GROWING 3D MAPS FROM 2D SLAM MAPS OBTAINED BY INDOOR SLAM WITH HETEROGENEOUS MOBILE ROBOTS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

BURAK ÇETİNKAYA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

DECEMBER 2015

Approval of the thesis:

GROWING 3D MAPS FROM 2D SLAM MAPS OBTAINED BY INDOOR SLAM WITH HETEROGENEOUS MOBILE ROBOTS

submitted by **BURAK ÇETİNKAYA** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. M. Gülbin Dural Ünver _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Gönül Turhan Sayan _____
Head of Department, **Electrical and Electronics Engineering**

Prof. Dr. Aydan M. Erkmen _____
Supervisor, **Electrical and Electronics Engineering Dept, METU**

Examining Committee Members:

Assoc. Prof. Dr. Afşar Saranlı _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Aydan M. Erkmen _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Aydın Alatan _____
Electrical and Electronics Engineering Dept., METU

Assoc. Prof. Dr. İlkay Ulusoy _____
Electrical and Electronics Engineering Dept., METU

Assoc. Prof. Dr. Sedat Nazlıbilek _____
Mechatronics Engineering Dept, Atilim University

Date: 11.12.2015

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: BURAK ÇETİNKAYA

Signature:

ABSTRACT

GROWING 3D MAPS FROM 2D SLAM MAPS OBTAINED BY INDOOR SLAM WITH HETEROGENEOUS MOBILE ROBOTS

Çetinkaya, Burak

M.S., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Aydan M. Erkmen

December 2015, 271 pages

This thesis focuses on growing 3D structure of an indoor environment from 2D maps constructed by Simultaneous Localization and Mapping (SLAM) algorithm at different height levels explored by heterogeneous vehicles. Vehicles differing mainly in their dynamics explore an indoor environment area using laser scanners at different height levels and creates 2D SLAM maps for each height level based on FastSLAM algorithm. This algorithm utilizes feature based scan matching and natural landmarks. These obtained 2D SLAM maps are then combined towards growing a 3D map in which three consecutive steps are followed namely; creating correspondences on the generated maps for correct alignment of the maps, generating correspondence points to use while connecting maps to each other and tiling the points which is the extrapolation method used between maps. Correspondences are to be established in the most adequate manner between the maps using topological adjacency relations via extracted key features from the maps so that these features can be matched and proper alignment between the maps can be established. Subsequently, regions that show similarities and dissimilarities between the maps are

obtained based on the affinity of the grids belong to corresponding regions. Although, similar regions from different height levels are easy to grow from 2D to 3D by connecting the boundary regions as an extrapolation approach, for dissimilar regions, it is hard to find correspondence points to be used efficiently in an extrapolation. The novel approach presented in this thesis focuses on the generation of correspondence points and their extrapolation between SLAM maps. Tree structures are established inside the dissimilar regions and connected vertices on these tree structures are utilized as correspondence points. Extrapolation is generated by tiling that involves triangular meshing between generated correspondences between maps which eventually become surfaces of the 3D structures completed 2D maps. The thesis concludes with an intensive discussion of actual implementation on hardware followed by a full sensitivity analysis of the 3D map growing system proposed.

Keywords: Simultaneous Localization and Mapping, Scan Matching, Map Growing, Key Point Detection, Rapidly Exploring Random Trees, Surface Reconstruction, Triangulation, Correspondence Points Generation, Tiling

ÖZ

FARKLI ROBOTLAR İLE YAPILAN ANLIK KONUMLAMA VE HARİTALAMA SONUCUNDA ORTAYA ÇIKAN 2 BOYUTLU HARİTALAR İLE 3 BOYUTLU HARİTALARIN OLUŞTURULMASI

Çetinkaya, Burak

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği

Tez Yöneticisi: Prof. Dr. Aydan M. Erkmen

Aralık 2015, 271 sayfa

Bu tez, farklı araçlar ile değişen yükseklik seviyelerinde yapılan anlık konumlama ve haritalama (SLAM) sonucunda elde edilen 2 boyutlu haritalar yardımıyla çevrenin 3 boyutlu yapısının oluşturulması üzerine odaklanmıştır. Farklı dinamikteki araçlar içerisinde bulundukları kapalı alanı farklı yükseklik seviyelerinde gezerek üzerilerindeki lazer tarayıcılar ile çevrelerinden ölçüm alarak ve SLAM algoritmasını kullanarak 2B haritalar oluştururlar. Bahsedilen algoritma çevredeki doğal şekilleri belirleyerek bunları farklı taramaları eşleme amaçlı kullanmaktadır. Farklı yükseklik seviyelerinden elde edilen 2B haritalar 3B harita geliştirmek için birbirini izleyen üç adım ile bir araya getirilirler; doğru hizalama için uygun eşleştirmelerin yapılması, haritalar arasındaki bağlantıların kurulabilmesi için birleşim noktalarının üretimi ve birleşim noktalarının tanıtılan dış değerleme yöntemi ile birleştirilmesi. Doğru hizalama sonuçları alabilmek için uygun eşleştirmeler haritalardan elde edilen köşe ve doğru şekillerinin topolojik komşuluk özellikleri kullanılarak yapılmaktadır. Sonrasında, harita üzerindeki alanları oluşturan örgüler arasındaki yakınlık bilgisi

kullanılarak haritalar arasında benzer ve benzer olmayan alanlar belirlenmektedir. Sınır bölgelerinin kullanılmasıyla farklı yükseklik seviyelerindeki benzer alanların büyütülmesi kolay bir işlem olsa da, benzer olmayan alanlar için uygun birleşme noktalarının bulunması ve birleştirilmesi zor bir işlemdir. Tez içerisinde sunulan özgün yöntem SLAM ile elde edilen haritalar için uygun birleşim noktalarının bulunması ve dış değerleme yöntemi ile birleştirilmesi üzerine yoğunlaşmaktadır. Benzer olmayan alanlar içerisinde ağaç yapıları oluşturularak bu yapılar üzerindeki birleşim noktaları kullanılmaktadır. Birleşim noktaları arasındaki bağlantılar üçgensel ağların oluşturulmasıyla elde edilen dış değerlemeler ile sağlanmaktadır. Bu oluşturulan üçgensel ağlar çevrede bulunan cisimlerin 3B haritadaki yüzey alanlarını oluşturmaktadır. Tez, sunulan yaklaşımın donanım üzerinde gerçeklenmesi ile yapılan deneylerin detaylı anlatımı ve hassasiyet analizinin yapılmasıyla sonlanmaktadır.

Anahtar Kelimeler: Anlık Konumlama ve Haritalama, Tarama Eşleme, Harita Geliştirme, Kritik Nokta Tespiti, Hızla Gelişen Rastlantısal Ağaçlar, Yüzeylerin Yeniden Yapılandırılması, Üçgenleştirme, Karşılık Noktalarının Tespiti, Döşeme

To my precious family,

ACKNOWLEDGEMENTS

I would like to enounce my deep gratitude to my supervisor Prof. Dr. Aydan M. Erkmen for her valuable supervision, advice, useful critics and discussions throughout this study.

I am also grateful to my thesis committee members Assoc. Prof. Dr. Afşar Saranlı, Prof. Dr. Aydın Alatan, Assoc. Prof. Dr. İlkay Ulusoy and Assoc. Prof. Dr. Sedat Nazlıbilek for their criticism and advices.

I would like to give my special thanks to Ayhan Özgür for his previous contributions on the topic, his shares and help for some parts of this study.

I cannot explain my feelings about every member of my family, especially to my mother Filiz Çetinkaya, to my father Metin Çetinkaya and my sister Simge Çetinkaya for their unconditional love. Without their encouragements and advices, that journey would be endless for me.

I am deeply grateful to Ali Mancar and Melis Kesik (Mancar) to choose me as the best man at the beginning of their journey.

I owe my great thanks to the fun gang that always makes me laugh, my roomie Baki Er, probably my oldest friend Kübra Mutlu, my “kirve”s Gökmen Cengiz and Can Görür.

I also want to thank Gizem Akar, Orçun Elitez, Çilem Çiçek, Kemal Arı, Özge Koçer, Çisot, Esra Özdemir, Büküş, Gizem Üstüner, Özge Tekin, Ulaş, Akın, Fırat, Utku, Tuğçe for their friendships, encouragement and support.

It is impossible for me to explain my gratitude to Nehir for her endless love, motivation support and patience. One thing that I learned: I am the luckiest person in

the world because she could find me. Even one “yes” from her makes me the happiest person in the world.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVATIONS

CCL	(Connected Component Labeling)
DT	(Delaunay Triangulation)
EKF	(Extended Kalman Filter)
IMU	(Inertial Measurement Unit)
LADAR	(LAser Detection And Ranging)
RRT	(Rapidly Exploring Random Tree)
SLAM	(Simultaneous Localization and Mapping)

CHAPTER 1

INTRODUCTION

1.1. Motivation

Mapping an environment autonomously by robotic devices is a critical and strategic approach where simultaneous localization and mapping (SLAM) is one of the prominent approaches generally used. 2D exploration of a 3D environment necessitates mapping while concurrently localizing within a 2D environment projected from 3D. 3D SLAM on the other hand, relies on 3D sensing of the actual environment and requires higher order measurements of the related robot motion. Today's sensor technology such as 3D LADAR or Microsoft Kinect is capable to generate 3D point cloud which is utilized in 3D mapping and localization process achieved by these sensory components [1–11]. However, working with 3D point cloud and high order motion measurements are computationally costly. Moreover, correcting both localization and mapping errors while working in high dimensional spaces is a difficult problem known as the curse of dimensionality. However, the use of 3D sensors is likely to be more expensive and less energy efficient than simple 2D sensors. Thus, the literature is equally divided between 3D SLAM and map growing from 2D to 3D.

Stacking and connecting 2D maps of different height levels in an environment needs the correct alignment orientation maps of the 2D maps. Actually, this method is commonly adopted in medical area since the medical imaging systems using equipments like X-rays, ultrasound and MR generate 2D parallel image slices of a 3D structure [12–18] which are then stacked and morphed to build the visualization of the 3D structure.

In the case of exploring an indoor environment reconstruction of the environment by stacking and merging 2D map can be achieved in one of the following cases

Heterogeneous 2D SLAM map slices of a 3D environment can be achieved either by; multiple land vehicles with different heights, single land vehicle with adjustable sensor height or an aerial vehicle flying at different altitudes. Common point in all is that they use simple 2D sensors and computational complexity in SLAM is considerably decreased. Most importantly, map growing that makes use of maps generated from these different platforms may have been obtained at different time and purposes.

Cleaning robots in the shopping malls can be thought as a real life application. Robots with various sizes fit different places of the malls and clean at different resolution creating 2D maps of different levels of details in the maps can create 2D maps of their environment and these maps can be used to grow a topological 3D map of the environment. After that this 3D map can be used while assigning tasks to each robot.

