center of the divide and conquer search space. The range images, called "scan1" up to "scan8", which were used in the indoor mapping experiment, are illustrated in Appendix A, in Figures A.11 to A.18.

Table 5.2 Viewpoint alignment results for the indoor map building experiment. The real *a priori* alignment value, given by the operator, was used as the center of the search space. The human operator of the sensor subjectively assessed the *a priori* alignment values (without using any measurement instruments). The dimensions of the search space were ± 5 meters in x- and y-directions, ± 0.625 meters along the z-direction and ± 30 degrees around the *a priori* heading direction. The results correspond to the alignment search space cell size of two centimeters along each of the search directions (for the ϕ -angle search direction, a corresponding angular threshold was used). Units are meters for the translation displacements and degrees for the rotation angle.

| | | |
|---|-----------------------------|----------------------------|
| Registration pair labels | 21 | 32 |
| Real <i>a priori</i> alignment given by the operator | 0.0, -14.0, 0.0, 0.0 | 0.0, -22.0, 0.0, 10.0 |
| Highest score alignment | 0.32, -13.70, 0.11, 0.88 | -0.93, -21.62, -0.15, 5.37 |
| Score cloud center alignment | 0.3, -13.68, 0.13, 0.93 | -0.91, -21.63, -0.14, 5.4 |
| Sample standard deviation of score cloud center alignment | 0.03, 0.03, 0.04, 0.14 | 0.04, 0.02, 0.02, 0.14 |
| 43 | 54 | 65 |
| 21.0, 0.0, 0.0, 10.0 | 0.0, 7.0, 0.0, 180.0 | 0.0, -12.0, 0.0, 90.0 |
| 20.74, 0.58, 0.11, -87.6 | 0.46, 7.58, 0.22, 178.24 | 0.32, -11.72, 0.05, 90.64 |
| 20.74, 0.59, 0.12, -87.53 | 0.45, 7.56, 0.08, 178.28 | 0.32, -11.72, 0.05, 90.65 |
| 0.02, 0.07, 0.02, 0.22 | 0.03, 0.02, 0.19, 0.18 | 0.06, 0.02, 0.02, 0.3 |
| 76 | 87 | X |
| 0.0, -20.0, 0.0, 0.0 | 0.0, -18.0, 0.0, 0.0 | X |
| -0.01, -20.2, -0.28, -1.99 | -1.42, -13.03, -0.13, 0.35 | X |
| -0.02, -20.21, -0.27, -1.96 | -1.42, -13.03, -0.127, 0.35 | X |
| 0.05, 0.03, 0.03, 0.2 | NOT COMPUTED | X |

Table 5.3 Viewpoint alignment results for the indoor map building experiment. The corrupted *a priori* alignment value was used as the center of the search space. The dimensions of the search space were ± 5 meters in x- and y-directions, ± 0.625 meters along the z-direction and ± 30 degrees around the *a priori* heading direction. The results correspond to the alignment search space cell size of two centimeters along each of the search directions (for the ϕ -angle search direction, a corresponding angular threshold was used). Units are meters for the translation displacements and degrees for the rotation angle.

| | | |
|---|----------------------------|----------------------------|
| Registration pair labels | 21 | 32 |
| Corrupted <i>a priori</i> alignment | 3.0, -14.0, 0.0, 15.0 | 2.0, -25.0, 0.0, -5.0 |
| Highest score alignment | 0.31, -13.71, 0.15, 0.88 | -0.92, -21.63, -0.17, 5.37 |
| Score cloud center alignment | 0.30, -13.67, 0.13, 0.93 | -0.91, -21.63, -0.14, 5.39 |
| Sample standard deviation of score cloud center alignment | 0.03, 0.03, 0.04, 0.14 | 0.04, 0.02, 0.02, 0.14 |
| 43 | 54 | 65 |
| 18.0, -2.0, 0.0, -75.0 | 2.0, 10.0, 0.0, 165.0 | 2.0, -10.0, 0.0, 105.0 |
| 20.73, 0.61, 0.11, -87.48 | 0.49, 7.57, 0.09, 178.24 | 0.29, -11.71, 0.05, 90.53 |
| 20.74, 0.59, 0.11, -87.52 | 0.43, 7.56, 0.06, 178.43 | 0.32, -11.72, 0.05, 90.65 |
| 0.02, 0.06, 0.02, 0.21 | 0.06, 0.03, 0.04, 0.26 | 0.06, 0.02, 0.02, 0.3 |
| 76 | 87 | X |
| -2.0, -17.0, 0.0, -15.0 | -2.0, -20.0, 0.0, 15.0 | X |
| -0.01, -20.2, -0.28, -1.99 | -1.33, -15.17, -0.13, 0.35 | X |
| -0.02, -20.21, -0.27, -1.96 | -1.31, -15.16, -0.12, 0.62 | X |
| 0.05, 0.03, 0.03, 0.2 | 0.02, 0.02, 0.02, 0.4 | X |

The viewpoint registration results documented in Tables 5.2 and 5.3 indicate that the *a priori* alignment values do not have an effect on the computed alignment values. This is, however, only partly true. The range image data used for the experiment was acquired in an indoor environment, which contains symmetric features. In another map building experiment, the *a priori* alignment value for the viewpoint pair "43" was corrupted by adding three meters to the x-coordinate. This caused the alignment search space to extend up to the location of the next pillar on the corridor. As there happened to be more overlap from the wall/ceiling

straight line segments part, the optimal alignment was computed at a location where the wrong pair of pillars in the corridor were in alignment. The result was a seven meter shift (the distance between the pillars) in the computed registration values.

The results for the registration pairs "21", "32", "43", "54" and "65" can be declared to be correct by visual inspection. During the visual inspection the wall planes and cylindrical pillars were both checked for alignment. The registration results for pair "76" seem to be in accordance with the predicted alignment despite the fact that no pillar objects were extracted from the "scan7" data set. Theoretically, it is very unreliable that the correct alignment can be computed along the direction of the corridor based on straight line features of the same orientation. When computing the optimal alignment between viewpoints "7" and "8", the wrong alignment contributed to the highest registration score value. Between the data sets, straight line segments from different sides of the corridor were found to be in alignment corresponding to the optimal registration result. This kind of situation could be avoided by applying simple heuristics that would prevent the comparison of wall planes from different sides of the corridor. Also, by assuming that the true *a priori* alignment information is accurate, the search space could be restricted to a small span (e.g. ± 0.5 meters) along the x-axis (perpendicular to the walls) of the previous viewpoint location. This could also have helped to find the correct alignment. Elevation maps and feature models, corresponding to the individual range scans and to the integrated models formed by means of *a priori* alignment information and computed registration values, are presented in Chapter 6 and Appendix D.

In Figure 5.7 the evolution of score distribution as a function of the search space cell size is depicted. The results have been stored during the computation of the relative alignment for the sub-map registration pair "21" (corrupted *a priori* alignment case, Table 5.3). The size of each of the six score grid maps is 10 meters square. The connected score cloud has been "grown" by starting from the cell in the four-dimensional search space having the highest score value. Within the score distribution images, the x- and y-coordinates of the highest score value and the center of gravity of the connected score cell cloud are marked with a simple and an ended cross hair respectively. The lengths of the vertical and horizontal lines of the ended cross hair correspond to the computed sample standard deviation of the score cloud. As already indicated by the sample standard deviation values in Tables 5.2 and 5.3, the sizes of the score clouds along the x- and y-directions are at the end of the search (corresponding to the cell size of two centimeters) very small. Literally, this would mean that the uncertainty of the registration result is small and almost equal in all cases independent of the "richness" of the feature sets used for registration computation. Normally this should not be the case. On the other hand, at the earlier stages of the search, a more visually trustworthy score distribution appears. Perhaps the whole history of score distribution evolution should be utilized to approximate the uncertainty of the final registration coordinate values. This remains the subject of future research.

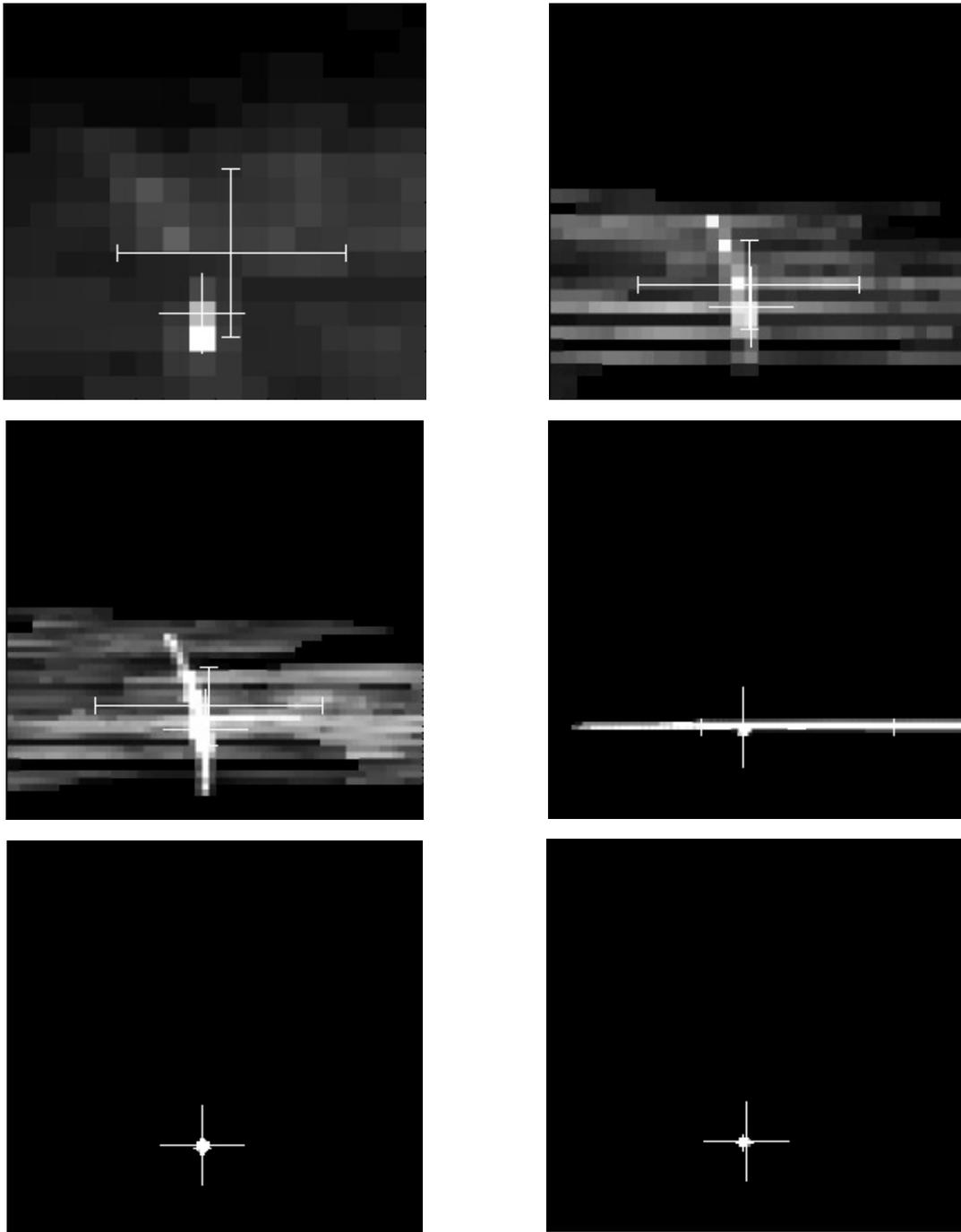


Figure 5.7 Registration score distribution for the sub-map pair "21" (corrupted *a priori* alignment case, Table 5.3). Score distribution for search space cell sizes of 0.64, 0.32, 0.16, 0.08, 0.04 and 0.02 meters are depicted in pictures starting from the upper left image up to the image on the right hand side in the bottom row. In the images, the distribution of the cumulative score values of the connected score cloud have been projected onto the x- and y-coordinate plane of the search space. The directions of the x- and y-coordinate axis are upwards and to the left respectively.

As discussed earlier, in the registration experiment for sub-map pair "43" different results have been obtained depending on the values of the *a priori* alignment. In the following two figures, the evolution of the connected score cloud distributions,

related to the largest and second largest search space cell score peak values, is depicted. The size of the search space with respect to both x- and y-coordinates, depicted in the images, is ± 5 meters around the *a priori* alignment coordinates.

In the top row images of Figures 5.8 and 5.9, there is one large cloud of connected score cells, within which the x- and y-coordinates of the largest and second largest peaks are marked with a simple cross-hair, respectively. The sample standard deviation and the location of the center of gravity of the large connected score cell cloud are the same in both figures. In the middle row images of Figures 5.8 and 5.9, it can be observed that the location of the maximum score cell changes place at the earlier stages of the search. As small cell sizes are reached, only the maximum score cloud survives and the others disappear. The size of search space cells, when the score cloud distribution condenses in the small neighborhood of the single highest peak, seems to be around 0.08 meters in the experiment. The effect of different parameters that determine the search space discretization level, when the search has reached a "mature" stage, should be better understood. It would then help to find a sensible estimate for the uncertainty of the computed score value. The uncertainty, computed at the finest discretization level, may not contain much information.

5.5 Conclusions

In Chapter 5, the registration, i.e. the relative location computation between two perception positions, was discussed. A method that is able to find the globally optimal alignment coordinates within the four-dimensional search space was implemented. The convergence properties of the method were studied through experiments. The problems related to the assessment of the precision of the registration result were adduced.

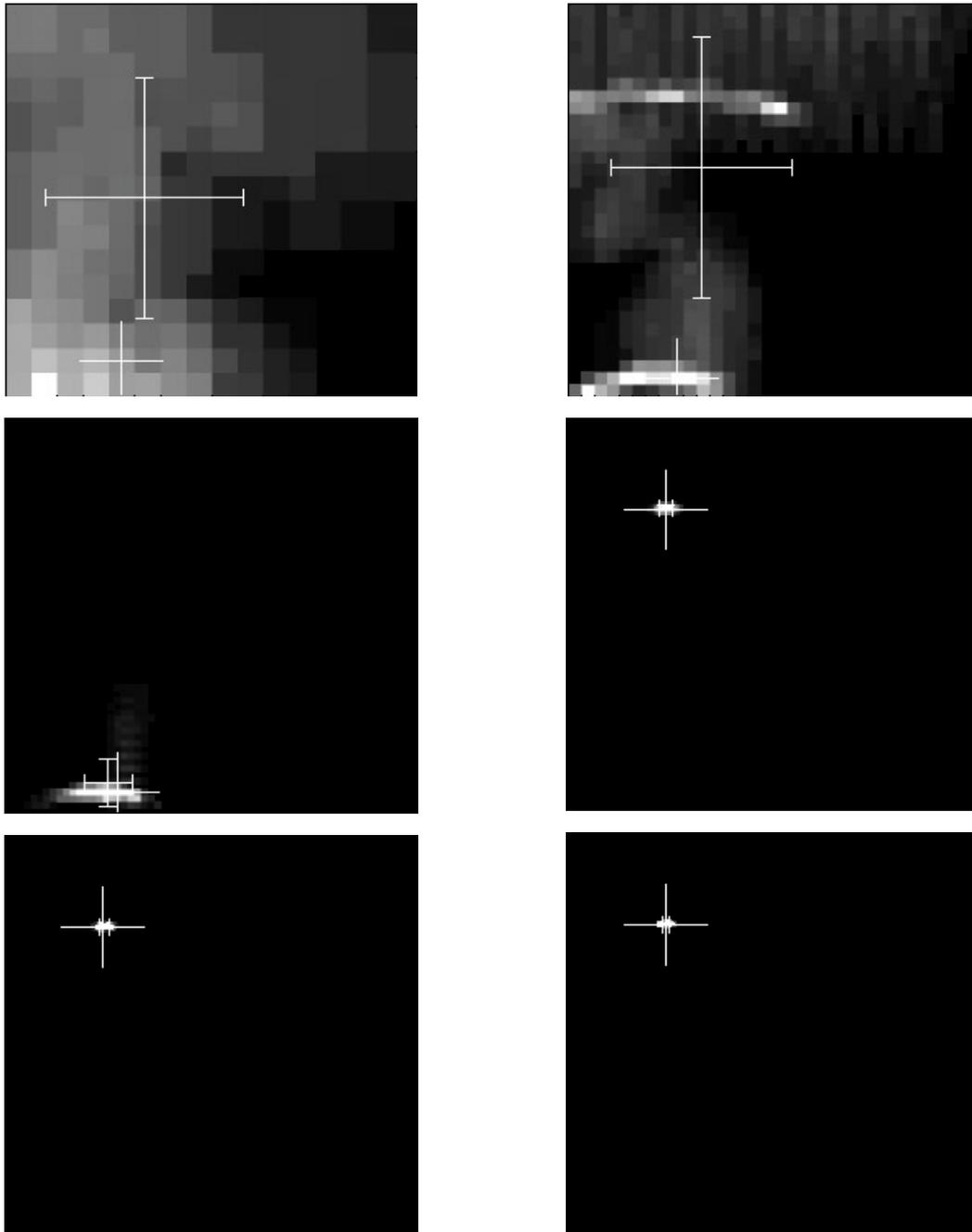
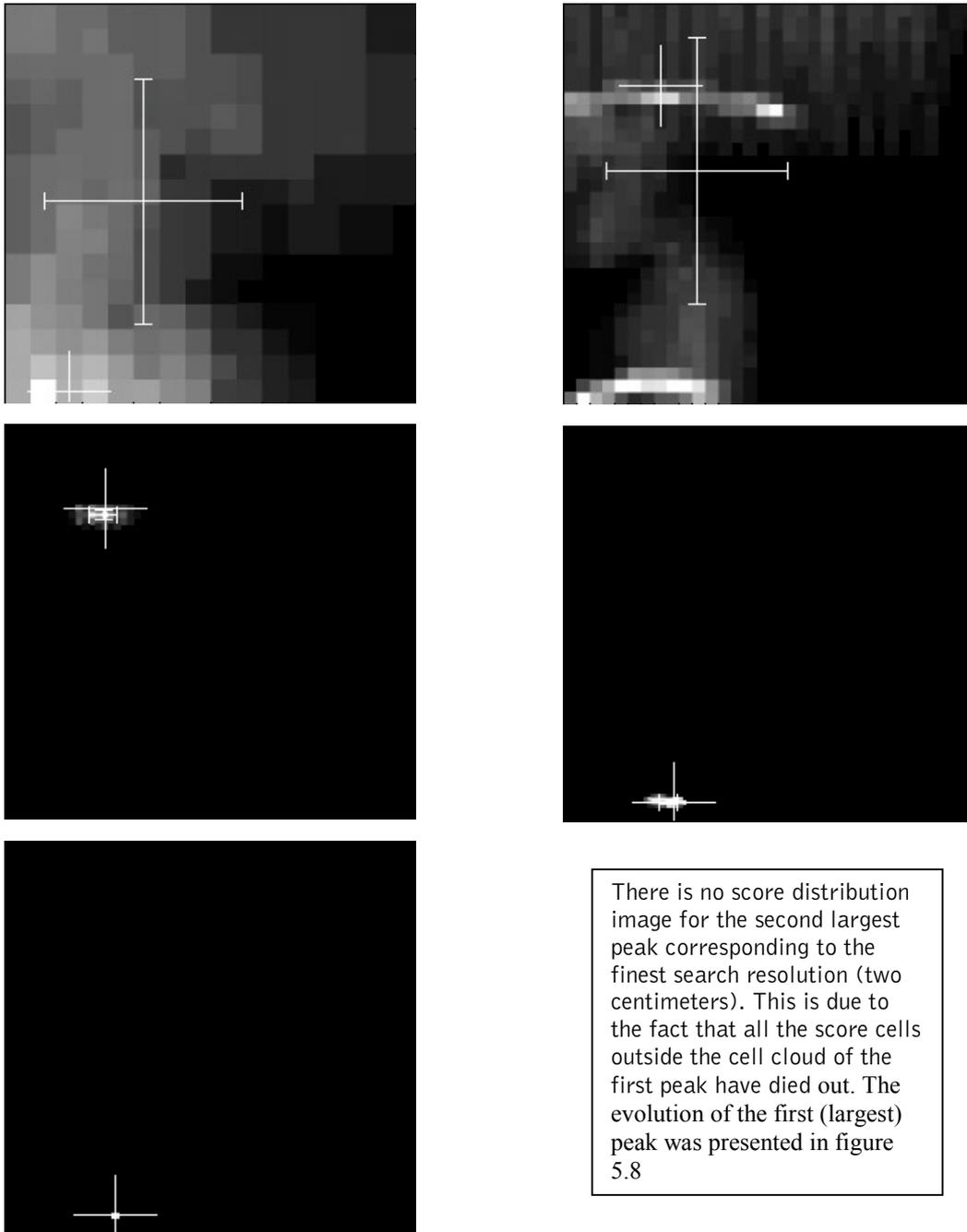


Figure 5.8 Distribution of the cell score cloud corresponding to the largest peak in the registration experiment for sub-map pair "43". Score distribution for search space cell sizes of 0.64, 0.32, 0.16, 0.08, 0.04 and 0.02 meters are depicted in the pictures starting from the upper left image up to the image on the right hand side in the bottom row. In the images, the distribution of the cumulative score values of the connected score cloud, corresponding to the largest peak score cell, have been projected on the x- and y-coordinate plane of the search space. The directions of the x- and y-coordinate axis are upwards and to the left respectively.



There is no score distribution image for the second largest peak corresponding to the finest search resolution (two centimeters). This is due to the fact that all the score cells outside the cell cloud of the first peak have died out. The evolution of the first (largest) peak was presented in figure 5.8

Figure 5.9 Distribution of the cell score cloud corresponding to the second largest remote peak in the registration experiment for sub-map pair "43". Score distribution for search space cell sizes of 0.64, 0.32, 0.16, 0.08, 0.04 and 0.02 meters are depicted in pictures starting from the upper left image up to the image on the right hand side in the bottom row. In the images, the distribution of the cumulative score values of the connected score cloud, corresponding to the second largest peak score cell, have been projected onto the x- and y-coordinate plane of the search space. (Note that the distance of the second peak with respect to the largest peak must exceed the given threshold in order to be accepted as another valid alignment hypothesis.) The directions of the x- and y-coordinate axis are upwards and to the left respectively.

Chapter 6

Environment modeling experiments

6.1 Introduction

The method for mapping large-scale environments has been tested by generating a metric model for three different kinds of environments. The first test site is a parking lot area nearby the Computer Science Building of the Helsinki University of Technology, depicted in Figure 6.1. The second test site covers part of the ground floor of the Computer Science building. The third test site, which represents unstructured, natural environments, is a small forest area.



Figure 6.1 Two camera views into the parking lot, which serves as the first test site of the map building experiments.

6.1.1 Description of the experimental system

The experimental system consists of the Riegl LMS-Z210 range imaging sensor, which was mounted on a manually maneuvered cart for mapping the parking lot area and the indoor office corridor spaces. In the forest test case, the sensor was mounted at the top of a tripod. In case of the cart, tilt sensors were used to measure the inclination angles of the sensor platform. When the tripod served as sensor platform, it was leveled horizontally so that the tilt angles were zero. Also, the operator of the sensor platform was asked to subjectively assess the translation displacement and heading direction change between the perception positions, where range data was collected. The four subjectively estimated transformation parameters served as *a priori* alignment information in the registration experiments.

6.1.1.1 The Riegl LMS-Z210 sensor

The LMS-Z210 is a sensor designed for the acquisition of 3D range images. A rotating mirror directs the internal laser range finder's transmit beam over a precise angular pattern. Based on the time-of-flight of the received beam, the distance to the nearest target in the direction determined by the scanning angles of the beam is measured. Dimensional drawings of the LMS-Z210 sensor are presented in Figure 6.2.

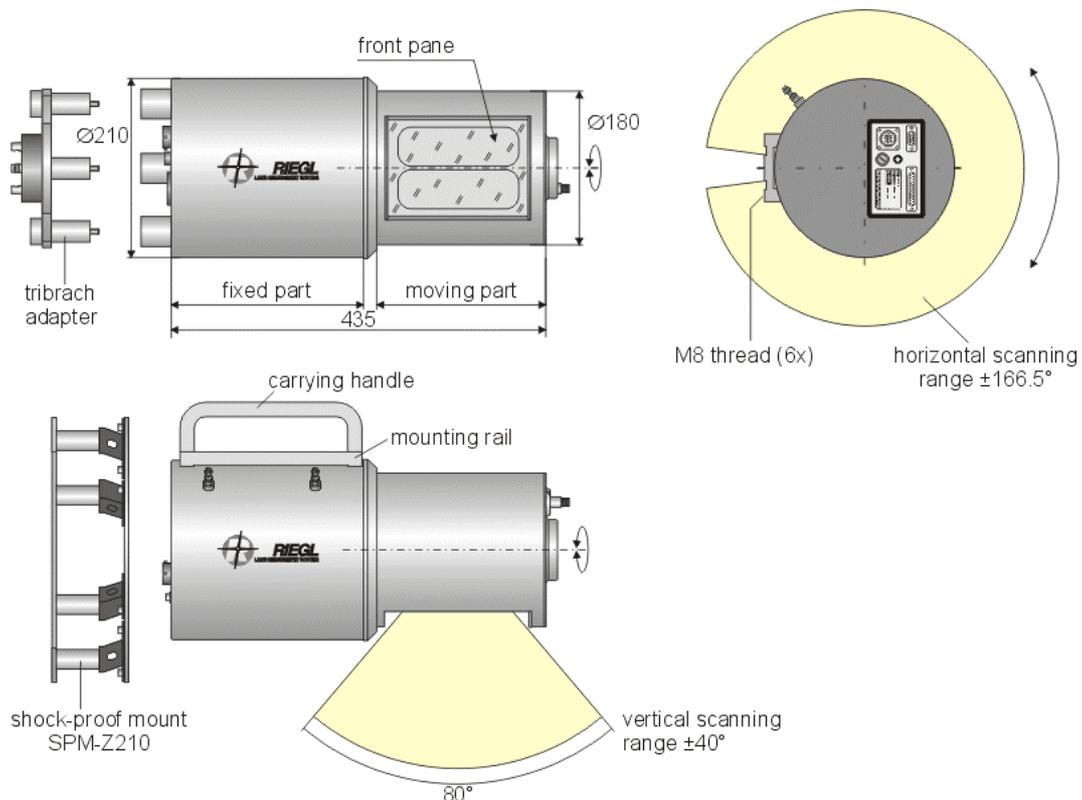


Figure 6.2 Dimensional drawings of the 3D measurement sensor used in the map building experiments. (For more information refer to: http://www.riegl.co.at/lms-z210/e_lms-z210.htm).

From a map building experiment point of view, the most important parameters of the Riegl range imaging sensor are given in Table 6.1. In order to make the device eye safe, the laser beam is spread so that its diameter is about four centimeters when it leaves the sensor housing. Then the diameter of the beam increases about 3 centimeters per 10 meters of measured distance. Due to this fact, it is somewhat difficult to measure through the clutter, especially in forest environments. Also, the wide beam complicates the measurement of small objects.

Table 6.1 Main technical data of the Riegl LMS-Z210 range scanner

| Parameter name | Value | Parameter name | Value |
|--|--------------------|--|---------------------------------------|
| Maximum measurement range (typical) | 350 meters | Fast scan angle readout resolution | 0.036 degrees |
| Basic measurement accuracy (standard deviation) | ± 0.025 meters | Slow (frame) scan angle readout resolution | 0.018 degrees |
| Laser beam diameter when it leaves the scanner housing | 0.042 meters | Measurement principle | Time-of-flight (first returned pulse) |
| Laser beam divergence | 3 mrad | Maximum measurement rate | 10kHz |

The readout resolution (0.036 degrees) of the scanner corresponds, in terms of spatial resolution, to an error of 2 centimeters at a distance of 30 meters from the sensor. At 30 meters' distance the beam diameter is 13 centimeters, which causes another source of error because the exact beam hit location can only be guaranteed to be within the 13-centimeter diameter beam pattern. In the feature object model computation experiments, described in Chapter 4, the distance measurement uncertainty (i.e. 2.5 centimeters) was used to represent the standard deviation of a single measured beam hit location. Obviously, this underestimates somewhat the true uncertainty, but due to the large number of measurement points, which were used to generate the environment models, the significance of knowing the uncertainty of a single measurement point accurately, becomes less important. The range images taken with the scanner are presented in Appendix A.

6.1.2 Description of the mapping method

The applied mapping method consists of three main phases. First, feature object models are computed by means of the range image data. In the test cases, when considering the parking lot area and the indoor office corridor environment, models for both the cylindrical nearly vertical objects and for the border lines between the wall/ceiling planes were extracted. In the forest mapping experiment, only the tree trunks were modeled. The extracted feature objects were described with respect to the reference frame of the corresponding range image, forming in this way a local feature map. In every experiment, a grid map containing the highest/lowest measured elevation value at the location of each grid element was also formed. The elevation data was used to augment the appearance of the models for subjective assessment of the results.

In the second phase of the method, an estimate for the relative alignment between relevant pairs of local feature maps was computed. In the parking lot and indoor environment experiments only consecutive pairs of sub-maps were registered with each other. In the forest mapping experiment, all possible registration pairs between the six sub-maps (altogether 15 pairs) were considered. However, for the forest mapping results, presented in Chapter 6.4, only 10 sub-map pairs were used to compute the optimal environment model. For these 10 sub-map pairs the chances that there would be enough features originating from the same physical objects were considered to be good enough. This kind of manual selection of sub-map pairs violates the autonomy requirement of the mapping method. In Chapters 6.4 and 7.1, possibilities to detect erroneous registration results, enabling thus fully autonomous map construction, will be discussed.

In the third mapping phase, an integrated map was formed by transporting the sub-map model data into a global reference frame. Two different techniques were used to compute the global coordinates for each sub-map. In the simplest technique, applied in the parking lot and indoor environment test cases, the global location of each sub-map is computed by incrementally connecting relative coordinate transformations between pairs of feature maps. In the second, statistically more sound technique, a linearized system of equations connecting the unknown global location estimates with the previously computed relative transformation parameters, is formed. By solving the linearized system of equations, optimal (in the least squares sense) global sub-map coordinates can be computed. The optimal method was applied in the forest mapping experiment.

Note, that, in the mapping experiments, the a-priori alignment estimate was not used to weight the perception based relative alignment observations. This was due to the very large uncertainty related to the subjectively assessed a-priori displacement estimates for the consecutive perception positions. In the experiments, the a-priori alignment information was utilized only to determine the location and size of the search space for the divide-and-conquer registration computation method. In Figure 6.3, a schematic diagram of the developed mapping method is presented.

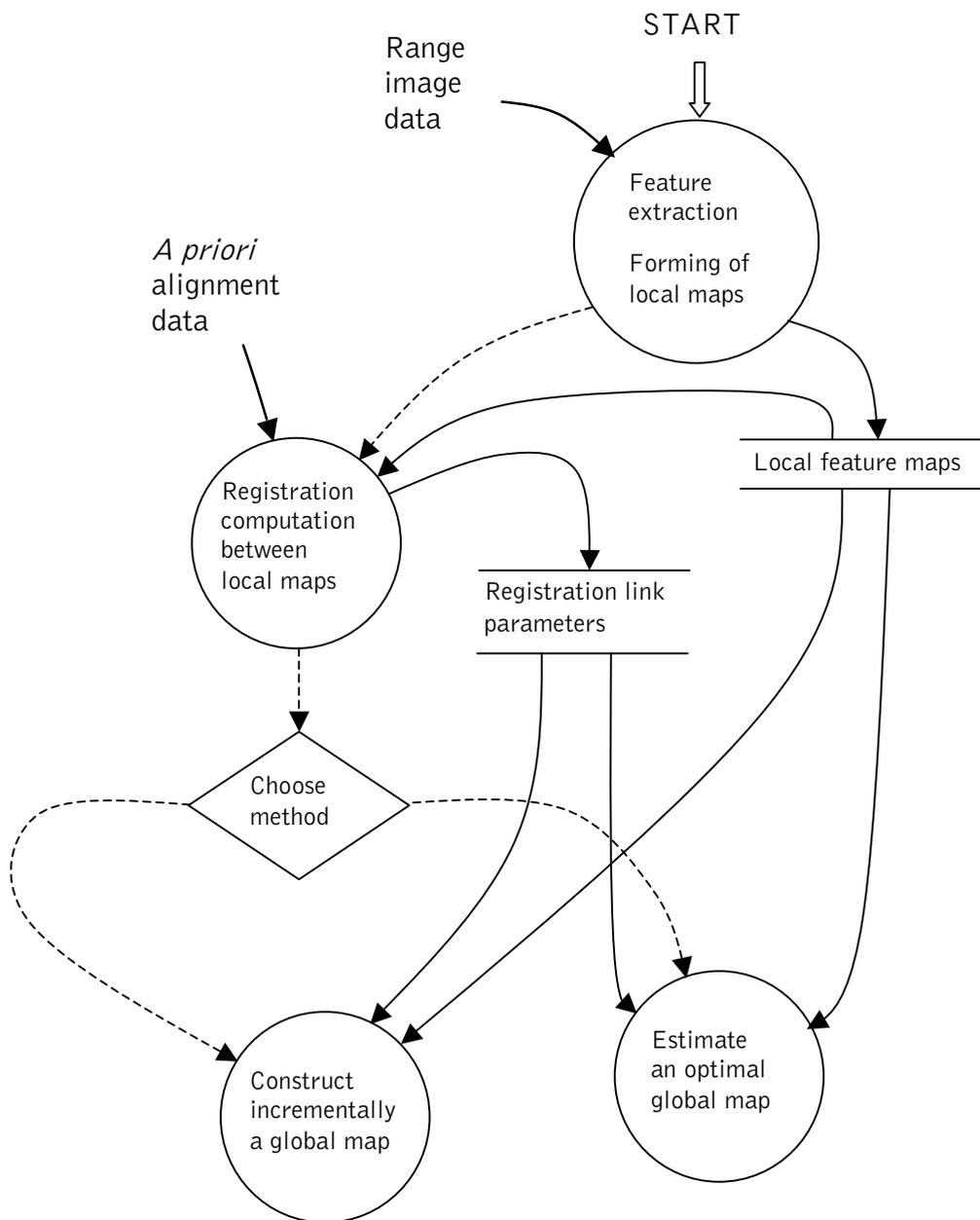


Figure 6.3 A schematic diagram of the environment mapping method. In the diagram, solid and dashed line arrows indicate data access and flow of control, respectively.

The detailed description of the three mapping phases can be found in Chapters 4.6 and 4.7 for the feature extraction phase and in Chapters 5.3.3 and 3.2.2 for the map registration and optimal map estimation phases respectively.

Within the experiments presented in this work, the computational complexity of the feature extraction and the sub-map registration phases was approximately equal. Both of the phases took, for mapping a particular test environment, in total tens of minutes or even multiple hours to complete. The third mapping phase, related to the construction of an integrated map, required only a few seconds of processing time. For example, in the forest mapping experiment, described in Chapter 6.4, the

extraction of the cylinder segments corresponding to the visible sections of the tree trunks took, altogether, about three hours for the six range images. In contrast, the computation of the relative alignment coordinates for the 10 sub-map pairs took about four hours. The computations were carried out on a 1.3 GHz pc-computer.

In the feature extraction phase the range images, each of 3000x720 data points, were processed systematically measurement by measurement. A computationally more efficient strategy would be the one where, for example, every fifth horizontal scan line was first segmented and the corresponding circle models estimated from the segment data. Then the initially ignored locations in the range image, situated in between the sparsely distributed valid segments, would be processed. With the new, coarse to fine processing principle considerable savings in computation time, without sacrificing performance too much, could be achieved. Similarly, if a more accurate *a priori* alignment estimate for each of the sub-map pairs would be available, the registration computation time could be easily dropped, even by a factor of 10. Nevertheless, the presented 3D simultaneous localization and map building method is, with these data quantities and dimensions of the space covered, clearly an off-line method.

6.2 Construction of an integrated map for the parking lot area

The main goal of the parking lot experiment is to demonstrate the general applicability of the method for large-scale semi-structured outdoor environments and also to give an example of how the non-Gaussian nature of the uncertainty, related to the computed relative alignments, affects the precision of the integrated model. The computed relative alignment values, corresponding to the alignment search space cell size of 0.156 meters for the five consecutive sub-map pairs, are given in Table 6.2. To compute the results for sub-map pair "FE", tree trunks served as the major set of common feature objects, whereas horizontal edge line segments, corresponding to the upper corners of building walls, dominated in the computation of the alignment estimate for the other sub-map pairs. This can also be seen in the uncertainty of the translation degrees of freedom, computed as the sample standard deviation of the connected registration search space cell cloud around the maximum score cell. For sub-map pair "FE", the largest uncertainty is along the z-axis whereas, for the other sub-map pairs, the uncertainty along the x- and y-axis is roughly 10 times larger than along the z-axis.