Another application can be though as an aerial vehicle used in territories that a human cannot enter, such as a damaged building after an earthquake. It is enough for this aerial vehicle to have a 2D range sensor (such as HOKUYO UTM-30LX/LN) and a barometric altitude sensor to know its height. After that it can match 2D scans with different heights and construct 2D map for each level. Lastly all 2D maps can be positioned and merged.

1.2.Objectives and Goals

Achieving 3D map growing requires solutions to the problems that are affecting building blocks of the approach which are;

- Construction of proper 2D maps, since they are the core elements of structure reconstruction that carries information about the structure itself,

- Correspondence problem, which consists of selecting and aligning correct maps for extrapolation,
- Branching problem, which is handling extrapolation of the separated regions at the same height level to the regions of other levels,
- Tiling problem which is about properly connecting the regions together even the regions reveal discontinuities between themselves.

In the present thesis aim is to manage mentioned problems by proposed novel map growing technique.

The first objective is to explore an unknown indoor environment with the robot and generate a 2D map of the environment. During the process, one sub-aim is to minimize the localization errors and eventually reach geometric structure of the environment at one level. By repeating the same process at different height levels, in other words “un-scanned” levels, of the environment various 2D maps are created. Important point here is that different platforms can scan the unknown indoor environment and create 2D maps from various height levels.

The second objective is about the growing 3D structure of the environment by surface reconstruction utilizing created 2D maps at the first part. Throughout growing part there are three sub objectives to achieve final result which are namely, creating correct correspondences between 2D maps, generating connection points and tiling the connection points.

In the creation of the correct correspondences, 2D maps are aimed to be aligned with respect to each other by using the key features extracted from each map. Additionally, these key features are used for designate similar and dissimilar regions between the maps. The purpose for separating the regions as similar and dissimilar is achieving smooth growing in the case of dissimilarities, or discontinuities, between the regions. Also, this separation allows the system to generate a feedback about the dissimilar regions to be re-explored.

Connection points or correspondence point sets are generated in two different ways considering the similar and dissimilar regions. For similar regions correspondence point sets are generated from the boundary parts of the regions while for dissimilar regions correspondence point sets are generated by creating tree structures inside the regions and using its vertices and links.

In the last sub objective, the aim is to achieve surfaces by connecting the correspondence points set, or in other words tiling them. Tiling of the points is achieved by generating triangular meshes with the correspondence points using their topological adjacency relations.

1.3.Methodology

1.3.1. Simultaneous Localization and Mapping (SLAM)

The first aim of the present thesis is to create 2D grid maps of the environment as mentioned in the objectives/goals section. The platform used for this purpose is a PIONEER robot with a laser scanner and inertial measurement unit (IMU). As auxiliary platforms, the PIONEER robot is mounted on two different manually controller carriers, one of which has adjustable height level and the other one has a fixed height level, in order to scan the environment from different height levels. There is no IMU sensors installed on these platforms, but this issue is handled in the SLAM algorithm by scan matching method.

At this initial part of the thesis, utilized platforms, PIONEER and the mobile sensor package, collect range and bearing information from the laser scanners. Additionally, there exist IMU data for the PIONEER. However, aim is to use only distance measurements from laser scanners and that is why feature based scan matching method is used for motion estimation with the selected SLAM algorithm.

In this thesis, FastSLAM algorithm with unknown data association is chosen as an approach to the SLAM problem which is in a way, combination of Kalman and

particle filters [19]–[23]. The algorithm handles localization and mapping in different parts. Localization is done by estimating the pose of each particle by using motion estimation obtained from feature based scan matching and landmarks which are extracted from the raw scan data. The mapping part is then applied depending on pose estimate of the most weighted particle and the scan data.

In addition to the FastSLAM, a feature based (feature to feature matching) scan matching algorithm is used to measure the motion of the platform [24]. This algorithm also reduces localization errors which eventually affect the structure of resulting map.

1.3.2. Map Growing

In this part, aim is to grow a 3D map from 2D grid maps constructed by FastSLAM algorithm. There are three steps taken for this purpose namely; creating correct correspondences, generating connection points and tiling.

1.3.2.1. Creating Correct Correspondences

Aim of this part is to solve correspondence problem by determining the affinity between the extracted key features from each map and using this information to align the maps. In order to achieve this, first line features are extracted from each map and a matching between these line features are established. Utilizing this matching result initial alignment is achieved. Afterwards key features are extracted from each map with Harris-Stephens corner detector. These key features are fed to the ICP (Iterative Closest Point) method for matching iteratively and obtain translation and rotation matrices necessary to optimize the alignment between the maps. Additionally, extracted corner and line features are used to designate similar and dissimilar regions among the maps. At first regions in each map are determined by CCL (Connected Component Labelling) method. Then, features located on regions are matched across the maps in order to measure the similarity of the regions. When the similar and dissimilar regions are obtained they are used for generating connection points. Also, detected dissimilar regions are labeled for re-exploration.

1.3.2.2.Generating Connection Points

At this part, connection points for map growing are aimed to be achieved. For similar regions boundary part are used for generation of connection points and their links. Therefore, boundary extraction method is applied on similar regions and on the extracted boundaries grid cells and their links between them are stored. For dissimilar regions, connection points are obtained from the tree structures created inside the regions which is achieved with the method known as RRT (Rapidly Exploring Random Tree). On the created tree vertices and their links between them are used as connection points.

1.3.2.3.Tiling Connection Points

In this part, the aim is to reconstruct surfaces by connecting the connection points appropriately via triangulation. The method used for this purpose is called DT (Delaunay Triangulation) which creates triangles while trying to maximize the angle values inside the triangles in order to avoid skinny triangles. Applying DT to generated connection points results in triangular meshes which are the surfaces that aimed to be reconstructed.

1.4.Contributions

As mentioned previously, there are various approaches to construction of 3D map problem which are using different methods to handle this problem. One of the methods in the literature is about working with 3D point clouds which enables the use of 3D measurements from environment directly. However, working with 3D point clouds requires working with higher order state spaces which is computationally costly and causes the problem known as curse of dimensionality.

There are some studies in medical imaging which are proposing alternative approaches such as constructing 3D structures by using 2D parallel images. In these systems the main idea is to use parallel images and reconstruct structures, mostly

organs, by stacking up the images and connect them properly. However, these studies work with continuous structures, so their approaches do not propose a solution for discontinuous structures.

In the present thesis, a novel approach for growing 3D maps by using 2D SLAM maps with mobile heterogeneous robots is proposed. The proposed approach takes 2D raw distance measurement data from different height levels of the environment, creates 2D maps of each height level and uses these maps to grow a 3D map of the environment without any model or prior information. At both continuous and discontinuous parts on the maps of each level, correspondence points are generated and tiled in order to achieve a smooth growing of 3D map. When the approaches in the literature are considered;

- Curse of dimensionality and computational complexity problems are avoided by working with 2D data,
- Discontinious structures can be reconstructed by using 2D maps.

In the present thesis, another use of the proposed novel method is to generate feedbacks for regions to re-examine, i.e. focus, by determining discontinuities at the extracted regions on the maps. This is achieved by comparing 2D map results from consecutive height levels with each other and inspecting the similar and dissimilar parts in order to determine the discontinuities on the structures.

1.5.Outline of the Thesis

The outline of the thesis can be presented as follows:

In Chapter 2, a survey of related studies done in SLAM methods, map growing methods are presented.

In Chapter 3, theoretical background and implementation of FastSLAM algorithm with feature based scan matching method are presented. This part also includes background information about probability theory and probabilistic filters used in FastSLAM.

In Chapter 4, map growing is presented. This chapter contains three sub parts namely; correspondence problem, generation of connection points and tiling problem.

In Chapter 5, sensitivity analysis is conducted to the proposed approach and results and discussions are presented.

In Chapter 6, experimental results and discussions are presented. In this chapter experimental setup is described and performance of the approach is evaluated with experimental results.

In Chapter 7, objective of the thesis and the results are summarized. Moreover, future works are presented.

In Chapter 8, references are presented.

CHAPTER 2

LITERATURE SURVEY

In this chapter, studies related to the goals of the thesis work are surveyed which are SLAM, map growing and surface/volume reconstruction. The first section, literature survey on simultaneous localization and mapping (SLAM) and different approaches to this famous synchronized problem is presented. In the second section, various studies related to map growing are provided. In these studies common problems encountered during map growing and surface reconstruction namely correspondence, branching and tiling problems are explained. Lastly, the literature survey is summarized and inspired studies are emphasized with the perspective of the thesis work.

2.1. Simultaneous Localization and Mapping (SLAM)

Simultaneous localization and mapping (SLAM) enables a platform to create a map of its environment and simultaneously localize itself relative to the map it creates. In SLAM problem, localization and mapping work are complementary to each other. Building a map requires pose estimation of the platform relative to the map, so that with the help of sensor measurements the map can be updated from the estimated pose.

In [25], Riisgaard and Blass gives a brief introduction to the SLAM problem by mentioning about different approaches and history of the problem. They also try to create an understanding about the SLAM process and provide the requested equipment for the related approach. In their work, they mostly focused on EKF-SLAM and provide detailed information about possible hardware options, landmark

extraction, data association and Extended Kalman Filter. Thesis work can be considered as a practical guide to the SLAM problem.

Durrant-Whyte et al. [19] [26], state the SLAM problem and related models used for the approach, and use statistical basis to establish a solution and adopt Kalman Filter based approach. Additionally, mathematical framework of system model, vehicle and landmark models, and observation models are defined in this statistical basis. In [26], implementation of SLAM problem for completely autonomous vehicles with distance measurement sensors on them and without the need of artificial infrastructures or a priori knowledge, is covered. Natural landmark observation and extraction as a geometric beacon, and data association, which uses the correlations between locations of the landmarks, are also discussed.

Starting from a statistical based system model for the SLAM problem, there exist various approaches which use different methodologies such as Extended Kalman Filter (EKF) based SLAM algorithm. Bailey et al. [27], provide detailed information about the non-linear SLAM cases and share the experiences on the EKF that is suitable for non-linear models. However, in their study [27] they mention two problems that they experience while working with EKF-SLAM. Firstly, linearization is not always as accurate as desired. Secondly they mention that the probability distribution in real life is non-Gaussian, so that calculated mean and variances may not be adequate.

Another approach to the SLAM problem is the Rao-Blackwellized particle filter based SLAM algorithm. This approach is introduced as an alternative to the EKF-SLAM and use particles with a calculated weight to represent the different possible states of the actual vehicle. Calculated weight of each particle is then used in resampling where particles with low weights are eliminated. Particle filter is easier to implement but has low efficiency in time and memory against EKF-SLAM. Additionally, particle filter solves the global localization problem by adding particles with random states to the particle set. Grisetti et al. [28] [29], provide their results on particle filter based SLAM and introduce an enhancement to it called Rao-

Blackwellized particle filter. With this method they aim to use adequate number of particles for building an accurate map. Moreover, they implement an adaptive resampling technique so that it provides a reasonable variety of particles to exits and prevent particle depletion.

FastSLAM is another statistical based approach to the SLAM problem which combines Kalman filter and particle filter together in one algorithm. In [20] [30], Montemerlo and Thrun introduce FastSLAM as an algorithm that uses modified particle filter to estimate location of the robot. Each particle has a defined number of landmarks and for each landmark there exist a Kalman filter that estimates its state. Montemerlo and Thrun also mention that with M landmarks and K independent Kalman filters the time complexity of the algorithm is calculated as O(M*logK) while in the EKF-SLAM case it is $O(M^2)$. Therefore, they state that FastSLAM is faster than existing EKF-SLAM algorithms. In [22], Guo et al. explain another advantage of the FastSLAM which is that data association can be done for each particle separately. This renders data association robust against noise of robot motion with particles with incorrect associations eliminated at the resampling step. Data association and place recognition in cyclic environments give the ability to use loop closure technique in which the integrated errors in localization can be eliminated. In [21], Hahnel et al. provide their experiences on creating maps of large scale cyclic environments. It is mentioned that FastSLAM is improved and number of particles required for mapping large scale environments is reduced. Also, they use scan matching technique for landmark extraction and these landmarks are used in loop closure.

2.2. Map Growing

Map growing or surface reconstruction subject aims to rebuild a structure or an environment from obtained 2D data. Throughout which, there are commonly three sub-problems namely; correspondence problem, which is about selecting and aligning correct maps or contours to connect, branching problem which is about handling separated regions at the same level for connection and tiling problem which is about connecting the regions together.

2.1.1. Correspondence Problem

Correspondence problem is mostly related to determining adjacency relationships between maps or contours at different height levels. Achieving relationship between different levels leads correct selection of maps or contours and proper alignment between them. Correspondence problem can be seen at different aspects in literature such as multi agent systems and medical imaging system where there are various maps or contours to relate and create correspondences.

In robotics multi agent systems can be seen in different applications in robotics, such as exploration of an environment, rescue missions etc. In [31], Burgard et al. work on a such system in which the aim is to explore an unknown environment by multiple robots. While exploring the environment, a common map which is constructed during the process is shared by the robots. Constructed maps are based on occupancy grid maps and integration of these maps is based on the assumption that relative positions of the maps are known. They use the probability in each cell for local maps to calculate the probability in each cell for total map with the following formula:

$$P(OCC_{x,y}) = \frac{odds_{x,y}}{1 + odds_{x,y}} \tag{2-1}$$

However, the assumption used in this study is not directly applicable in a real life experiment. In this thesis, relative position of local maps is unknown.

In [32], Birk and Carpin introduces an approach called Adaptive Random Walk which looks for stochastic search of transformations between provided maps. In this approach, at each step of the random walk, a random configuration is created and an overlapping function is used to measure the goodness of the matching. In this approach a map is chosen as fixed and the other one is moved around by selecting different transformations. Transformations are chosen stochastically by using a Gaussian distribution. However, according to Adluru et al. [33], usage of Gaussian distribution is a drawback of the Adaptive Random Walk method because if initially local maps are placed far apart from each other distribution may not converge. This

resembles global localization problem for which Gaussian filters cannot find a solution.

In [33], the proposed method is called ViRtual-SLAM (VR-SLAM) in which Sequential Monte Carlo estimation (Rao-Blackwellized particle filter) is employed for a virtual robot. Also, trajectory of as single robot is utilized which makes their technique more robust when minimal overlap between maps occurs. They also compare their method to mapping part of particle filter based SLAM for multi agent systems and it is mentioned that proposal distribution and importance weights have different aspects in design. In proposal distribution, they use structural registration (shape matching) to measure the motion. For the perception model, they use a correlation matrix to detect the similarity between grid maps. It is known that particle filter has a solution for convergence issue, and as Adluru et al. mention this can be an improvement from Adaptive Random Walk method which uses Gaussian distribution.

In an alternative approach to correspondence issue, Wang at el. [34] realize a multi robot system to explore a large scale unknown environment. In their study, robots are communicating by using RTM (Robot Technology Middleware) and aim to generate a global map by utilizing local maps through map merging. For this purpose, an integration of Scale-Invariant Feature Transform (SIFT) feature matching with Iterative Closest Point (ICP) algorithm is used. As the name implies SIFT features are invariant to the rotation and scaling which makes it possible to have a robust matching between these features. SIFT features are extracted by using difference-of-Gaussian (DoG) image which is calculated by differentiating nearby scales separated by a constant. Formulation used for this method is given as:

$$\begin{aligned} D(x,y,\sigma) &= \big(G(x,y,k\sigma) - G(x,y,\sigma)\big) * \mathrm{I(x,y)} \\ &= L(x,y,k\sigma) - L(x,y,\sigma) \end{aligned} \tag{2-2}$$

Where D(x,y,σ) is the DoG image, L(x,y,σ) is the convolution result of input image I(x,y) and variable-scale Gaussian G(x,y,σ) which is formulated as:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (2\text{-}3)$$

They use SIFT matching as an initial step for map correspondence, to obtain some prior information, and then complete it by using ICP algorithm. ICP is used here for optimizing the initial matching result and obtain a better overlap for the local maps.

Meyers and Skinner [14] present an approach to the problem of reconstructing 3D structures with planar contours. As they mentioned surface reconstruction from contours is significant for biomedical research, solid modelling and industrial inspection. Their proposed method produces triangular meshes for tiling the data points of the contours in order to create surfaces. They mainly focused on obtaining correct correspondences between adjacent contours and implemented two solutions for correspondence problem. One of them is based on Soroka's object-understanding system which assembles elliptical cylinders from contours. This system generates ellipses that centers along a linear axis for each contour and use their length variation to overcome correspondence problem. Other solution is based on a method that uses minimum-cost spanning tree in order to compute correspondences of the contours from different layers simultaneously.

In this thesis, in order to find the transformation between different local maps a similar approach to the Wang et al. [34] is utilized. Line and corner features used for that purpose are directly obtained from maps. Then feature matching is applied to initially align maps with respect to each other and afterwards ICP algorithm is used to optimize the result. Considering stochastic approaches on map merging [32] [33], random steps can take longer time to converge and there is also a possibility that algorithm may not converge. Additionally, instead of taking random steps, utilizing features which are already obtained in the SLAM solution, provides low computational cost and robustness since the location and properties of the features, or landmarks, are also calculated and updated several times in the algorithm.

2.1.2. Branching and Tiling Problems

Branching and tiling problems are rarely encountered in the robotics literature unlike previously mentioned subjects. On the other hand, there are several studies implemented for medical applications which are considering these problems together with surface reconstruction subject [35] [36] [37] [38] [39].

In [38], Sareen et al. focus on prosthesis device which a silicon made artificial facial feature worn to cover facial deformity. At first, the geometry of the patient's facial features is scanned with 3D scanners. Outcome of this process is 3D point data cloud which is large to handle, so they reduce it to reconstruct facial surface accurately. An integrated contour-based algorithm which extracts sequence of contours from dense unstructured point cloud is used to tile the contours and generate the facial surface from the extracted contours. Reconstructed facial surface is then used for modelling and fitting the prosthetic device. In here the extracted contours resemble the 2D map constructed in SLAM algorithm. The only difference here is the contours are really dense and the facial model is known, so that deformed parts can be reconstructed.

Zeng et al. [39][36][37], present surface and volumetric intensity reconstruction of a bone from 2D X-ray images. In [36] and [37], surface of a bone is reconstructed from 2D X-ray images by following three stages. In the first stage, affine registration, transformation between point distribution model (PDM) and the X-ray images are estimated by using Iterative Closest Point (ICP) algorithm. The result of first stage, obtained point correspondences are then utilized in second stage which includes regularized morphing where a template surface is estimated form PDM by using a Mahalanobis distance based stabilizer. At the final stage estimated template surface is used in shape deformation method where 2D-3D correspondences are established based on iterative non-rigid 2D point matching which uses symmetric injective nearest neighbor mapping operator (SIN-MO) and 2D thin plate splines (2D-TPS) based deformation iteratively to find a fraction of best matched point pair for tiling. In [39], Zheng present a volumetric intensity reconstruction in addition to the surface reconstruction of a bone structure. The proposed method in this study utilizes 3D surface model reconstructed from 2D images as a carrier to obtain the volumetric

intensity reconstruction by Partial Least Squares Regression (PLSR) where input vectors are iteratively mapped to the output vectors into a common vector space so that the covariance between these projected vectors is maximized.

Pontier et al. [35], introduce a new method called skeleton-based implicit model in order to reconstruct the shapes from data point cloud organized in parallel planar sections. This method constructs 3D skeleton of the main structure which consists of planar elements and is grown by the 2D Voronoi skeleton of each section. In addition to the skeleton a field function, which is a decreasing function of a distance to the skeleton, is used to define an object. Field function assigns potential for each point on 2D Voronoi skeleton of each section and generates a scalar field function to surround the skeleton. Advantage of this approach is that algorithm does not operate iteratively and there is no need for a priori knowledge to build a model or in this case skeleton.

Wang et al. [15] present an approach to surface reconstruction from contours in their study. Their method which is based on two-dimensional constrained Delaunay triangulation provides solutions for correspondence, branching and tiling problems together. They claimed that Delaunay triangulation provides nearest connections optimally and tiling problem can be handled with this approach. In order to overcome the branching problem they propose adding intermediate points to the edges of the generated triangles when vertices of triangles are on the same contours. Intermediate points are inserted on the edge of the triangle which is in the middle of the two regions which are on the same contours.

Coyle et al. [16] introduces a method that approach all three problems, corresponding, branching and tiling, simultaneously. They are claiming that it is accomplished by implementing a set of constraints and deriving rules for correspondence and tiling such that these rules and constraints ensures regions to obey physical constructs.

2.3. Summary

In this chapter related studies in the literature that are adopted in the present thesis are surveyed. Outcomes can be summarized as follows:

- FastSLAM technique is found adequately efficient for simultaneous localization and mapping approach compared to other techniques. The reason is that it uses advantages sides of both Kalman filter which is linear and two variables are enough to define probability density, and particle filter, which has a solution to the global localization problem.
- Map growing or surface reconstruction from 2D maps to 3D space is encountered more in medical area than robotic area. Therefore, the present thesis work brings an approach that consists map growing to the robotic literature.
- After examining related studies, map growing is decided to be implemented considering the three common problems.
    - At first, correct correspondences between local maps are created and these maps are properly aligned. This is achieved by using line and corner features extracted directly from local maps for an initial alignment and then using the ICP method to optimize the result. (correspondence problem)
    - Secondly, for connectivity of the regions on both same and different levels, correspondence points are generated considering boundary points and points obtained by using the vertices of tree structures constructed inside the regions. (branching problem)
    - Lastly, extracted correspondence points are tiled by using Delaunay triangulation method. (tiling problem)

CHAPTER 3

SIMULTANEOUS LOCALIZATION AND MAPPING

In the present thesis, stochastic based simultaneous localization and mapping (SLAM) algorithm is preferred since it allows the efficient exploration of an unknown environment without getting lost and create a 2D map. Among the presented approaches for implementation of SLAM, FastSLAM algorithm is found adequate to utilize in this thesis, since it uses advantageous parts of both Kalman and particle filter. Additional to the FastSLAM algorithm, feature based scan matching algorithm which extracts features and landmarks from raw laser scan data is used.