Table 6.2 Viewpoint point alignment results for the five consecutive registration pairs of the parking lot map building experiment. The size of the search space along x- and y- coordinate directions was ± 10 meters, along the z-axis ± 1.25 meters and around the *a priori* heading direction ± 100 degrees. The dimensions of a cell in the discrete alignment search space corresponded to 0.156 meters when the results in the table were recorded. Units are meters for the x-, y- and z-translation displacement coordinates and degrees for the z-rotation displacement angle.

| Registration pair labels | FE | GF |
|---|---------------------------|---------------------------|
| <i>A priori</i> alignment | 11.0, 0.0, 0.0, -90.0 | 30.0, 0.0, 0.0, 0.0 |
| Highest score alignment | 12.33, 0.08, 0.08, -87.85 | 33.05, -0.55, 0.39, -0.98 |
| Score cloud center alignment | 12.39, 0.13, 0.07, -87.23 | 33.79, 0.73, 0.39, -2.66 |
| Sample standard deviation of score cloud center alignment | 0.22, 0.54, 0.62, 3.15 | 2.29, 2.91, 0.16, 3.95 |
| HG | IH | JI |
| 32.0, 2.0, 0.0, 0.0 | 0.0, 15.0, 0.0, 0.0 | 0.0, 20.0, 0.0, 0.0 |
| 36.61, 0.36, 0.23, -2.54 | 0.70, 16.17, -0.08, 5.27 | 1.33, 20.70, -0.39, -3.71 |
| 37.35, 1.09, 0.21, -1.48 | 0.56, 15.38, -0.14, 8.49 | 0.61, 21.19, -0.42, -3.09 |
| 3.38, 2.50, 0.15, 3.46 | 1.07, 2.36, 0.63, 10.83 | 2.79, 1.79, 0.18, 3.57 |

The locations of sub-maps "F", "G", "H", "I" and "J" with respect to the global frame of reference (i.e. reference frame of sub-map "E") are computed by incrementally concatenating the relative coordinate transformations between consecutive sub-map pairs. Before the relative registration coordinate values can be summed up, they have to be transformed to the common (global) frame of reference:

$$\begin{aligned}
 x_g &= \cos(\phi_{tot})x_r - \sin(\phi_{tot})y_r \\
 y_g &= \sin(\phi_{tot})x_r + \cos(\phi_{tot})y_r \\
 z_g &= z_r \\
 \phi_g &= \phi_r
 \end{aligned} \tag{6.1}$$

where ϕ_{tot} is the total, accumulated z-rotation (heading) angle of the second sub-map in the sub-map pair described with respect to the global reference frame. The subscript "r" indicates relative displacement coordinates referenced with respect to the second sub-map in the sub-map pair. The subscript "g" corresponds to the

relative displacement coordinates, transformed to the global reference frame of the integrated environment model.

Similarly, the transformed covariance matrix of the resulting global location parameters can be approximated by first taking the partial derivatives of the non-linear coordinate transformation equations, given by Equation 6.1, with respect to the relative location parameters and with respect to the global heading angle. The Jacobian matrix, consisting of the partial derivatives, can then be used to transform the covariance matrix of the registration parameters to the global reference frame:

$$\mathbf{C}_g^* = \mathbf{J}\mathbf{C}_r^*\mathbf{J}^T = \mathbf{J} \begin{bmatrix} \sigma_{x_r}^2 & \sigma_{x_r}\sigma_{y_r} & \sigma_{x_r}\sigma_{\phi_r} & \sigma_{x_r}\sigma_{z_r} & \sigma_{x_r}\sigma_{\phi_{tot}} \\ \sigma_{y_r}\sigma_{x_r} & \sigma_{y_r}^2 & \sigma_{y_r}\sigma_{\phi_r} & \sigma_{y_r}\sigma_{z_r} & \sigma_{y_r}\sigma_{\phi_{tot}} \\ \sigma_{\phi_r}\sigma_{x_r} & \sigma_{\phi_r}\sigma_{y_r} & \sigma_{\phi_r}^2 & \sigma_{\phi_r}\sigma_{z_r} & \sigma_{\phi_r}\sigma_{\phi_{tot}} \\ \sigma_{z_r}\sigma_{x_r} & \sigma_{z_r}\sigma_{y_r} & \sigma_{z_r}\sigma_{\phi_r} & \sigma_{z_r}^2 & \sigma_{z_r}\sigma_{\phi_{tot}} \\ \sigma_{\phi_{tot}}\sigma_{x_r} & \sigma_{\phi_{tot}}\sigma_{y_r} & \sigma_{\phi_{tot}}\sigma_{\phi_r} & \sigma_{\phi_{tot}}\sigma_{z_r} & \sigma_{\phi_{tot}}^2 \end{bmatrix} \mathbf{J}^T \quad (6.2)$$

where \mathbf{J} is the Jacobian matrix:

$$\mathbf{J} = \begin{bmatrix} \frac{dx_g}{dx_r} & \frac{dx_g}{dy_r} & \dots & \frac{dx_g}{d\phi_{tot}} \\ \frac{dx_r}{dy_g} & \frac{dy_r}{dy_g} & & \frac{d\phi_{tot}}{dy_g} \\ \frac{dx_r}{d\phi_{tot}} & \frac{dy_r}{d\phi_{tot}} & \ddots & \frac{d\phi_{tot}}{d\phi_{tot}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{d\phi_{tot}}{dx_r} & \frac{d\phi_{tot}}{dy_r} & & \frac{d\phi_{tot}}{d\phi_{tot}} \end{bmatrix} = \begin{bmatrix} \cos(\phi_{tot}) & -\sin(\phi_{tot}) & 0 & 0 & -\sin(\phi_{tot}) - \cos(\phi_{tot}) \\ \sin(\phi_{tot}) & \cos(\phi_{tot}) & 0 & 0 & \cos(\phi_{tot}) - \sin(\phi_{tot}) \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.3)$$

Now, the globally referenced 4x4 covariance matrix \mathbf{C}_g is given by the upper left, 4x4 sub-matrix of \mathbf{C}_g^* . The variance of the global sub-map coordinate values can be approximated by summing the globally referenced variance parameters of each of the concatenated sub-map pairs, given by the diagonal elements of \mathbf{C}_g . The globally referenced registration parameters are given in Table 6.3. (The corresponding locally referenced registration parameters were given in Table 6.2.)

Table 6.3 Viewpoint point alignment results for the five consecutive registration pairs of the parking lot map building experiment. The presented values correspond to the registration parameters, given in Table 6.2, after projecting them into the global reference frame. Units are meters for the x-, y- and z-translation displacement coordinates and degrees for the z-rotation displacement angle.

| Registration pair labels | FE | GF |
|---|---------------------------|---------------------------|
| <i>A priori</i> alignment | 11.0, 0.0, 0.0, -90.0 | 0.0, -30.0, 0.0, 0.0 |
| Highest score alignment | 12.33, 0.08, 0.08, -87.85 | 0.69, -33.04, 0.39, -0.98 |
| Score cloud center alignment | 12.39, 0.13, 0.07, -87.23 | 2.37, -33.71, 0.39, -2.66 |
| Sample standard deviation of score cloud center alignment | 0.22, 0.54, 0.62, 3.15 | 2.91, 2.28, 0.16, 3.95 |
| HG | IH | JI |
| 2.0, -32.0, 0.0, 0.0 | 15.0, 0.0, 0.0, 0.0 | 20.0, 0.0, 0.0, 0.0 |
| 1.11, -36.59, 0.23, -2.53 | 16.15, -1.09, -0.08, 5.27 | 20.74, 0.08, -0.39, -3.71 |
| 1.16, -37.35, 0.21, -1.48 | 15.36, -0.93, -0.14, 8.49 | 21.11, 2.02, -0.42, -3.08 |
| 2.49, 3.38, 0.15, 3.46 | 2.34, 1.12, 0.63, 10.83 | 1.73, 2.82, 0.18, 3.57 |

And finally, the global coordinates and related uncertainties of the five sub-maps, computed by summing the individual registration coordinate and variance terms respectively, are given in Table 6.4.

Table 6.4 Global locations of sub-maps "F", "G", "H", "I" and "J" described with respect to the reference frame of sub-map "E". The global location estimates were computed by incrementally concatenating the relative alignment values, given in Table 6.3. Units are meters for the x-, y- and z-translation displacement coordinates and degrees for the z-rotation displacement angle.

| Sub-map label | F | G |
|--|-----------------------------|-----------------------------|
| Global, <i>a priori</i> alignment based sub-map location | 11.0, 0.0, 0.0, -90.0 | 11.0, -30.0, 0.0, -90.0 |
| Global, maximum score value-based sub-map location | 12.33, 0.08, 0.08, -87.85 | 13.02, -32.97, 0.47, -88.82 |
| Global, score cloud center based sub-map location | 12.39, 0.13, 0.07, -87.23 | 14.75, -33.58, 0.46, -89.89 |
| Standard deviation of the score cloud center based location estimate | 0.22, 0.54, 0.62, 3.15 | 2.92, 2.35, 0.64, 5.05 |
| H | I | J |
| 13.0, -62.0, 0.0, -90.0 | 28.0, -62.0, 0.0, -90.0 | 48.0, -62.0, 0.0, -90.0 |
| 14.13, -69.56, 0.70, -91.36 | 30.2, -70.65, 0.62, -86.09 | 51.02, -70.56, 0.23, -89.80 |
| 15.91, -70.93, 0.67, -91.37 | 31.27, -71.86, 0.54, -82.88 | 52.38, -69.84, 0.12, -85.96 |
| 3.84, 4.11, 0.66, 6.12 | 4.50, 4.26, 0.91, 12.44 | 4.82, 5.11, 0.92, 12.94 |

To illustrate the quality of the integrated global model, two different metric representations are constructed. First, a rectangular grid over the mapped area is formed and for each grid cell the maximum elevation value, recorded at the location, is stored. The incrementally constructed elevation map of the parking lot area is shown in Figure 6.4. For computing the global sub-map coordinates for the integrated model, the "maximum score location" values from Table 6.2 were used. These estimates corresponds to the coordinates of the center of the cell in the registration search space having the highest score value.

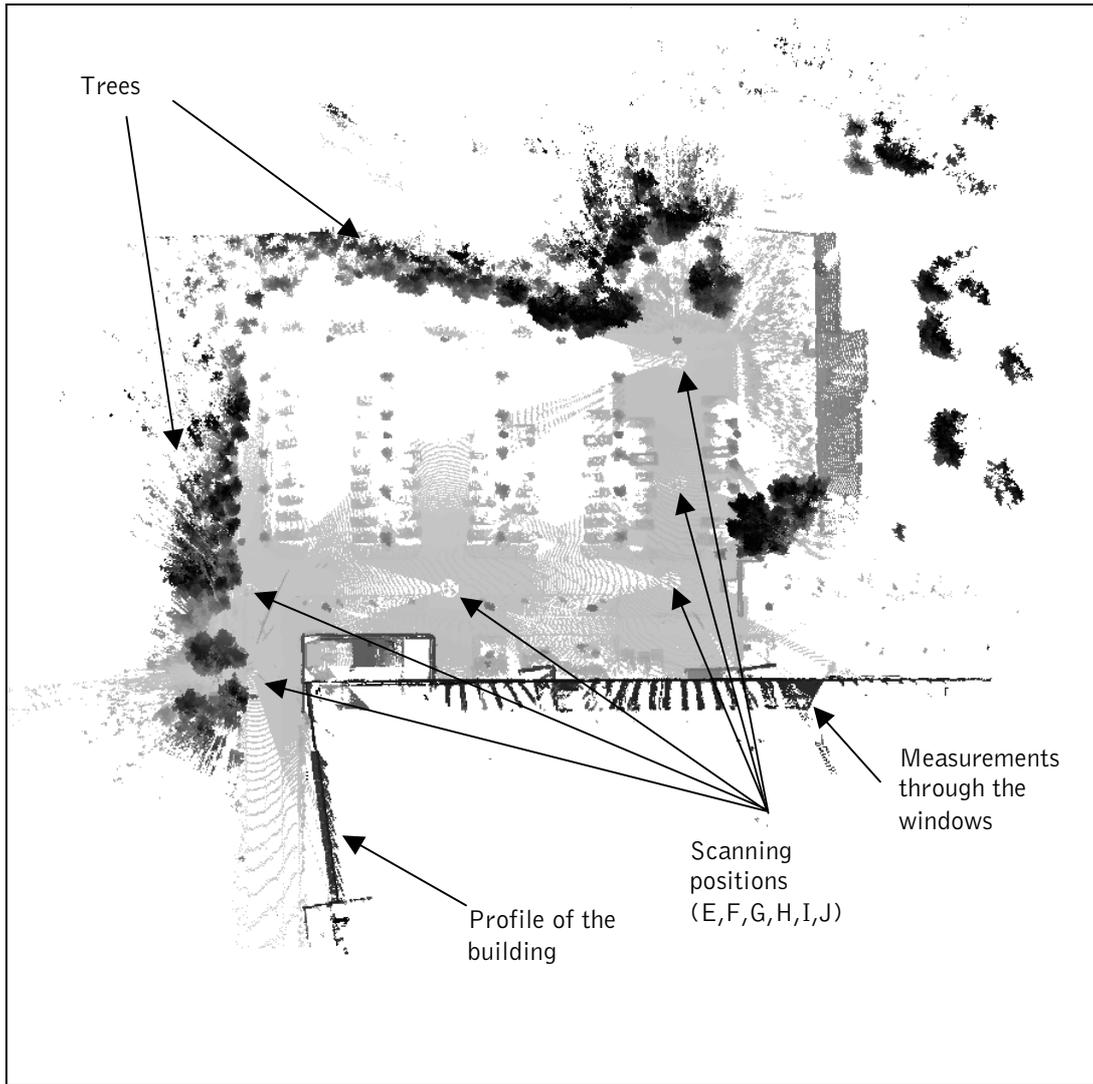


Figure 6.4 The size of the grid map is 900x900 pixels corresponding to an area of 180 meters square. White color corresponds to no measured elevation value. Otherwise, the darker the color, the higher the elevation.

In addition to the elevation map, visible sections of the tree trunks and other elongated cylindrical objects as well as the straight line segments corresponding to the upper end corners of large wall planes are modeled on a feature map. The feature map, formed by integrating the data from the six sub-maps corresponding to the parking lot area, is shown in Figure 6.5.

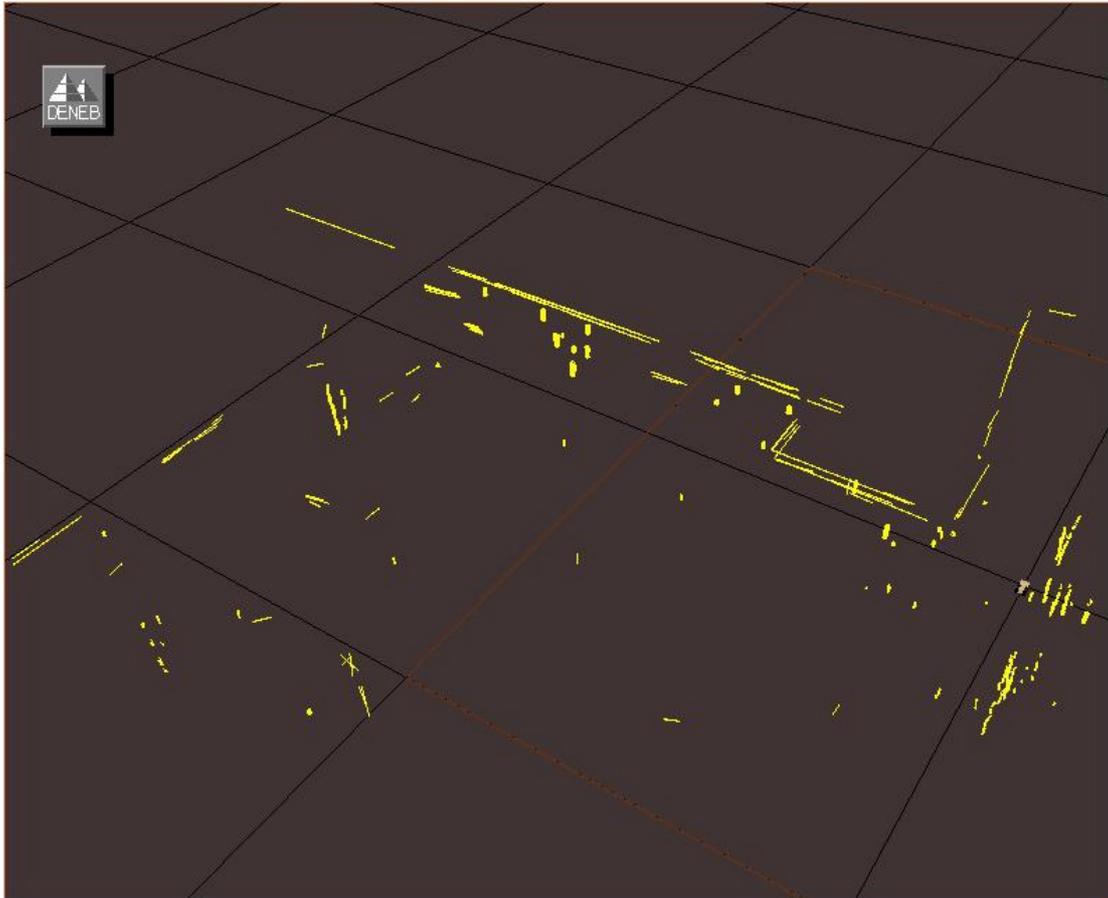


Figure 6.5 Feature map of the parking lot area near the Computer Science Building. The profile of the building and small groups of trees can be identified in the figure. Also, a number of outlier features are present in the figure.

The outlier, cylinder segment features, extracted from the wall of the building in the upper middle part of Figure 6.5 are due to the more relaxed circle segment computation parameters. The “error symmetry threshold” parameter in particular, given in Table 4.2, which is used to classify narrow elongated objects as either circular or more rectangular has great impact on the sensitivity of the method. In the parking lot and forest mapping experiments, more asymmetry in the distribution of the measurement points around the fitted circle model was allowed. In this way the chances of being able to model true cylindrical objects from a larger distance increases, as does the risk for faulty detections. The best values for the computation configuration parameters depend on the preferences of the user. In practice, they are determined empirically. In Figure 6.5, a significant spread of the wall/ceiling corner lines along the vertical, z-direction can be identified. This is, probably, due to the small error in the measured tilt angles of the sensor platform. Even a small angular error causes significant translation displacements when the measured features are distant from the perception sensor.

Corresponding model illustrations for the situations where the integrated maps were constructed based on *a priori* alignment data and the center of connected score cloud alignment estimates are presented in Appendix D, Figures D.1 to D.4. Moreover, in Figures D.7 to D.18, the model data for the individual sub-maps is shown.

6.3 Modeling the interior of a building

As the second test case study, an integrated model was incrementally built for an office corridor environment. Both an elevation and a feature map were constructed to demonstrate the applicability of the method. The computed, locally referenced relative alignment coordinates between consecutive sub-maps were already presented in Chapter 5.4.1, Table 5.3. The global coordinates and related uncertainties of the seven sub-maps are given in Table 6.5.

Table 6.5 Global locations of sub-maps "2", "3", "4", "5", "6", "7" and "8" described with respect to the reference frame of sub-map "1". The global location estimates were computed by incrementally concatenating the relative alignment values computed for the consecutive sub-map pairs "21", "32", "43", "54", "65", "76" and "87". The size of a cell in the registration search space, when the search was stopped, corresponds to two centimeters. Units are meters for the x-, y- and z-translation displacement coordinates and degrees for the z-rotation displacement angle.

| Sub-map label | 2 | 3 |
|--|----------------------------|-----------------------------|
| Global, <i>a priori</i> alignment based sub-map location | 3.0, -14.0, 0.0, 15.0 | 11.4, -37.6, 0.0, 10.0 |
| Global, maximum score value-based sub-map location | 0.31, -13.72, 0.15, 0.88 | -0.27, -35.36, -0.02, 6.25 |
| Global, score cloud center-based sub-map location | 0.30, -13.67, 0.13, 0.93 | -0.27, -35.32, -0.01, 6.32 |
| Standard deviation of the score cloud center-based location estimate | 0.03, 0.03, 0.04, 0.13 | 0.05, 0.04, 0.05, 0.20 |
| 4 | 5 | 6 |
| 29.5, -36.5, 0.0, -65.0 | 39.4, -34.1, 0.0, 100.0 | 48.9, -30.4, 0.0, 205.0 |
| 20.26, -32.50, 0.09, -81.23 | 27.81, -31.83, 0.17, 97.01 | 39.40, -30.11, 0.22, 187.54 |
| 20.28, -32.45, 0.10, -81.20 | 27.81, -31.72, 0.16, 97.22 | 39.40, -29.92, 0.21, 187.87 |
| 0.05, 0.08, 0.05, 0.29 | 0.06, 0.10, 0.07, 0.39 | 0.06, 0.11, 0.07, 0.49 |
| 7 | 8 | X |
| 43.5, -14.1, 0.0, 190.0 | 42.0, 5.9, 0.0, 205.0 | X |
| 36.76, -10.08, -0.06, 185.54 | 36.61, 5.15, -0.19, 185.90 | X |
| 36.66, -9.90, -0.05, 185.91 | 36.40, 5.31, -0.18, 186.53 | X |
| 0.08, 0.12, 0.07, 0.53 | 0.08, 0.12, 0.08, 0.66 | X |

It should be noted, that the relative alignment estimates, presented in Tables 5.3 and 6.5, were computed by using corrupted *a priori* alignment estimates. As already discussed in Chapter 5.4.1, the sub-map registration method should be able to find practically the same alignment coordinates irrespective of the *a priori* alignment estimate. The only requirement is that the correct solution resides within the alignment search space. The search space, on the other hand, is centered around the *a priori* alignment coordinates. In connection with the indoor mapping demonstration, two problem examples, often encountered in practical modeling implementations, are presented.

The difference between the global sub-map coordinates, computed by using the two alternative registration data sets, is much smaller than in the parking lot mapping case. This is primarily due to the fact that the registration computation was continued up to a more fine-grained level (0.02 versus 0.156 meters) within the discrete registration search space than in the parking lot mapping experiment. At fine-grained cell sizes, the cell cloud concentrates at a very small neighborhood of the corresponding maximum score cell, which means that the two estimates, inevitably, are very closely located.

To better illustrate the computed feature map, the wall/ceiling corner lines were extended up to the zero height elevation level, thus forming rectangular wall planes. A view into the 3D model objects, extracted from the range image "scan1", seen against a camera image taken about 30 centimeters above the range image acquisition position, can be seen in Figure 6.6.



Figure 6.6 A partial, mixed reality view of a 3D feature model. The wall plane objects and the cylinder models for the pillars have been extracted from the range image labeled "scan1". The complete set of feature objects extracted from the range image can be seen in Figure D.19.

The incrementally constructed elevation map and two images of the integrated feature map of the office corridor environment are shown in Figures 6.7, 6.8 and 6.9 respectively. In the elevation map, the lowest measured elevation value is recorded (in the parking lot experiment, the highest elevation value was stored instead). The integrated maps were generated from eight range images labeled "1", "2", "3", "4", "5", "6", "7" and "8".

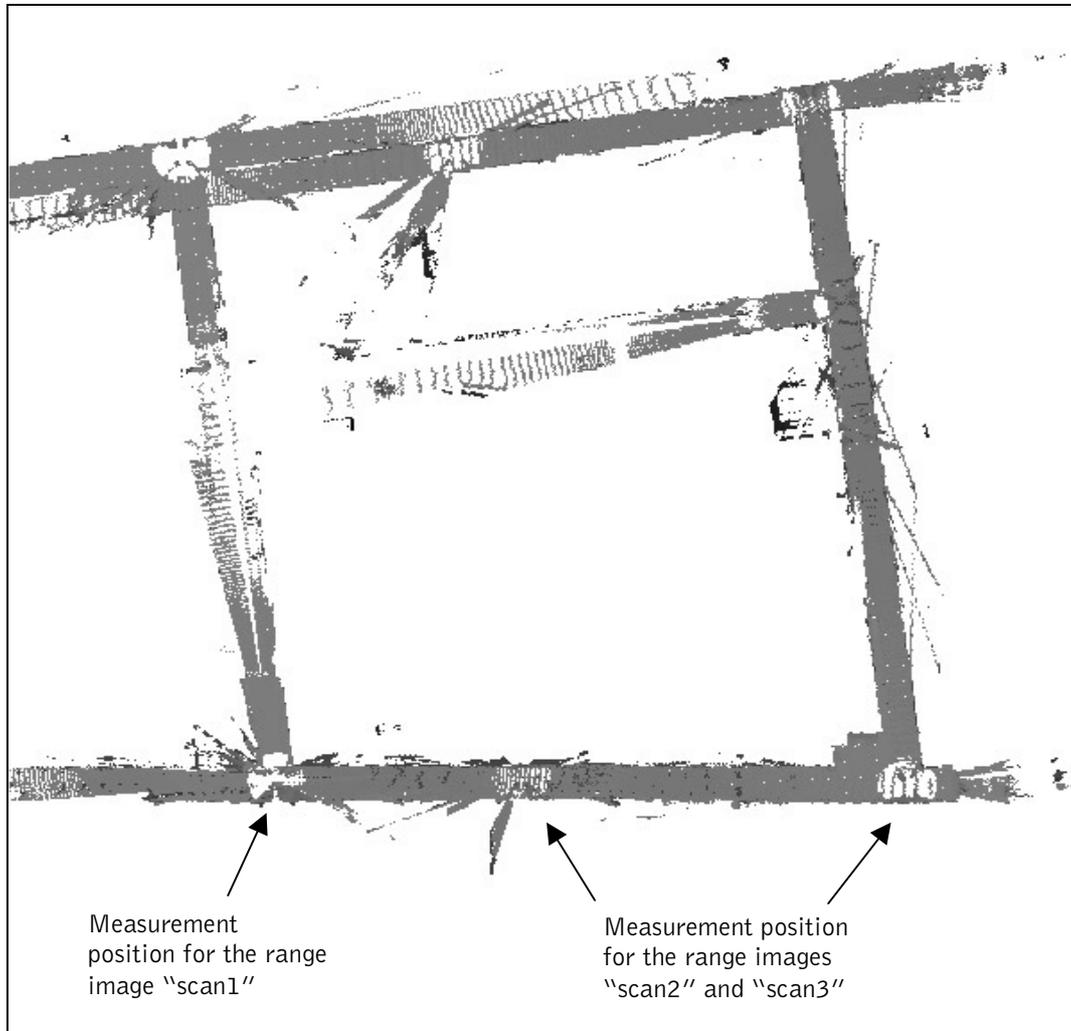


Figure 6.7 Elevation map of the office corridor environment. The size of the grid map is 600x600 pixels corresponding to an area of 60 meters square. In the elevation map white color means no recorded elevation value, otherwise the darker the color the higher the elevation.

The erroneous registration computed between the sub-maps "8" and "7" can be clearly seen in the upper left part of the elevation map. The false registration was due to the fact that only parallel wall planes were compared with each other. Also, the size of the search space was large enough, enabling the confusion of the left wall with the right one.

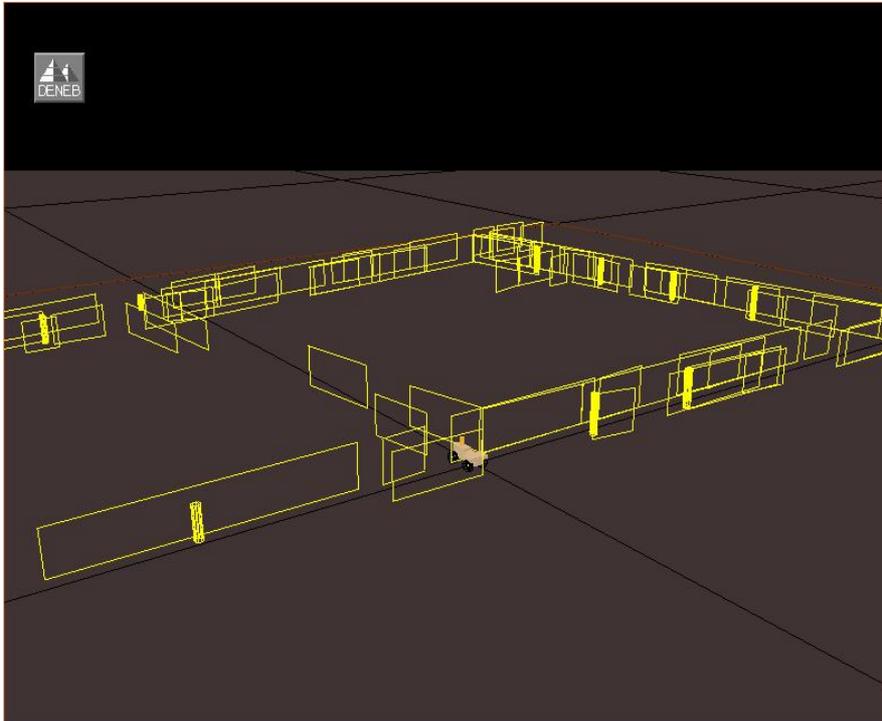


Figure 6.8 A feature map of a ground floor corridor at the Computer Science Building of HUT. The cylindrical supporting pillars of 0.38 meters in diameter and the wall planes of a height of about 2.3 meters can be seen in the figure. Also, the model of a mobile robot at the location, where the first range image was measured, can be identified.

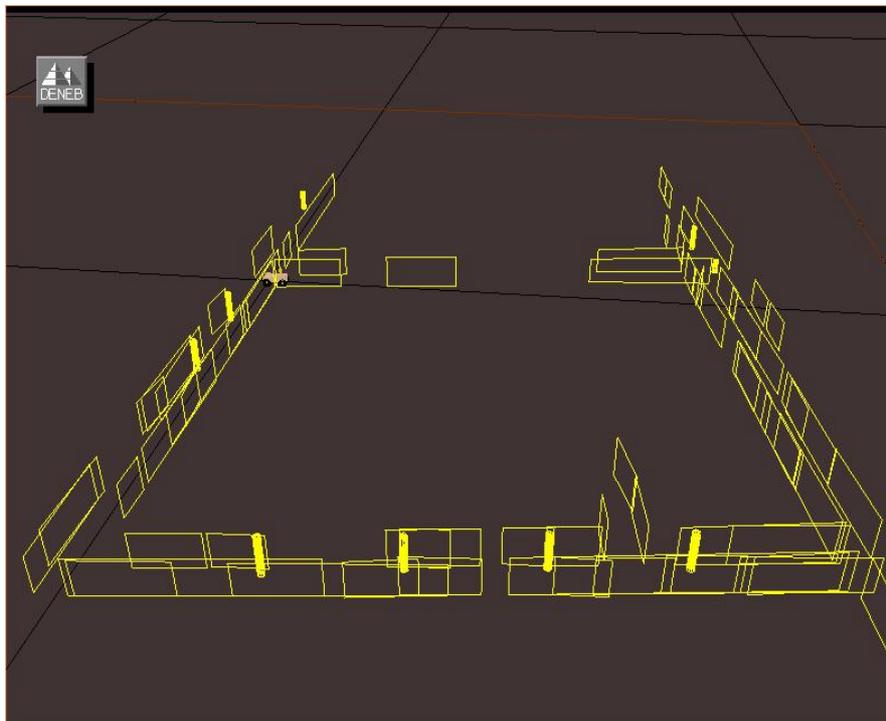


Figure 6.9 Another view into the feature map of the ground floor corridor at the Computer Science Building of HUT.

As can be seen in the images, only parts of the wall planes are modeled. This is due to the relatively large distances between the measurement positions and occlusion (due to other architectural objects and people). Also, the range images labeled "9" and "A" were not used to construct the integrated model due to the lack of common features to facilitate correct registration. The dismissed range images, which can be seen in appendix A Figures A.19 and A.20, were acquired in an open hall. Successful computation of relative alignment estimates for the range image pairs "98", "A9" and "1A" would have been required to close the loop of mutually connected sub-maps. On the whole, the mapping method and the utilized sensor technology are not at their best in this kind of structured indoor environment. In the next chapter, the utilization of the method to map a natural forest environment will be discussed.

6.4 Generating a globally optimal model for the forest terrain

In the last experiment, a small forest area was mapped. The range images "scanT", "scanS", "scanR", "scanQ", "scanO" and "scanN" served as measurement data for model building. A metric-topological layout of the "forest" test area is illustrated in Figure 6.10.

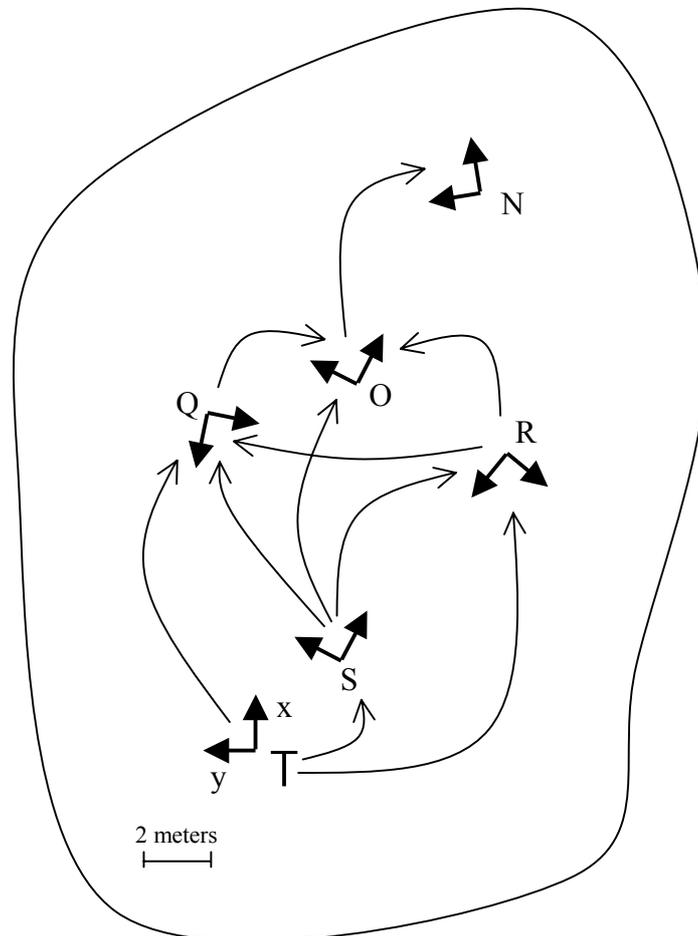


Figure 6.10 A rough metric topological map of the forest area. The viewpoint locations are marked with the coordinate frames and the sub-map pairs, for which relative alignment parameters were computed, are indicated with the arrows.

In the figure, the range image acquisition positions are indicated with xy-coordinate frames along with the corresponding range image label. The 10 sub-map pairs for which a relative alignment was computed are indicated with the arrows. The manually-drawn rough map was used to determine the *a priori* alignment relations between the sub-maps. No measuring devices were used to determine distances within the map. The operator of the Riegl range-imaging sensor subjectively assessed the rough relative displacements between the measuring locations while maneuvering the sensor.

The relative alignment coordinates and the corresponding uncertainties computed in the forest mapping test case are given in Table 6.6.

Table 6.6 Viewpoint alignment results for the forest map building experiment. In the table, the *a priori* alignment values as well as the registration coordinates and the related uncertainties are given. Units are meters for the x-, y- and z-translation displacement coordinates and degrees for the z-rotation displacement angle.

| Registration pair labels | ST | RS |
|---|----------------------------|----------------------------|
| <i>A priori</i> alignment | 3.0, -1.0, 1.0, -30.0 | 4.0, -2.0, 1.0, 180.0 |
| Highest score alignment | 4.42, -1.46, 0.76, -38.79 | 4.09, -3.69, 0.29, 161.37 |
| Score cloud center alignment | 4.43, -1.46, 0.68, -38.71 | 4.08, -3.69, 0.14, 161.30 |
| Sample standard deviation of score cloud center alignment | 0.03, 0.03, 0.29, 0.27 | 0.04, 0.03, 0.15, 0.24 |
| RT | QR | QS |
| 5.0, -5.0, 1.5, 140.0 | 1.0, -5.0, 0.0, 30.0 | 5.0, 2.0, 1.0, 180.0 |
| 5.30, -6.90, 0.71, 122.77 | -0.34, -5.97, 0.63, 5.57 | 6.36, 1.83, 0.56, 166.58 |
| 5.29, -6.90, 0.74, 122.61 | -0.35, -5.96, 0.65, 5.41 | 6.34, 1.83, 0.55, 166.54 |
| 0.08, 0.03, 0.29, 0.48 | 0.04, 0.02, 0.05, 0.20 | 0.04, 0.03, 0.04, 0.25 |
| QT | OQ | OR |
| 8.0, -2.0, 2.0, 160.0 | -3.0, 1.0, 1.0, 140.0 | -2.0, -6.0, 1.0, 140.0 |
| 10.53, -3.94, 1.25, 128.07 | -3.36, -3.48, 0.21, 136.78 | -3.34, -9.76, 0.80, 142.40 |
| 10.52, -3.97, 1.27, 127.99 | -3.36, -3.48, 0.21, 136.77 | -3.35, -9.77, 0.71, 142.35 |
| 0.03, 0.05, 0.04, 0.31 | 0.02, 0.02, 0.03, 0.15 | 0.03, 0.03, 0.04, 0.19 |
| OS | NO | X |
| 7.0, 3.0, 1.5, 320.0 | 5.0, 1.0, 1.0, 30.0 | X |
| 10.41, 4.44, 1.20, 303.42 | 4.97, 1.73, 0.72, 38.14 | X |
| 10.39, 4.46, 1.26, 303.45 | 4.97, 1.74, 0.74, 38.18 | X |
| 0.03, 0.04, 0.04, 0.17 | 0.02, 0.02, 0.03, 0.32 | X |

Before using the measured relative map coordinates and the related uncertainties to solve the optimal map, they have to be transformed to the global reference frame (i.e. reference frame of sub-map "T"). The transformed values can then be

assigned to the $\bar{\mathbf{d}}$ -vector and \mathbf{C} -matrix to solve the Equations 3.20 and 3.21. The projection of the relative registration coordinates and the related uncertainties to the global reference frame was discussed in Chapter 6.2.