In this chapter, we will introduce the implementation of FastSLAM and feature based scan matching algorithm. In the first part a theoretical background about the probability theory and probabilistic filters used for all stochastic SLAM approaches is provided. The main structure of our implemented FastSLAM algorithm will then follow together with the detailed breakdown into blocks that constitute the infrastructure of the algorithm.

3.1. Theoretical Background

Stochastic approaches in robotics or mostly known as probabilistic robotics is a relatively new field which models uncertainty issues from sources, such as noises, limitations, environmental effects on sensors and actuators, modelling errors, approximations. Probability theory is utilized at this point to deal with uncertainty and represent the possibilities with belief. Therefore, it is thought that before proceeding into further details of stochastic FastSLAM algorithm it can be good to provide brief theoretical overview about probability theory and probabilistic filters.

3.1.1. Probability Theory

Probability theory mathematically explains probability of a potential result of a specific event to occur. Let for a specific event, x denote a potential result and X denote the set of potential results, or commonly known as random variable, such that possibility for that event to result in x is given as

$$P(X = x)\,, or\ P(x) \tag{3-1}$$

Probability is non-negative and sum of all possible probabilities is equal to 1. That is

$$0 \leq P(X = x) \leq 1 \tag{3-2}$$

In order to compute probabilities in continuous spaces, a function called probability distribution functions (PDF) is utilized. This function carries information about the relative likelihood of a random variable to take a specified value. A common PDF used in robotics is one dimensional normal distribution with mean μ and variance σ^2 which is given as

$$\mathrm{p(x)} = (2\pi\sigma^2)^{-\frac{1}{2}}\ e^{(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2})} \tag{3-3}$$

For multi-dimensional case the normal distribution becomes

$$\mathrm{p(x)} = det(2\pi\Sigma)^{-\frac{1}{2}}\ e^{(-\frac{1}{2}(x-\mu)^T(\Sigma)^{-1}(\mathrm{x}-\mu))} \tag{3-4}$$

Where μ is mean vector and Σ is covariance matrix which is positive semi definite and symmetric. In the case of two random variables, let them to be X and Y, joint distribution is utilized which is defined as

$$\mathrm{p(x, y)} = \mathrm{p(X = x\ and\ Y = y)} \tag{3-5}$$

For the case in which X and Y are independent

$$p(x, y) = p(x)p(y) \tag{3-6}$$

If result of a random variable is known, probability of the other random variable is given as (if p(y)>0)

$$p(x|y) = \frac{p(x, y)}{p(y)} \tag{3-7}$$

This is called conditional probability. By using the conditional probability, theorem of total probability is defined as

$$p(x) = \int p(x|y)p(y) \qquad \text{(continuous)} \tag{3-8}$$

$$p(x) = \sum_{y} p(x|y)p(y) \qquad \text{(discrete)} \tag{3-9}$$

According to Bayes rule, when p(y) > 0 and p(x) > 0

$$p(x|y) = \frac{p(x, y)}{p(y)} = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\int p(y|x')p(x')} \qquad \text{(continuous)} \tag{3-10}$$

$$p(x|y) = \frac{p(x, y)}{p(y)} = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\sum_{x'} p(y|x')p(x')} \qquad \text{(discrete)} \tag{3-11}$$

If one would like x to be inferred from y, p(x) becomes the prior probability distribution, p(x|y) becomes posterior probability distribution and y becomes measurement data. However, according to Bayes rule, p(x|y) is computed by using p(y|x) which is called inverse conditional probability and means that for a given state, probability of a measurement data to be obtained.

3.1.2. Probabilistic Filters

During robot navigation, robotic explorations and environmental interactions, present robot state is dependent on all past states, control inputs and measurements. In this case probabilistic distribution can be constructed as follows

$$p(x_t|x_{0:t-1}, z_{1:t-1}, u_{1:t}) \tag{3-12}$$

Where x is the robot state, z is the measurements and u is the control input. If it is assumed that present state depends on only previous state, measurement and control input then it is called Markov model. In this case probability distribution becomes

$$p(x_t|x_{0:t-1}, z_{1:t-1}, u_{1:t}) = p(x_t|x_{t-1}, u_t) \tag{3-13}$$

The distribution above is regarded as state transition or motion model for the robot. In other aspect, it is enough to know only present state from which the measurements are taken. Then the probability distribution for the measurements becomes

$$p(z_t|x_{0:t-1}, z_{1:t-1}, u_{1:t}) = p(z_t|x_t) \tag{3-14}$$

This is regarded as measurement or sensor model of the robot.

Belief is one of the key concepts in probabilistic robotics. Belief provides internal knowledge of the environment. It is calculated with a stepwise method, first prediction then measurement update steps. In prediction step the following belief function is calculated.

$$bel'(x_t) = p(x_t|z_{1:t-1}, u_{1:t}) \tag{3-15}$$

In measurement update state from bel'(.) function bel(.) function, which is given below, is calculated

$$bel(x_t) = p(x_t|z_{1:t}, u_{1:t}) \tag{3-16}$$

The general approach for calculating the belief is the Bayes Filter which is a recursive approach that uses measurements and control data. Pseudo Code for this filter is presented at Table 3-1.

Table 3-1 Pseudo code of Bayes filter algorithm

1:	**Algorithm Bayes_Filter**$(bel(x_{t-1}), u_t, z_t)$
2:	for all x_t do
3:	$bel'(x_t) = \int p(x_t\|u_t, x_{t-1})\ bel(x_{t-1})dx$
4:	$bel(x_t) = \eta p(z_t\|x_t) bel'(x_t)$
5:	end for
6:	return $bel(x_t)$

3.2. Implementation of FastSLAM

FastSLAM, as mentioned earlier, is a stochastic technique used for simultaneous localization and mapping which includes both particle and Kalman filter.

Kalman filter is a linear, Gaussian based Bayes and parametric filter whose parameters are μ and Σ . In the case of FastSLAM, a discrete Kalman filter is utilized with a particle filter which is non-parametric and uses finite set of samples instead of parametric closed form expressions. These samples are called particles and each particle has its own state space, observed landmarks' state space and map of the area

stored inside it. Estimation of particle and its landmarks' states are achieved by the use of Kalman filter and in FastSLAM algorithm state spaces are given as

$$ps_t^i = \begin{pmatrix} px_t^i \\ py_t^i \\ p\theta_t^i \end{pmatrix} \tag{3-17}$$

$$ls_t^{i,n} = \begin{pmatrix} lx_t^{i,n} \\ ly_t^{i,n} \end{pmatrix} \tag{3-18}$$

Where px_t^i, py_t^i and $p\theta_t^i$ are the pose estimates of the i[th] particle of the robot at time t and $lx_t^{i,n}, ly_t^{i,n}$ are the pose estimates of the n[th] landmark of the i[th] particle.

State and measurement equations are governed by linear difference equations as follows

$$ps_t^i = A_t ps_{t-1}^i + B_t u_t + \varepsilon_t \tag{3-19}$$

$$z_t = C_t ps_t^i + \delta_t \tag{3-20}$$

Where A_t describes the state change from time t-1 to t, B_t describes the change in control input from time t-1 to t, C_t describes the map between state and observations, ε_t and δ_t are process and measurement noise respectively. Errors are assumed to be Gaussian distributed with covariance R_t and Q_t which are given as

$$R_t = \begin{bmatrix} \varepsilon_{rot1,t} & 0 & 0 \\ 0 & \varepsilon_{trans,t} & 0 \\ 0 & 0 & \varepsilon_{rot2,t} \end{bmatrix} \tag{3-21}$$

$$Q_t = \begin{bmatrix} \varepsilon_{r,t} & 0 \\ 0 & \varepsilon_{\theta,t} \end{bmatrix} \tag{3-22}$$

Where $\varepsilon_{rot1,t}, \varepsilon_{rot2,t}$ and $\varepsilon_{trans,t}$ are Gaussian noises for robot motion which is modelled as rotational-translational-rotational motion. $\varepsilon_{r,t}, \varepsilon_{\theta,t}$ are also Gaussian noises for range and bearing of the measurement sensor.

State transition probability distribution $\mathrm{p}(ps_t^i|u_t, ps_{t-1}^i)$ and measurement probability distribution $\mathrm{p}(z_t|ps_{t-1}^i)$ are given by

$$\mathrm{p}(ps_t^i|u_t, ps_{t-1}^i) = det(2\pi R_t)^{-\frac{1}{2}}\ e^{(-\frac{1}{2}\left(ps_t^i - A_t ps_{t-1}^i - B_t u_t\right)^T (R_t)^{-1}(ps_t^i - A_t ps_{t-1}^i - B_t u_t))} \quad (3\text{-}23)$$

$$\mathrm{p}\left(z_t|ps_t^i\right) = det(2\pi Q_t)^{-\frac{1}{2}}\ e^{(-\frac{1}{2}\left(z_t - C_t ps_t^i\right)^T (Q_t)^{-1}(z_t - C_t ps_t^i))} \quad (3\text{-}24)$$

In this distribution functions given u_t and z_t are described as follows

$$u_t = \begin{bmatrix} \delta_{rot1,t} \\ \delta_{trans,t} \\ \delta_{rot2,t} \end{bmatrix} \quad (3\text{-}25)$$

$$z_t = \begin{bmatrix} r_t \\ \theta_t \end{bmatrix} \quad (3\text{-}26)$$

Control inputs, $\delta_{rot1,t}$, $\delta_{trans,t}$and $\delta_{rot2,t}$ are computed with the help of feature based scan matching method. In the scan matching method, at first feature sets which are represented with f_t are extracted from each scan. Then feature sets of consecutive scans are compared and pose differences which are represented with Δf_t are computed. These pose differences are utilized to calculate the control inputs.

As mention earlier, Kalman filter, whose pseudo code is given in Table 3-2**,** is executed for each particle separately.

Table 3-2 Pseudo code of Kalman filter algorithm

1:	**Algorithm Kalman_Filter**$(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t)$
2:	$\mu'_t = A_t\mu_{t-1} + B_t u_t$
3:	$\Sigma'_t = A_t\Sigma_{t-1}A^T{}_t + R_t$
4:	$K_t = \Sigma'_t C^T{}_t (C_t \Sigma'_t C^T{}_t + Q_t)^{-1}$
5:	$\mu_t = \mu'_t + K_t(z_t - C_t\mu'_t)$
6:	$\Sigma_t = (I - K_t C_t)\Sigma'_t$
7:	**return** μ_t, Σ_t

On the top of the Kalman filter, particle filter is utilized to calculate weights of particles and according to these weights resample the particle set. Pseudo code of the general particle filter is given in Table 3-3**.** Here X_t is the particle set at time t.

Table 3-3 Pseudo code of particle filter

1:	**Algorithm Particle_Filter**(X_{t-1}, u_t, z_t)
2:	$X'_t = X_t = \varnothing$
3:	**for** i = 1 to M do
4:	sample $ps_t^i \sim p(ps_t^i \mid u_t, ps_{t-1}^i)$ (achieved with KF)
5:	$w_t^i = p(z_t \mid ps_t^i)$

Table 3-3 (continued)

6:	$X'_t = X'_t + < ps_t^i, w_t^i >$
7:	**end for**
8:	**for** i = 1 to M do
9:	draw a particle with probability α w_t^i
10:	add ps_t^i to X_t
11:	**end for**
12:	**return** X_t

Combining the equations and method stated above FastSLAM algorithm whose pseudo code is presented in Table 3-4 is constructed.

Table 3-4 Pseudo code of implemented FastSLAM algorithm

1:	**Algorithm FastSLAM**($X_{t-1}, z_t, f_{t-1}, lm_{t-1}, m_{t-1}$)
2:	$X_t = X_{t-1}$,
3:	$f_t = \boldsymbol{feature\_extraction}(z_t)$
4:	$\Delta f_t = \boldsymbol{scan\_matching}(f_t, f_{t-1})$
5:	$u_t = \boldsymbol{inverse\_odometer\_motion}(\Delta f_t, ps_{t-1}^i)$

Table 3-4 (continued)

6: **for** i = 1 to N do

7: $[ps_t^i, p\Sigma_t^i] = \boldsymbol{prediction}(ps_{t-1}^i, p\Sigma_{t-1}^i, u_t)$

8: $[ls_t^i, l\Sigma_t^i] = \boldsymbol{landmark\_extraction}(f_t, ps_t^i)$

9: $[mls_t^i, lm_t^i] = \boldsymbol{data\_association}(ps_t^i, f_t, ls_t^i, lm_{t-1}^i)$

10: **if** mls_t^i is not empty

11: $J_t^i = \boldsymbol{jacobians}(ps_t^i, mls_t^i)$

12: $[ps_t^i, p\Sigma_t^i] = \boldsymbol{particle\_update}(ps_t^i, p\Sigma_t^i, J_t^i)$

13: $[ls_t^i, l\Sigma_t^i] = \boldsymbol{landmark\_update}\ (ls_t^i, l\Sigma_t^i, mls_t^i, J_t^i)$

14: $w_t^i = \boldsymbol{particle\_weight}(p\Sigma_t^i, J_t^i)$

15: $X_t^i = < ps_t^i, p\Sigma_t^i, w_t^i >$

16: **end if**

17: **end for**

18: **if** mls_t is not empty

19: $X_t = \boldsymbol{particle\_resampling}(X_t)$

20: **end if**

Table 3-4 (continued)

21:	$m_t = \boldsymbol{occupancyGridMapping}(X_t, m_{t-1}, z_t)$
22:	**return** X_t, f_t, lm_t, m_t

FastSLAM algorithm takes followings as inputs:

- Set of particles, with their weights, at time t-1 represented by X_{t-1}
- Sensor measurement or scan at time t, represented by z_t
- Feature set for all particles at time t-1, represented by f_{t-1}
- Landmark memory of each particle at time t-1, represent by lm_{t-1}

At first algorithm extracts features from the sensor measurements and store them in f_t. Then by using scan matching algorithm pose differences which are represented by Δf_t are calculated. Pose difference are used to calculate the control inputs, represented by u_t, which are used in the prediction step for each particle to estimate their states, ps_t^i and covariance matrix $p\Sigma_t^i$. According to estimated particle states, landmarks are extracted by using extracted features, and their state estimations and covariance matrices are stored in ls_t^i and $l\Sigma_t^i$. Then new extracted landmarks are associated with the ones in the memory of corresponding particle, lm_t^i and if a matching exists matched landmarks are stored in mls_t^i.

Measurement update step is active only if there is a matching case and when it is activated state of both particles and their landmarks are updated. For this purpose jacobian of landmark observation with respect to both particle state and landmark states are computed, J_t^i and used in the update steps. Finally weight of each particle, w_t^i is calculated according to jacobian and particle states and used while resampling the particles.

In Figure 3-1 implementation of FastSLAM algorithm is illustrated.

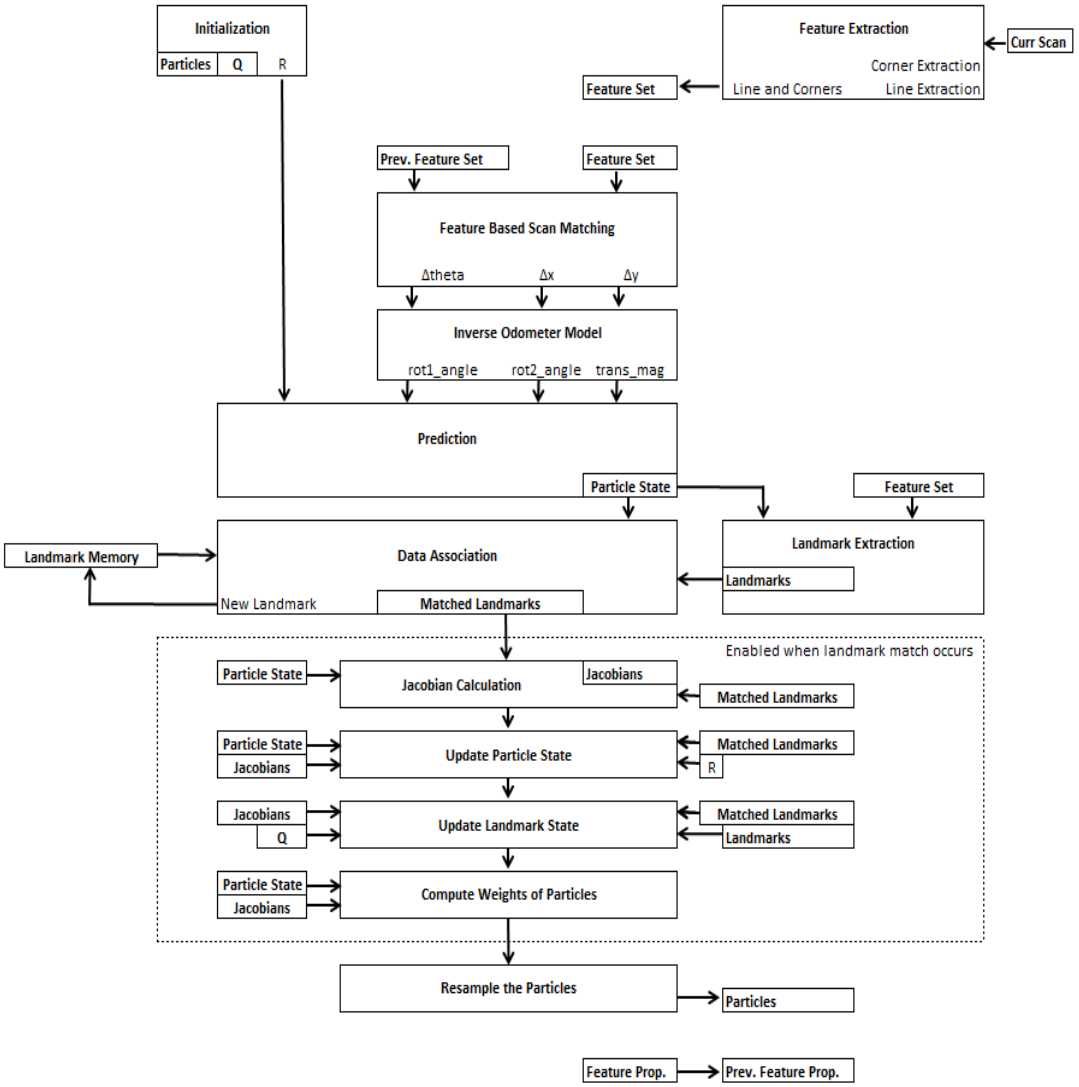


Figure 3-1 Implementation of FastSLAM algorithm

3.2.1. Feature Extraction

Feature extraction is an important part of the FastSLAM algorithm since extracted features are used in both control input calculation and landmark extraction. In this thesis, indoor environments are addressed and polygonal shapes are examined. Therefore, implemented feature extraction method uses raw sensor measurements to extract lines and corners along with their properties.

Nguyen et al. [40], compare different line extraction algorithms in their study and they achieve a result given at Table 3-5 . Considering the results at Table 3-5, Split

and merge algorithm with clustering has low complexity, high speed and accuracy while extracting line features. Therefore, split and merge with clustering, which is the first step in the method, is implemented to extract lines which are then used to extract corners.

Table 3-5 Comparison of various line extraction algorithms done by Nguyen et al. [40]

Algorithm	Complexity	Speed $[Hz]$	N.Lines	Correctness		Precision	
				TruePos [%]	FalsePos [%]	$\sigma_{\Delta r}$ $[cm]$	$\sigma_{\Delta\alpha}$ $[deg]$
Split-Merge + Clus.	$N \times logN$	1470	641	86.0	8.9	1.95	0.74
Incremental	$S \times N^2$	344	561	77.8	5.9	2.04	0.72
Incremental + Clus.		617	567	79.2	5.1	2.04	0.76
Line Regression	$N \times N_f$	364	577	76.4	10.1	1.99	0.80
LR + Clus.		384	562	75.8	8.4	1.97	0.79
RANSAC	$S \times N \times N.Trials$	29	749	75.6	31.5	1.68	0.77
RANSAC + Clus.		93	547	70.7	12.2	1.37	0.70
Hough Transform	$S \times N \times NC + S \times NR \times NC$	8	825	82.0	32.5	1.63	0.76
HT + Clus.		9	600	79.5	10.0	1.51	0.67
EM	$S \times N1 \times N2 \times N$	0.6	1153	78.6	53.7	2.09	0.97
EM + Clus.		0.7	709	80.3	23.1	1.58	0.73

3.2.1.1. Raw Sensor Data

In the present thesis, a laser scanner, which is SICK OEM1000 given in Figure 3-2, is utilized to obtain measurements from the environment.

Figure 3-2 Laser scanner used in the present thesis, SICK OEM1000

In each scan SICK OEM1000 provides range measurements from 360 degree area and its maximum range is approximately 80 meters. However, in the case of small

roll and pitch movements of robot, measurements taken from long distances are affected because of reflectance from floor or ceiling. Therefore, maximum range is limited to 10 meters for SICK OEM1000 in order to avoid a reflectance problem as much as possible. Moreover, by limiting the range of the sensor max range errors is avoided. Additionally, readings from 180 degree area which is in front of the robot are utilized in the configuration since there are fixed obstacles behind the robot which can cause problem for feature detection and so on. SICK OEM1000 updates its scan at 20Hz and outputs readings from its RS232 serial channel at 115200 baud rate.