In the table, the *a priori* alignment values are also given. The dimensions of the search space around the *a priori* alignment coordinates were ± 5.0 meters in x- and y-directions, ± 1.25 meters along the z-direction and ± 60.0 degrees around the *a priori* heading direction. The size of a cell in the four-dimensional registration space, when the search was stopped, was 0.02 meters.

In the table, the uncertainty related to the x- and y- alignment coordinates is small in all cases. The uncertainty of the z-coordinate is relative large in the registration pairs "ST", "RS" and "RT". This can be explained by the characteristics of the features, i.e. tree trunks, used for registration computation. As they are more or less vertically oriented, the registration score will also be more widely distributed along the vertical, z-coordinate direction. According to Table 6.6, the *a priori* estimates of the relative translation displacements between the scanning positions deviate from the computed values from a few centimeters up to 4.5 meters, while the difference between the *a priori* and computed heading angles deviates from a few degrees up to 32 degrees. In all cases, the computed values seem to be within the boundaries of the divide and conquer search space.

A check whether all the computed registration values are really correct can be done by studying the residual values of the optimal map. The residual values are computed by subtracting registered relative alignment coordinates from the relative alignment values derived from the optimal global coordinates of the corresponding sub-maps in the sub-map pair under consideration. The residual value for the sub-map pair (i, j) , corresponding to the absolute value of the difference between measured and computed relative sub-map coordinates, is given as

$$e_{(i,j)} = \left| \bar{\mathbf{d}}_{ij} - (\mathbf{x}_i - \mathbf{x}_j) \right| \quad (6.4)$$

where $\bar{\mathbf{d}}_{ij}$ corresponds to the observation of the i^{th} sub-map location with respect to the j^{th} sub-map location and \mathbf{x}_i and \mathbf{x}_j are the estimates of the global sub-map locations. Now, if all the residual values are about the same order of magnitude as the uncertainty of the corresponding observation, the integrated optimal map can be presumed to be correct within the given precision.

As mentioned before, the locally referenced relative alignment values have to be transformed to the global reference frame before the optimal map can be computed. For the transformation, or projection, of the relative alignment data, the global heading angle of the second sub-map in each sub-map pair, corresponding to the optimal sub-map locations, is used. Indeed, the map-building task is carried out in phases, where one new sub-map is added to the sub-map data pool at each phase. To project the locally referenced relative alignment observations to the global reference frame, the optimal sub-map locations, computed during the previous phase, are utilized. In Table 6.7, the observations of the relative alignment coordinates given in Table 6.6 are presented with respect to the global frame of reference. In the table, the estimates, computed from the registration score cell cloud, are given. The relative alignment values and the

related covariance information were projected to the global reference frame by utilizing the global heading angle information computed in the second last map building phase.

Table 6.7 Globally referenced score cloud center alignment values for the forest test site. The accumulated, global heading angles used to project the locally referenced data to the global reference frame correspond to the optimal global map computed by means of the five sub-maps, "T, S, R, Q and O". Units are meters for the x-, y- and z-translation displacement coordinates and degrees for the z-rotation displacement angle.

| | | |
|---|---------------------------|---------------------------|
| Registration pair labels | ST | RS |
| Score cloud center alignment | 4.43, -1.46, 0.68, -38.71 | 0.83, -5.44, 0.14, 161.30 |
| Sample standard deviation of score cloud center alignment | 0.03, 0.03, 0.29, 0.27 | 0.04, 0.03, 0.15, 0.24 |
| RT | QR | QS |
| 5.29, -6.90, 0.74, 122.61 | 5.21, 2.92, 0.65, 5.41 | 6.08, -2.58, 0.55, 166.54 |
| 0.08, 0.03, 0.29, 0.47 | 0.03, 0.03, 0.05, 0.19 | 0.03, 0.04, 0.04, 0.25 |
| QT | OQ | OR |
| 10.52, -3.97, 1.27, 127.99 | 4.81, -0.52, 0.21, 136.77 | 10.03, 2.45, 0.71, 142.34 |
| 0.03, 0.05, 0.04, 0.31 | 0.02, 0.02, 0.03, 0.15 | 0.02, 0.03, 0.04, 0.19 |
| OS | NO | X |
| 10.88, -3.09, 1.26, 303.45 | 1.26, -5.11, 0.74, 38.18 | X |
| 0.02, 0.04, 0.04, 0.17 | 0.02, 0.02, 0.03, 0.32 | X |

The computation of the global map of the forest area was carried out gradually by starting with the sub-maps "T" and "S" for which the first integrated map was computed. Then the sub-map "R" was added to the sub-map data set and the second map building phase was started. The relative observation coordinates linking sub-map "R" to sub-map "S" were projected to the global reference frame by using the optimal global heading angle of sub-map "S" computed during the first map building phase. In the third phase, sub-map "Q" was added to the sub-map data set and the observation coordinates linking it to sub-maps "R", "S" and "T" as well as other locally referenced relative observation coordinates within the

map (i.e. "RS") were projected to the global reference frame by using the global heading angle estimates of sub-maps "S" and "R", computed during the second phase of the map building process. This process was repeated altogether five times until all the six sub-maps were included into the computation of the global map. In Table 6.8, the global locations of the sub-maps, included in the environment model at each of the five mapping phases, are presented.

Table 6.8 Optimal sub-map locations with respect to the global frame of reference. In the table, the global locations of the sub-maps, which were considered for the computation of the optimal integrated map at the given mapping phase, are presented. The mapping phases correspond to the situations where a new sub-map data set was acquired and the optimal map was recomputed. Altogether, the results of five optimal map computation iterations are presented. They correspond to the sub-map sets "T,S", "T,S,R", "T,S,R, Q", "T,S,R,Q,O" and "T,S,R,Q,O,N". Units are meters for the translation degrees of freedom (x, y, z) and degrees for the z-rotation angle ϕ .

| Sub-map label | S | R |
|---|----------------------------|----------------------------|
| Optimal sub-map coordinates corresponding to the sub-map set "T, S" | 4.43, -1.46, 0.68, -38.71 | X |
| Optimal sub-map coordinates corresponding to the sub-map set "T, S, R" | 4.42, -1.46, 0.64, -38.68 | 5.29, -6.90, 0.78, 122.62 |
| Optimal sub-map coordinates corresponding to the sub-map set "T, S, R, Q" | 4.42, -1.46, 0.71, -38.66 | 5.30, -6.90, 0.65, 122.61 |
| Optimal sub-map coordinates corresponding to the sub-map set "T, S, R, Q, O" | 4.42, -1.46, 0.50, -38.66 | 5.30, -6.90, 0.74, 122.61 |
| Optimal sub-map coordinates corresponding to the sub-map set "T, S, R, Q, O, N" | 4.42, -1.46, 0.50, -38.66 | 5.30, -6.90, 0.74, 122.61 |
| Q | O | N |
| X | X | X |
| X | X | X |
| 10.52, -3.98, 1.27, 127.96 | X | X |
| 10.52, -3.98, 1.28, 128.01 | 15.33, -4.49, 1.55, 264.76 | X |
| 10.52, -3.98, 1.28, 128.01 | 15.33, -4.49, 1.55, 264.76 | 16.61, -9.60, 2.30, 302.94 |

The corresponding uncertainty information is given in Table 6.9. In Appendix D, Figures D.23 to D.28, the gradual evolution of the feature-based 3D model representation of the forest map is depicted.

Table 6.9 Uncertainties in the optimal, global sub-map locations given in Table 6.8. The values correspond to the standard deviation of the sub-map coordinates. Units are meters for the uncertainty of the translation degrees of freedom (x, y, z) and degrees for the uncertainty of the z-rotation angle ϕ .

| Sub-map label | S | R |
|-----------------------------|------------------------|------------------------|
| Sub-maps "T, S" | 0.03, 0.03, 0.29, 0.27 | X |
| Sub-maps "T, S, R" | 0.02, 0.02, 0.22, 0.18 | 0.04, 0.02, 0.22, 0.24 |
| Sub-maps "T, S, R, Q" | 0.02, 0.02, 0.06, 0.14 | 0.02, 0.02, 0.06, 0.13 |
| Sub-maps "T, S, R, Q, O" | 0.02, 0.02, 0.05, 0.13 | 0.02, 0.02, 0.05, 0.13 |
| Sub-maps "T, S, R, Q, O, N" | 0.02, 0.02, 0.05, 0.13 | 0.02, 0.02, 0.05, 0.13 |
| Q | O | N |
| X | X | X |
| X | X | X |
| 0.02, 0.02, 0.04, 0.17 | X | X |
| 0.02, 0.02, 0.04, 0.15 | 0.02, 0.02, 0.05, 0.15 | X |
| 0.02, 0.02, 0.04, 0.15 | 0.02, 0.02, 0.05, 0.15 | 0.03, 0.03, 0.06 0.36 |

Now, the residual values can be computed by using the relative observation coordinates from Table 6.7 and the global sub map coordinates given in Table 6.8 for the fifth mapping phase. The residual values along with the uncertainties of the relative observation coordinates are presented in Table 6.10.

Table 6.10 Comparison of the residual values of the optimal map solution with the uncertainties of the measured relative sub-map alignment coordinates. Units are meters and degrees for the values corresponding to the translation displacement coordinates and the heading angle, respectively. Due to the small size of the residual value, a third decimal is given.

| | | |
|--|-----------------------------|----------------------------|
| Registration pair labels | ST | RS |
| Residual values | 0.003, 0.003, 0.186, 0.048 | 0.000, 0.004, 0.101, 0.024 |
| Sample standard deviation of observed values | 0.03, 0.03, 0.29, 0.27 | 0.04, 0.03, 0.15, 0.24 |
| RT | QR | QS |
| 0.010, 0.004, 0.002, 0.005 | 0.007, 0.001, 0.113, 0.014 | 0.006, 0.019, 0.231, 0.121 |
| 0.08, 0.03, 0.29, 0.47 | 0.03, 0.03, 0.05, 0.19 | 0.03, 0.04, 0.04, 0.25 |
| QT | OQ | OR |
| 0.007, 0.002, 0.004, 0.016 | 0.003, 0.008, 0.069, 0.0164 | 0.005, 0.030, 0.106, 0.194 |
| 0.03, 0.05, 0.04, 0.31 | 0.02, 0.02, 0.03, 0.15 | 0.02, 0.03, 0.04, 0.19 |
| OS | NO | X |
| 0.004, 0.017, 0.209, 0.028 | 0.000, 0.000, 0.000, 0.000 | X |
| 0.02, 0.04, 0.04, 0.17 | 0.02, 0.02, 0.03, 0.32 | X |

The residual values in Table 6.10 are well within the three times standard deviation uncertainty bounds determined during the sub-map registration phase, except in the case of the z-coordinate direction for the sub-map pairs "QS" and "OS". This is a consequence of the characteristics of the feature sets used for registration computation. Tree trunks are usually nearly vertically oriented, which naturally increases the uncertainty of the computed relative alignment values along the z-axis direction. The large variation of the uncertainty in the observation data allows, correspondingly, large adjustments along the z-axis during the optimal map computation. Nevertheless, it is claimed that the constructed optimal map can be declared the correct one. The possibility of automatically detecting false registration parameters, when they are to be added to the global map building process, will be discussed in Chapter 7.1.

In Figure 6.11, an elevation map, corresponding to the globally optimal sub-map locations, is shown. The elevation map was formed by first recording the lowest beam hit value at each grid location within each of the six sub-maps and then transporting the elevation values into the global frame of reference.

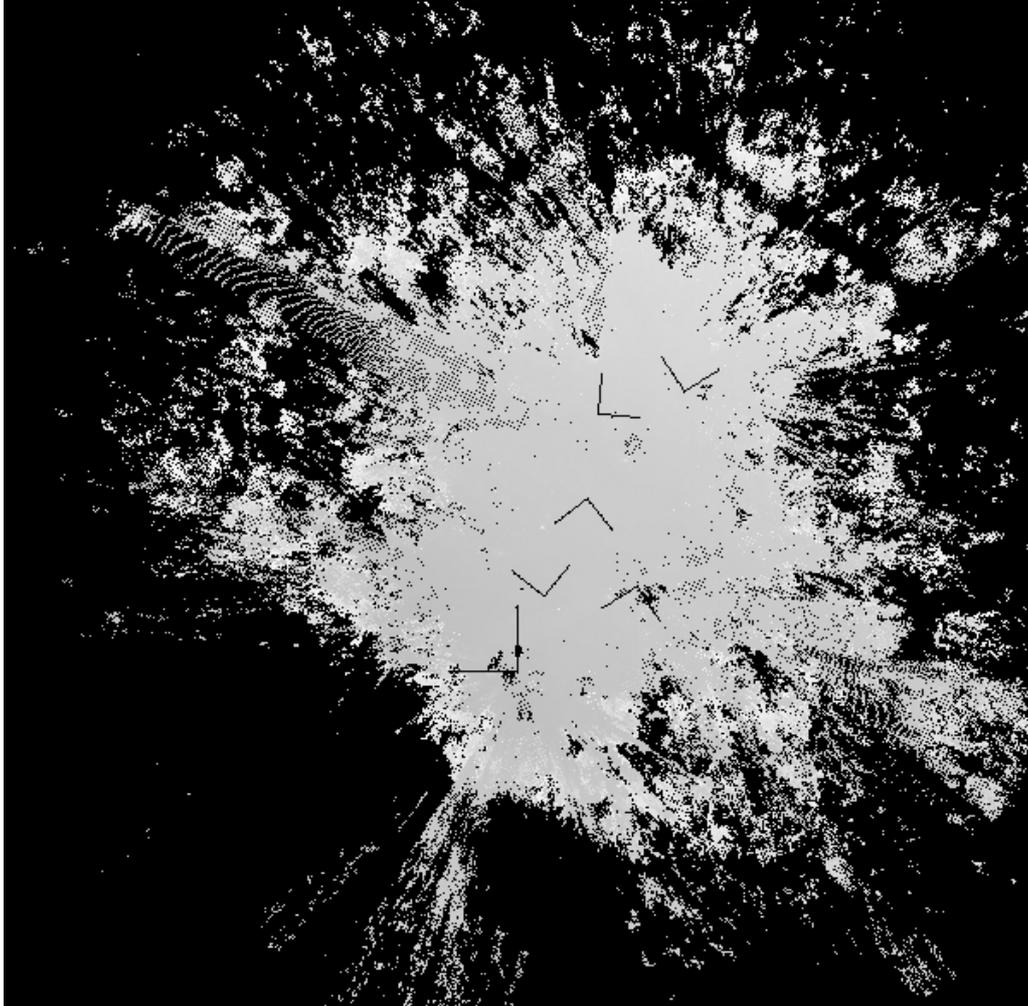


Figure 6.11 The size of the grid map is 600x600 pixels corresponding to an area of 60 meters square. In the image, black indicates no registered elevation value. Otherwise, the lighter the color, the higher the elevation. For each x- and y-grid cell location the lowest measured elevation value was stored. The first measurement position, corresponding to the global frame of reference of the integrated map, is indicated by the larger xy-coordinate axis symbol. The other five scanning positions are marked with smaller symbols. The z-axis of the coordinate frames is pointing from the paper upwards.

In Figure 6.12, a bird's-eye view into the 3D realization of the combined elevation/feature map is shown.

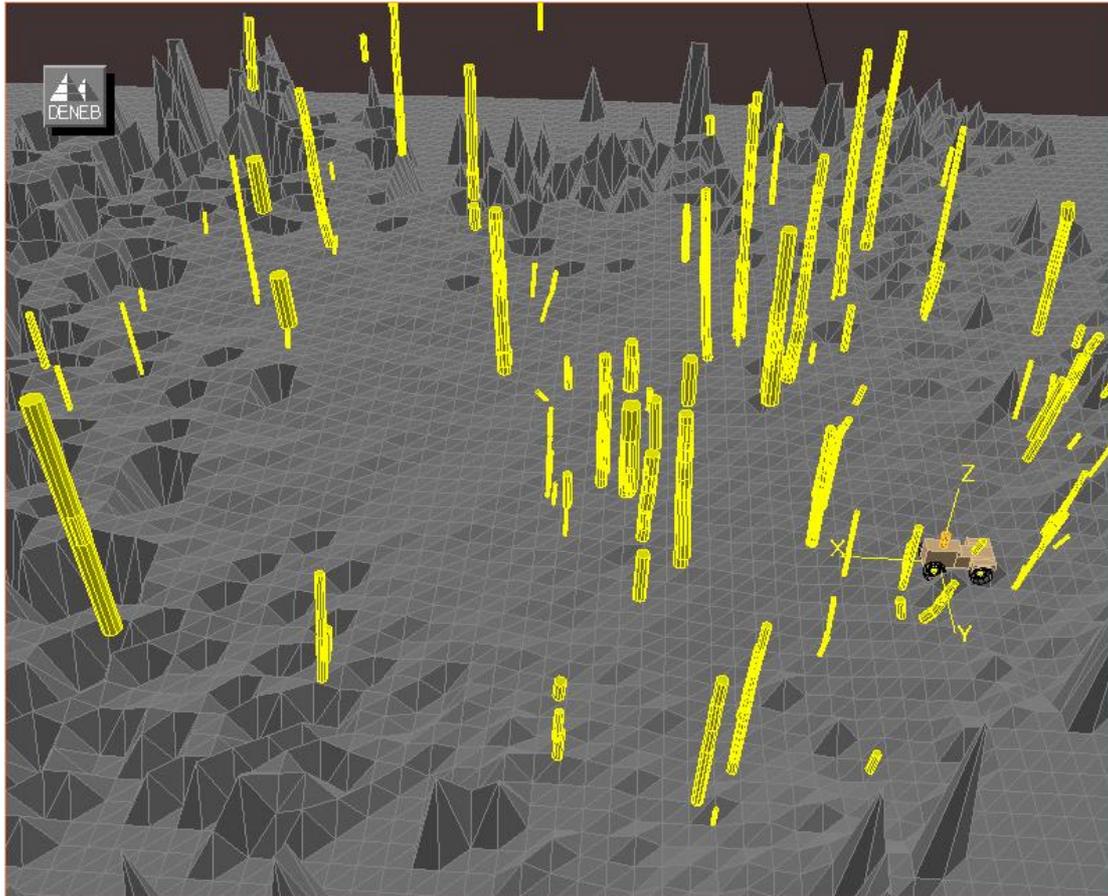


Figure 6.12 A bird’s-eye view into the combined 3D elevation/feature model of the small forest area. The size of a single grid element in the elevation map is 0.6 meters. A model of a mobile robot (not used in the experiments) is placed at the position where the range image “scanT” was acquired. That location, where the coordinate frame is illustrated as well, corresponds to the origin of the global coordinate system of the model.

In the model, cylinder segment approximations are used to model the visible sections of the tree trunks. The radius of each cylinder segment within a sub-map was determined as the average of the computed circle model radius values for the particular section of the tree trunk. For the 3D elevation map representation, a grid cell size of 0.6 meters was used. The elevation value of the larger grid cell was computed as the average of the five smallest elevation values of the grid cells within the corresponding area patch of the original grid map with a 0.1-meter cell size. The cylinder segments and elevation values were then moved to the global map to their corresponding global locations. No smoothing or merging operations were carried out to “clean” the model appearance of the tree trunks.

In Figure 6.13, a top down look to the neighborhood of the vehicle model is depicted. In Figure 6.14 a view from behind the model of the mobile robot is shown. And finally, two closer views into the 3D model and corresponding real views into the forest environment are presented in Figures 6.15 to 6.18.

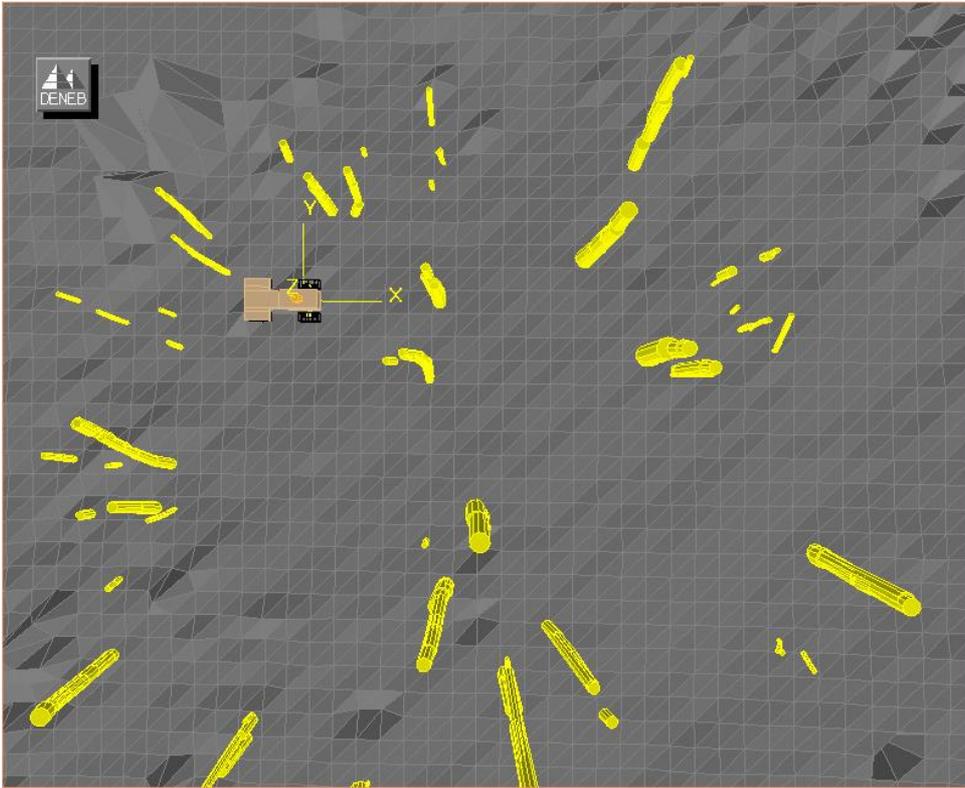


Figure 6.13 A closer top down view into the combined 3D elevation/feature map. The size of a single grid element in the elevation map is 0.6 meters.

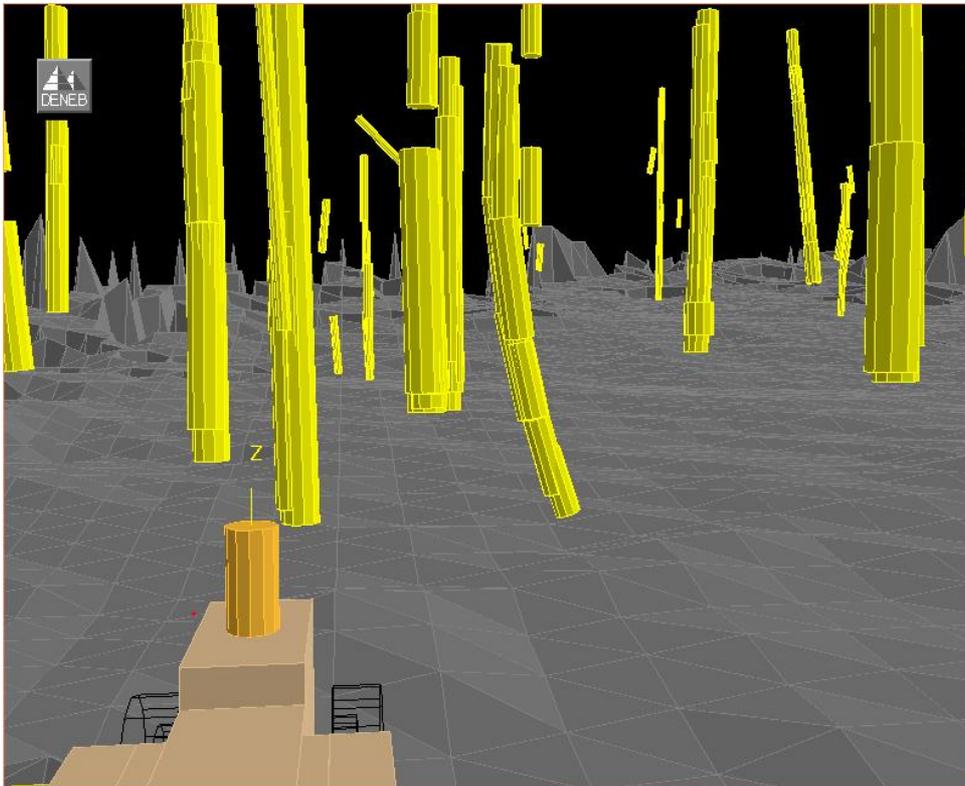


Figure 6.14 A view from behind the model of the mobile robot.

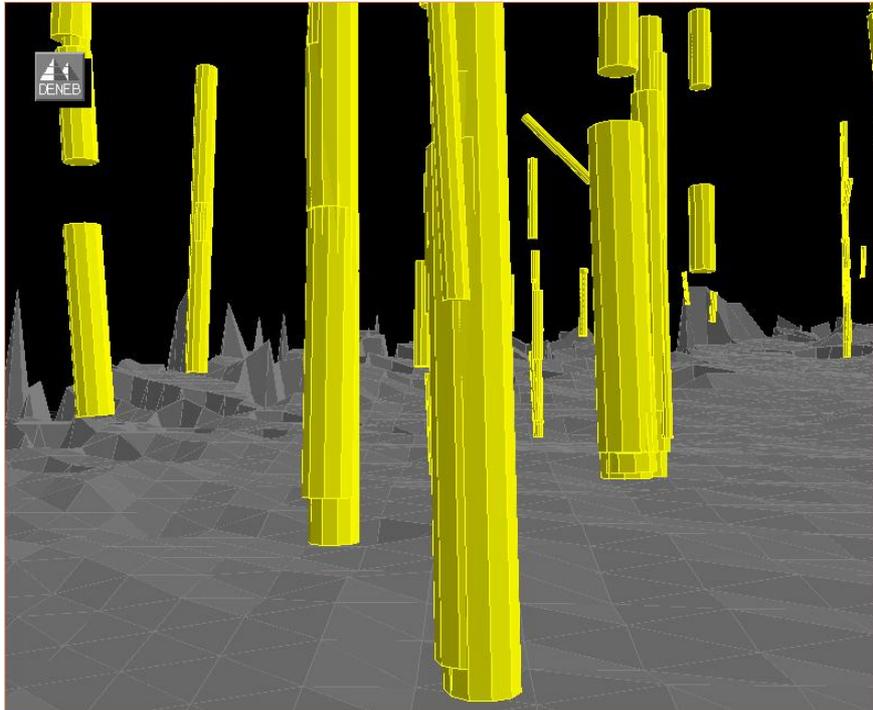


Figure 6.15 A close look into the combined 3D elevation/feature map. The virtual camera view is taken roughly towards the x-axis direction from above the origin of the global frame (i.e. scanning position "T").



Figure 6.16 A camera image taken from above scanning location "T" into the same scene corresponding to the model view, shown in Figure 6.15.

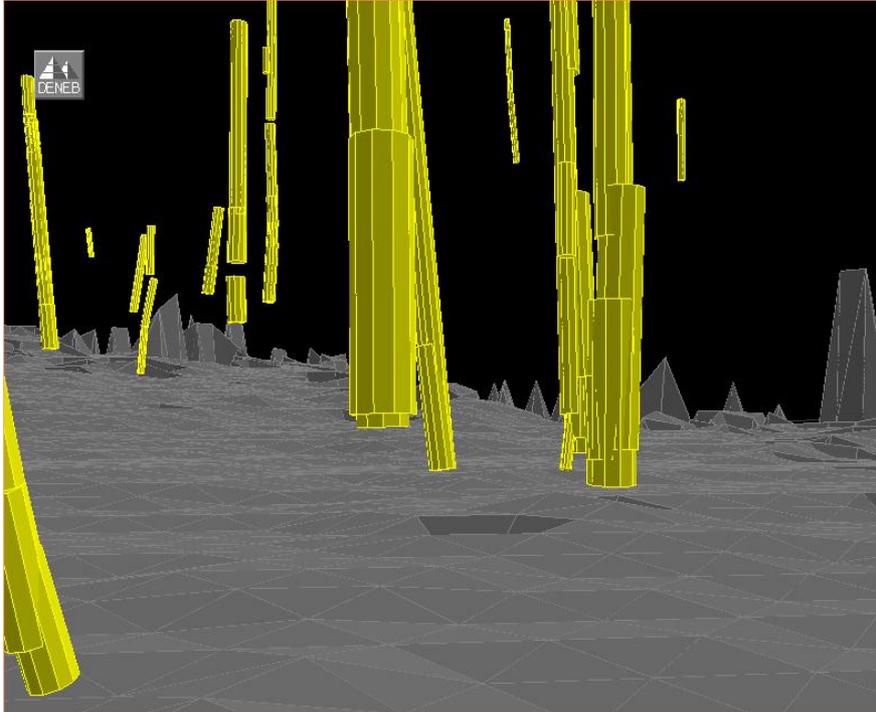


Figure 6.17 The virtual camera view is taken from above scanning position "T" into a direction about 45 degrees to the right with respect to the optical axis of the view, shown in Figures 6.15 and 6.16.



Figure 6.18 A camera image taken from above scanning location "T" into the same scene corresponding to the model view, shown in Figure 6.17.

In Figures 6.15 and 6.17, the cylinder segments, extracted from individual range images, have been moved to the global frame and overlap each other in the images. In most cases, the overlapping cylinder segments seem to be of about the same size

and are approximately aligned with each other. However, some clearly too thin cylinder segments are found at nearly the same location as the actual size tree trunk model as well. This is due to noisy and incomplete range data from the tree trunk objects, thus "misleading" the feature extraction process.

The generated 3D model for the forest environment consists of the original feature data mapped into a common reference frame. For an end-user application, such as high level, cognitive communication between human and mobile robot, it would be desirable to merge the overlapping feature data originating from the same physical object. In this way, for example, a single model object would be assigned for a single tree yielding a "cleaner" virtual representation of the operation environment. Alternatively, for another application, the original range image data could first be transformed into a polygonal mesh model and then mapped into the global reference frame, thereby modeling the outer contour of the visible objects.

6.5 Conclusions

The developed simultaneous localization and map building (SLAM) method was tested by mapping three different kinds of test environments. Integrated models for the parking lot environment and the indoor office corridor environment were constructed by using a simple incremental method. The results indicate that the method can be applied to construct a 3D-model representation for semi-structured and structured environments. However, it was recognized that the method might not be the ideal solution for mapping structured indoor environments.

The principal mapping experiment, to demonstrate the capability of the method, was carried out in a forest environment. In the experiment, an, in the least squares sense, optimal integrated map was built from six range images, acquired at different positions within the test site. The generated environment model seems to be sufficient, for example, to serve as a common representation of the corresponding real environment, conceivable by both man and his machine. Consequently, such an abstracted, feature-based model representation might offer the basis for high level dialog between people and robots.

Chapter 7

Key problems in automatic map construction

7.1 Evaluation of the correctness of the map

It would be desired to be able to evaluate the correctness of the individual sub-map registration results. Olson proposed that the share of score mass concentrated in the local neighborhood of the selected registration peak value should be over a certain percentage of the total score mass to indicate a correct registration [Olson, 2000]. The total score mass would be computed as the sum of the score masses of the pruned and remaining score cells. However, it seems that the share of score mass concentrated around a particular peak depends on the initial size of the search space and the distribution and amount of feature points in the maps to be aligned. Also, the share of features, for which no correct pair in the other map exists (i.e. outliers), affects the score distribution during the divide and conquer search.

As demonstrated in the forest mapping experiment, a bigger number of parallel registration dependencies can be utilized to gain information about the correctness of the entire set of registration dependencies of the global map as a whole. By studying the residual values, computed by means of the estimated global sub-map locations and the corresponding observations of the relative map alignment coordinates, a measure for the correctness of the map can be formed. According to the measure, the constructed integrated map for the forest terrain, discussed in Chapter 6.4, was declared to be correct within given levels of precision. Now, let us add four new hypothesis of a correct observation to the relative alignment data set presented in Chapter 6.4, Table 6.6. The residual values after testing the four new alignment hypotheses, one at a time, are presented in Table 7.1.

Table 7.1 Residual values recomputed for the sub-map pairs, presented in Chapter 6.4, Table 6.10. The sub-map pair labels of the added relative alignment observation hypothesis are given in the leftmost column after the "Residual values" text. Units are meters and degrees for the translation displacement coordinates and the heading angle, respectively.

| | | |
|----------------------------|------------------------------|-----------------------------|
| Registration pair labels | ST | RS |
| Residual values, "NQ" | 0.164, 0.042, 0.265, 0.565 | 0.081, 0.052, 0.001, 0.650 |
| Residual values, "NR" | 0.284, 0.208, 0.542, 3.413 | 0.993, 0.424, 0.410, 2.599 |
| Residual values, "NS" | 0.301, 0.363, 0.210, 2.812 | 0.757, 0.520, 0.443, 5.079 |
| Residual values, "OT" | 0.015, 0.006, 0.162, 0.000 | 0.003, 0.006, 0.098, 0.035 |
| Std of observed values | 0.03, 0.03, 0.29, 0.27 | 0.04, 0.03, 0.15, 0.24 |
| RT | QR | QS |
| 0.232, 0.083, 0.178, 0.105 | 0.570, 0.151, 0.090, 2.508 | 0.477, 0.180, 0.334, 2.999 |
| 0.722, 0.205, 0.045, 0.793 | 1.140, 1.252, 0.075, 7.263 | 0.159, 0.851, 0.578, 4.822 |
| 0.443, 0.168, 0.146, 2.286 | 0.248, 0.204, 0.019, 0.658 | 0.521, 0.339, 0.180, 4.261 |
| 0.004, 0.001, 0.023, 0.055 | 0.010, 0.003, 0.116, 0.024 | 0.005, 0.014, 0.226, 0.098 |
| 0.08, 0.03, 0.29, 0.47 | 0.03, 0.03, 0.05, 0.19 | 0.03, 0.04, 0.04, 0.25 |
| QT | OQ | OR |
| 0.314, 0.241, 0.027, 2.588 | 0.916, 0.112, 0.214, 2.878 | 0.361, 0.286, 0.026, 0.207 |
| 0.441, 1.041, 0.005, 8.082 | 0.379, 0.749, 0.107, 3.986 | 1.534, 1.979, 0.182, 11.087 |
| 0.219, 0.042, 0.012, 1.602 | 0.579, 0.461, 0.151, 3.623 | 0.346, 0.234, 0.018, 3.127 |
| 0.018, 0.010, 0.022, 0.055 | 0.006, 0.013, 0.076, 0.044 | 0.011, 0.039, 0.110, 0.232 |
| 0.03, 0.05, 0.04, 0.31 | 0.02, 0.02, 0.03, 0.15 | 0.02, 0.03, 0.04, 0.19 |
| OS | NO | X |
| 0.432, 0.320, 0.389, 0.254 | 4.187, 2.563, 0.991, 58.927 | X |
| 0.531, 1.573, 0.177, 8.676 | 7.013, 1.529, 0.848, 90.626 | X |
| 1.093, 0.772, 0.840, 8.017 | 5.015, 8.079, 0.692, 145.120 | X |
| 0.005, 0.027, 0.207, 0.078 | 0.000, 0.000, 0.000, 0.000 | X |
| 0.02, 0.04, 0.04, 0.17 | 0.02, 0.02, 0.03, 0.32 | X |

The computed relative alignment coordinates for the sub-map pairs "NQ", "NR" and "NS" are known to be incorrect due to the fact that there was simply not enough common tree trunk objects visible in the corresponding range images to enable correct relative alignment computation. The suspicion of the faultiness of the new registration hypothesis can be confirmed by comparing the recomputed

residual values of the other sub-map pair registration results with the corresponding observation uncertainties. For all the sub-map pairs at least one of the components of the four-dimensional residual value considerably exceeds the three times standard deviation threshold computed from the observation uncertainty.

The fourth new relative alignment observation, between sub-maps "O" and "T" was, in Chapter 6.4, left out of the input data set for computing the optimal map for the forest terrain. This was due to the fact that it was assumed that there would not be enough common feature objects to enable the computation of correct registration parameters. Now, if we look at the residual values, after adding the alignment observation of the sub-map pair "OT" into the integrated map, we can see that the residual values appear to be about the same size as in Table 6.10. The results indicate that the computed relative alignment coordinates are indeed correct within the precision of the corresponding observations. This was also confirmed by manual inspection of the aligned feature sets.

It should be noted, however, that the technique, described above, works only in case there are already parallel observation link paths in the map with the new observation link path. For example it would not have been possible to test the correctness of the observation for the sub-map pair "NO" when it was added to the integrated environment model. This is due to the fact that sub-map "N" had, at the time of residual computation, no other registration link paths established with the other sub-maps.

7.2 Treatment of multiple map hypotheses

In the registration experiments it was demonstrated how the location of the relative alignment of the maximum score alignment may "jump" between different alignment hypotheses until it settles down at one of them (compare Figure 5.8 in Chapter 5.4.1). Such kind of a phenomenon may reflect the existence of ambiguous features in the environment. For more robust and error tolerant mapping, a map administration method should be adopted to be able to keep the multiple map alignment hypotheses until more data is acquired at a later stage of the mapping.