3.2.1.2. Split and Merge Algorithm

Split and merge algorithm is one of the efficient and easy to implement line extraction method that commonly used in robotics. Technique can be explained in following steps

- **First Split Phase**: In this phase an adaptive clustering is utilized to create group similar points. This is done by computing distance between two consecutive points and then comparing the result with a threshold. If distance is greater than the threshold points are split into different regions, otherwise they are grouped together. Threshold is not fixed for all point pairs since density of the points is not uniform and that is why as range of the points increase, threshold is getting larger. This is called adaptive clustering. At the end of this phase, different regions are constructed with the points which are in the neighborhood of each other.
- **First Merge Phase:** In this phase small regions, which are constructed with a few points are eliminated. Then, regions which are closer to each other are detected by using differences between start and end points of each region, and a threshold which is larger than the one used in the first split phase. At the end of this phase updated regions are obtained.

In Figure 3-3 result of first split and merge phases is shown. From a real scan taken from SICK OEM1000, measurements are shown in Cartesian coordinate system in

which raw measurements are shown in blue and grouped measurements are shown in red. There are a few outlier points that are left out emphasized with red circles.

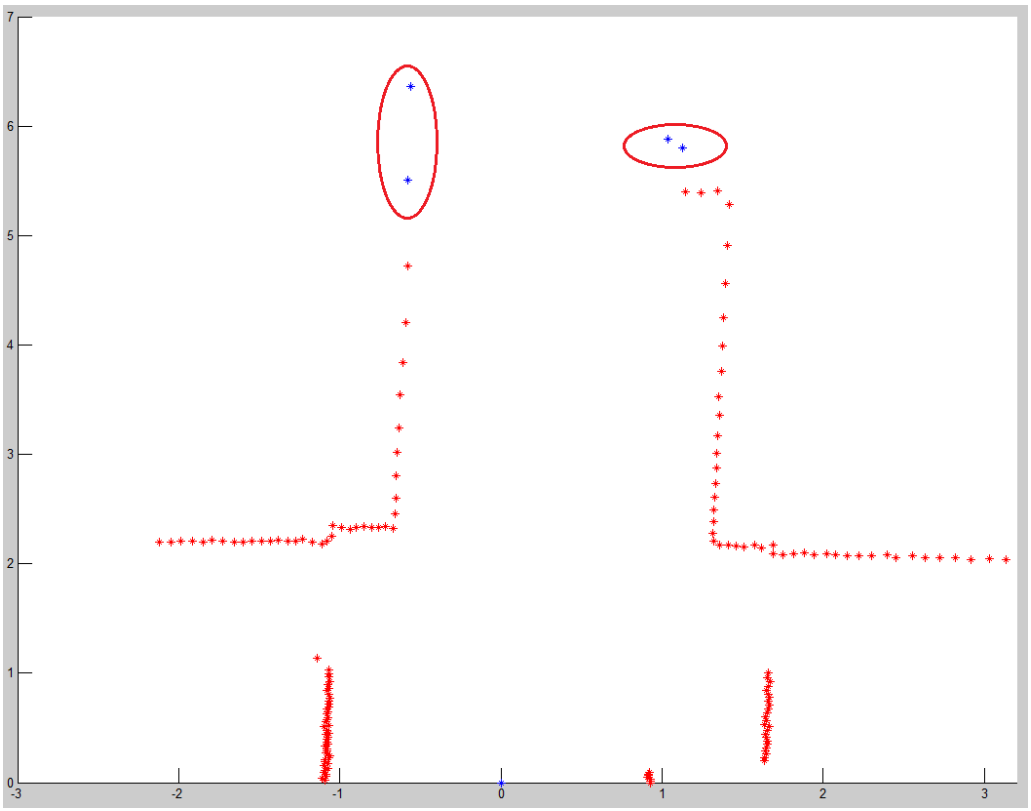


Figure 3-3 Result of first split and merge phases

- **Second Split Phase:** In this phase, points in the constructed regions are fitted on a best fit line which is done by using orthogonal distance from a point to a line. Among all points in a region the one with the largest orthogonal distance is found and compared with a threshold which specifies the split point. If orthogonal distance of this point is greater than the threshold region is split into two at this point. This is done recursively and at the end linear groups are obtained.

Let the equation of a straight line to be $y = mx + n$ and given k different points. Best line constructed with given points is achieved by least square method. In this method, initially mean of x and y values of each point are calculated.

$$x' = \frac{\sum_{i=1}^{k} x_i}{\mathrm{k}} \tag{3-27}$$

$$y' = \frac{\sum_{i=1}^{k} y_i}{\mathrm{k}} \tag{3-28}$$

Then slope of the best fit line is computed as

$$m = \frac{\sum_{i=1}^{k} (x_i - x')(y_i - y')}{\sum_{i=1}^{k} (x_i - x')^2} \tag{3-29}$$

Interception point at vertical axis is computed as

$$n = \mathrm{y}' - mx' \tag{3-30}$$

In order to test the line of best fit, randomly generated points are fed to the method. In Figure 3-4 blue stars are the randomly generated points and red line is the line of best fit for given points.

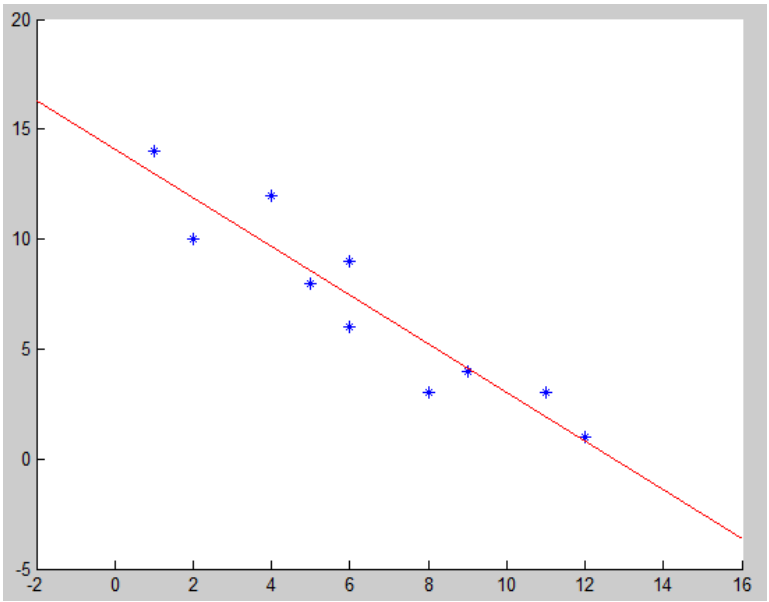


Figure 3-4 Result of line of best fit method

Figure 3-5 illustrates the second split phase of the algorithm. A split point which has the largest distance to the initially constructed line (shown in green) is found if exists. Then from split point regions is separated into two different regions. New lines are constructed for these regions and again split points for each region are searched. This is done iteratively and terminated when no split points are found for the regions. At the end, linear regions are obtained.

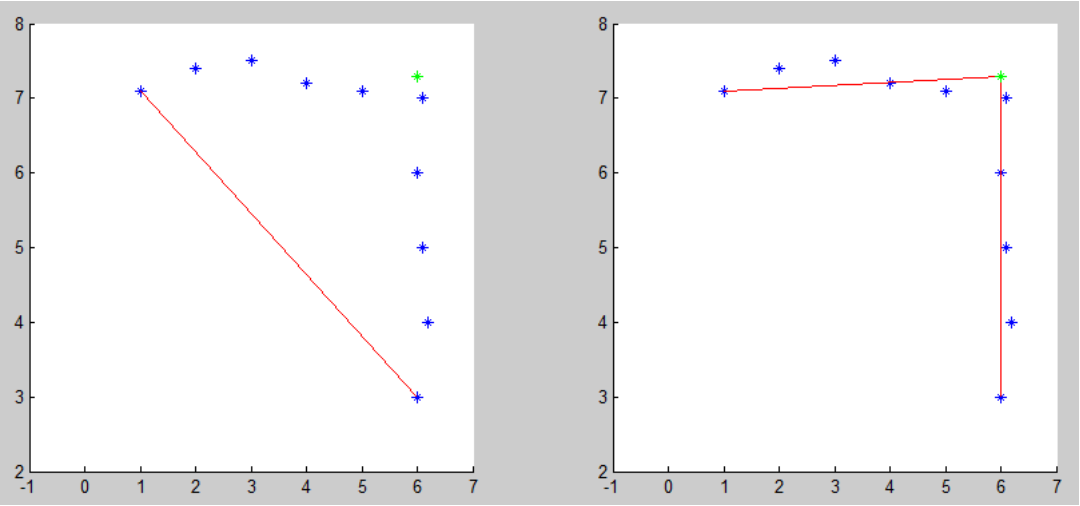


Figure 3-5 Second split phase of Split and Merge algorithm

- **Median Filter:** Median filter is generally used to reduce the noise in an image or signal. In here, it is used to reduce the deformations in the line regions so that better line features are extracted.
- **Second Merge Phase:** In this phase, at first small linear regions are eliminated. After that, two different procedure is activated which are extracting corners and merging linear regions. For corner extraction, slope difference between two consecutive linear regions is computed. If it is larger than a defined threshold, distance between those linear regions is checked. If linear regions are close to each other, then a corner is created with these two linear regions.

  For merging linear regions, slope difference between consecutive linear regions is compared to be smaller than a defined threshold. If so, again

the distance between the linear regions are checked and if regions are close to each other these two regions are combined.

Extracted lines and corners by split and merge are given in Figure 3-6.

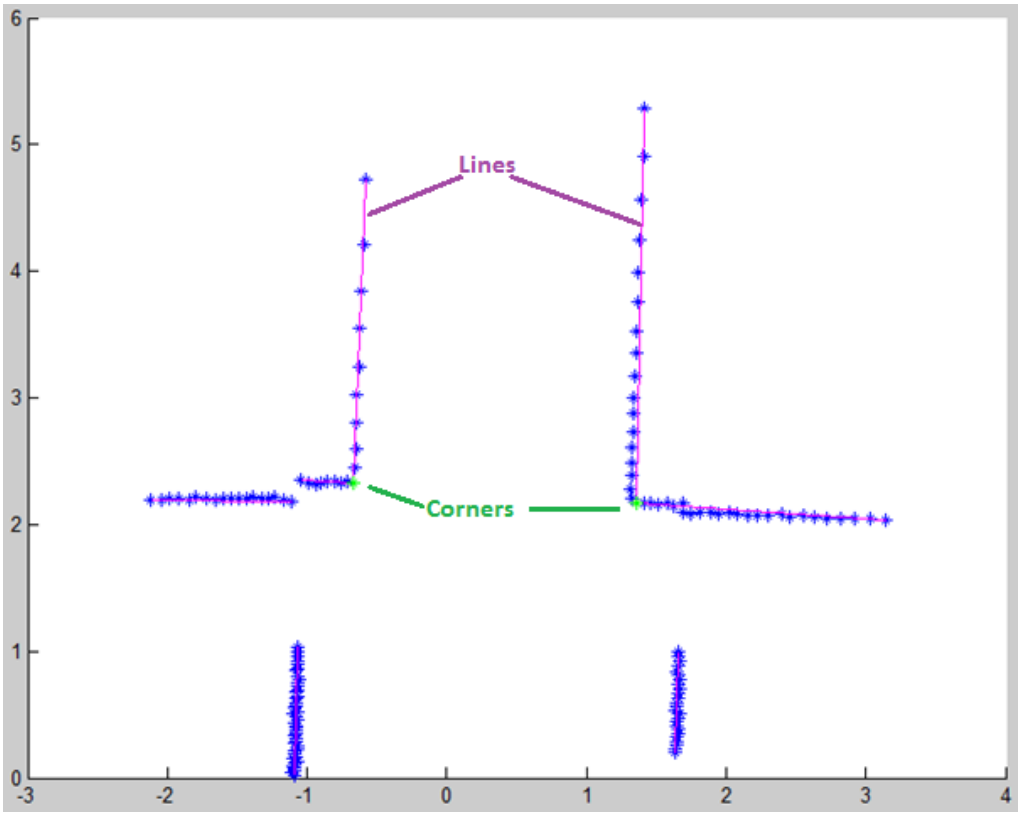


Figure 3-6 Extracted lines and corners by split and merge algorithm

3.2.1.3. Feature Property Extraction

As mentioned earlier, extracted features are used in both scan matching and landmark extraction methods. Therefore, features should carry useful properties that define them clearly. Selected properties for the features are as follows:

- **Line center point:** This property describes the position of the line in local frame.
- **Line angle:** This property describes the angle, or slope, of the line in local frame.

- **Line length:** This property describes length of the line. Length of a line provides information about the reliability of that line. Longer lines are regarded as more reliable than the smaller ones.
- **Angle difference between lines:** This property describes angle difference between each line and all remaining lines extracted in a local map. Also it provides differential information about the local map which is invariant from rotations and transformations.
- **Center point difference between lines:** This property describes the center point distances between each line and all remaining lines. This property also provides differential information about the local map and used in scan matching.
- **Both sided ended lines:** This is crucial information regarding the lines since being both side ended guarantees that line feature is completely seen in the scan. Therefore, other properties of the line become more reliable and this changes the weight given to this line when using its properties.

3.2.2. Scan Matching

In the present thesis, feature based scan matching method which has low computational complexity due to being a non-iterative process is utilized. Method basically calculates a relation between two consecutive local maps which are consist of line structures and provides pose differences as result.

3.2.2.1. Structure of Scan Matching

Implemented scan matching method uses geometric properties of the lines features to relate two consecutive local maps. Therefore, extracted features and their properties are sourced to this method as inputs. Moreover, it is assumed that implemented method can work properly with the features extracted from two local maps obtained at several time step differences. In other words, local maps that are obtained at different time steps, which differ in a large amount, provide low geometric relation which results in undesired pose difference calculations and eventually undesired control inputs for prediction step.

Scan matching can be summarized with following steps:

1. **Line Angle Matching:** In this step, aim is to detect angle pattern matches of line features between local maps. For that purpose, angle differences between line features of each local map are compared to each other. If angle difference between two line pairs is below a defined threshold then corresponding line matches gain weight.
2. **Line Center Point Distance Matching:** In this step same procedure mentioned in the first step applies by changing only angle difference with center point differences.
3. **Detecting both side ended lines**: In this step, there exists a list of line matches with their weight. All the lines in the list are checked whether they are both side ended and if so, weight multipliers of these lines are increased so that for the final weight calculation these lines can have higher weights. As mentioned before, both side ended lines are regarded as more reliable than other lines.
4. **Detecting long lines**: In this step, all the matched and both side ended lines are checked if their lengths are above a defined threshold so that longer lines which are regarded as reliable just like both side ended lines, can be detected and weight multiplier of these lines can be increased.
5. **Evaluations:** In the last step, weights of line matches are multiplied with the corresponding weight multipliers and final weights are calculated. After that, each line from a local map is matched with a line from the other local map which gives the highest weight. Finally, pose differences between each matched lines are calculated by using angle and center point differences. Final results are computed according to average of all pose differences with the weights coming from corresponding match.

Pseudo code of the scan matching algorithm is presented in Table 3-6**.**

Table 3-6 Pseudo code of implemented scan matching algorithm

1: **Algorithm Scan Matching**(f_t, f_{t-1})

2: $wf_t = 0$

3: $mwf_t = 0$

4: $matchedLines = \varnothing$

5: $\Delta f_t = 0$

6: **for** i = 1 to M do

7: $temp\_matchedWeight = 0$

8: $temp\_matchedLines = \varnothing$

9: **for** j = 1 to N do

10: **if** $abs(f_t^i.line\_angle\_diff - f_{t-1}^j.line\_angle\_diff) < Threshold$ **or**

11: $abs(f_t^i.center\_point\_dist - f_{t-1}^j.center\_point\_dist) < Threshold$

12: $wf_t \sim abs(f_t^i.line\_angle\_diff - f_{t-1}^j.line\_angle\_diff)$

13: $wf_t \sim wf_t + abs(f_t^i.center\_point\_dist - f_{t-1}^j.center\_point\_dist)$

14: $mwf_t = 0$

15: **if** f_t^i.both_side_ended

Table 3-6 (continued)

16: $mwf_t = mwf + BSE\_coeff$

17: **if** f_t^j.both_side_ended

18: $mwf_t = mwf + BSE\_coeff$

19: **end if**

20: **else**

21: **if** f_t^j.both_side_ended

22: $mwf_t = mwf + BSE\_coeff$

23: **end if**

24: **end if**

25: **if** f_t^i.line_length > line_length_threshold

26: $mwf_t = mwf + BSE\_coeff$

27: **if** f_t^j.line_length > line_length_threshold

28: $mwf_t = mwf + BSE\_coeff$

29: **end if**

30: **else**

Table 3-6 (continued)

31:	**if** f_t^j.line_length > line_length_threshold
32:	$mwf_t = mwf + BSE\_coeff$
33:	**end if**
34:	**end if**
35:	**end if**
36:	**if** $temp\_matchedWeight < mwf * wf_t$
37:	$temp\_matchedWeight = mwf * wf_t$
38:	$temp\_matchedLines = < i, j, temp\_matchedWeight >$
39:	**end if**
40:	**end for**
41:	$matchedLines = matchedLines + temp\_matchedLines$
42:	**end for**
43:	**for** j = 1 to N do
44:	$indexNumber =$ ***find*** $matchedLines(:,2)$ ***equal to*** j
45:	**if** $indexNumber > 1$

Table 3-6 (continued)

46: $indexNumberMax = \boldsymbol{find\ max\ of}\ matchedLines(indexNumber, 3)$

47: **delete** $indexNumber(indexNumberMax)$

48: **delete** $matchedLines(indexNumber)$

49: **end if**

50: **end for**

51: $sumOfWeights = \boldsymbol{sum\ all}\ matchedLines(:,3)$

52: **for** k = 1 to **rank**$(matchedLines, 1)$

53: $i = matchedLines(k, 1)$

54: $j = matchedLines(k, 2)$

55: $weight = matchedLines(k, 3)$

56: $\Delta f_t = \Delta f_t + \boldsymbol{pose\_difference}\left(f_t^i, f_t^j\right) * weight/sumOfWeights$

57: **end for**

58: **return** Δf_t

3.2.2.2. Advantages and Results

First advantage of the implemented feature based scan matching technique is low computational cost as mentioned earlier. One other advantage is utilizing differential feature properties which are more robust to translational and rotational errors. Moreover, including both side ended lines and length of the lines into the equation while calculating line matches also provides robustness against the possible measurement errors. It is possible that a couple of samples can be missed in a measurement which can cause extraction of two different line regions or a shorter line when compared to correct measurement case. Also, line regions that cannot be seen entirely are hard to match since their center point may not be calculated accurately.

In order to test the scan matching algorithm two local maps are taken and line features from these maps are extracted. Then extracted features are matched by using scan matching algorithm. Outcome of the algorithm which is pose difference of features is used to move the previously extracted features to see whether they can overlap with the currently extracted features. Result of this test can be seen in Figure 3-7.

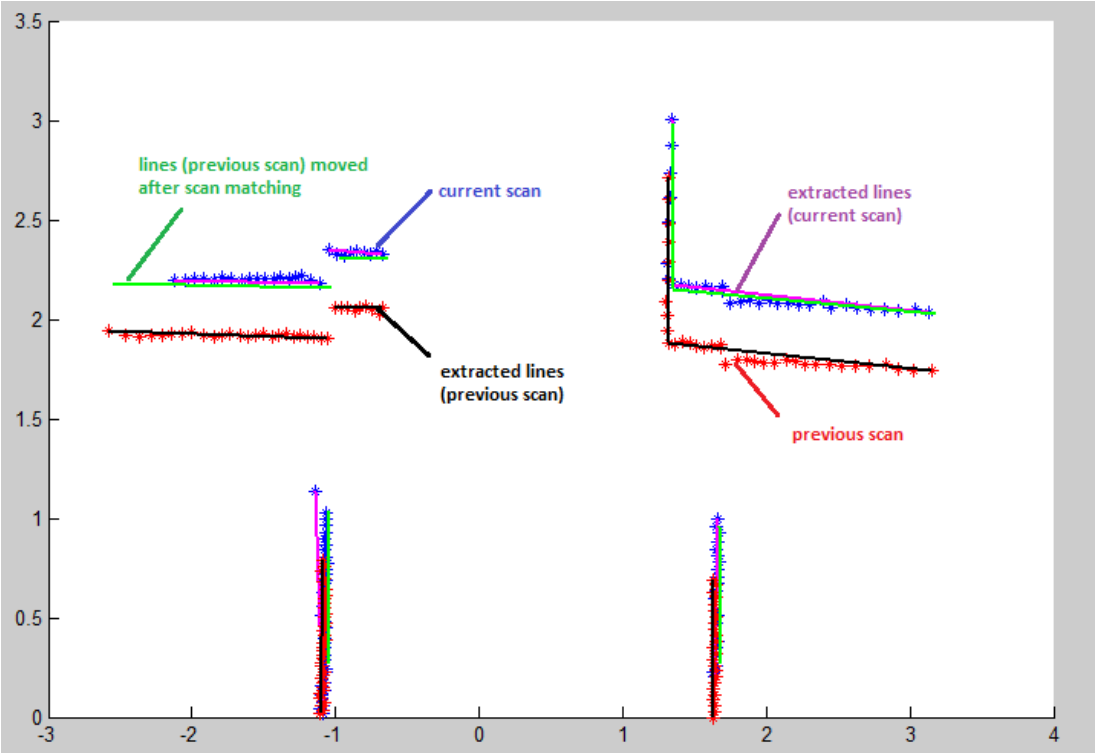


Figure 3-7 Test results of Scan Matching algorithm with two local maps

3.2.3. Inverse Odometer Motion

Inverse odometer motion constructs model for the motion of robotic agent by calculating rotation and translation values. This model takes pose difference and previous state estimate as inputs and creates control input for the robot which is critical for the prediction step as output. Motion of the robot is modelled as first rotation which adjusts the robot motion direction, then translation which defines position change in a linear path and finally second rotation which adjusts the final direction of the robot. As given in Figure 3-8 δ_{rot1} represents the first rotation, δ_{trans} represents the translation and δ_{rot2} represents the second rotation.