Chapter 8

Summary and conclusions

In this work a method for 3D localization and mapping of large-scale environments was presented. Both structured (indoor) environments and natural outdoor environments were considered. The measurement data was collected with a new type of two axes range imaging sensor. The achieved results are partly specific to this category of perception systems, especially with respect to the feature extraction phase of the method. On the other hand, computation of the relative alignment (i.e. registration) between a pair of sub-maps is a generic technique applicable to models constructed by means of other mapping approaches and perception systems.

The discussed mapping method consists of three main phases: first, cylinder segment objects and straight-line objects are extracted from the range image data. The extracted feature objects are presented with respect to the corresponding local frame of reference situated at the location where the range image was acquired. The features represented with respect to the perception position frame form a sub-map. Second, the relative coordinate transformation between the sub-maps is determined by means of 3D point data sampled from the geometry of the feature objects. In probabilistic terms, the sub-map registration computation corresponds to the measurement update step of a mobile robot localization method. And, lastly, the parallel network of relative alignment coordinates between sub-maps was used to compute an optimal integrated model.

In the experimental part of the work, three different test environments were mapped. In the parking lot and office corridor environments the relative alignment coordinates between consecutive sub-map frames were incrementally connected to construct an integrated global model. For the forest environment, 10 relative alignment links were computed among the six sub-map frames, which enabled the computation of an optimal global map. Relatively accurate results were achieved although room for improvements and warrants for future research remain. In

particular, a better understanding of the actual uncertainty, related to the estimate of relative alignment coordinates between two sub-maps, could further improve the precision of the integrated map. Also, the relative alignment coordinates could be recomputed by means of the original 3D point coordinates of the range measurements or, for example, by means of the elevation map data. In that case, the relative alignments, computed by means of the feature maps, would serve as an initial, approximate solution. In fact, commercial software packages are available for implementing the raw measurement data-based registration computation.

There are two main applications planned for the utilization of the constructed maps. First, the large-scale integrated map can serve as the basic navigational reference of an autonomous mobile robot. With the map the robot can reset the accumulated error in its location estimate while maneuvering in the environment. For this purpose, a 2D profile of the 3D map is formed. The second application is related to the development of a high-level communication interface for future service robots. Cognitive communication between people and robots requires common understanding of the operation environment. The models, constructed with the presented method can offer the basic media for the human-machine dialog. Beyond the robotics field of science, the developed simultaneous localization and map building method could be applied elsewhere such as in forest inventory. However, at the present stage of the method and with the applied range sensor, the precision requirements of tree trunk models might be difficult to meet.

References and literature

Atiya, S. and Hager, G. D. [1993]. *Real-Time Vision-Based Robot Localization*. IEEE Transactions on Robotics and Automation, Vol. 9, No. 6, December, 1993, pp. 785-800.

Ayache, N. and Faugeras O. D. [1989]. *Maintaining Representations of the Environment of a Mobile Robot*. IEEE Transactions on Robotics and Automation, Vol. 5, No. 6, December, 1989, pp. 804-819.

Bailey, T., Nebot, E. M., Rosenblatt, J. K. and Durrant-Whyte, H. F. [2000]. *Data Association for Mobile Robot Navigation: A Graph Theoretic Approach*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 2512-2517.

Bar-Shalom, Y. and Fortmann, T.E. [1988]. *Tracking and Data Association*, Academic Press, 1988.

Besl, P. J. and Jain, R. C. [1988]. *Segmentation Through Variable-Order Surface Fitting*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 10, No. 2, March, 1988, pp. 167-192.

Betgé-Brezetz, S., Chatila, R. and Devy, M. [1994a]. *Natural Scene Understanding for Mobile Robot Navigation*. Proceedings of the IEEE International Conference on Robotics and Automation, 1994, pp.730-736.

Betgé-Brezetz, S., Chatila, R., Devy, M., Fillatreau, P. and Hashashibi, F. [1994b]. *Adaptive Localization of an Autonomous Mobile Robot in Natural Environments*. Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, Las Vegas, NV, October, 1994, pp. 77-84.

Betgé-Brezetz, S., Chatila, R. and Devy, M. [1995]. *Object-based Modeling and Localization in Natural Environments*. Proceedings of the IEEE International Conference on Robotics and Automation, 1995, pp. 2920-2927.

Betgé-Brezetz, S., Hébert, P., Chatila, R. and Devy, M. [1996]. *Uncertain Map Making in Natural Environments*. Proceedings of the IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, April, 1996, pp.1048-1053.

Borenstein, J., Everett, H. R. and Feng, L. [1996]. *"Where Am I", Sensors and Methods for Mobile Robot Positioning*. Report of the University of Michigan, Edited and compiled by J. Borenstein, April, 1996, 277 p.

Boyer, K. L., Mirza, M. J. and Ganguly, G. [1994]. *The Robust Sequential Estimator: A General Approach and its Application to Surface Organization in Range Data*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 10, October, 1994, pp. 987-1001.

Bulata, H., and Devy, M. [1996]. *Incremental Construction of a Landmark-Based and Topological Model of Indoor Environments by a Mobile Robot*. Proceedings of the IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, April, 1996, pp.1054-1060.

Burgard, W., Fox, D., Hennig, D. and Schmidt, T. [1996]. *Estimating the Absolute Position of a Mobile Robot Using Position Probability Grids*. Proceedings of the 13th National Conference on Artificial Intelligence [AAAI], 1996.

Burgard, W., Fox, D. and Thrun, S. [1997]. *Active Mobile Robot Localization by Entropy Minimization*. Proceedings of the Second EUROMICRO workshop on Advanced Mobile Robots, 1997, pp. 155 -162.

Burgard, W., Derr, A., Fox, D. and Cremers, A.B. [1998]. *Integrating Global Position Estimation and Position Tracking for Mobile Robots: The Dynamic Markov Localization Approach*. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Victoria, B.C, Canada, October, 1998, pp. 730-735.

Burgard, W., Fox, D., Jans, H., Matenar, C. and Thrun, S. [1999]. *Sonar-Based Mapping with Mobile Robots Using EM*. Proceedings of the 16th International Conference on Machine Learning [ICML'99], 1999.

Burgard, W., Moors, M., Fox, D., Simmons, R. and Thrun, S. [2000]. *Collaborative Multi-Robot Exploration*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 476-481.

Byrne, J. and Singh, S. [1998]. *Precise Image Segmentation for Forest Inventory*. Research Report CMU-RI-TR-98-14, 1998, 30p

Carmichael, O. and Hebert, M. [1998]. *Unconstrained Registration of Large 3D Point Sets for Complex Model Building*. Proceedings of IEEE/RSJ International Conference on Robots and Systems [IROS], Victoria, B. C., Canada, October, 1998, pp. 360-367.

Carpenter, J., Clifford, P. and Fearnhead, P. [1999]. *Improved Particle Filter for Non-linear Problems*. IEE Proceedings of Radar, Sonar and Navigation, Vol. 146, No. 1 , February, 1999, pp. 2 -7.

Cassandra, A.R., Kaelbling, L.P. and Kurien, J.A. [1996]. *Acting under Uncertainty: Discrete Bayesian Models for Mobile-Robot Navigation*. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 1996, pp. 963-972.

Chatila, R., Lacroix, S. and Giralt, G. [1997]. *A Case Study in Machine Intelligence: Adaptive Autonomous Space Rovers*. Proceedings of first International Conference on Field and Service Robotics, Canberra, Australia, December, 1997, pp. 101-108.

Chong, K.S. and Kleeman, L. [1997]. *Sonar Based Map Building for a Mobile Robot*. Proceedings of the IEEE International Conference on Robotics and Automation, 1997, pp. 1700-1705.

Cox, I. J. [1991]. *Blanche – An Experiment in Guidance and Navigation of an Autonomous Robot Vehicle*. IEEE Transactions on Robotics and Automation, Vol. 7, No. 2, April, 1991, pp. 193-204.

Csorba, M. [1997]. *Simultaneous Localisation and Map Building*. PhD thesis, University of Oxford, 1997.

Csorba, M., Uhlmann, J.K. and Durrant-Whyte, H.F. [1997]. *A Sub Optimal Algorithm for Automatic Map Building*. Proceedings of the American Control Conference, Albuquerque, New Mexico, June, 1997.

Deans, M. C. and Hebert, M. [2000]. *Invariant Filtering for Simultaneous Localization and Mapping*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 1042-1047.

Dellaert, F., Burgard, W., Fox, D. and Thrun, S. [1999a]. *Using the Condensation Algorithm for Robust, Vision-based Mobile Robot Localization*. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999, pp. 588-594.

Dellaert, F., Fox, D., Burgard, W. and Thrun, S. [1999b]. *Monte Carlo Localization for Mobile Robots*. IEEE International Conference on Robotics and Automation, 1999, pp. 1322 -1328.

Devy, M. and Parra, C. [1999]. *Topological Modeling for Outdoor Mobile Robots*. LAAS Report, No. 99144, March, 1999, 7 p.

Dissanayake, G., Hebert M., Stentz A. and Durrant-Whyte, H. [1997]. *Map Building and Terrain-Aided Localisation in an Underground Mine*. Proceedings of first International Conference on Field and Service Robotics, Canberra, Australia, December, 1997, pp. 34-40.

Dissanayake, G., Durrant-Whyte, H. and Bailey, T. [2000]. *A Computationally Efficient Solution to the Simultaneous Localization and Map Building [SLAM] Problem*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 1009-1014.

Duckett, T., Marsland, S. and Shapiro, J. [2000]. *Learning Globally Consistent Maps by Relaxation*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 3841-3846.

Durrant-Whyte, H.F. [1988]. *Integration, Coordination and Control of Multi-Sensor Robot Systems*. Kluwer Academic Publishers, 1998, 236 p.

Everett, H.R. [1995]. *Sensors for Mobile Robots; Theory and Application*. A K Peters, Ltd., p. 528.

Feddema, J.T. and Little, C.Q. [1997]. *Rapid World Modeling: Fitting Range Data to Geometric Primitives*. Proceedings of the IEEE International Conference on Robotics and Automation, 1997, pp. 2807–2812.

Feder, H. J. S., Leonard, J. J. and Smith C. M. [1998a]. *Adaptive Concurrent Mapping and Localization Using Sonar*. Proceedings of IEEE/RSJ International Conference on Robots and Systems [IROS], Victoria, B. C., Canada, October, 1998.

Feder, H. J. S., Leonard, J. J. and Smith C. M. [1998b]. *Incorporating Environment Measurements in Navigation*. In IEEE AUV, Cambridge, MA, August, 1998, pp. 115-122.

Feder, H. J. S, Leonard, J. J. and Smith, C. M. [1999]. *Adaptive mobile robot navigation and mapping*. International Journal of Robotics Research, Special Issue on Field and Service Robotics, Volume 18, Number 7, July, 1999, pp. 650-668.

Flynn, P.J. and Jain, A.K. [1988]. *Surface Classification: Hypothesis Testing and Parameter Estimation*. Proceedings CVPR '88, Computer Society Conference on Vision and Pattern Recognition, 1988, pp 261-267

Forsberg, J.; Larsson, U.; Wernersson, Å. [1995]. *Mobile Robot Navigation Using the Range-Weighted Hough Transform*. IEEE Robotics & Automation Magazine, Vol.2, No.1, March, 1995 , pp. 18-26.

Forsman, P. [1994] *Integration of intensity and range measurements; an approach based on quasi coaxial camera and range sensor*, Proceedings of the 3rd Int. Conference on Automation, Robotics and Computer Vision, Singapore, 8-11 November, 1994, pp. 202-206.

Forsman, P. [2001] *Feature Based Registration of 3D Perception Data for Indoor and Outdoor Map Building*, Proceedings of the 3rd Int. Conference on Field and Service Robotics, Finland, 11-13 June, 2001, pp. 367-372.

Forsman, P. and Halme, A. [1989]. *A model-based interactive robot control method for on-site task planning and execution*, Proceedings of the 20th International Symposium on Industrial Robots, Tokyo, Japan, October, 1989, pp. 297-304.

Forsman, P. and Halme, A. [1994]. *A two-stage method for selective modeling of unknown teleoperation environments*, Proceedings of the 3rd IEEE International Workshop on Robot and Human Communication, Nagoya, Japan, 18-20 July, 1994, pp. 333-338.

Forsman, P. and Halme, A. [1995]. *Computer graphics overlay on the camera view for interactive modeling of teleoperation environments*, 7th International Conference on Advanced Robotics [ICAR'95], Spain, September 20-22, 1995.

Fox, D., Burgard, W. and Thrun, S. [1998]. *Active Markov Localization for Mobile Robots*. Robotics and Autonomous Systems, No. 25, 1998, pp. 195-207.

Fox, D., Burgard, W., Dellaert, F. and Thrun, S. [1999a]. *Monte Carlo Localization: Efficient Position Estimation for Mobile Robots*. Proceedings of the 16th National Conference on Artificial Intelligence, [AAAI-99], Orlando, Florida, 1999.

Fox, D., Burgard, W. and Thrun, S. [1999b]. *Markov Localization for Mobile Robots in Dynamic Environments*. Journal of Artificial Intelligence Research, 11, 1999, pp. 391-427.

Ghahramani, Z. and Roweis, S. T. [1999a]. *Learning Nonlinear Dynamical Systems using an EM Algorithm*. In Kearns, M.S., Solla, S. A. and Cohn, D. A. [eds.], Advances in Neural Information Processing Systems 11, Cambridge, MA: MIT Press, 1999. [To appear].

Ghahramani, Z. and Roweis, S. T. [1999b]. *Probabilistic Models for Unsupervised Learning*. Tutorial presented at the 1999 Neural Information Processing Systems Conference. <http://www.gatsby.ucl.ac.uk/~zoubin/Nipstutorial.html>, 1999.

Guivant, J., Nebot, E. and Baiker, S. [2000]. *High Accuracy Navigation Using Laser Range Sensors in Outdoor Applications*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 3817-3822.

Gutmann, J.-S. and Schlegel, C. [1996]. *AMOS: Comparison of Scan Matching Approaches for Self-Localization in Indoor Environments*. Proceedings of the First Euromicro Workshop on Advanced Mobile Robot, 1996 , pp. 61 -67.

Gutmann, J.-S., Burgard, W., Fox, D. and Konolige, K. [1998]. *An Experimental Comparison of Localization Methods*. IEEE/RSJ International Conference on Intelligent Robots and Systems, 1998, pp. 736 -743.

Gutmann, J.-S. and Konolige, K. [1999]. *Incremental Mapping of Large Cyclic Environments*. Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation, 1999, pp. 318 -325.

Halme, A., Leppänen, I., Ylönen, S. and Kettunen, I. [2001] *Workpartner – Centaur Like Service Robot for Outdoors Applications*, Proceedings of the 3rd Int. Conference on Field and Service Robotics, Finland, 11-13 June, 2001, pp. 217-223.

Haralick, R.M., Hyonam, J., Chung-Nan, L., Xinhua, Z., Vinay, G.V. and Man, B.K. [1989]. *Pose Estimation from Corresponding Point Data*. IEEE Transactions on Systems, Man and Cybernetics, Vol. 19, No. 6, November/December, 1989, pp. 1426-1446.

Hébert, P., Laurendeau, D. and Poussart, D. [1993]. *Scene Reconstruction and Description: Geometric Primitive Extraction from Multiple View Scattered Data*. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1993, pp. 286-292.

Hébert, P., Laurendeau, D. and Poussart, D. [1994]. *Surface Profile Description: Reliable Geometric Primitive Extraction*. Proceedings of the 12th IAPR International Conference on Pattern Recognition, 1994, pp. 258-263.

Hébert, P., Betge-Brezets, S. and Chatila, R. [1995]. *Probabilistic map learning: Necessity and Difficulties*. LAAS Report 95542, December, 1995, 14 p.

Hébert, P., Betge-Brezets, S. and Chatila, R. [1996]. *Decoupling Odometry and Exteroceptive Perception in Building a Global World Map of a Mobile Robot: The Use of Local Maps*. Proceedings of the IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, April, 1996, pp. 757-764.

Hoover, A. [1996]. *The Space Envelope Representation for 3D Scenes*. Ph.D. Dissertation, University of South Florida, Tampa, Florida, December, 1996, 170p.

Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P. J., Bunke H., Goldgof, D. B., Bowyer, K., Eggert, D. W., Fitzgibbon A. and Fisher, R. B. [1996]. *An Experimental Comparison of Range Image Segmentation Algorithms*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 7, July, 1996, pp. 673-689.

Howard, A. and Kitchen, L. [1997]. *Vision-Based Navigation Using Natural Landmarks*. Proceedings of the first International Conference on Field and Service Robotics, Canberra, Australia, December, 1997, pp. 348-355.

Huber, D. F. and Hebert, M. [1999]. *A New Approach to 3-D Terrain Mapping*. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 1999, pp. 1121-1127.

Huber, D., Carmichael, O. and Hebert, M. [2000]. *3-D Map Reconstruction from Range Data*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 891-897.

Högström, T. and Wernersson, Å. [1998]. *On Segmentation, Shape Estimation and Navigation Using 3D Laser Range Measurements of Forest Scenes*. Preprints of 3rd IFAC Symposium on Intelligent Autonomous Vehicles, Madrid, Spain, March, 1998, pp. 547-552.

Jensfelt, P., Wijk, O., Austin, D. J. and Andersson, M. [2000]. *Experiments on Augmenting Condensation for Mobile Robot Localization*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 2518-2524.

Jiang, X. and Bunke, H. [1994]. *Fast Segmentation of Range Images into Planar Regions by Scan Line Grouping*. Machine Vision and Applications, vol. 7, no. 2, 1994, pp. 115-122.

Jiang, X. and Bunke, H. [1996]. *Robust Edge Detection in Range Images Based on Scan Line Approximation*. Technical Report IAM-96-016, University of Bern, 1996.

Johnson, A. E. and Hebert, M. [1999]. *Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No. 5, May, 1999, pp. 433-449.

Julier, S.J., Uhlmann, J.K. and Durrant-Whyte, H.F. [1995]. *A New Approach for Filtering Nonlinear Systems*. Proceedings of the American Control Conference, Seattle, Washington, 1995.

Julier, S.J. [1997]. *Process Models for the Navigation of High-Speed Land Vehicles*. PhD thesis, University of Oxford, 1997.

Julier, S.J. and Uhlmann, J.K. [1997a]. *A Non-Divergent Estimation Algorithm in the Presence of Unknown Correlations*. Proceedings of the American Control Conference, Albuquerque, New Mexico, June, 1997.

Julier, S.J. and Uhlmann, J.K. [1997b]. *A New Extension of the Kalman Filter to Nonlinear Systems*. The Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls, Orlando, Florida, 1997.

Julier, S.J., Uhlmann, J.K. and Durrant-Whyte, H.F. [2000]. *A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators*. IEEE Transactions on Automatic Control, Vol. 45, No. 3, March, 2000, pp. 477-482.

Juujärvi, J., Heikkonen, J., Brandt, S., and Lampinen, J. [1998]. *Digital Image Based Tree Measurement for Forest Inventory*, Intelligent Robots and Computer Vision XVII: Algorithms, Techniques, and Active Vision, Casasent, D. P., editor, Proc. SPIE 3522, pp. 114-123, 1998.

Konolige, K. and Chou, K. [1999]. *Markov Localization Using Correlation*. Proceedings of the International Joint Conference on Artificial Intelligence [IJCAI'99], Stockholm, 1999.

Kumar, S., Han, S., Goldgof, D. and Bowyer, K. [1995]. *On Recovering Hyperquadrics from Range Data*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 17, No. 11, November, 1995, pp. 1079-1083

Känsälä, K., Vähä, P., Kerva, J. and Saavalainen, P. [1998]. *Workhorse with flexible remote control unit for lumberjacks*. Proceedings of the 2nd International Conference on Machine Automation, ICMA'98, Tampere, Finland, September, 1998, pp. 305-309.

Kweon, I. S. and Kanade, T. [1992]. *High-Resolution Terrain Map from Multiple Sensor Data*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 2, February, 1992, pp. 278-292.

Lacroix, S., Chatila, R., Fleury, S., Herrb, M. and Siméon T. [1994]. *Autonomous Navigation in Outdoor Environment: Adaptive Approach and Experiment*. Proceedings of the IEEE International Conference on Robotics and Automation, San Diego, California, May, 1994, pp. 426-432.

Lacroix, S., Mallet, A., Chatila, R. and Gallo, L. [1999]. *Rover Self Localization in Planetary-Like Environments*. Proceedings of the 5th International Symposium on Artificial Intelligence, Robotics and Automation in Space [I-SAIRAS 99], Noordwijk, The Netherlands, June, 1999.

Lehtinen, H., Viitanen, J., Kaikkonen, J., Kaarmila, P. and von Numers, T.[1998]. *Autonomous Vehicles Evolving*. VTT Automation Technology Review 1998, Kelhä, V., Editor, December, 1998, pp. 6-11.

Lenser, S. and Veloso, M. [2000]. *Sensor Resetting Localization for Poorly Modeled Mobile Robots*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 1225-1232.

Leonard, J., Durrant-Whyte, H. and Cox, I. J. [1990]. *Dynamic Map Building for an Autonomous Mobile Robot*. IEEE International Workshop on Intelligent Robots and Systems, IROS'90, 1990, pp. 89-96.

Leonard, J. J. and Durrant-Whyte, H. F. [1991]. *Mobile Robot Localization by Tracking Geometric Beacons*. IEEE Transactions on Robotics and Automation, Vol. 7, No. 3, June, 1991, pp. 376-382.

Leonard, J. J. and Feder, H. J. S. [1999a]. *A computationally efficient method for large-scale concurrent mapping and localization*. Proceedings of the Ninth International Symposium on Robotics Research, Salt Lake City, Utah, October, 1999.

Leonard, J. J. and Feder, H. J. S. [1999b]. *Decoupled Stochastic Mapping*. MIT Marine Robotics Laboratory Technical Memorandum 99-1, December, 1999, 40p.

Lu, F. and Milios, E.E. [1994]. *Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans*. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1994, pp. 935-938.

Lu, F. and Milios, E.E. [1995]. *Optimal Global Pose Estimation for Consistent Sensor Data Registration*. Proceedings of the IEEE International Conference on Robotics and Automation, 1995, pp. 93-100.

Lu, F. and Milios, E. [1997]. *Globally Consistent Range Scan Alignment for Environment Mapping*. Autonomous Robots, Vol. 4, No. 4, 1997, pp. 333-349.

Luck, J., Little, C. and Hoff, W. [2000]. *Registration of Range Data Using a Hybrid Simulated Annealing and Iterative Closest Point Algorithm*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 3739-3744.

Madhavan, R, Dissanayake, M.W.M.G. and Durrant-Whyte, H.F. [1998]. *Autonomous Underground Navigation of an LHD using a Combined ICP-EKF Approach*. Proceedings of the 1998 IEEE International Conference on Robotics and Automation, Leuven, Belgium, August, 1998, pp. 3703-3708.

Martin, M. and Moravec, H. [1996]. *Robot Evidence Grids*. CMU Robotics Institute Technical Report, CMU-RI-TR-96-06, March, 1996, 48p.

Masuda, T. and Yokoya, N. [1994]. *A Robust Method for Registration and Segmentation of Multiple Range Images*. Proceedings of the Second CAD-Based Vision Workshop, 1994, pp. 106-113.

Matthies, L. and Shafer, S. [1987]. *Error modeling in stereo navigation*. IEEE Journal of Robotics and Automation, vol. RA-3, no. 3, , June 1987, pp. 239-248.

Maybeck, P.S. [1979]. *Stochastic Models, Estimation, and Control*. Volume 1, Academic Press, 1979, 423p.

Maybeck, P.S. [1982]. *Stochastic Models, Estimation, and Control*. Volume 2, Academic Press, 1972, 289p.

Moravec, H.P. [1988]. *Sensor fusion in certainty grids*. AI Magazine, summer 1988, pp. 61-74.

Moravec, H.P. [1999]. *Robust Navigation by Probabilistic Volumetric Sensing*. Proposal, Carnegie Mellon University, The Robotics Institute, <http://cart.frc.ri.cmu.edu/users/hpm/project.archive/robot.papers/1999/ARPA.proposal.99/ARPA.990108.html>, 1999.

Moutarlier, P. and Chatila R. [1989]. *Stochastic Multisensory Data Fusion for Mobile Robot Location and Environment Modeling*. Proceedings of the 5th International Symposium on Robotics Research, August, 1989, pp. 85-94.

Murphy, K. P. [1998]. *Switching Kalman Filters*. Technical Report, U.C. Berkeley, Department of Computer Science, 1998, 16p.

Murrieta-Cid, R., Parra, C., Devy, M. and Briot, M. [1998]. *Scene Modelling from 2D and 3D Sensory Data Acquired from Natural Environments*. LAAS Report, No. 98418, October, 1998, 8 p.

Mäkelä, H., Lehtinen, H., Rintanen, K. and Koskinen, K. [1995]. *Navigation System for LMD Machines*. Preprints of the 2nd IFAC Conference on Intelligent Autonomous Vehicles, Helsinki, Finland, 1995, pp. 314-319.

Ng, I.; Illingworth, J. and Jones, G. [1995]. *A Novel Method for Segmentation of Cones and Cylinders from Geometrically Fused Depth Maps*. Fifth International Conference on Image Processing and its Applications, pp. 544-548, 1995.

Olson, C.F. [1998]. *A Probabilistic Formulation for Hausdorff Matching*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1998, pp. 150-156.

Olson, C.F. [1999]. *Subpixel Localization and Uncertainty Estimation Using Occupancy Grids*. Proceedings of the IEEE International Conference on Robotics and Automation, Detroit, Michigan, 1999, pp. 1987-1992.

Olson, C.F. [2000]. *Probabilistic Self-Localization for Mobile Robots*. IEEE Transactions on Robotics and Automation, Vol. 16, No. 1, February, 2000, pp. 55-66.

Parra, C., Murrieta-Cid, R., Devy, M. and Briot, M. [1998]. *3-D Modelling and Robot Localization from Visual and Range Data in Natural Scenes*. LAAS Report, No. 98239, May, 1998, 20 p.

Payeur, P., Hébert, P., Laurendeau, D. and Gosselin, C.M. [1997]. *Probabilistic Octree Modeling of a 3-D Dynamic Environment*. Proceedings of the IEEE International Conference on Robotics and Automation, Albuquerque, New Mexico, April, 1997, pp. 1289-1296.

Powell, M.W., Bowyer, K.W., Jiang, X and Bunke, H. [1998]. *Comparing Curved-Surface Range Image Segmenters*. Sixth International Conference on Computer Vision, 1998, pp. 286 -291.

Prassler, E.A. and Milios, E.E. [1995]. *Position Estimation Using Equidistance Lines*. Proceedings of the IEEE International Conference on Robotics and Automation, 1995, pp. 85-92.

Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P. [1992]. *Numerical Recipes in C*. Cambridge University Press, Cambridge, 2nd edition, 1992, 994 p.

Quek, F., Jain, R. and Weymouth, T. E. [1993]. *An Abstraction-Based Approach to 3-D Pose Determination from Range Images*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15, No. 7, July, 1993, pp. 722-736.

Rencken, W. D. [1993]. *Concurrent Localisation and Map Building for Mobile Robots Using Ultrasonic Sensors*. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Yokohama, Japan, July 26-30, 1993, pp. 2192-2197.

Rose, K. [1998]. *Deterministic Annealing for Clustering, Compression, Classification, Regression, and Related Optimization Problems*. Proceedings of the IEEE, Vol. 86, No. 11, November, 1998, pp. 2210-2239.

Roweis, S. and Ghahramani, Z. [1999]. *A Unifying Review of Linear Gaussian Models*. Neural Computation 11, 1999, pp. 305-345.

Roy, N., Burgard, W., Fox, D. and Thrun, S. [1999]. *Coastal Navigation - Mobile Robot Navigation with Uncertainty in Dynamic Environments*. IEEE International Conference on Robotics and Automation, pp. 35 -40.

Schiele, B. and Crowley, J. [1994]. *A Comparison of Position Estimation Techniques Using Occupancy Grids*. Robotics and Autonomous Systems 12, 1994, pp. 163-171.

Schonherr, F., Hertzberg, J. and Burgard, W. [1999]. *Probabilistic Mapping of Unexpected Objects by a Mobile Robot*. Proceedings of the International Conference on Intelligent Robots and Systems, IROS'99, 1999, pp. 474 -481.

Schultz, A.C., Adams, W. and Yamauchi, B. [1999]. *Integrating Exploration, Localization, Navigation and Planning with a Common Representation*. Autonomous Robots 6, 1999, pp. 293-308.

Sequeira, V., Goncalves, J.G.M. and Ribeiro, M.I. [1995]. *3D Environment Modeling Using Laser Range Sensing*. Robotics and Autonomous Systems, 16, 1995, pp. 81-91.

Shatkay, H. [1999]. *Learning Hidden Markov Models with Geometrical Constraints*. To appear in UAI99, 1999.

Sinha, S. S. and Schunck, B. G. [1992]. *A Two-Stage Algorithm for Discontinuity-Preserving Surface Reconstruction*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 1, January, 1992, pp. 36-55.

Smith, R., Self, M. and Cheeseman, P. [1990]. *Estimating uncertain spatial relationships in robotics*. In I.J. Cox and G.T. Wilfong, editors, Autonomous Robot Vehicles, Springer-Verlag, 1990, pp. 167-193.

Smith, C. M. and Leonard, J. J. [1997]. *A Multiple-Hypothesis Approach to Concurrent Mapping and Localization for Autonomous Underwater Vehicles*. Proceedings of first International Conference on Field and Service Robotics, Canberra, Australia, December, 1997, pp. 249-256.

Sutherland, K. T. and Thompson, W. B. [1994]. *Localizing in Unstructured Environments: Dealing with the Errors*. IEEE Transactions on Robotics and Automation, Vol. 10, No. 6, December, 1994, pp. 740-754.

Taylor, R. M. and Probert, P. J. [1996]. *Range Finding and Feature Extraction by Segmentation of Images for Mobile Robot Navigation*. Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, April, 1996, pp. 95-100.

Thrun, S. [1998a]. *Learning Metric-Topological Maps for Indoor Mobile Robot Navigation*. Artificial Intelligence, 99[1], 1998, pp.21-71.

Thrun, S. [1998b]. *Bayesian Landmark Learning for Mobile Robot Localization*. Machine Learning, 33[1], 1998, 33p.

Thrun, S., Burgard, W. and Fox, D. [1998a]. *A Probabilistic Approach to Concurrent Mapping and Localization for Mobile Robots*. Autonomous Robots, No. 5, 1998, pp. 253-271.

Thrun, S., Gutmann, S., Fox, D., Burgard, W. and Kuipers, B. [1998b]. *Integrating Topological and Metric Maps for Mobile Robot Navigation: A Statistical Approach*. Proceedings of AAAI-98. 1998.

Thrun, S., Buecken, A., Burgard, W., Fox, D., Froehlinghaus, T., Henning, D., Hofmann, T., Krell, M. and Schmidt, T. [1998c]. *Map Learning and High-Speed Navigation in RHINO*. In AI-based Mobile Robots: Case Studies of Successful Robot Systems. Kortenkamp, D. and Bonasso, R.P. and Murphy, R. [eds.], MIT Press, 1998, pp. 21-52.

Thrun, S., Fox, D. and Burgard, W. [2000a]. *Monte Carlo Localization With Mixture Proposal Distribution*. In Proceedings of the AAAI National Conference on Artificial Intelligence, Austin, TX, 2000. AAAI.

Thrun, S., Burgard, W. and Fox, D. [2000b]. *A Real-Time Algorithm for Mobile Robot Mapping with Applications to Multi-Robot and 3D Mapping*. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, April, 2000, pp. 321-329.

Thrun, S., Fox, D. and Burgard, W. [2000c]. *Robust Monte Carlo Localization for Mobile Robots*. Research Report CMU-CS-00-125, 2000, 39p.

Uhlmann, J. K., Julier, S. J., Csorba, M. and Durrant-Whyte, H. F. [1996]. *A Culminating Advance in the Theory and Practice of Data Fusion, Filtering and Decentralized Estimation*, 1996. Available from <http://ait.nrl.navy.mil/people/uhlmann/CovInt.html>.

Umeda, K., Ikushima K. and Arai, T. [1996]. *Fusion of Range Image and Intensity Image for 3D Shape Recognition*. Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, April, 1996, pp. 680-685.

Weiss, G., Wetzler, C. and von Puttkamer, E. [1994]. *Keeping Track of Position and Orientation of Moving Indoor Systems by Correlation of Range-Finder Scans*. Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems '94. 'Advanced Robotic Systems and the Real World', IROS '94, 1994, pp. 595-601.

Whaite, P. and Ferrie, F. P. [1997]. *Autonomous Exploration: Driven by Uncertainty*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 3, March, 1997, pp. 193-205.

von Wichert, G. [1998]. *Mobile Robot Localization Using a Self-Organized Visual Environment Representation*. Robotics and Autonomous Systems, vol. 25, pp. 185-194.

Yu, X, Bui, T.D. and Krzyzak, A. [1994]. *Robust Estimation for Range Image Segmentation and Reconstruction*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 5, May 1994, pp. 530-538.

Yamauchi, B., Schultz, A. and Adams, W. [1998]. *Mobile Robot Exploration and Map-Building with Continuous Localization*. Proceedings of the IEEE International Conference on Robotics and Automation, 1998, pp. 3715 -3720.

Zhang, Z. [1994]. *Iterative Point Matching for Registration of Free-Form Curves and Surfaces*. International Journal of Computer Vision, 13[2], 1994, pp. 119-152.

Appendices

- Appendix A Range image data
- Appendix B Feature extraction from range images
- Appendix C Registration of pairs of feature sub-maps
- Appendix D Integrated environment model data
- Appendix E State estimation for discrete-time, linear, stochastic systems
- Appendix F State estimation for discrete-time, non-linear, stochastic systems
- Appendix G Distribution approximation filter

Parking lot test environment (called "outdoor")

Illustrations of the six pseudo-colored range images for constructing an environment model for the parking lot area next to the Computer Science Building at the Helsinki University of Technology are presented in the following images. The range images have been measured by means of the Riegl LMS-Z210 range scanner.

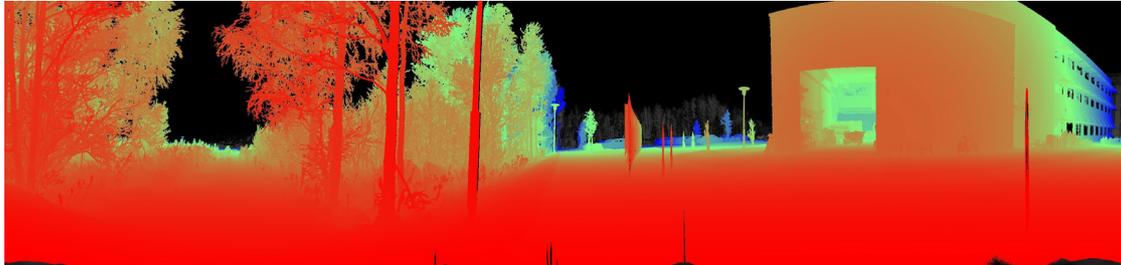


Figure A.1 An illustration of the range image "scanE". Ranges between 2.0 and 40.0 meters have been pseudo-colored from red to blue, respectively.

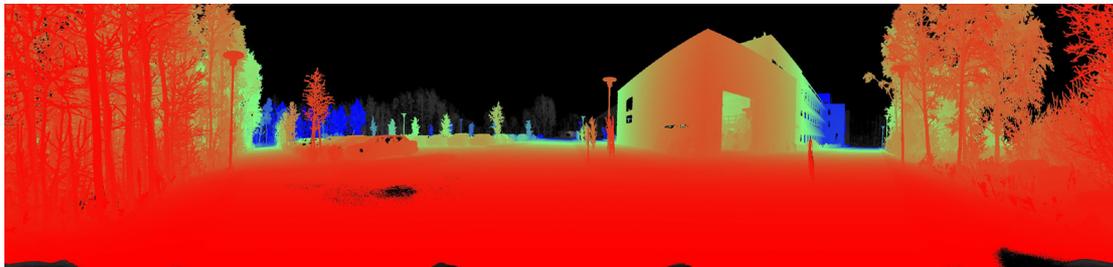


Figure A.2 An illustration of the range image "scanF". Ranges between 2.0 and 60.0 meters have been pseudo-colored from red to blue, respectively.

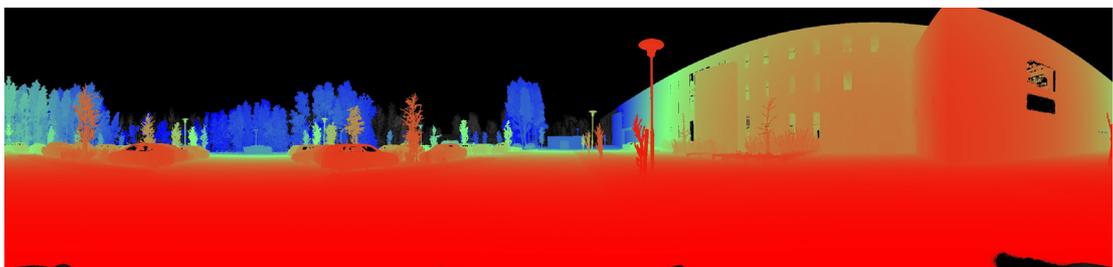


Figure A.3 An illustration of the range image "scanG". Ranges between 2.0 and 60.0 meters have been pseudo-colored from red to blue, respectively.

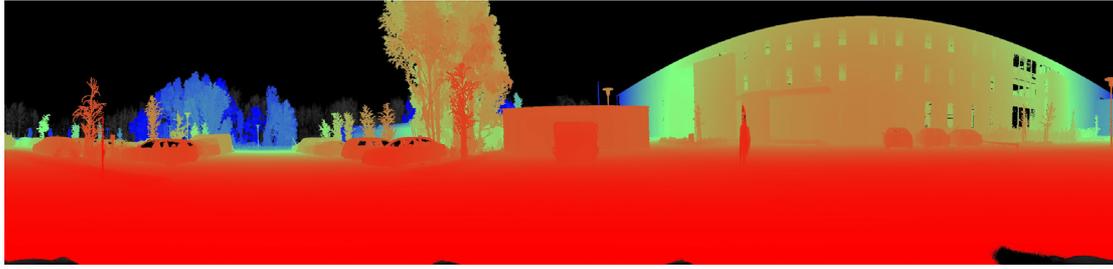


Figure A.4 An illustration of the range image "scanH". Ranges between 2.0 and 60.0 meters have been pseudo-colored from red to blue, respectively.

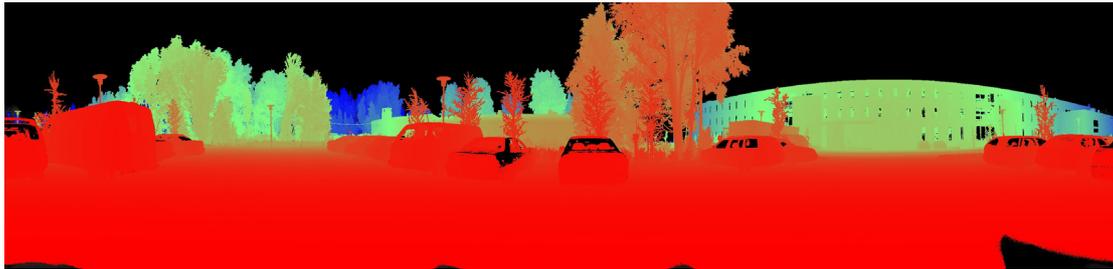


Figure A.5 An illustration of the range image "scanI". Ranges between 2.0 and 80.0 meters have been pseudo-colored from red to blue, respectively.

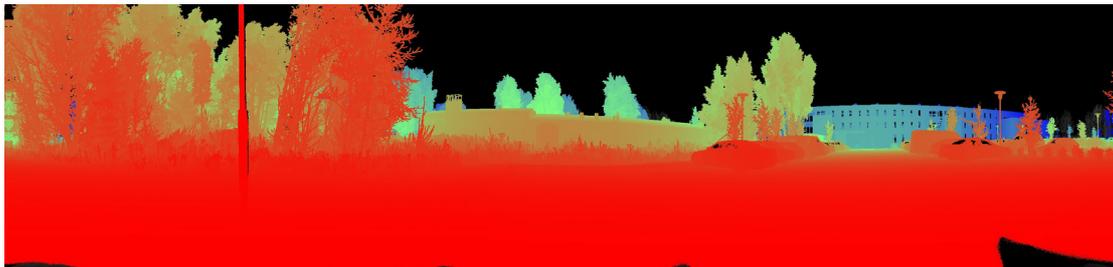


Figure A.6 An illustration of the range image "scanJ". Ranges between 2.0 and 80.0 meters have been pseudo-colored from red to blue, respectively.

In order to get a better understanding of the data and the environment to be modeled, camera images, taken from the same environment, are presented in the following four figures.



Figure A.7 A camera image taken from the origin of the coordinate frame of "scanE" towards the location where the range data set, labeled F, was measured.



Figure A.8 A camera image taken from the origin of the coordinate frame of "scanF".



Figure A.9 A camera image taken from the origin of the coordinate frame of "scanF" along the driveway on which the range images "scanG" and "scanH" were measured.



Figure A.10 A camera image taken from the origin of the coordinate frame of "scanJ" across the parking lot towards the location (near the building) where the range image "scanG" was measured.

Office corridor test environment (called "indoor")

Illustrations of the 10 (originally pseudo-colored) range images for constructing an environment model for a part of the ground floor of the Computer Science Building at the Helsinki University of Technology are presented in the following images. The range images have been measured by means of the Riegl LMS-Z210 range scanner.

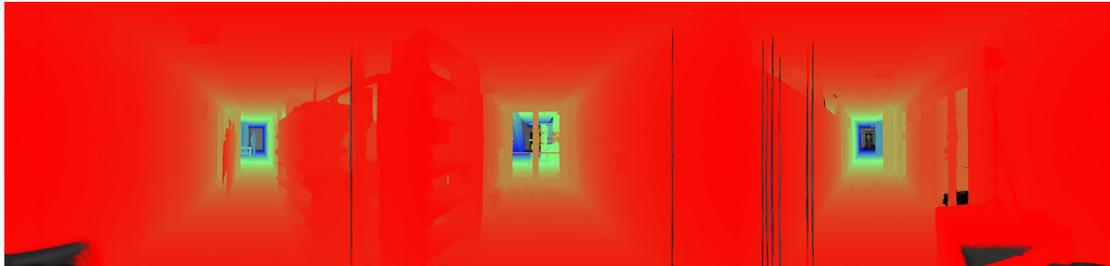


Figure A.11 An illustration of the range image "scan1". Ranges between 1.0 and 20.0 meters have been pseudo-colored from red to blue, respectively.

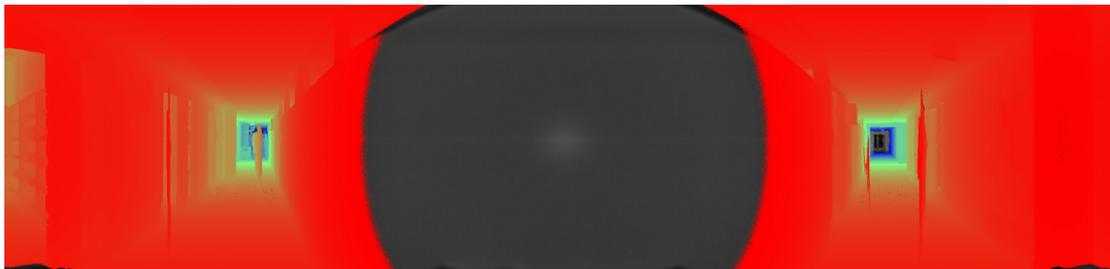


Figure A.12 An illustration of the range image "scan2". Ranges between 1.0 and 20.0 meters have been pseudo-colored from red to blue, respectively.

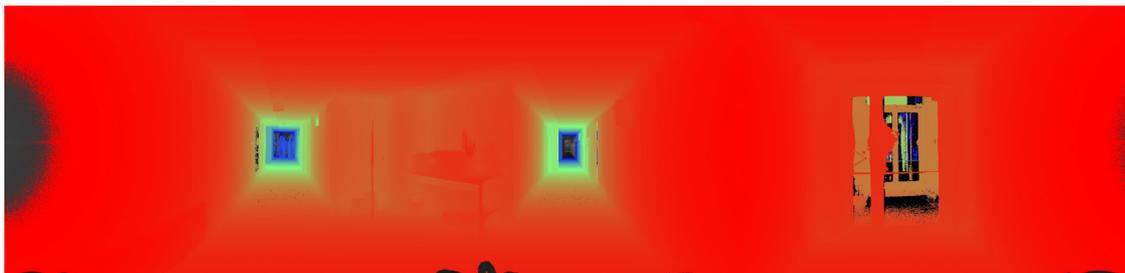


Figure A.13 An illustration of the range image "scan3". Ranges between 1.0 and 20.0 meters have been pseudo-colored from red to blue, respectively.



Figure A.14 An illustration of the range image "scan4". Ranges between 1.0 and 20.0 meters have been pseudo-colored from red to blue, respectively.



Figure A.15 An illustration of the range image "scan5". Ranges between 1.0 and 20.0 meters have been pseudo-colored from red to blue, respectively.

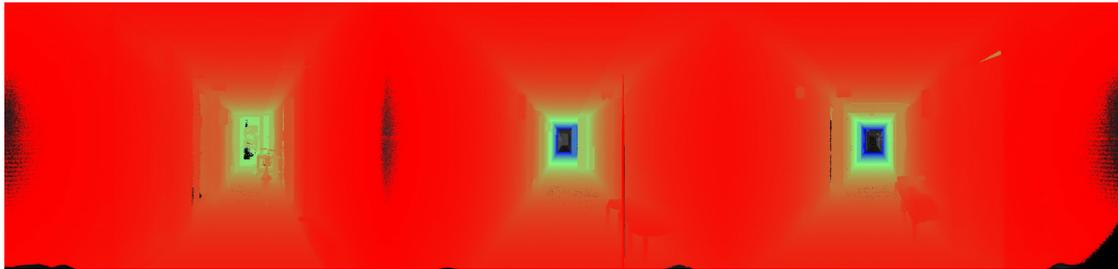


Figure A.16 An illustration of the range image "scan6". Ranges between 1.0 and 20.0 meters have been pseudo-colored from red to blue, respectively.



Figure A.17 An illustration of the range image "scan7". Ranges between 1.0 and 20.0 meters have been pseudo-colored from red to blue, respectively.

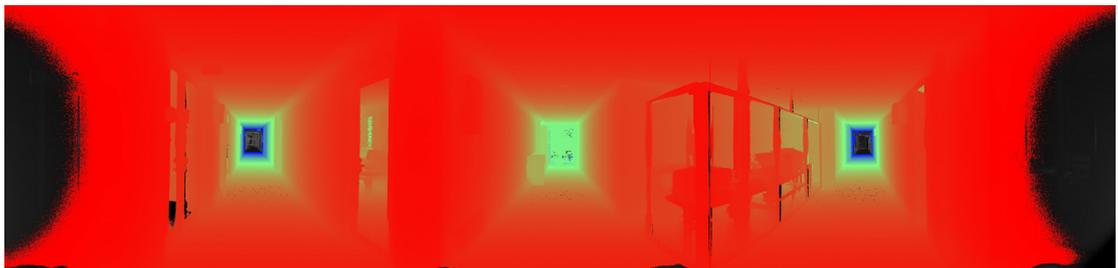


Figure A.18 An illustration of the range image "scan8". Ranges between 1.0 and 20.0 meters have been pseudo-colored from red to blue, respectively.

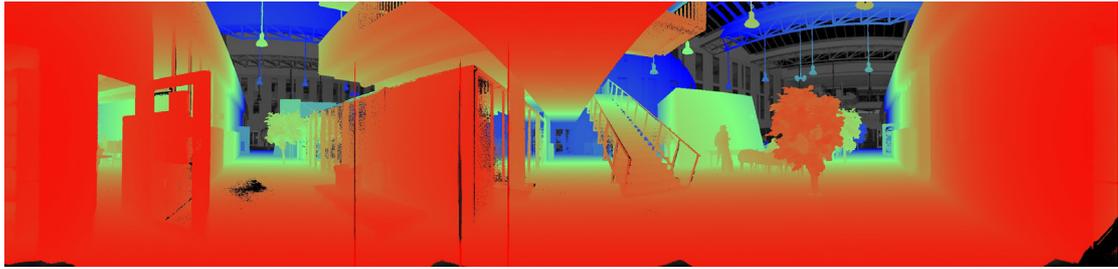


Figure A.19 An illustration of the range image "scan9". Ranges between 1.0 and 20.0 meters have been pseudo-colored from red to blue, respectively.

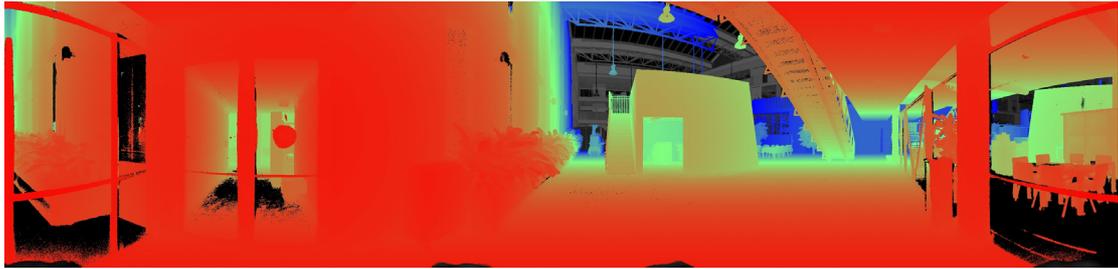


Figure A.20 An illustration of the range image "scanA". Ranges between 1.0 and 20.0 meters have been pseudo-colored from red to blue, respectively.

And then three camera images taken from the same location as "scan6", presented in Figure 16, are shown on the next page. The camera images shown in the Figures 20, 21 and 22 are taken against the corridors on the left, center and right respectively, visible in Figure 16.



Figure A.21 A camera image taken from the origin of the coordinate frame of "scan6" against the leftmost corridor visible in the range image.



Figure A.22 A camera image taken from the origin of the coordinate frame of "scan6" against the center corridor visible in the range image.



Figure A.23 A camera image taken from the origin of the coordinate frame of "scan6" against the rightmost corridor visible in the range image.

Forest test environment (called "forest")

Pseudo-colored illustrations of the six range images for constructing an environment model for a small forest area are presented in the following figures. The range images have been measured by means of the Riegl LMS-Z210 range scanner.

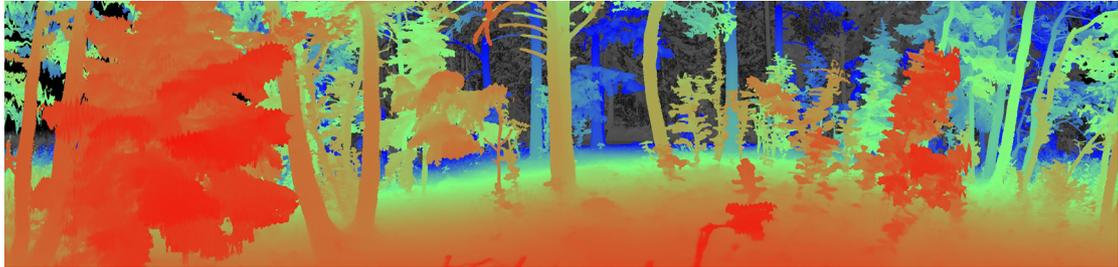


Figure A.24 An illustration of the range image "scanT". Ranges between 0.5 and 10.0 meters have been pseudo-colored from red to blue, respectively.

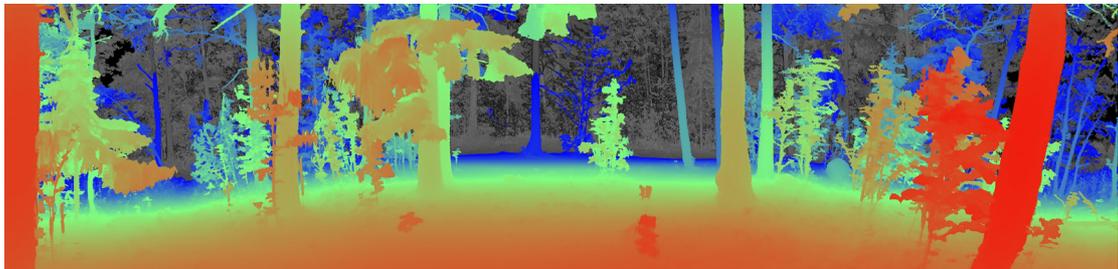


Figure A.25 An illustration of the range image "scanS". Ranges between 0.5 and 10.0 meters have been pseudo-colored from red to blue, respectively.

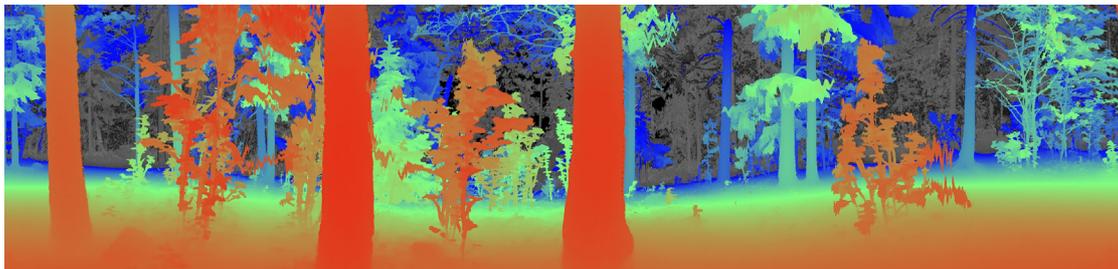


Figure A.26 An illustration of the range image "scanR". Ranges between 0.5 and 10.0 meters have been pseudo-colored from red to blue, respectively.

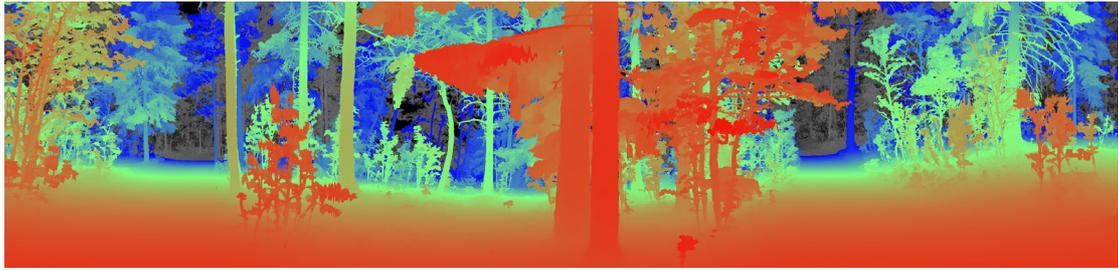


Figure A.27 An illustration of the range image "scanQ". Ranges between 1.0 and 15.0 meters have been pseudo-colored from red to blue, respectively.

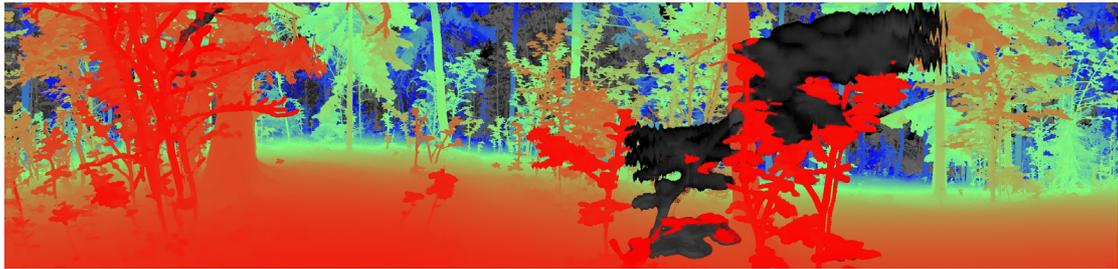


Figure A.28 An illustration of the range image "scan0". Ranges between 1.0 and 15.0 meters have been pseudo-colored from red to blue, respectively.

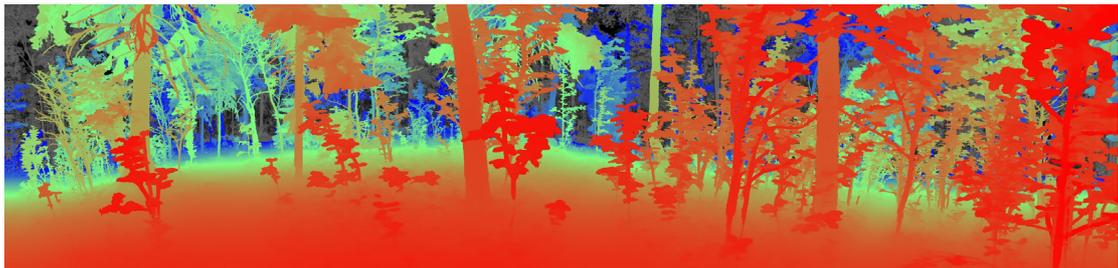


Figure A.29 An illustration of the range image "scanN". Ranges between 1.0 and 15.0 meters have been pseudo-colored from red to blue, respectively.

Extraction of cylinder segments (or more generally truncated cone segments) from range images

In the results, the 3000x720 binary images, corresponding to the initial segmentation of the range data and to the estimates of valid circle segment models, are presented. In the circle segment images, the width of each horizontal white line, corresponding to a valid circle segment, has been determined from the radius of the estimated circle. The corresponding range images have been illustrated in Appendix A.

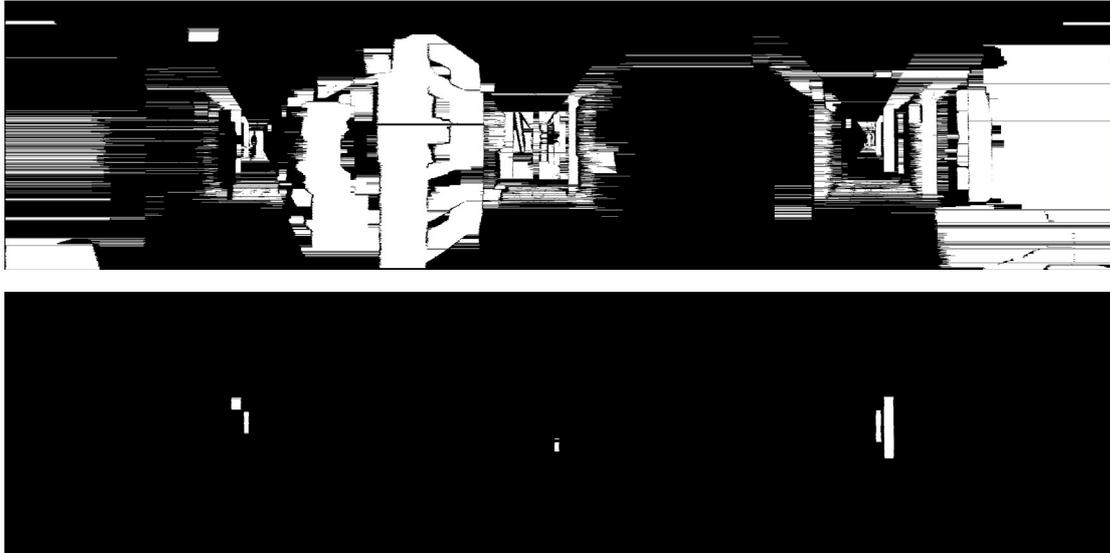


Figure B.1 Initial segmentation of the range profile data "scan1" in the upper image. In the lower image, the valid circle models, computed from the segments in the upper image, are shown.



Figure B.2 Initial segmentation of the range profile data "scan2" in the upper image. In the lower image, the valid circle models, computed from the segments in the upper image, are shown.



Figure B.3 Initial segmentation of the range profile data "scanT" in the upper image. In the lower image, the valid circle models, computed from the segments in the upper image, are shown.

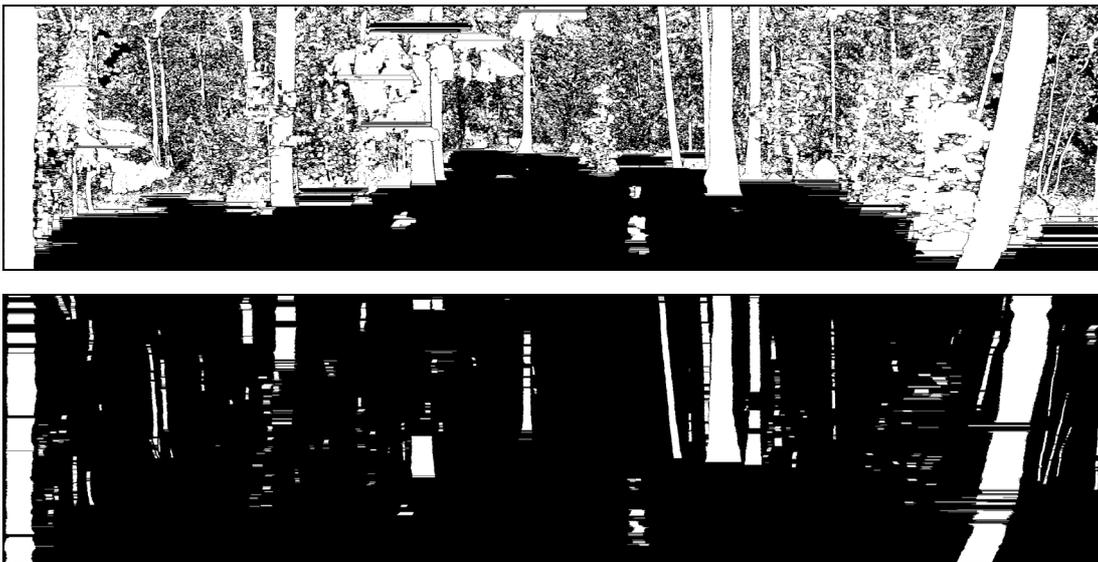


Figure B.4 Initial segmentation of the range profile data "scanS" in the upper image. In the lower image, the valid circle models, computed from the segments in the upper image, are shown.



Figure B.5 Initial segmentation of the range profile data "scanR" in the upper image. In the lower image, the valid circle models, computed from the segments in the upper image, are shown.



Figure B.6 Initial segmentation of the range profile data "scanQ" in the upper image. In the lower image, the valid circle models, computed from the segments in the upper image, are shown.

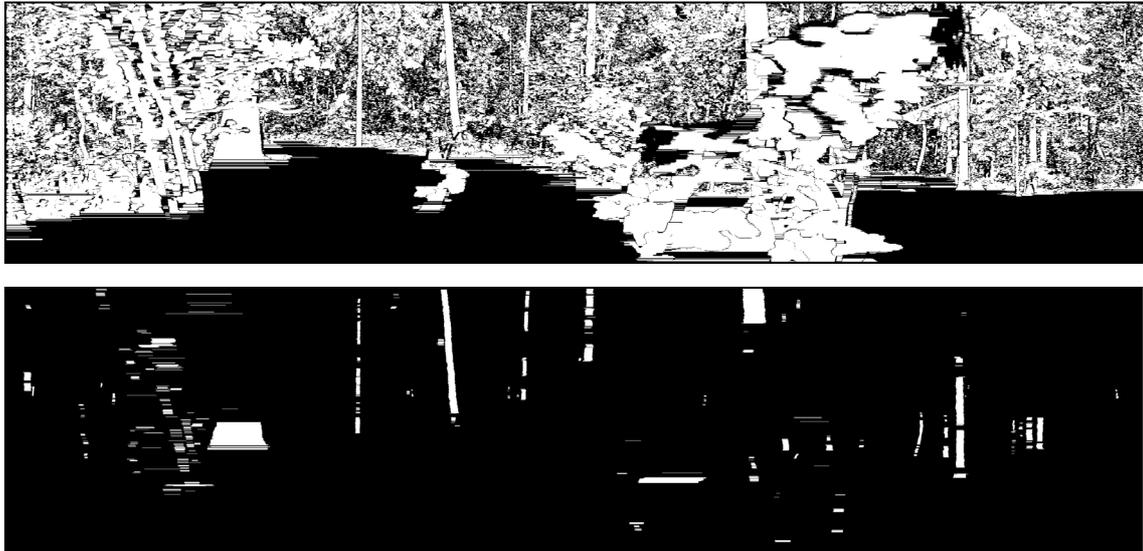


Figure B.7 Initial segmentation of the range profile data "scan0" in the upper image. In the lower image, the valid circle models, computed from the segments in the upper image, are shown.



Figure B.8 Initial segmentation of the range profile data "scanN" in the upper image. In the lower image, the valid circle models, computed from the segments in the upper image, are shown.

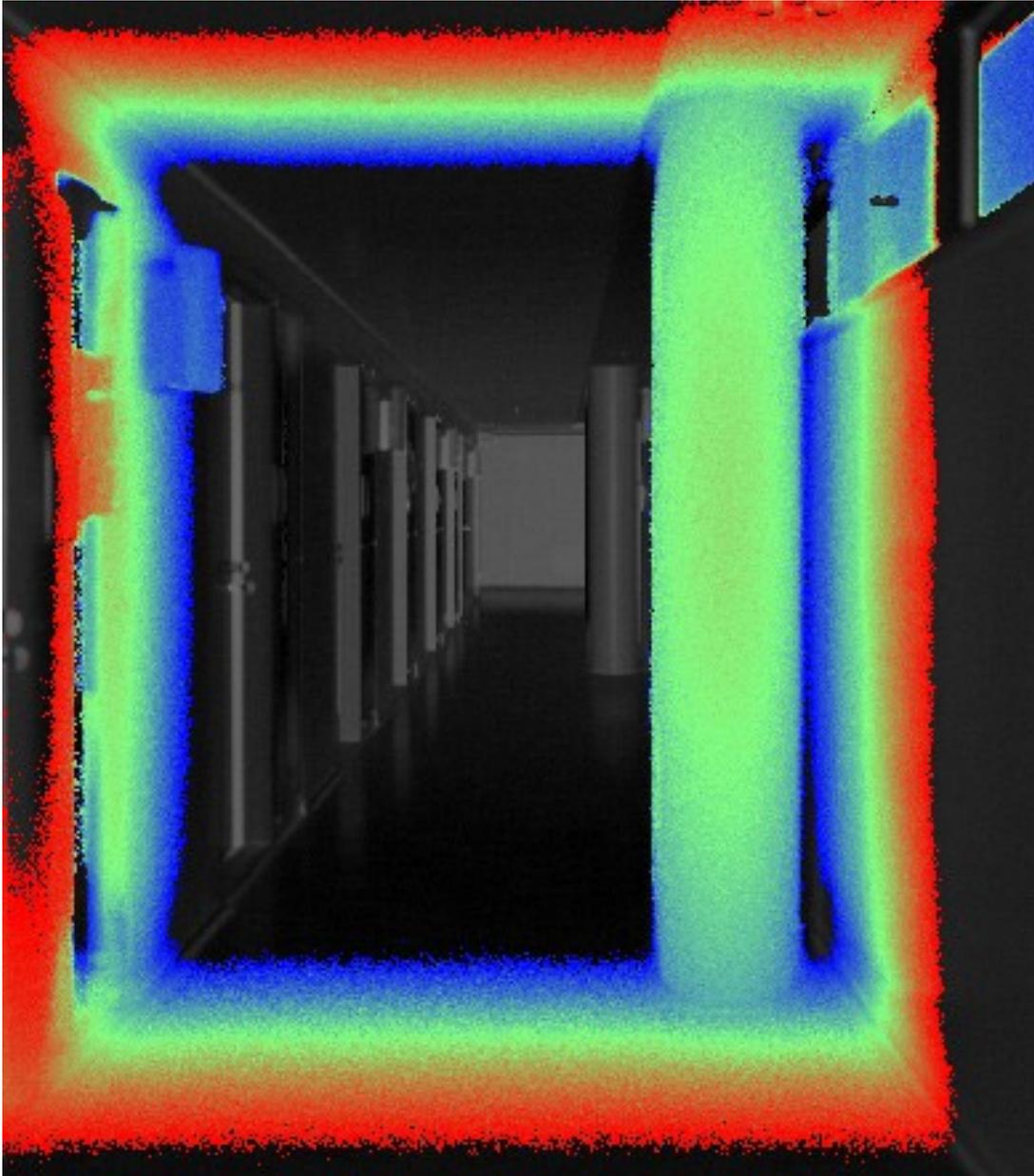


Figure B.9 A closer look of the pillar object at about 3.7 meters' distance from the range scanner.

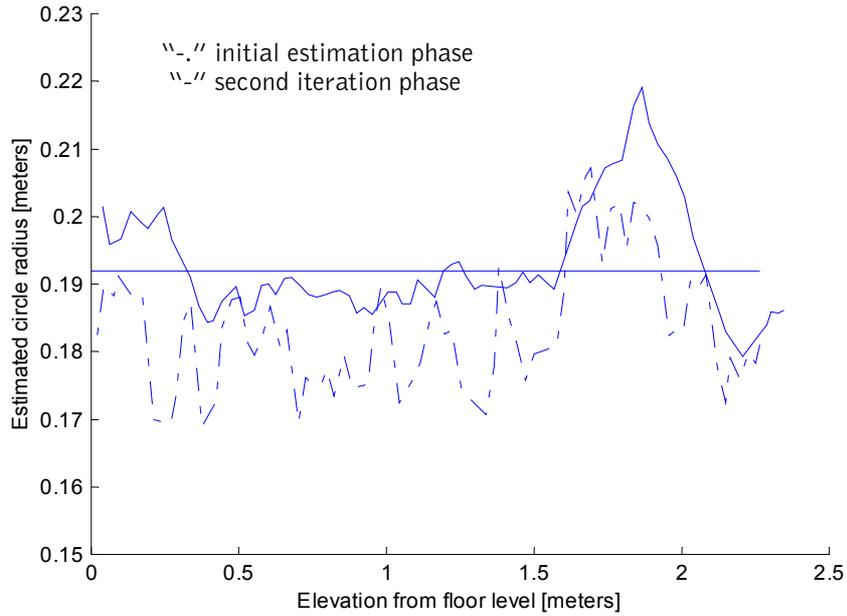


Figure B.10 The computed circle model radius as the function of its elevation from floor level. The distance of the pillar object from the range scanner was about 14.5 meters. The projection plane for the initial circle model estimation phase was tilted 30 degrees along the measurement direction.

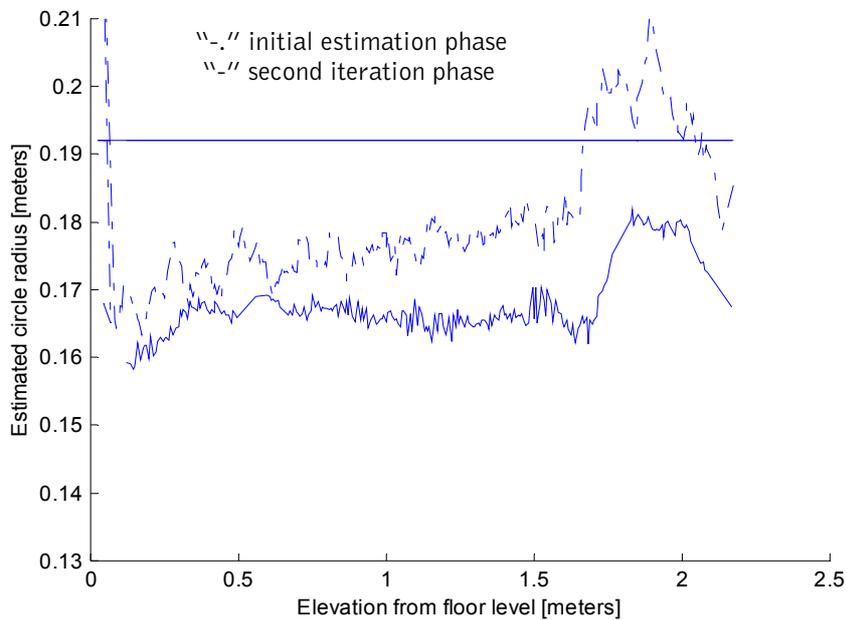


Figure B.11 The computed circle model radius as the function of its elevation from floor level. The distance of the pillar object from the range scanner was about 3.7 meters. The projection plane for the initial circle model estimation phase was tilted 30 degrees along the measurement direction.

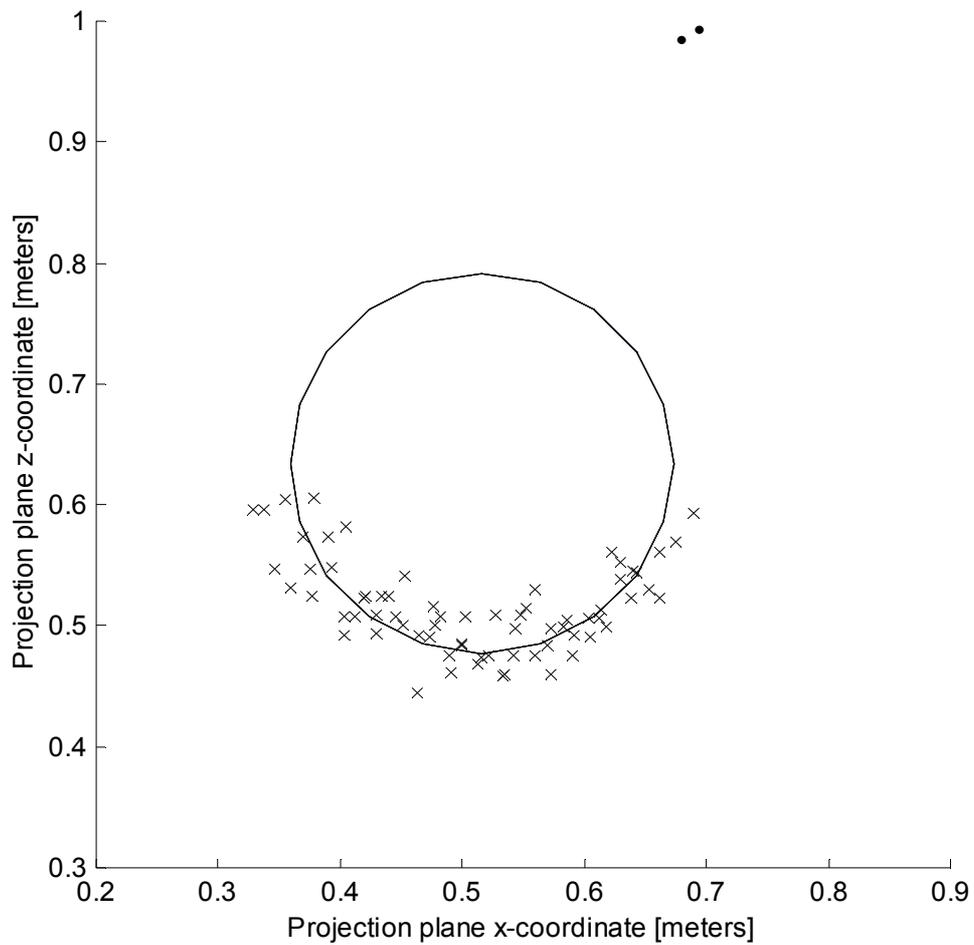


Figure B.12. Estimated circle model and the projected measurement data points corresponding to the initial estimation phase. The pillar object to be modeled was located at about 14.5 meters' distance from the sensor. For the results the projection plane was tilted 30 degrees sideways with respect to the measurement direction. The estimated radius of the circle is 0.157 meters. Data points, classified as valid are marked with a cross and data points classified as outlier with a dot.

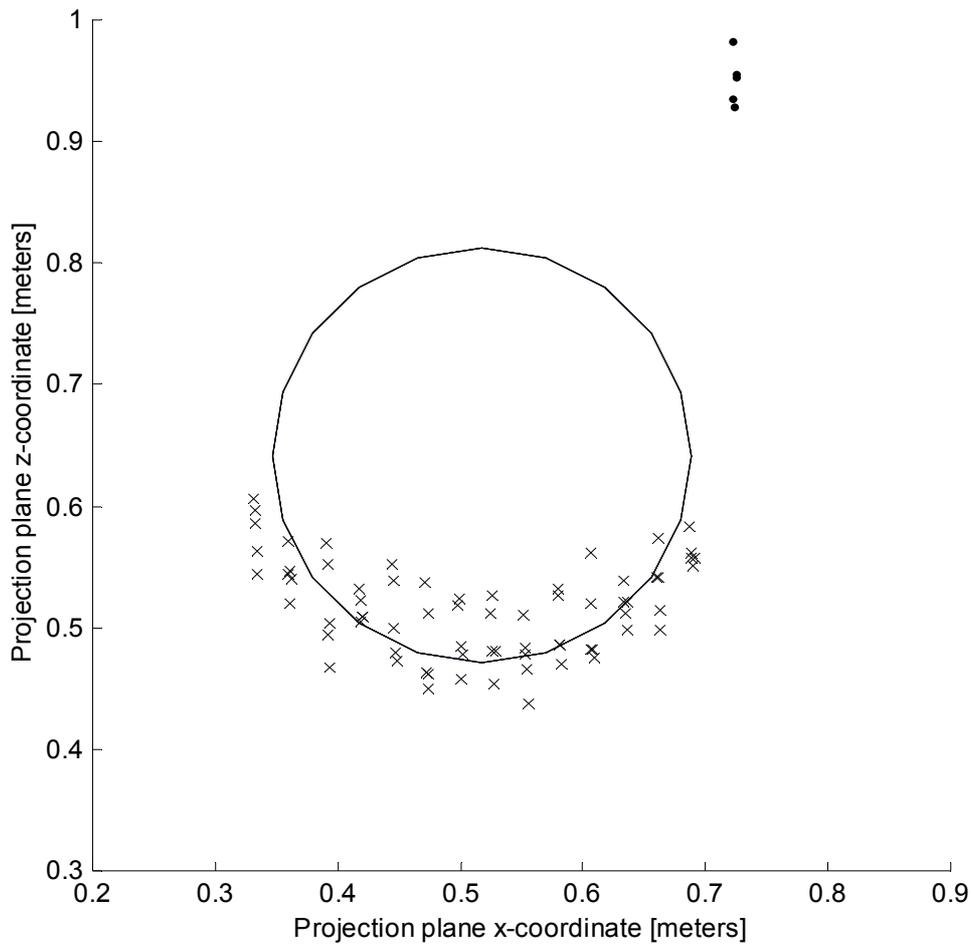


Figure B.13. Estimated circle model and the projected measurement data points corresponding to the initial estimation phase. The pillar object to be modeled was located at about 14.5 meters' distance from the sensor. For the results the projection plane was tilted 30 degrees along the measurement direction. The estimated radius of the circle is 0.170 meters. Data points, classified as valid are marked with a cross and data points classified as outlier with a dot.

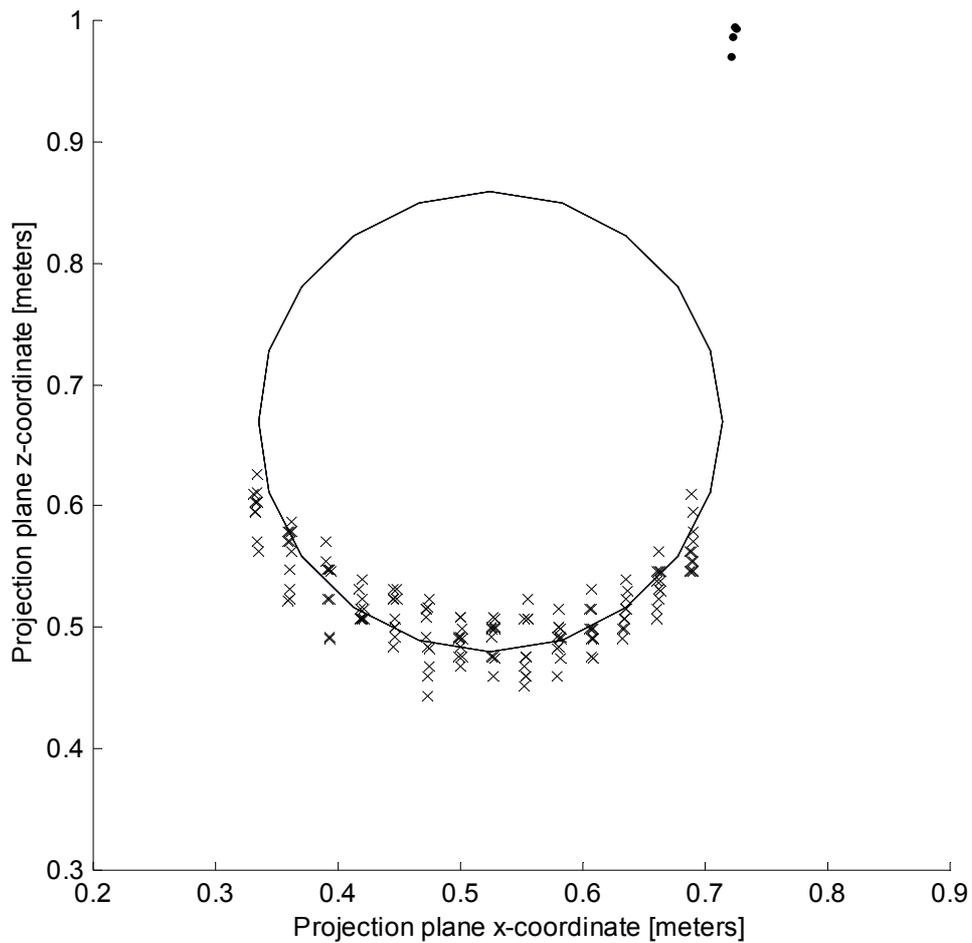


Figure B.14. Estimated circle model and the projected measurement data points corresponding to the final estimation phase of the test case shown in Figure B.12. The estimated radius of the circle is 0.190 meters. Altogether 154 (inlier) data points were used to estimate the circle model parameters, which is about two times more points than in the initial estimation phase, depicted in Figure B.12. Data points, classified as valid are marked with a cross and data points classified as outlier with a dot.

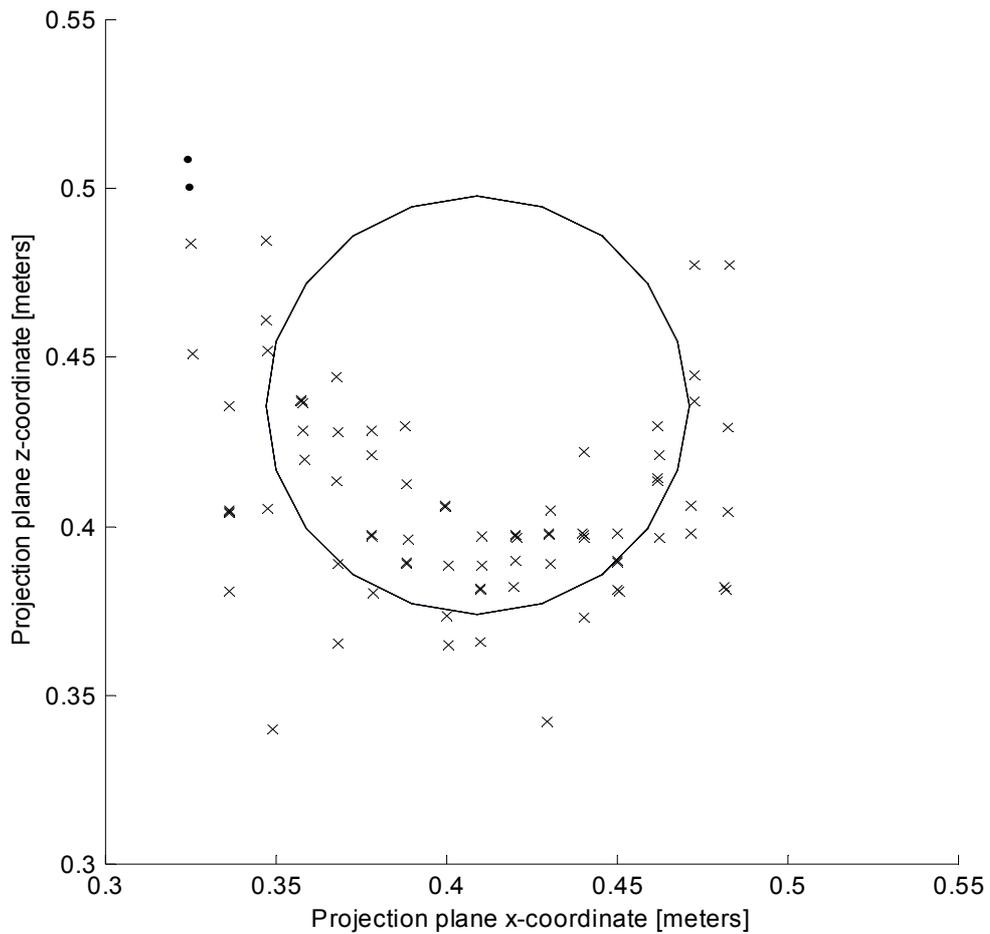


Figure B.15. Estimated circle model and the projected measurement data points corresponding to the initial estimation phase. The target object corresponds to a thin tree at about 5.3 meters' distance from the sensor. Measurement points within the segment under consideration and two neighboring segments, both above and below the current segment, were used to compute the circle model parameters. The estimated radius of the circle is 0.062 meters. Data points, classified as valid are marked with a cross and data points classified as outlier with a dot.

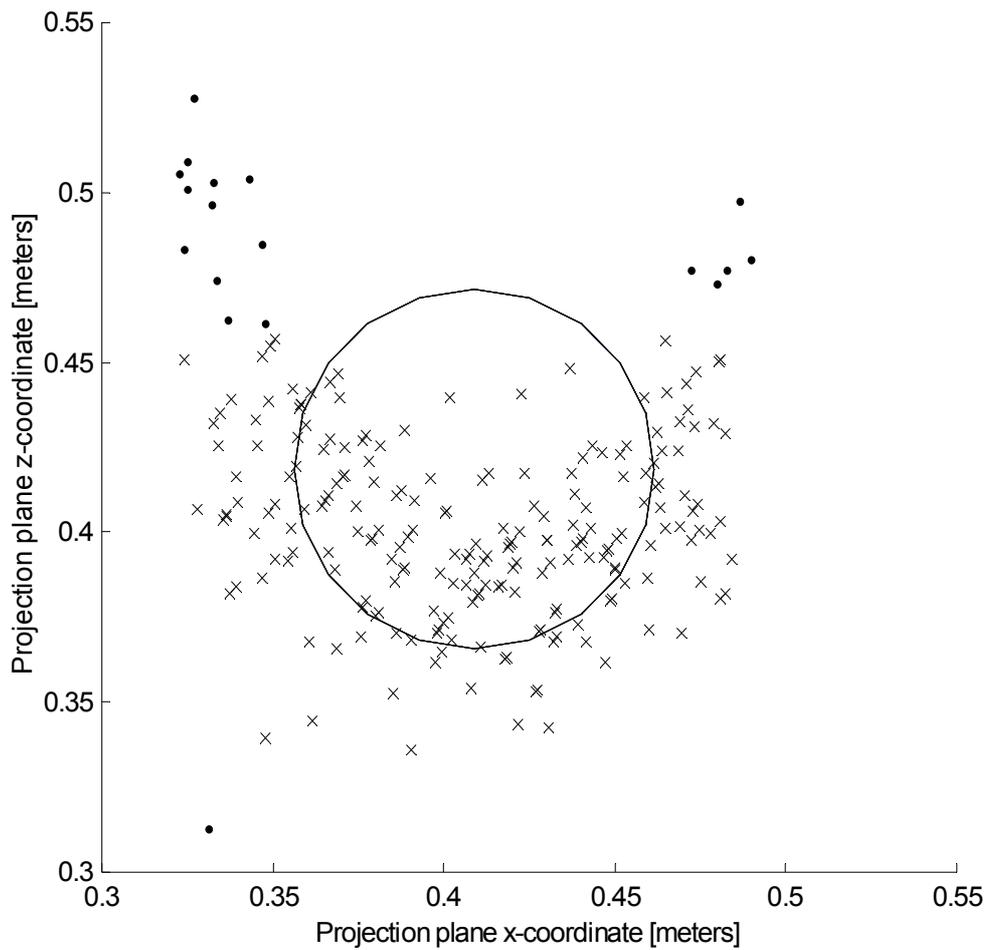


Figure B.16. Estimated circle model and the projected measurement data points corresponding to the final estimation phase of the test case shown in Figure B.15. The estimated radius of the circle is 0.053 meters, which is quite close to the ground truth measured value of 0.055 meters. Altogether 213 (inlier) data points were used to estimate the circle model parameters, which is about three times more points than in the initial estimation phase, depicted in Figure B.15. Data points, classified as valid are marked with a cross and data points classified as outlier with a dot.

Extraction of straight line segments, corresponding to the upper end of vertical walls

In the following figures, the output of the first phase of the straight-line segment extraction method in the parking lot mapping experiment, are presented. A 3D virtual model presentation of the straight line segments, computed from the free space to unknown/occupied space borderline data, can be seen in Appendix D, Figures D.7 to D.12.

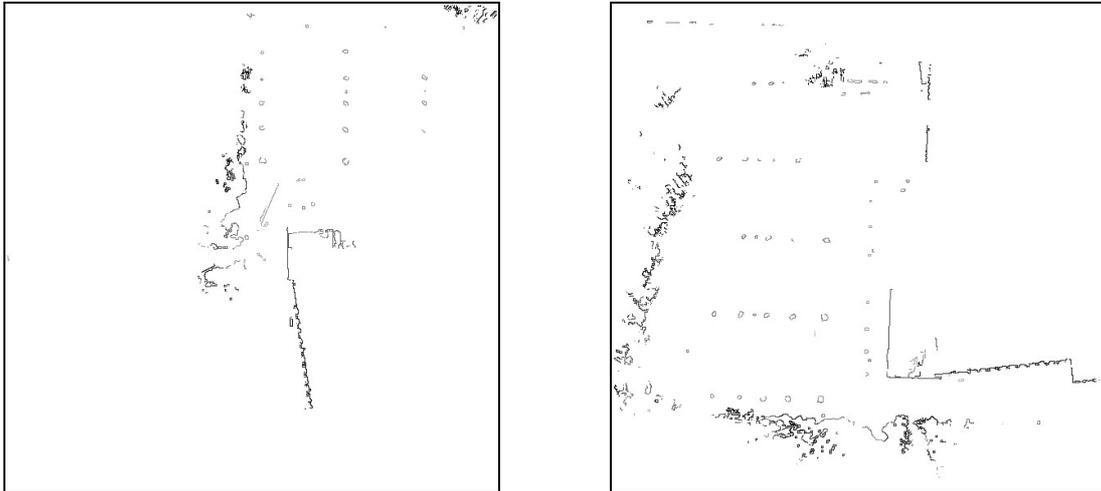


Figure B.17 A 2D projection of the free space to unknown/occupied space border computed from the 3D occupancy grid. The darker the shade of gray, the higher the highest location where a laser beam hit was recorded. In the image on the left hand side are results for sub-map "E", and on the right for sub-map "F".

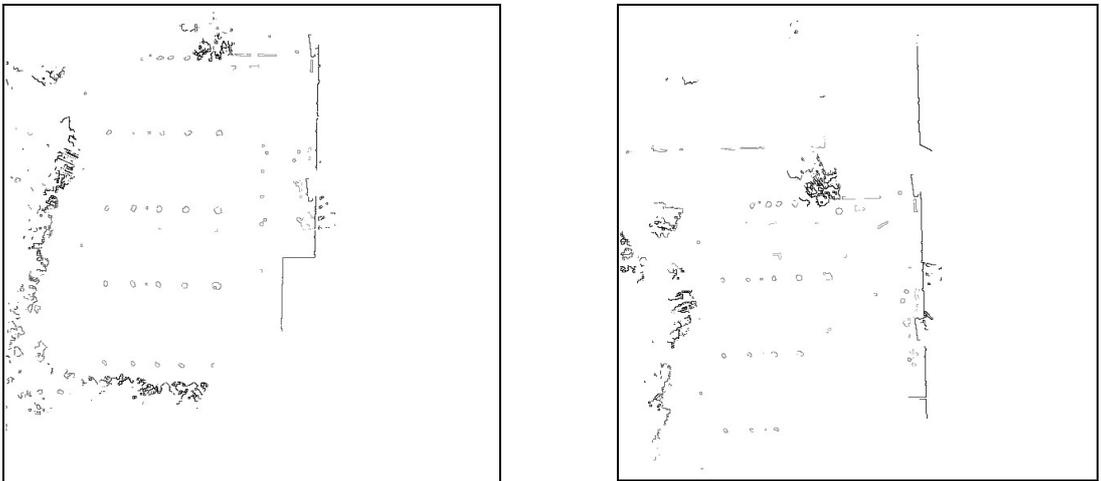


Figure B.18 A 2D projection of the free space to unknown/occupied space border computed from the 3D occupancy grid. The darker the shade of gray, the higher the highest location where a laser beam hit was recorded. In the image on the left hand side are results for sub-map "G", and on the right for sub-map "H".

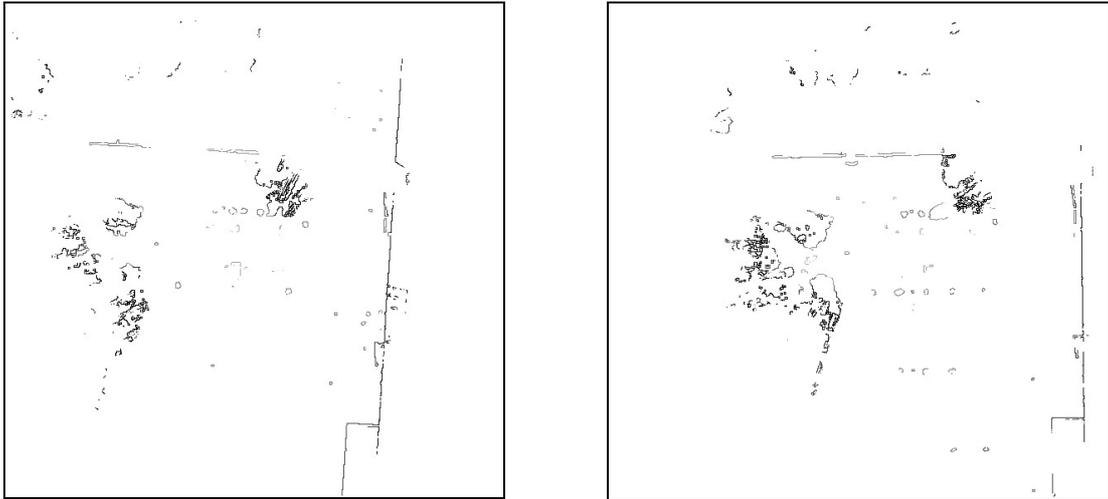


Figure B.19 A 2D projection of the free space to unknown/occupied space border computed from the 3D occupancy grid. The darker the shade of gray, the higher the highest location where a laser beam hit was recorded. In the image on the left hand side are results for sub-map "I", and on the right for sub-map "J".

In Figure C.1 the evolution of score distribution as a function of the search space cell size is depicted. The results have been stored during the computation of the relative alignment for sub-map registration pair "ST" in the forest mapping test case. The dimensions of the score grid are 10 meters square. Within the score distribution images, also the x- and y-coordinates of the highest score value and the center of gravity of the connected score cell cloud are marked with a simple and an ended cross hair respectively. The lengths of the vertical and horizontal lines of the ended cross hair correspond to the computed sample standard deviation of the score cloud. In Figure C.2, the evolution of score distribution as a function of the search space cell size is depicted for sub-map registration pair "RS". Note that only cylinder segment features were used for registration computation in the forest mapping test case.

For the generation of the score distribution images for the sub-map pair "ST" different *a priori* alignment coordinates than those given in Chapter 6.4, Table 6.6 were used to determine the center of the search space. The coordinates for the expected relative displacement along the x-, y- and z-coordinate axis directions were 4.0, 0.0 and 1.0 meters (instead of 3.0, -1.0 and 1.0, given in Table 6.6). For the heading direction of the sub-map frame "S" with respect to the reference frame of the sub-map "T" no change was predicted. We can now compare the computed x- and y-coordinates, given in Table 6.6, with the *a priori* values. The difference between the computed and the *a priori* values is 0.43 and -1.46 meters along the x- and y- axis of the sub-map frame "T", respectively. The result can be confirmed by considering the score distribution images in Figure C.1. In the images it is shown how the search for the most probable, relative alignment value converges to the middle-right part of the search space. This means that the computed alignment coordinate for the x-axis should be roughly equal to the predicted displacement whereas for the y-axis direction the difference should be about 1.5 meters in the negative direction. This is in accordance with the numeric values (0.43, -1.46).

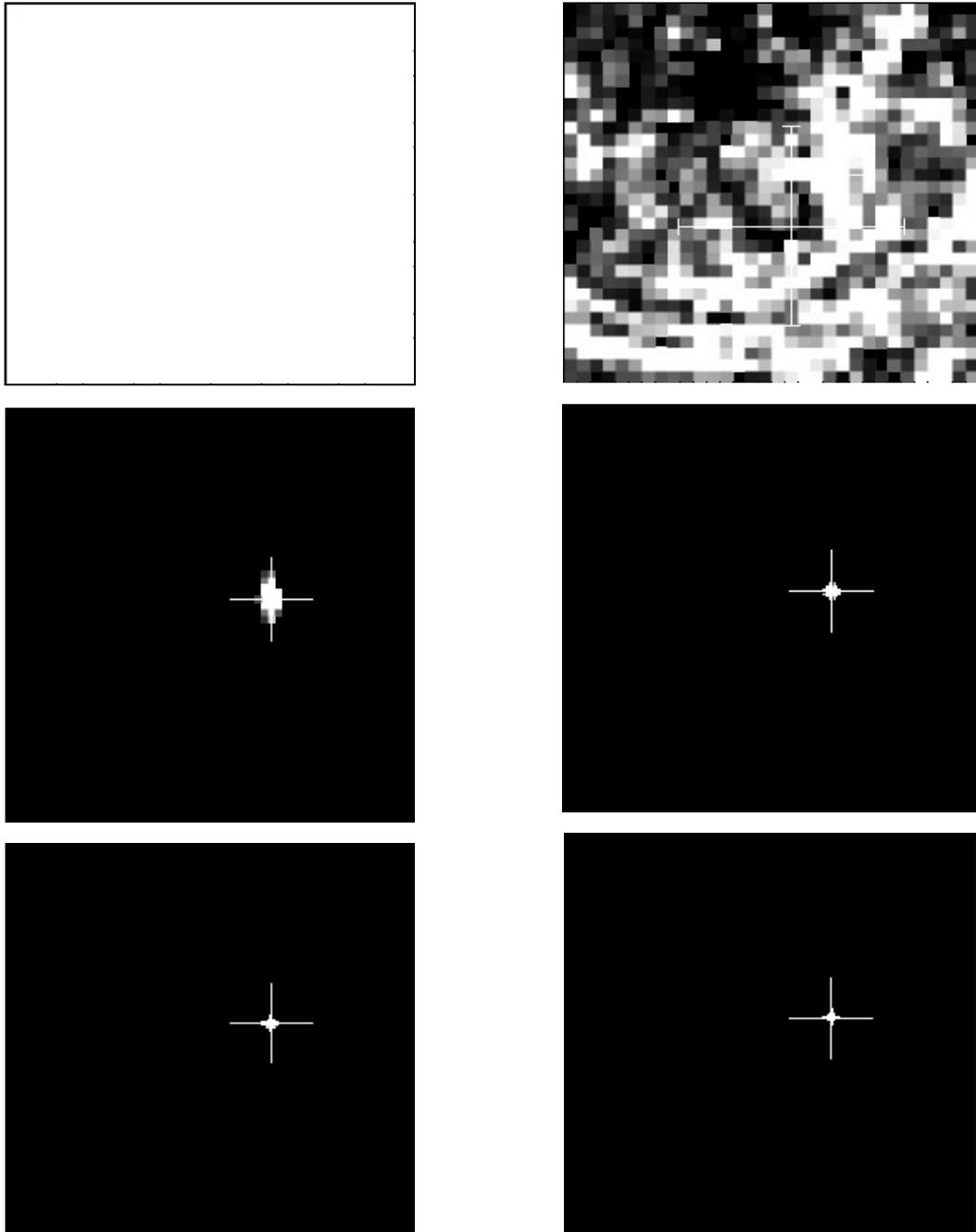


Figure C.1 Score distribution for search space cell sizes of 0.64, 0.32, 0.16, 0.08, 0.04 and 0.02 meters is depicted for registration pair "ST" in pictures starting from the upper left image up to the image on the right hand side of the bottom row. In the images, the distribution of the cumulative score values of the connected score cloud have been projected onto the x- and y-coordinate plane of the search space. Note that in the uppermost left image corresponding to the coarsest search level, the score distribution is almost even across the search space. This is probably due to the cluttered, outlier rich data sets corresponding to the sub-maps "S" and "T". The directions of the x- and y-coordinate axis are upwards and to the left respectively.

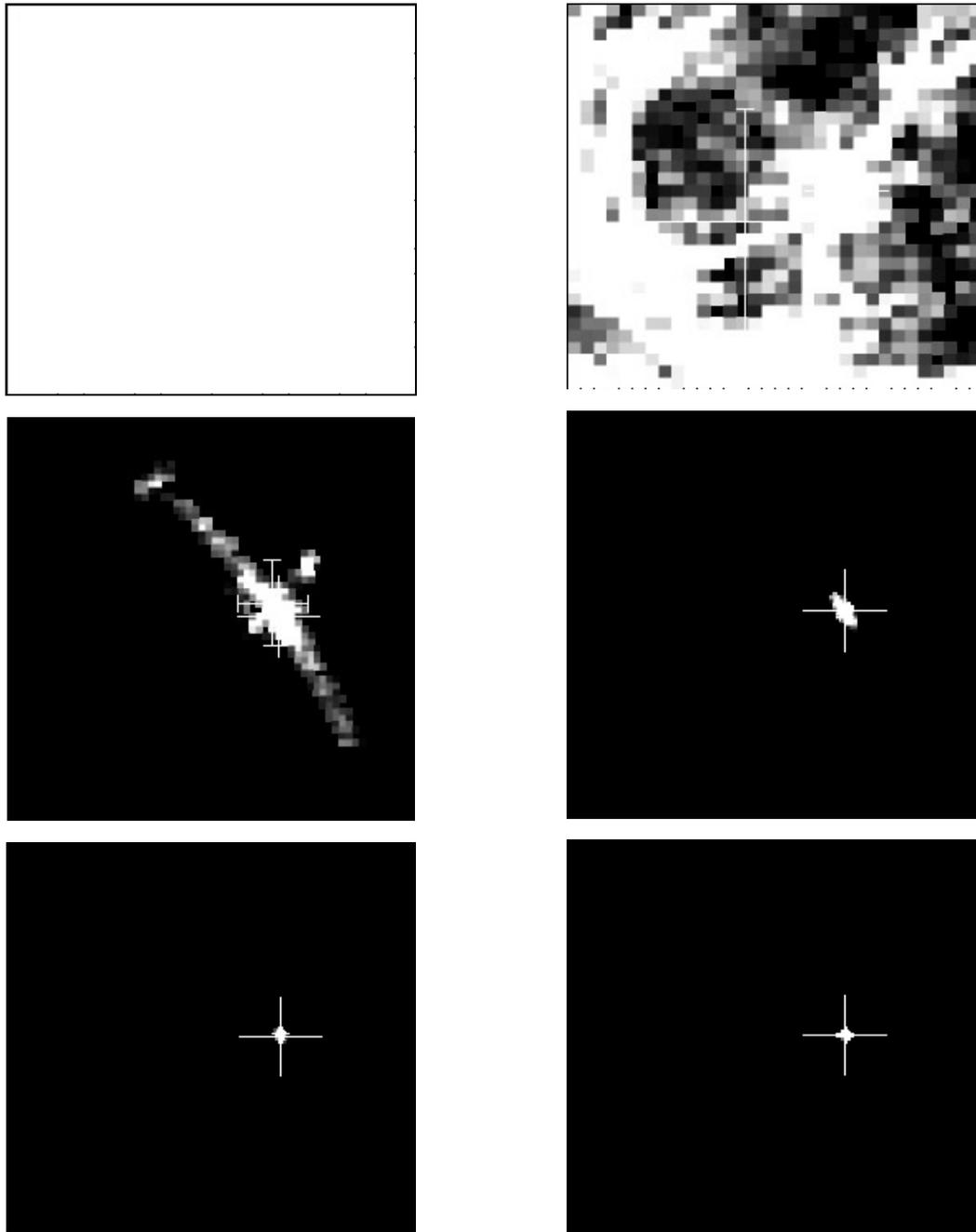


Figure C.2 Score distribution for search space cell sizes of 0.64, 0.32, 0.16, 0.08, 0.04 and 0.02 meters is depicted for the registration pair "RS" in pictures starting from the upper left image up to the image on the right hand side of the bottom row. In the images, the distribution of the cumulative score values of the connected score cloud have been projected onto the x- and y-coordinate plane of the search space. Note that in the uppermost left image corresponding to the coarsest search level, the score distribution is almost even across the search space. This is probably due to the cluttered, outlier rich data sets corresponding to the sub-maps "R" and "S". The directions of the x- and y-coordinate axis are upwards and to the left respectively.

In the following six figures, integrated model data for the parking lot area is illustrated. The models have been built by incrementally adding new sub-map data to the global model. Both a 3D view into the integrated feature model as well as the integrated elevation map are presented. The results are given with respect to three different viewpoint registration (i.e. alignment) results: *a priori* alignment, center of registration score cloud alignment and the alignment corresponding to the maximum score value when the final discretization threshold was reached. After presenting the integrated model images, the model data corresponding to each of the sub-maps, labeled E,F,G,H,I and J, are presented. By inspecting the illustrations of the integrated maps it can be concluded, that the best quality map is acquired, when the relative alignment value, corresponding to the highest score peak, is used to transport the sub-map model data into the common reference frame (Figures D5 and D6). This is mainly due to the fact that the search for the maximum score alignment was stopped when the dimensions of a cell in the discrete registration search space was 0.156 meters. At such a coarse search level a large number of cells "survive" yielding large and non-normal registration error distributions. In such a case, however, the maximum score cell is already located relatively close to the actual maximum score alignment value.

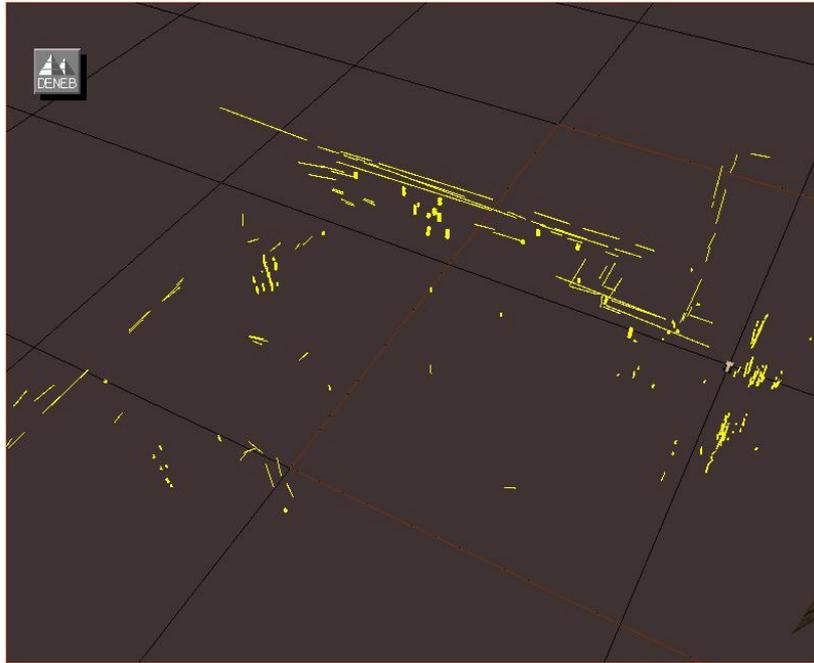


Figure D.1 An illustration of the incrementally constructed feature model of the parking lot area. Registration values corresponding to *a priori* alignment coordinates were used to compute the global locations of the sub-maps.

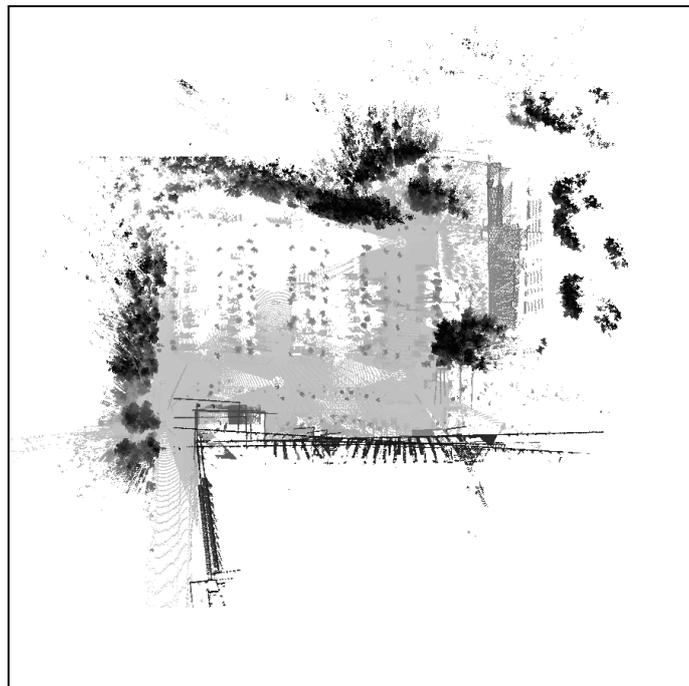


Figure D.2 An illustration of the incrementally constructed elevation map of the parking lot area. White corresponds to no measured elevation value, otherwise, the darker the color, the higher the underlying elevation. Registration values corresponding to *a priori* alignment coordinates were used to compute the global locations of the sub-maps. The size of the elevation grid map is 180 meters square, represented by 900x900 pixels.

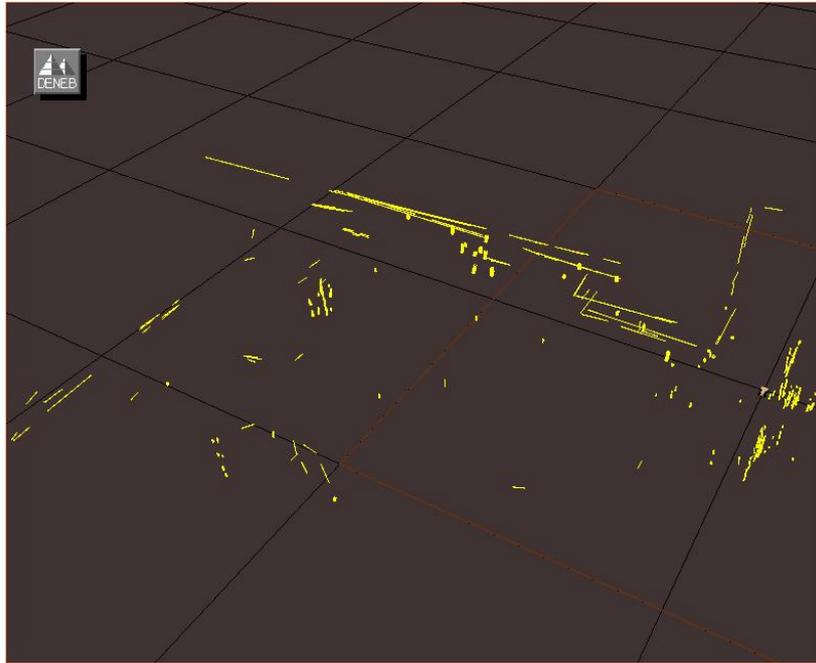


Figure D.3 An illustration of the incrementally constructed feature model of the parking lot area. Registration coordinates corresponding to the center of gravity of the connected registration score cells around the maximum score value were used to compute the global locations of the sub-maps. The size of the cell in the registration space was 0.156 meters, at the time the results were recorded.

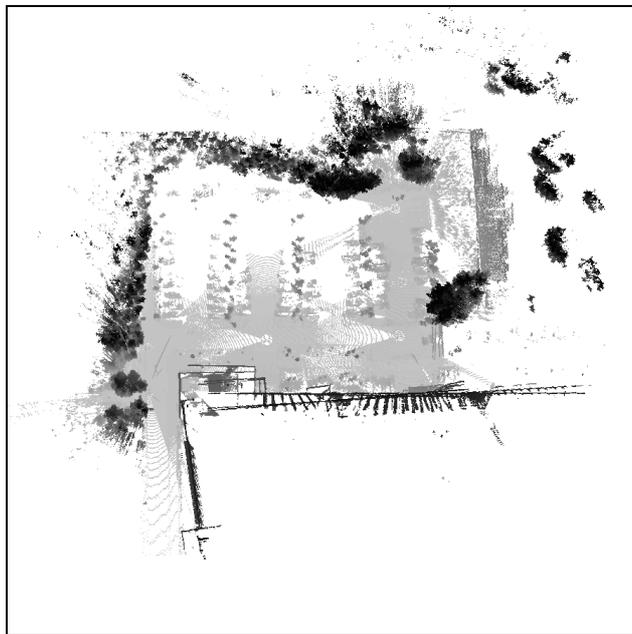


Figure D.4 An illustration of the incrementally constructed elevation map of the parking lot area. White corresponds to no measured elevation value, otherwise, the darker the color, the higher the underlying elevation. Registration coordinates corresponding to the center of gravity of the connected registration score cells around the maximum score value were used to compute the global locations of the sub-maps. The size of the cell in the registration space was 0.156 meters, at the time the results were recorded. The size of the elevation grid map is 180 meters square, represented by 900x900 pixels.

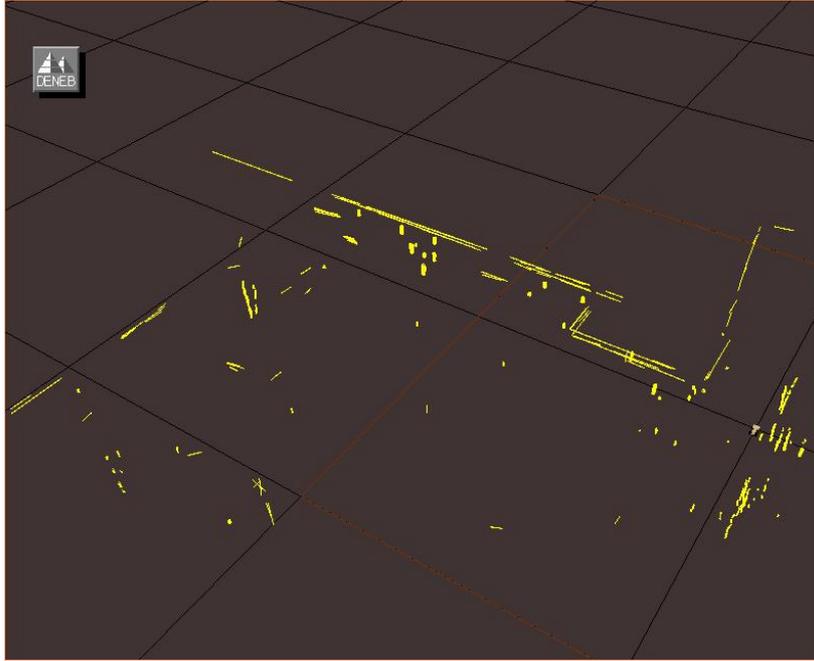


Figure D.5 An illustration of the incrementally constructed feature model of the parking lot area. Registration coordinates corresponding to the maximum score value were used to compute the global locations of the sub-maps. The size of the cell in the registration space was 0.156 meters, when the results were recorded.

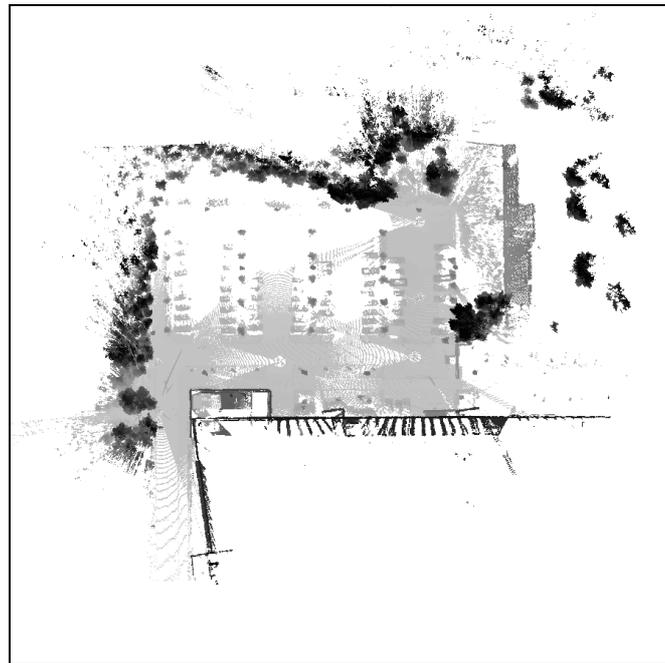


Figure D.6 An illustration of the incrementally constructed elevation map of the parking lot area. White color corresponds to no measured elevation value, otherwise the darker the color the higher the underlying elevation. Registration coordinates corresponding to the maximum score value were used to compute the global locations of the sub-maps. The size of the cell in the registration space was 0.156 meters, at the time the results were recorded. The size of the elevation grid map is 180 meters square, represented by 900x900 pixels.

The local feature maps, corresponding to single range images, are presented in the following six figures. The corresponding range images can be seen in Appendix A. The model of a mobile robot is shown for reference at the location where the particular range image was measured. (To be precise, the back axle of the robot is located at the perception position.)

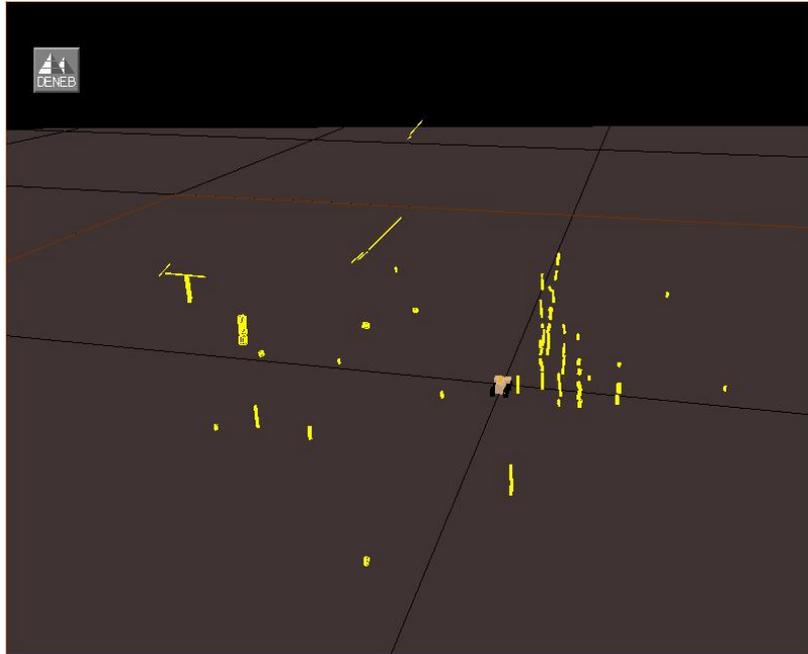


Figure D.7 An illustration of the feature model of the parking lot area corresponding to the range image labeled "scanE".

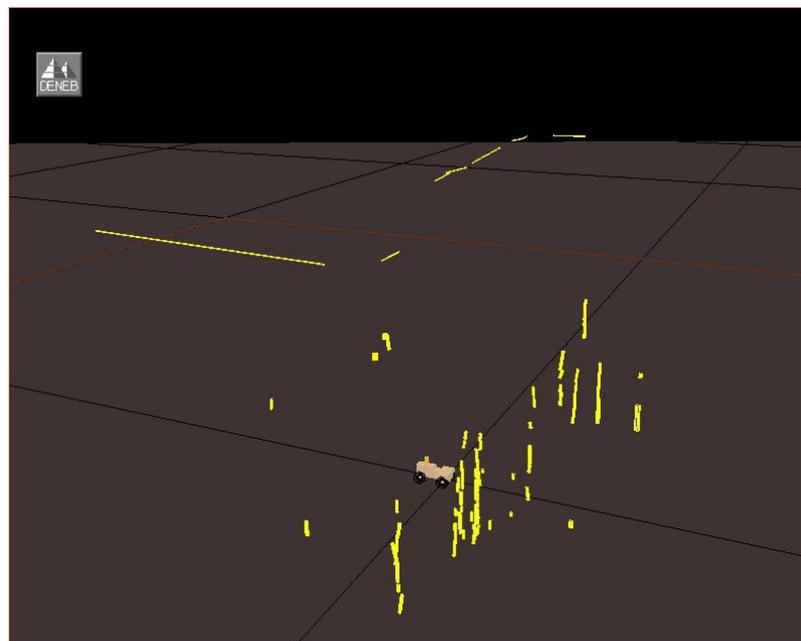


Figure D.8 An illustration of the feature model of the parking lot area corresponding to the range image labeled "scanF".

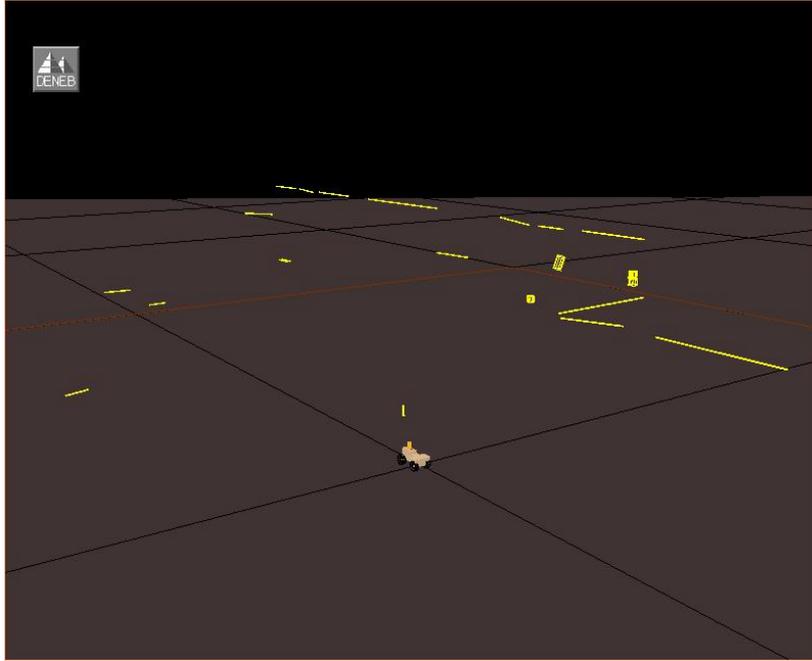


Figure D.9 An illustration of the feature model of the parking lot area corresponding to the range image labeled "scanG".

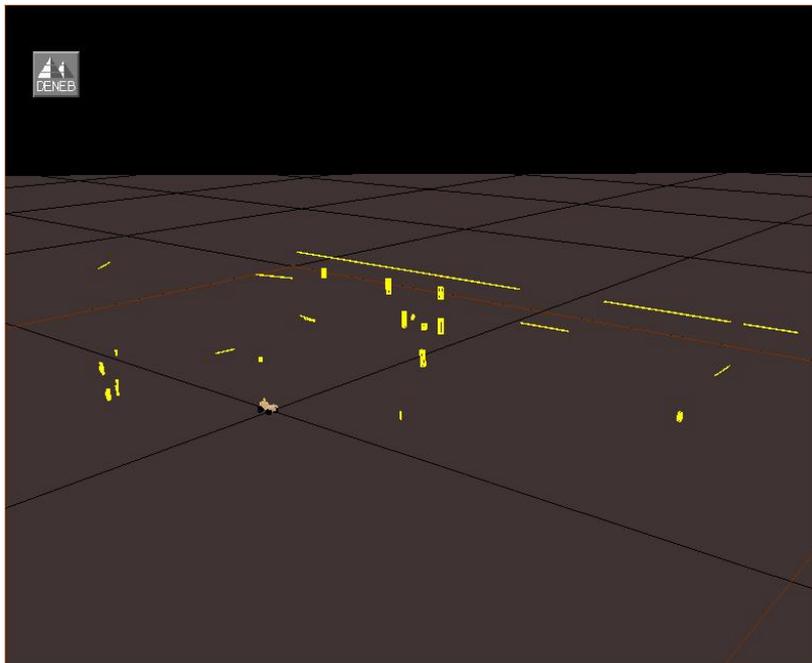


Figure D.10 An illustration of the feature model of the parking lot area corresponding to the range image labeled "scanH".

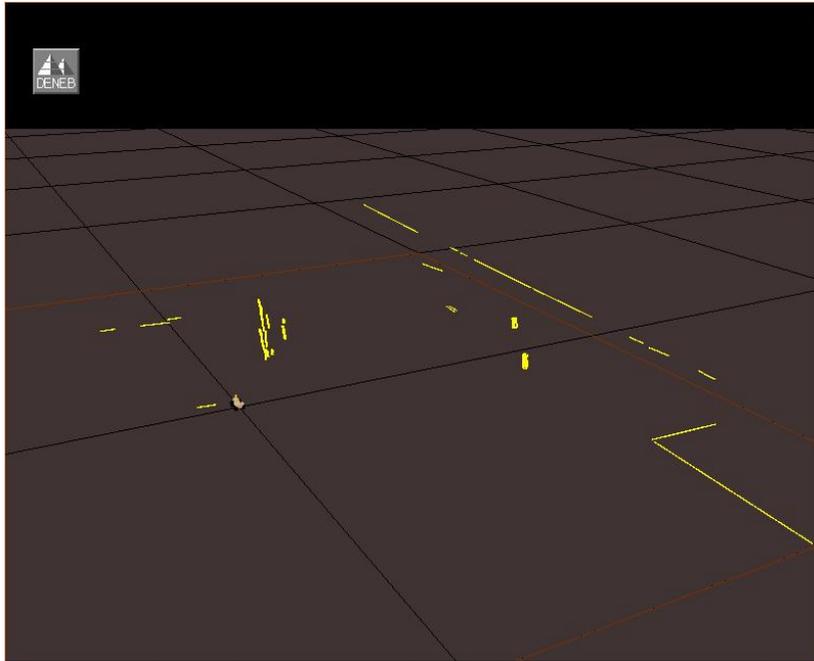


Figure D.11 An illustration of the feature model of the parking lot area corresponding to the range image labeled "scanI".

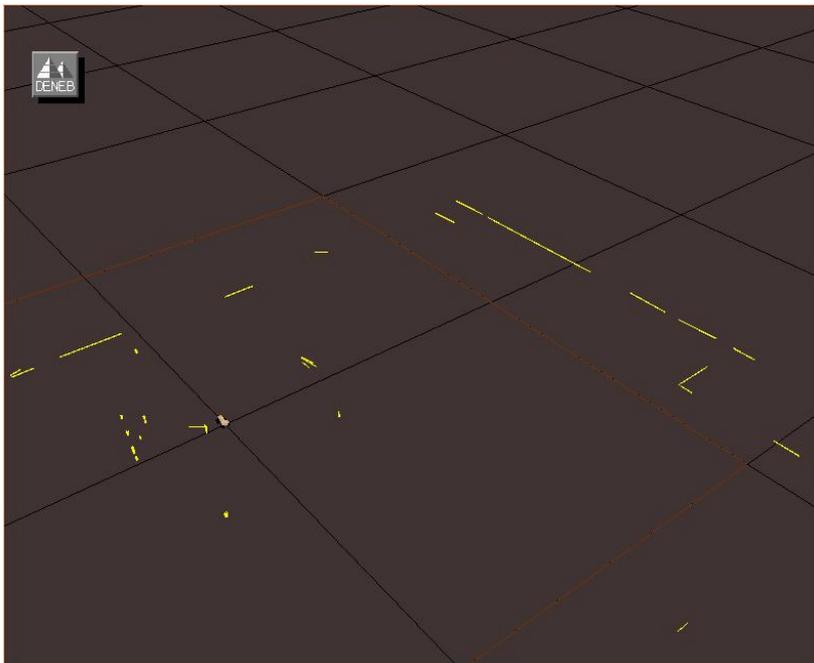


Figure D.12 An illustration of the feature model of the parking lot area corresponding to the range image labeled "scanJ".

The local elevation maps corresponding to single range images are presented in the following six figures. The size of the elevation grid maps is 120 meters square, represented by 600x600 pixels.

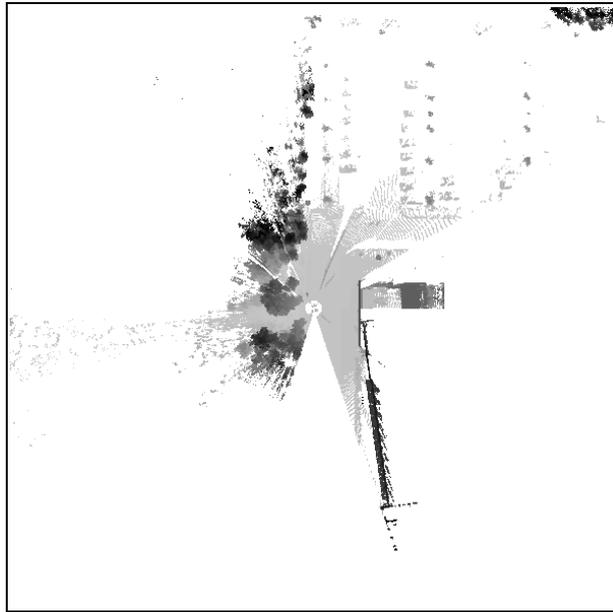


Figure D.13 An illustration of the elevation map of the parking lot area corresponding to the range image "scanE". White corresponds to no measured elevation value, otherwise, the darker the color, the higher the underlying elevation.

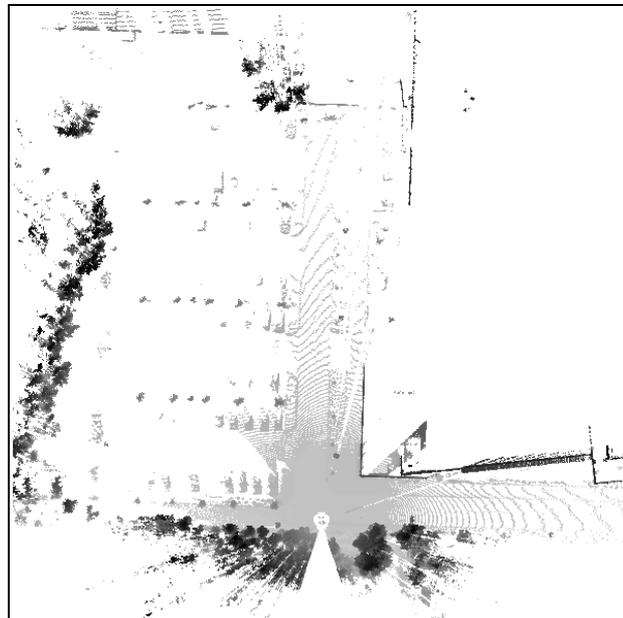


Figure D.14 An illustration of the elevation map of the parking lot area corresponding to the range image "scanF". White corresponds to no measured elevation value, otherwise, the darker the color, the higher the underlying elevation.

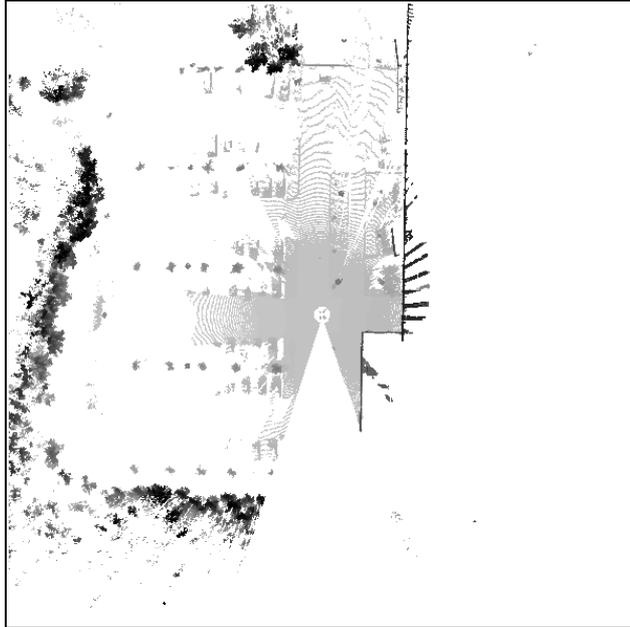


Figure D.15 An illustration of the elevation map of the parking lot area corresponding to the range image "scanG". White corresponds to no measured elevation value, otherwise, the darker the color, the higher the underlying elevation.

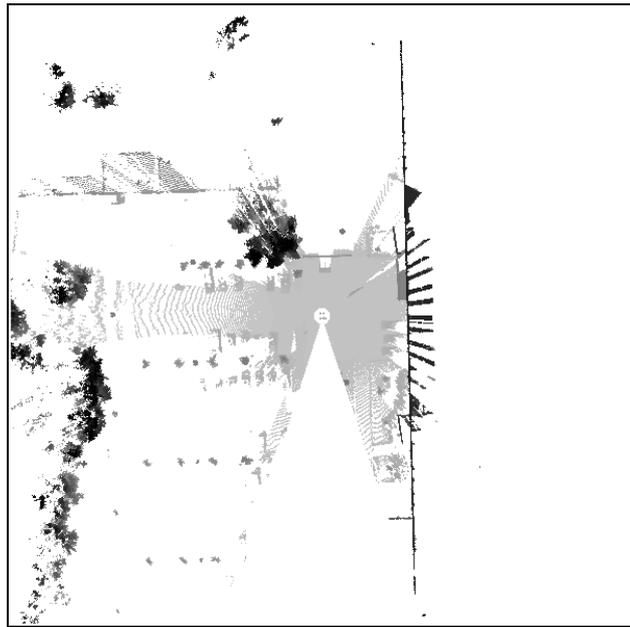


Figure D.16 An illustration of the elevation map of the parking lot area corresponding to the range image "scanH". White corresponds to no measured elevation value, otherwise, the darker the color, the higher the underlying elevation.

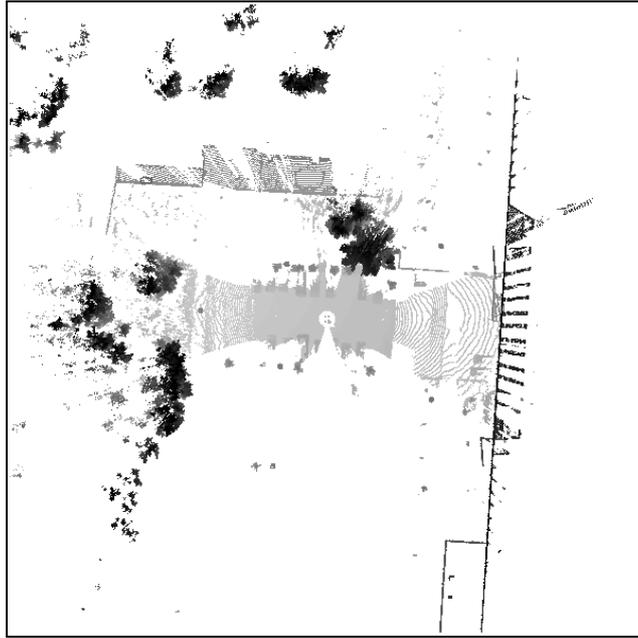


Figure D.17 An illustration of the elevation map of the parking lot area corresponding to the range image "scanI". White corresponds to no measured elevation value, otherwise, the darker the color, the higher the underlying elevation.

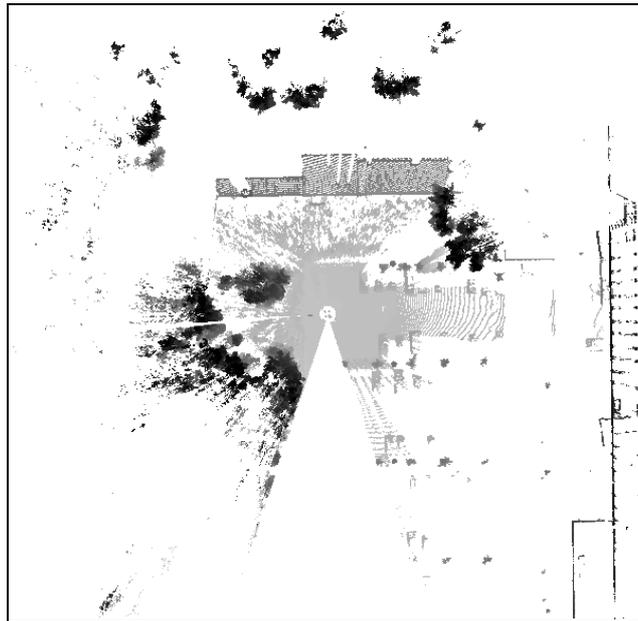


Figure D.18 An illustration of the elevation map of the parking lot area corresponding to the range image "scanJ". White corresponds to no measured elevation value, otherwise, the darker the color, the higher the underlying elevation.

For the “indoor” map building experiment, the feature sub-maps labeled “1”, “2”, “3” and “4” are illustrated in Figures D.19-D.22. These feature sets were involved in the computation of the evolution of the registration score distribution in Chapter 5.4.1. The model of a mobile robot is shown for reference at the location where the particular range image was measured. (To be precise, the back axle of the robot is located at the perception position.)

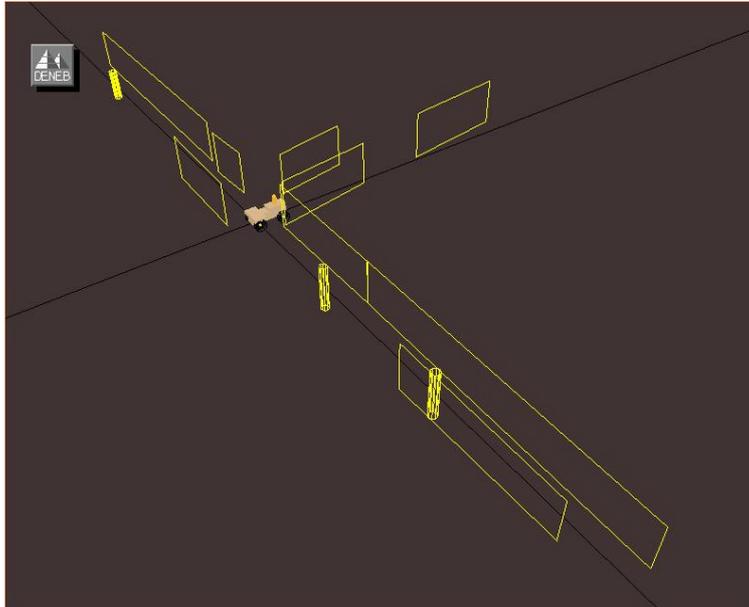


Figure D.19 An illustration of the feature model of the indoor environment corresponding to the range image labeled “scan1”.

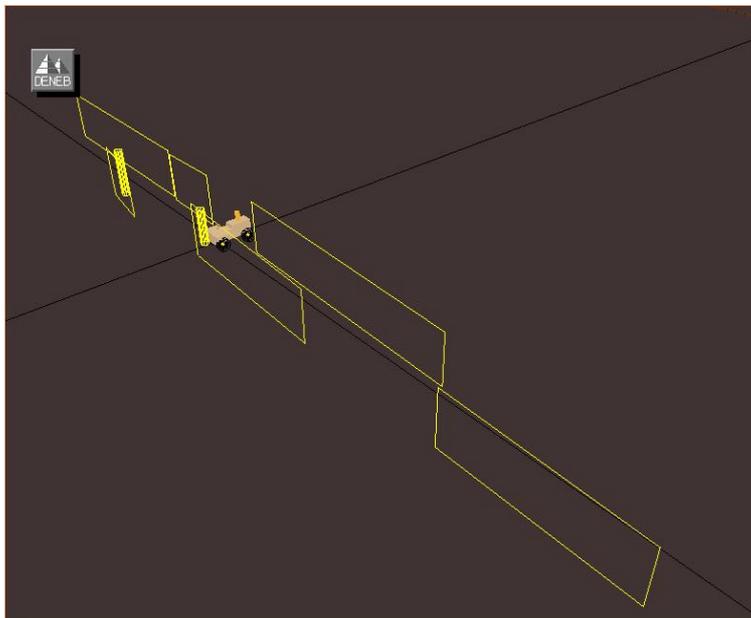


Figure D.20 An illustration of the feature model of the indoor environment corresponding to the range image labeled “scan2”.

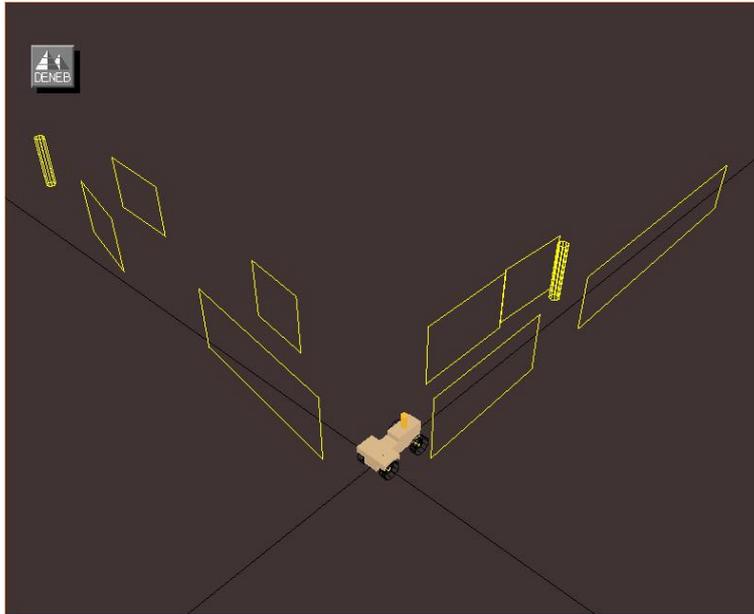


Figure D.21 An illustration of the feature model of the indoor environment corresponding to the range image labeled "scan3".

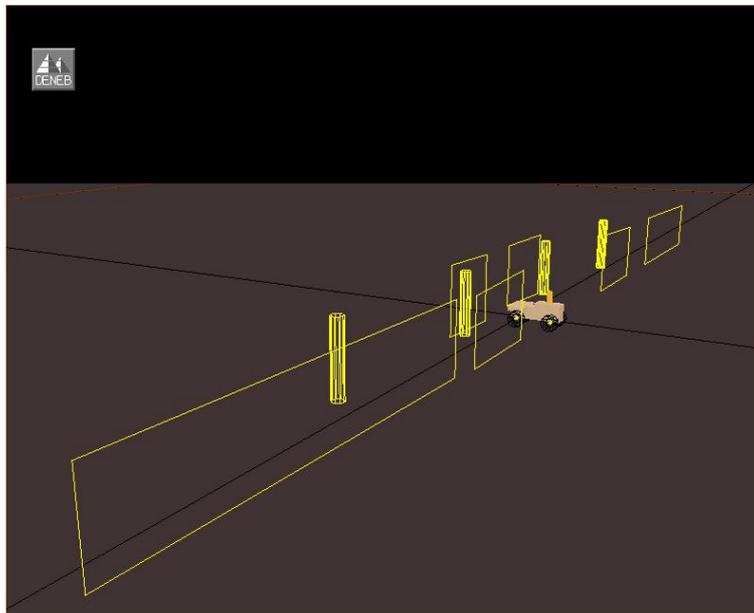


Figure D.22 An illustration of the feature model of the indoor environment corresponding to the range image labeled "scan4".

The evolution of the combined map of the forest environment, consisting of the ground elevation information and the cylinder approximations of the visible section of tree trunks, is depicted in Figures D.23 to D.28. The images correspond to the integrated, optimal map constructed from the sub-map sets "T", "T,S", "T,S,R", "T,S,R,Q", "T,S,R,Q,O" and "T,S,R,Q,O,N" respectively. In the images, the size of a grid cell in the elevation map is 0.6 meters. (Note also, that each grid cell consists of two triangles.) The location parameters of the viewpoint to the virtual model, with respect to the global reference frame, are 17.24, 9.44 and 5.61 meters for the x-, y- and z-coordinates, respectively, and -140.505 and -11.667 degrees for the pan and tilt angles, respectively. The field of view of the virtual camera is 61 degrees. The position, which also corresponds to the origin of the integrated environment model, is indicated by the coordinate frame in the figure. The CAD model of a mobile robot (not used in the experiments) is also drawn at the same location.

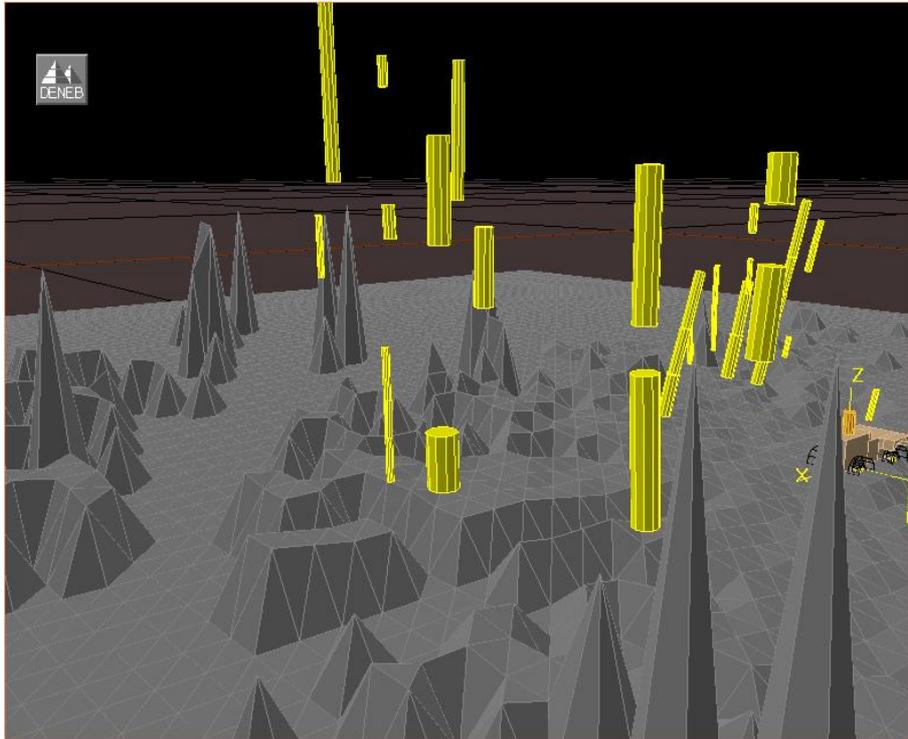


Figure D.23 A view into the 3D representation of the environment model for the forest test site. The model corresponds to the data acquired from the origin of the measurement position "T". The peaks in the elevation map, due to multiple hits to the trees at the corresponding location, can also be seen in the figure.

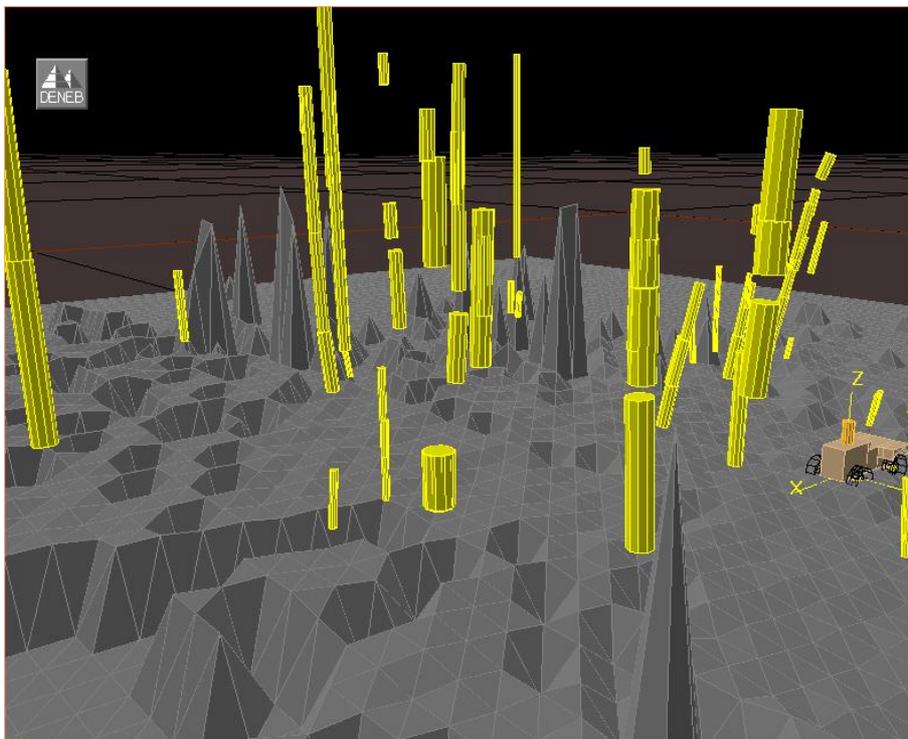


Figure D.24 A view into the 3D representation of the environment model for the forest test site. The model has been constructed from the sub-maps "T" and "S".

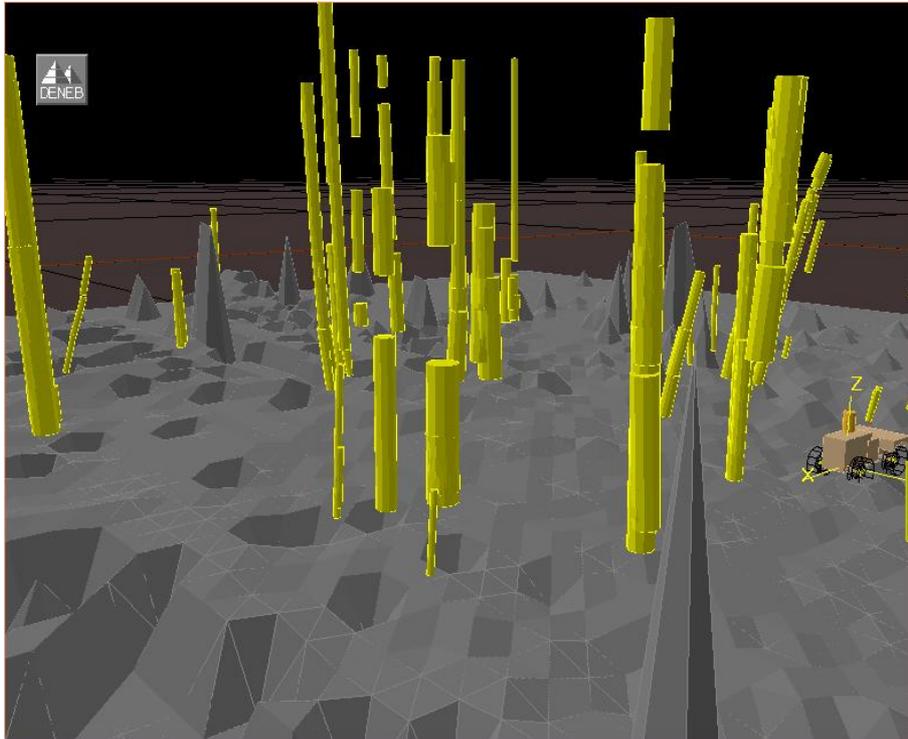


Figure D.25 A view into the 3D representation of the environment model for the forest test site. The model has been constructed from the sub-maps "T", "S" and "R".

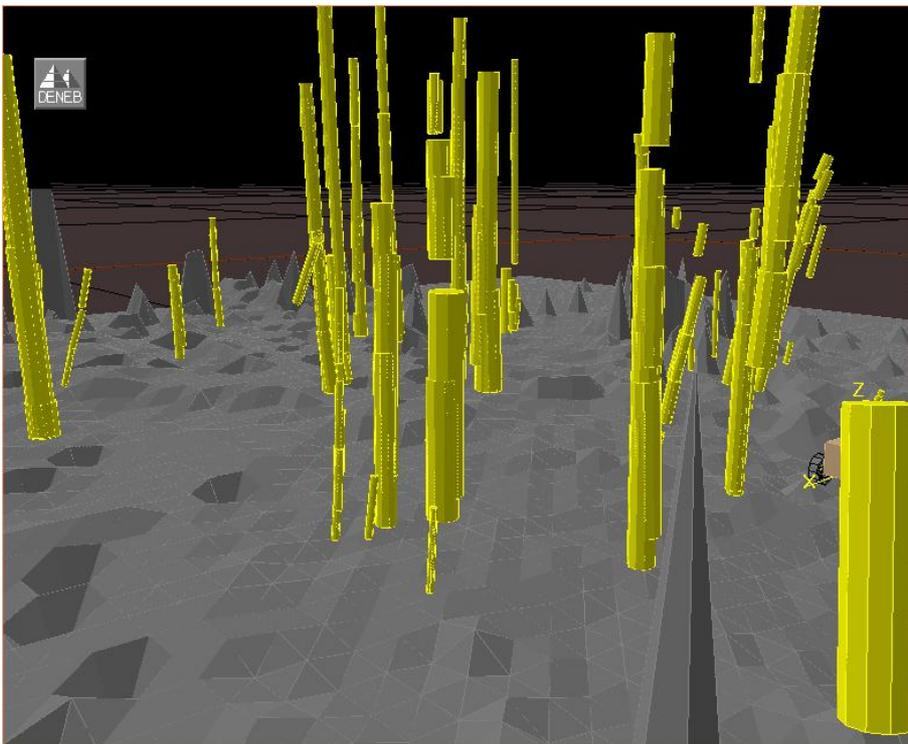


Figure D.26 A view into the 3D representation of the environment model for the forest test site. The model has been constructed from the sub-maps "T", "S", "R" and "Q".

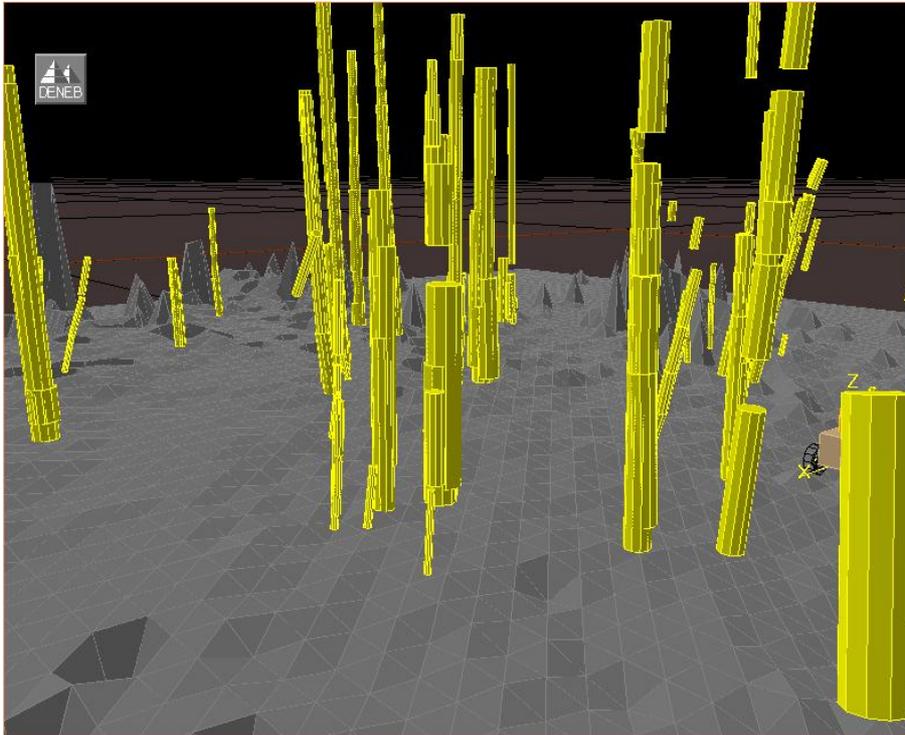


Figure D.27 A view into the 3D representation of the environment model for the forest test site. The model has been constructed from the sub-maps "T", "S", "R", "Q" and "O".

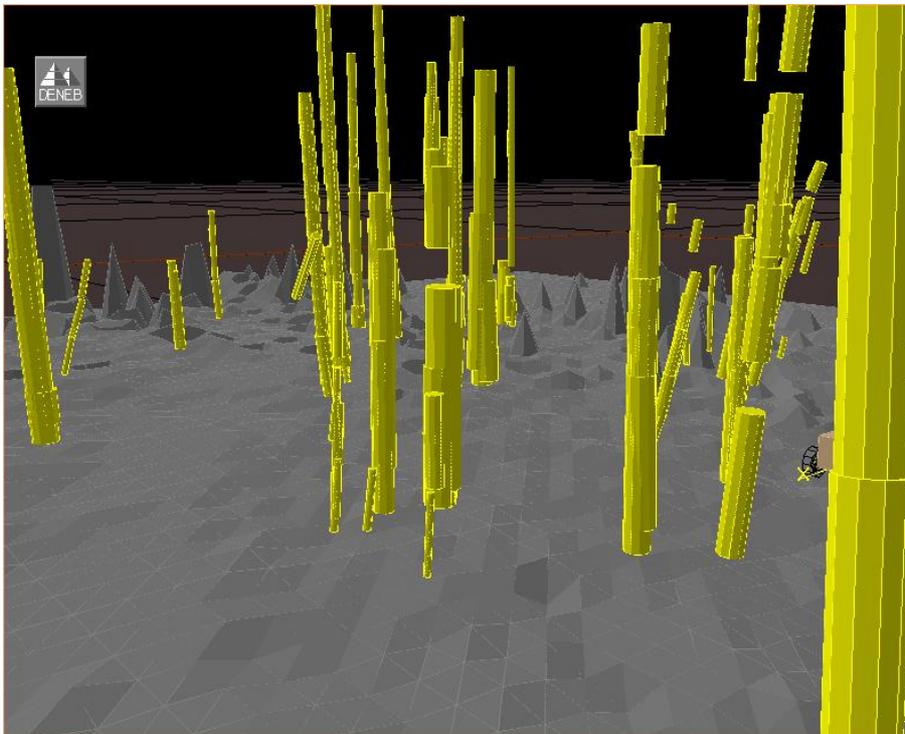


Figure D.28 A view into the 3D representation of the environment model for the forest test site. The model has been constructed from the sub-maps "T", "S", "R", "Q", "O" and "N".

The Kalman filter is the basic method for maintaining and updating dynamic, stochastic systems with linear state prediction and observation models. Now, let us assume that the system dynamics can be modeled with the following linear equation, which transforms the state from the (time) instant k to $k+1$,

$$\mathbf{x}(k+1) = \mathbf{F}(k)\mathbf{x}(k) + \mathbf{G}(k)\mathbf{u}(k) + \mathbf{v}(k) \quad (\text{E.1})$$

where the control input $\mathbf{u}(k)$ and the matrices $\mathbf{F}(k)$ and $\mathbf{G}(k)$ are assumed to be known [Maybeck, 1979] and [Bar-Shalom and Fortmann, 1988]. The process noise is modeled as a zero-mean, white, random sequence with a known covariance matrix, $\mathbf{Q}(k)$. Similarly, assume that the measurements of the state can be modeled as a linear function of the state as described by the measurement equation,

$$\mathbf{z}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{w}(k) \quad (\text{E.2})$$

where $\mathbf{w}(k)$ is again assumed to be a zero-mean, independent, random noise sequence with a known covariance matrix, $\mathbf{R}(k)$. An estimate of the state vector $\mathbf{x}(k)$ given the measurement data up to the time index k can be expressed as

$$\hat{\mathbf{x}}(k|k) \triangleq E\{\mathbf{x}(k)|Z^k\} \quad (\text{E.3})$$

which is the conditional mean of the state given the measurements

$$Z^k \triangleq \{\mathbf{z}(j), j = 1, \dots, k\} \quad (\text{E.4})$$

The associated conditional state error covariance matrix is

$$\mathbf{P}(k|k) \triangleq E\{[\mathbf{x}(k) - \hat{\mathbf{x}}(k|k)][\mathbf{x}(k) - \hat{\mathbf{x}}(k|k)]^T | Z^k\} \quad (\text{E.5})$$

The recursive Kalman filter, which propagates the estimate $\hat{\mathbf{x}}(k|k)$ and the associated covariance matrix $\hat{\mathbf{P}}(k|k)$ to the corresponding variables at the next instant of time $\hat{\mathbf{x}}(k+1|k+1)$ and $\hat{\mathbf{P}}(k+1|k+1)$, is given by the following equations [Bar-Shalom and Fortmann, 1988]. The equation for the one step prediction of the state can be acquired by applying the operator of expectation to the linear model of system dynamics, Equation E.1, conditioned on Z^k , which gives

$$\hat{\mathbf{x}}(k+1|k) = \mathbf{F}(k)\hat{\mathbf{x}}(k|k) + \mathbf{G}(k)\mathbf{u}(k) \quad (\text{E.6})$$

The uncertainty in the state estimate after prediction is acquired by subtracting the state estimate after prediction (E.6) from the system dynamics equation (E.1)

$$\tilde{\mathbf{x}}(k+1|k) \triangleq \mathbf{x}(k+1) - \hat{\mathbf{x}}(k+1|k) = \mathbf{F}(k)\tilde{\mathbf{x}}(k|k) + \mathbf{v}(k) \quad (\text{E.7})$$

The covariance of the predicted estimate is given as the expected value of the second power of the prediction error conditioned on all the accumulated data up to the time index k ,

$$\mathbf{P}(k+1|k) \stackrel{\Delta}{=} E\{\tilde{\mathbf{x}}(k+1|k)\tilde{\mathbf{x}}(k+1|k)^T | Z^k\} = \mathbf{F}(k)\mathbf{P}(k|k)\mathbf{F}(k)^T + \mathbf{Q}(k) \quad (\text{E.8})$$

An estimate for the predicted measurement is computed by taking the expected value of the linear measurement equation (E.2) at time $k+1$, conditioned on the measurement data up to time index k ,

$$\hat{\mathbf{z}}(k+1|k) = \mathbf{H}(k+1)\hat{\mathbf{x}}(k+1|k) \quad (\text{E.9})$$

Subtracting the estimate of the predicted measurement, Equation E.9, from the true measurement value, Equation E.2, at discrete time $k+1$, conditioned on Z^k , yields the error in the predicted measurement,

$$\tilde{\mathbf{z}}(k+1|k) \stackrel{\Delta}{=} \mathbf{z}(k+1) - \hat{\mathbf{z}}(k+1|k) = \mathbf{H}(k+1)\tilde{\mathbf{x}}(k+1|k) + \mathbf{w}(k+1) \quad (\text{E.10})$$

The covariance of the predicted measurement can now be computed,

$$\begin{aligned} \mathbf{P}_{zz} &= \mathbf{S}(k+1) \stackrel{\Delta}{=} E\{\tilde{\mathbf{z}}(k+1|k)\tilde{\mathbf{z}}(k+1|k)^T | Z^k\} \\ &= \mathbf{H}(k+1)\mathbf{P}(k+1|k)\mathbf{H}(k+1)^T + \mathbf{R}(k+1) \end{aligned} \quad (\text{E.11})$$

The cross-covariance between the predicted state and the predicted measurement, Equations E.7 and E.10 respectively, at time $k+1$ given the measurements up to time k is,

$$\begin{aligned} \mathbf{P}_{xz} &= E\{\tilde{\mathbf{x}}(k+1|k)\tilde{\mathbf{z}}(k+1|k)^T | Z^k\} \\ &= E\{\tilde{\mathbf{x}}(k+1|k)[\mathbf{H}(k+1)\tilde{\mathbf{x}}(k+1|k) + \mathbf{w}(k+1)]^T | Z^k\} \\ &= \mathbf{P}(k+1|k)\mathbf{H}(k+1)^T \end{aligned} \quad (\text{E.12})$$

The filter gain for the measurement update equation can now be given as a function of the covariance matrix of the predicted measurement, Equation E.11, and the covariance matrix between the predicted state and the predicted measurement, Equation E.12,

$$\mathbf{W}(k+1) = \mathbf{P}_{xz}\mathbf{P}_{zz}^{-1} = \mathbf{P}(k+1|k)\mathbf{H}(k+1)^T\mathbf{S}^{-1}(k+1) \quad (\text{E.13})$$

The measurement update equations can now be written as,

$$\hat{\mathbf{x}}(k+1|k+1) = \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\mathbf{v}(k+1) \quad (\text{E.14})$$

where $\mathbf{v}(k+1)$, called the innovation, is the difference between the actual and the predicted measurement,

$$\mathbf{v}(k+1) \stackrel{\Delta}{=} \tilde{\mathbf{z}}(k+1|k) \stackrel{\Delta}{=} \mathbf{z}(k+1) - \hat{\mathbf{z}}(k+1|k) = \mathbf{z}(k+1) - \mathbf{H}(k+1)\hat{\mathbf{x}}(k+1|k) \quad (\text{E.15})$$

The measurement update equation above can be derived, for example, by applying the Principle of Orthogonality to the error in the state estimate [Bar-Shalom and Fortmann, 1988]. The covariance matrix for the state estimate conditioned on the data up to and including the discrete time $k+1$ can now be computed,

$$\begin{aligned} \mathbf{P}(k+1|k+1) &\stackrel{\Delta}{=} E\{\tilde{\mathbf{x}}(k+1|k+1)\tilde{\mathbf{x}}(k+1|k+1)^T | Z^{k+1}\} \\ &= \mathbf{P}(k+1|k) - \mathbf{P}(k+1|k)\mathbf{H}(k+1)^T \mathbf{S}^{-1}(k+1)\mathbf{H}(k+1)\mathbf{P}(k+1|k) \\ &= \mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}(k+1)\mathbf{W}(k+1)^T = \mathbf{P}_{xx} - \mathbf{P}_{xz}\mathbf{P}_{zz}^{-1}\mathbf{P}_{zx} \end{aligned} \quad (\text{E.16})$$

At every discrete instant of time, denoted by k , all the information acquired in the past is summarized by the sufficient statistic $\mathbf{x}(k|k)$ and its associated covariance matrix. This follows from the whiteness property of the process noise, which allows preservation of the state's Markov property.

The Kalman filter yields the best linear estimate given zero-mean, statistically independent noise sequences. The best linear estimator is that which minimizes the mean-square error. If the noises obey a Gaussian distribution, the estimate will also be optimal.

In what follows, the extended Kalman filter (EKF), for the estimation of non-linear stochastic systems, is described [Bar-Shalom and Fortmann, 1988]. (The Kalman filter for the estimation of linear stochastic systems was outlined in Appendix E.) The simultaneous localization and map building formulation, discussed in Chapter 3.2.1.1, is utilized as the case example.

In order to be able to apply the techniques of linear estimation for the non-linear motion model of the sensor platform, Equation 3.7, it will be expanded in the Taylor series around the approximate conditional mean given the measurements up to and including the (time) instant k , $\hat{\mathbf{x}}_r(k|k)$. Higher order terms than the first derivate are neglected,

$$\mathbf{x}_r(k+1) = \mathbf{f}(\hat{\mathbf{x}}_r(k|k), \mathbf{u}(k), 0, k) + \nabla \mathbf{f}_{\mathbf{x}_r} \tilde{\mathbf{x}}_r(k|k) + \nabla \mathbf{f}_v \mathbf{v}(k) \quad (\text{F.1})$$

where $\tilde{\mathbf{x}}_r(k|k)$ is the uncertainty in the state estimate before carrying out the displacement of the sensor and $\mathbf{v}(k)$ corresponds to the uncertainty in the process model itself. In the equation $\nabla \mathbf{f}_{\mathbf{x}_r}$ is the Jacobian of $\mathbf{f}(\cdot)$ with respect to \mathbf{x}_r and $\nabla \mathbf{f}_v$ is the Jacobian of $\mathbf{f}(\cdot)$ with respect to \mathbf{v} both evaluated at the current estimate of the state, $\hat{\mathbf{x}}_r(k|k)$. By taking the expected value of F.1, an equation for the estimate of the predicted state can be formed,

$$\begin{aligned} \hat{\mathbf{x}}_r(k+1|k) &\stackrel{\Delta}{=} E\{\mathbf{x}_r(k+1)|Z^k\} \\ &\approx E\{\mathbf{f}(\hat{\mathbf{x}}_r(k|k), \mathbf{u}(k), 0, k) + \nabla \mathbf{f}_{\mathbf{x}_r} \tilde{\mathbf{x}}_r(k|k) + \nabla \mathbf{f}_v \mathbf{v}(k)\} \end{aligned} \quad (\text{F.2})$$

As the uncertainties $\tilde{\mathbf{x}}_r(k|k)$ and $\mathbf{v}(k)$ are assumed to correspond to zero mean random variables and so their expected values are zeroes, Equation F.2 becomes

$$\hat{\mathbf{x}}_r(k+1|k) = \mathbf{f}(\hat{\mathbf{x}}_r(k|k), \mathbf{u}(k), 0, k) \quad (\text{F.3})$$

The uncertainty of the estimate after prediction is given by subtracting Equation F.3 from F.1 and, again, assuming that the expected values of the higher order terms of the Taylor expansion are negligible,

$$\begin{aligned} \tilde{\mathbf{x}}_r(k+1|k) &= \mathbf{x}_r(k+1) - \hat{\mathbf{x}}_r(k+1|k) \\ &\approx \nabla \mathbf{f}_{\mathbf{x}_r} \tilde{\mathbf{x}}_r(k|k) + \nabla \mathbf{f}_v \mathbf{v}(k) \end{aligned} \quad (\text{F.4})$$

And the covariance (or rather mean square error) associated with the predicted state is given by,

$$\begin{aligned} \mathbf{P}(k+1|k) &\stackrel{\Delta}{=} E\{\tilde{\mathbf{x}}_r(k+1|k)\tilde{\mathbf{x}}_r(k+1|k)^T|Z^k\} \\ &\approx \nabla \mathbf{f}_{\mathbf{x}_r} \mathbf{P}_{rr}(k|k) \nabla \mathbf{f}_{\mathbf{x}_r}^T + \nabla \mathbf{f}_v \mathbf{Q}(k) \nabla \mathbf{f}_v^T \end{aligned} \quad (\text{F.5})$$

where $\mathbf{P}(k|k)_{rr}$ is the covariance of the state estimate of the mobile sensor platform before it was moved to another location between the (time) instants k and $k+1$ and $\mathbf{Q}(k)$ is the known covariance of the uncertainty related to the state prediction (motion) model.

Similarly, the non-linear measurement Equation 3.8 can be expanded in the Taylor series around the state estimate of the stochastic map $\hat{\mathbf{x}}(k+1|k)$, composed of the predicted state estimate of the mobile (robot) platform, $\hat{\mathbf{x}}_r(k+1|k)$, and the state estimates of the static map objects, $\hat{\mathbf{x}}_1(k|k) \dots \hat{\mathbf{x}}_N(k|k)$,

$$\mathbf{z}(k+1) = \mathbf{h}(\hat{\mathbf{x}}(k+1|k), 0, k) + \nabla \mathbf{h}_x \tilde{\mathbf{x}}(k+1|k) + \nabla \mathbf{h}_w \mathbf{w}(k+1) + \text{higher order terms} \quad (\text{F.6})$$

where $\tilde{\mathbf{x}}(k+1|k)$ and $\mathbf{w}(k+1)$ are assumed to be small and zero mean random variables. If it is further assumed that the higher order terms in the Taylor series are negligible, an equation for the predicted measurement at the (time) instant $k+1$ can be given by,

$$\begin{aligned} \hat{\mathbf{z}}(k+1|k) &\approx E\{\mathbf{h}(\hat{\mathbf{x}}(k+1|k), 0, k) + \nabla \mathbf{h}_x \tilde{\mathbf{x}}(k+1|k) + \nabla \mathbf{h}_w \mathbf{w}(k+1)\} \\ &= \mathbf{h}(\hat{\mathbf{x}}(k+1|k), 0, k) \end{aligned} \quad (\text{F.7})$$

where $\nabla \mathbf{h}_x$ is the Jacobian of $\mathbf{h}(\cdot)$ with respect to \mathbf{x} and $\nabla \mathbf{h}_w$ is the Jacobian of $\mathbf{h}(\cdot)$ with respect to \mathbf{w} , both evaluated at the predicted estimate of the state, $\hat{\mathbf{x}}(k+1|k)$.

By subtracting F.7 from F.6, the uncertainty related to the predicted observation becomes,

$$\begin{aligned} \mathbf{v} &\stackrel{\Delta}{=} \tilde{\mathbf{z}}(k+1|k) = \mathbf{z}(k+1) - \hat{\mathbf{z}}(k+1|k) \\ &\approx \nabla \mathbf{h}_x \tilde{\mathbf{x}}(k+1|k) + \nabla \mathbf{h}_w \mathbf{w}(k+1) \end{aligned} \quad (\text{F.8})$$

where \mathbf{v} corresponds to the innovation, which is the difference between the actual and the predicted observations. The corresponding covariance matrix of the predicted estimate is given by,

$$\begin{aligned} \mathbf{P}_{zz} &\stackrel{\Delta}{=} E\{\tilde{\mathbf{z}}(k+1|k) \tilde{\mathbf{z}}(k+1|k)^T | Z^k\} \\ &\approx \nabla \mathbf{h}_x \mathbf{P}(k+1|k) \nabla \mathbf{h}_x^T + \nabla \mathbf{h}_w \mathbf{R}(k+1) \nabla \mathbf{h}_w^T \end{aligned} \quad (\text{F.9})$$

where $\mathbf{P}(k+1|k)$ is the covariance of the stochastic map state estimate before incorporating the new observation information and $\mathbf{R}(k)$ is the known covariance of the uncertainty related to the observation model. The cross-covariance between the predicted state and the predicted measurement at the (time) instant $k+1$ given the measurements up to the (time) instant k can be computed as:

$$\begin{aligned}
 \mathbf{P}_{\mathbf{z}\mathbf{z}} &= E\left\{\tilde{\mathbf{x}}(k+1|k)\tilde{\mathbf{z}}(k+1|k)^T | Z^k\right\} \\
 &= E\left\{\tilde{\mathbf{x}}(k+1|k)\left[\nabla\mathbf{h}_{\mathbf{x}}\tilde{\mathbf{x}}(k+1|k) + \nabla\mathbf{h}_{\mathbf{w}}\mathbf{w}(k+1)\right]^T | Z^k\right\} \\
 &= \mathbf{P}(k+1|k)\nabla\mathbf{h}_{\mathbf{x}}^T
 \end{aligned} \tag{F.10}$$

An update for the state estimate can now be computed as,

$$\hat{\mathbf{x}}(k+1|k+1) = \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\mathbf{v}(k+1) \tag{F.11}$$

where $\mathbf{v}(k+1)$ is given by Equation F.8, and $\mathbf{W}(k+1)$ is the filter gain,

$$\mathbf{W}(k+1) = \mathbf{P}_{\mathbf{z}\mathbf{z}}\mathbf{P}_{\mathbf{z}\mathbf{z}}^{-1} = \mathbf{P}(k+1|k)\nabla\mathbf{h}_{\mathbf{x}}^T \left(\nabla\mathbf{h}_{\mathbf{x}}\mathbf{P}(k+1|k)\nabla\mathbf{h}_{\mathbf{x}}^T + \nabla\mathbf{h}_{\mathbf{w}}\mathbf{R}(k+1)\nabla\mathbf{h}_{\mathbf{w}}^T\right)^{-1} \tag{F.12}$$

In order to justify the application of the first order extended Kalman filter (EKF) for the estimation of non-linear systems, it must be assumed that the errors due to the truncated second and higher order terms of the Taylor expansion, (Equation F.1), are negligible. If this is not the case, more terms in the Taylor series expansion can be retained. For example, by including the first and second order derivatives from the Taylor series, the Gaussian second order filter can be attained [Maybeck, 1982]. Applying the EKF to the estimation of non-linear systems requires the evaluation of the Jacobian matrices of the state prediction and observation models. For the Gaussian second order filter the Hessian matrices need to be evaluated in addition to the Jacobian matrices. In both cases, the implementation of the filters may be difficult in practice, especially if the system is composed of many states and is highly non-linear [Julier, 1997]. In Appendix G, an alternative filtering method for nonlinear stochastic systems, is discussed.

The difference between the *distribution approximation filter* algorithm and the extended Kalman filter is related to the propagation of the information about the random state variables through the non-linear state prediction and measurement equations. With the extended Kalman filter, discussed in Appendix F, the propagation of the state estimate and the associated covariance matrix are described by Equations F.3 and F.5. The computation of the predicted measurement and the associated covariance matrix are given by Equations F.7 and F.9 respectively. The application of these formulae requires the computation of Jacobians of the non-linear process model and observation model equations. The derivation and evaluation of the Jacobian matrices can be a difficult task in practice [Julier and Uhlmann, 1997b]. Moreover, in case of highly non-linear models $\mathbf{f}(\cdot)$ and/or $\mathbf{h}(\cdot)$, successful application of the EKF algorithm would require very small time steps between the linearizations increasing the computational load.

Indeed, the intuition behind the *distribution approximation filter* (DAF) is that "it is easier to approximate a (Gaussian) probability distribution than it is to approximate an arbitrary non-linear function or transformation" [Julier, 1997] and [Julier et al., 2000]. As an example, let us consider the propagation of the mean $\hat{\mathbf{x}}_r(k|k)$ and the covariance $\mathbf{P}_{rr}(k|k)$ of the state of the mobile robot through the non-linear process model $\mathbf{f}(\mathbf{x}_r(k), \mathbf{u}(k), \mathbf{v}(k), k)$. First an augmented state vector composed of the n -dimensional original state vector $\mathbf{x}_r(k)$ and the q -dimensional vector of the noise terms $\mathbf{v}(k)$ is formed,

$$\mathbf{x}^a(k) = \begin{bmatrix} \mathbf{x}_r(k) \\ \mathbf{v}(k) \end{bmatrix} \quad (\text{G.1})$$

The process model, Equation 3.7, can now be rewritten,

$$\mathbf{x}_r(k+1) = \mathbf{f}(\mathbf{x}^a(k), \mathbf{u}(k), k) \quad (\text{G.2})$$

The propagation of information is based on the computation of the so-called sigma points, which are drawn deterministically from,

$$\hat{\mathbf{x}}^a(k|k) = \begin{pmatrix} \hat{\mathbf{x}}_r(k|k) \\ \mathbf{0}_{q \times 1} \end{pmatrix} \quad (\text{G.3})$$

$$\mathbf{P}^a(k|k) = \begin{bmatrix} \mathbf{P}_{rr}(k|k) & \mathbf{P}_{x,v}(k|k) \\ \mathbf{P}_{x,v}(k|k) & \mathbf{Q}(k) \end{bmatrix} \quad (\text{G.4})$$

where the $q \times 1$ vector of zeroes is due to the assumption of zero mean process noise, the covariance matrix of which is $\mathbf{Q}(k)$. With the off-diagonal terms $\mathbf{P}_{x,v}(k|k)$, correlations between the state errors and process noises can be incorporated as required. To approximate the $n+q$ -dimensional random variable $\mathbf{x}^a(k)$ with mean $\hat{\mathbf{x}}^a(k|k)$ and covariance $\mathbf{P}^a(k|k)$ a number of $2(n+q)+1$ sigma point vectors are drawn by the following equations,

$$\begin{aligned}
\chi_0^a(k|k) &= \hat{\mathbf{x}}^a & W_0 &= \kappa / (n + q + \kappa) \\
\chi_i^a(k|k) &= \hat{\mathbf{x}}^a + \left(\sqrt{(n + q + \kappa) \mathbf{P}^a(k|k)} \right)_i & W_i &= 1 / (2(n + q + \kappa)) \\
\chi_{i+n}^a(k|k) &= \hat{\mathbf{x}}^a - \left(\sqrt{(n + q + \kappa) \mathbf{P}^a(k|k)} \right)_i & W_{i+n} &= 1 / (2(n + q + \kappa))
\end{aligned} \tag{G.5}$$

where $\kappa \in \mathfrak{R}$, $\left(\sqrt{(n + q + \kappa) \mathbf{P}^a(k|k)} \right)_i$ is the i^{th} row or column of the square root of the $(n+q) \times (n+q)$ matrix $(n + q + \kappa) \mathbf{P}^a(k|k)$ and W_i is the weight that is associated with the i^{th} sigma point. The transformation of the mean $\hat{\mathbf{x}}^a(k|k)$ and the associated covariance matrix $\mathbf{P}^a(k|k)$ through the non-linear transformation $\mathbf{f}(\cdot)$ is carried out by first instantiating each sigma point through the process model,

$$\chi_i^a(k+1|k) = \mathbf{f}(\chi_i^a(k|k), \mathbf{u}(k), k) \tag{G.6}$$

The mean of the predicted robot state can now be computed as a weighted sum of the transformed sigma points,

$$\hat{\mathbf{x}}^a(k+1|k) = \sum_{i=0}^{2(n+q)} W_i \chi_i^a(k+1|k) \tag{G.7}$$

And the associated covariance is computed as follows,

$$\mathbf{P}^a(k+1|k) = \sum_{i=0}^{2(n+q)} W_i [\chi_i^a(k+1|k) - \hat{\mathbf{x}}^a(k+1|k)] [\chi_i^a(k+1|k) - \hat{\mathbf{x}}^a(k+1|k)]^T \tag{G.8}$$

Note that from the transformed augmented state and the associated covariance matrices, the parts related to the actual system state $\mathbf{x}_r(k)$ are of interest. These parts correspond to the n uppermost terms in the augmented state vector $\hat{\mathbf{x}}^a(k+1|k)$ and to the $n \times n$ upper left corner sub-matrix of $\mathbf{P}^a(k+1|k)$.

Similarly, in order to compute the predicted observation, the transformed sigma points are instantiated through the observation model,

$$\mathbf{z}_i(k+1|k) = \mathbf{h}(\chi_i^a(k+1|k), k) \tag{G.9}$$

The mean of predicted observations can now be computed as the weighted sum of the transformed sigma points,

$$\hat{\mathbf{z}}(k+1|k) = \sum_{i=0}^{2(n+q)} W_i \mathbf{z}_i(k+1|k) \tag{G.10}$$

In the case of additive and independent observation noise with covariance $\mathbf{R}(k+1)$, the predicted innovation covariance can be computed as,

$$\mathbf{P}_{zz}(k+1|k) = \mathbf{R}(k+1) + \sum_{i=0}^{2(n+q)} W_i [\mathbf{Z}_i(k+1|k) - \hat{\mathbf{z}}(k+1|k)] [\mathbf{Z}_i(k+1|k) - \hat{\mathbf{z}}(k+1|k)]^T \quad (\text{G.11})$$

In Equation G.9 the observation noise is left outside function $\mathbf{h}(\cdot)$ due to its additive nature. However, if the observation is introduced in a non-linear fashion, or is correlated with uncertainty of the system state, then the augmented state vector, G.1, and the associated covariance matrix, G.4, can be further expanded to include the observation noise terms and the noise covariance and correlation terms respectively. And finally, the cross correlation matrix between the uncertainties of the predicted state and the predicted measurement is given by,

$$\mathbf{P}_{xz}(k+1|k) = \sum_{i=0}^{2(n+q)} W_i [\boldsymbol{\chi}_i^a(k+1|k) - \hat{\mathbf{x}}^a(k+1|k)] [\mathbf{Z}_i(k+1|k) - \hat{\mathbf{z}}(k+1|k)]^T \quad (\text{G.12})$$

To update the predicted state with the observation the same update formulae as used with the extended Kalman filter, given by Equations F.11 and F.12, can now be exploited.

In order to guarantee that the uncertainty of the estimates propagated through the non-linear transformations will be positive semi-definite, the algorithm can be modified such that the covariances are evaluated about $\boldsymbol{\chi}_0^a(k|k)$ and $\mathbf{Z}_0(k+1|k)$ instead of $\hat{\mathbf{x}}^a(k+1|k)$ and $\hat{\mathbf{z}}(k+1|k)$. With this modification, the prediction is the same as that of the truncated second order filter [Julier et al., 2000]. However, to obtain a significant advance over the Kalman filter, it is necessary to work with non-Gaussian distributions [Julier, 1997]. Methods that employ a numerical approximation of the probability distributions are a possible alternative. These methods are discussed in Chapter 3.3.

HELSINKI UNIVERSITY OF TECHNOLOGY AUTOMATION TECHNOLOGY LABORATORY RESEARCH REPORTS

- No. 10 Zhang, X., Halme, A.,
A summary of the study of bioelectrochemical fuel cell by using *Saccharomyces cerevisiae*, January 1994.
- No. 11 Yang, H.,
Using landmarks for the vehicle location measurement, February 1994.
- No. 12 Halme, A., Zhang, X.,
Experimental study of bioelectrochemical fuel cell using bacteria from Baltic sea, February 1995.
- No. 13 Zhang, X.,
Aspects of modelling and control of bioprocesses: Application of conventional approach, and functional state concept, October 1995.
- No. 14 Yang, H.,
Vision methods for outdoor mobile robot navigation, November 1995.
- No. 15 Wang, Y.,
Spherical rolling robot, February 1996.
- No. 16 Hartikainen, K.,
Motion planning of a walking platforms designed to locomote on natural terrain, November 1996.
- No. 17 Zhang, X., Halme, A.,
Effect of size and structure of a bacteria fuel cell on the electricity production and energy conversion rate, March 1997.
- No. 18 Visala, A.,
Modeling of nonlinear processes using Wiener-NN representation and multiple models, November 1997.
- No. 19 Xu, B.,
An interactive method for robot control and its application to deburring, November 1998.
- No. 20 Zhang, X., Halme A.,
A biofilm reactor for a bacteria fuel cell system, August 1999.
- No. 21 Vainio M.,
Intelligence through interactions – Underwater robot society for distributed operations in closed aquatic environment, October 1999.
- No. 22 Appelqvist P.,
Mechatronics design of a robot society – A case study of minimalist underwater robots for distributed perception and task execution, November 2000.