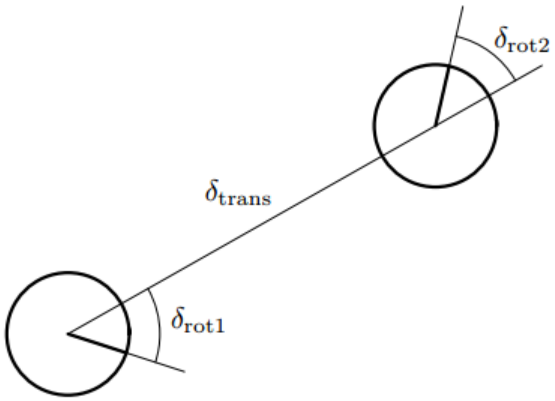


Figure 3-8 Odometer model for the robot motion

$\delta_{rot1}, \delta_{trans}$ and $\delta_{rot2}$are computed as follows

$$\delta_{rot1,t} = \arctan\left(\frac{\Delta y_t}{\Delta x_t}\right) - \theta_{t-1} \tag{3-31}$$

$$\delta_{trans,t} = \sqrt{\Delta {y_t}^2 + \Delta {x_t}^2} \tag{3-32}$$

$$\delta_{rot2,t} = \theta_t - \theta_{t-1} - \delta_{rot1,t} \tag{3-33}$$

3.2.4. Prediction

In this part, pose of the robot is estimated according to given previous state of the robot and control inputs. The state transition function is used to update the mean of previous state given as

$$ps_t^i = A_t ps_{t-1}^i + B_t u_t + \varepsilon_t \tag{3-34}$$

Covariance is updated by using only A_t since it considers that linear matrix A_t represents the dependence of current state to previous state.

Formulations for the odometer motion model are given as

$$ps_t^i = \begin{pmatrix} px_t^i \\ py_t^i \\ p\theta_t^i \end{pmatrix} \tag{3-35}$$

$$px_t^i = px_{t-1}^i + \delta_{trans,t}\cos(\delta_{rot1,t} + p\theta_{t-1}^i) \tag{3-36}$$

$$py_t^i = py_{t-1}^i + \delta_{trans,t}\sin(\delta_{rot1,t} + p\theta_{t-1}^i) \tag{3-37}$$

$$p\theta_t^i = p\theta_{t-1}^i + \delta_{rot1,t} + \delta_{rot2,t} \tag{3-38}$$

In prediction step, two different Jacobian matrixes are used namely, G_v which represents particle pose estimate with respect to particle state and $G_u$which represents particle pose estimate with respect to control inputs. Pseudo code of the prediction algorithm is presented at Table 3-7

$$G_{v,t}^i = \begin{bmatrix} \frac{\partial px_t^i}{\partial px_t^i} & \frac{\partial px_t^i}{\partial py_t^i} & \frac{\partial px_t^i}{\partial p\theta_t^i} \\ \frac{\partial py_t^i}{\partial px_t^i} & \frac{\partial py_t^i}{\partial py_t^i} & \frac{\partial py_t^i}{\partial p\theta_t^i} \\ \frac{\partial p\theta_t^i}{\partial px_t^i} & \frac{\partial p\theta_t^i}{\partial py_t^i} & \frac{\partial p\theta_t^i}{\partial p\theta_t^i} \end{bmatrix} \tag{3-39}$$

$$G_{u,t}^i = \begin{bmatrix} \frac{\partial px_t^i}{\partial \delta_{rot1,t}} & \frac{\partial px_t^i}{\partial \delta_{rot2,t}} & \frac{\partial px_t^i}{\partial \delta_{trans,t}} \\ \frac{\partial py_t^i}{\partial \delta_{rot1,t}} & \frac{\partial py_t^i}{\partial \delta_{rot2,t}} & \frac{\partial py_t^i}{\partial \delta_{trans,t}} \\ \frac{\partial p\theta_t^i}{\partial \delta_{rot1,t}} & \frac{\partial p\theta_t^i}{\partial \delta_{rot2,t}} & \frac{\partial p\theta_t^i}{\partial \delta_{trans,t}} \end{bmatrix} \tag{3-40}$$

Table 3-7 Pseudo code of implemented prediction algorithm

1: **Algorithm Prediction**$(ps_{t-1}^{i}, p\Sigma_{t-1}^{i}, u_t)$

2: $$ps_t^i = \begin{pmatrix} px_{t-1}^i + \delta_{trans,t}\cos(\delta_{rot1,t} + p\theta_{t-1}^i) \\ py_{t-1}^i + \delta_{trans,t}\sin(\delta_{rot1,t} + p\theta_{t-1}^i) \\ p\theta_{t-1}^i + \delta_{rot1,t} + \delta_{rot2,t} \end{pmatrix}$$

3: $$G_{v,t}^i = \begin{bmatrix} 1 & 0 & -\delta_{trans,t}\sin(\delta_{rot1,t} + p\theta_{t-1}^i) \\ 0 & 1 & \delta_{trans,t}\cos(\delta_{rot1,t} + p\theta_{t-1}^i) \\ 0 & 0 & 1 \end{bmatrix}$$

4: $$G_{u,t}^i = \begin{bmatrix} -\delta_{trans,t}\sin(\delta_{rot1,t} + p\theta_{t-1}^i) & \cos(\delta_{rot1,t} + p\theta_{t-1}^i) & 0 \\ \delta_{trans,t}\cos(\delta_{rot1,t} + p\theta_{t-1}^i) & \sin(\delta_{rot1,t} + p\theta_{t-1}^i) & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

5: $$p\Sigma_t^i = G_{v,t}^i p\Sigma_{t-1}^i {G_{v,t}^i}^T + G_{u,t}^i p\Sigma_{t-1}^i {G_{u,t}^i}^T$$

6: **return** ps_t^i

3.2.5. Landmark Extraction

Landmarks have important role in the approaches to the SLAM problem. They are regarded as geometric beacons and mostly used for robust localization. Landmarks can be classified in two categories namely, structured landmarks and natural landmarks. Structured landmarks have specific indicators, specific color or a signal, that carry information about themselves which can be used to identify the landmark or locate it directly. Identifying a landmark from its indicators also eliminates the data association problem which is difficult to deal with. GPS systems are good examples for this type of landmarks. However structured landmarks are not always available in an unknown environment. Especially for the indoor environments where

communication with an external positioning system is prohibited, such as GPS, the structured landmarks are not available.

Natural landmarks, on the other hand, do not have any special indicators and mostly they are extracted from the environment. Therefore, directly accessing to a landmark is not a concern for this case. However, extraction of landmarks is a difficult problem to handle since extracted landmarks should have indicative properties so that wrong associations which can cause algorithm to diverge can be avoided.

In this thesis, landmarks are extracted by using the line features extracted earlier for the scan matching method. Geometric structures constructed by these lines are first examined in order to see that they are proper to be used as landmarks. At this point, corners and both side ended lines are the key features to select proper geometric structures. Then distinct properties of these geometric structures are extracted and stored for each structure for better description and recognition. These distinct properties can be listed as follow:

- Number of corner features in the structure
- Position of the corner features
- Slope difference between line features that construct corner features.
- Slope difference between both side ended lines which do not construct corner features.
- Position of the center points of both side ended lines.
- Distance difference between center points of both side ended lines.

In Figure 3-9, properties that are extracted from a landmark are given.

For a structure to be a landmark it is necessary that this structure should be detected several times. Number of successive detection is determined at the initialization phase. This approach prevents the usage of wrong landmarks and protects the algorithm from diverging.

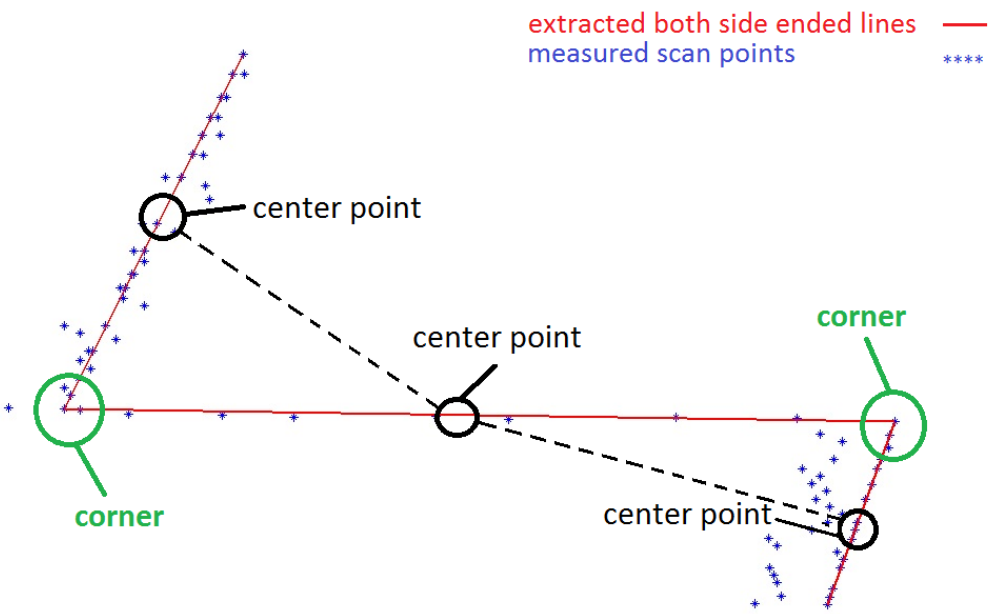


Figure 3-9 Property extraction from a structure detected as landmark

3.2.6. Data Association

This part mainly handles all the work related to landmarks. There are two memories one of which stores landmarks that have been seen several times and the other one stores the temporary landmarks that are either new or seen recently. Data association part controls these two memories both of which include landmark states along with their distinct properties.

Data association part is fed with newly extracted landmarks at each step. These landmarks are first compared with the ones in the memory that keeps accepted landmarks, in other words true landmarks. Comparison is not only done over states of the landmarks, but the extracted properties that define the landmarks distinctly are also compared. As a result of this comparison it is decided that either extracted landmark is previously observed and now it is observed again or this landmark is seen recently.

In the case that extracted landmark is seen recently it is compared with the ones in the second memory which includes temporary landmarks. If a match is found in the

second memory then the number of observation for corresponding landmark is incremented. When this number reaches initially determined limit then related landmark is transferred to the first memory and afterwards it is regarded as a true landmark.

If a landmark is observed for the first time then it is initialized and stored in the second memory with other temporary landmarks. Initialization is done by computing the state of the landmark and storing its state along with its properties. Landmark state is initialized by first sampling the state of the particles according to their proposal distribution and then adding position of both extracted corners and center point of the lines. Afterwards, center point of total feature is calculated to obtain the state of the new landmark. In Figure 3-10 a landmark initialization is given.

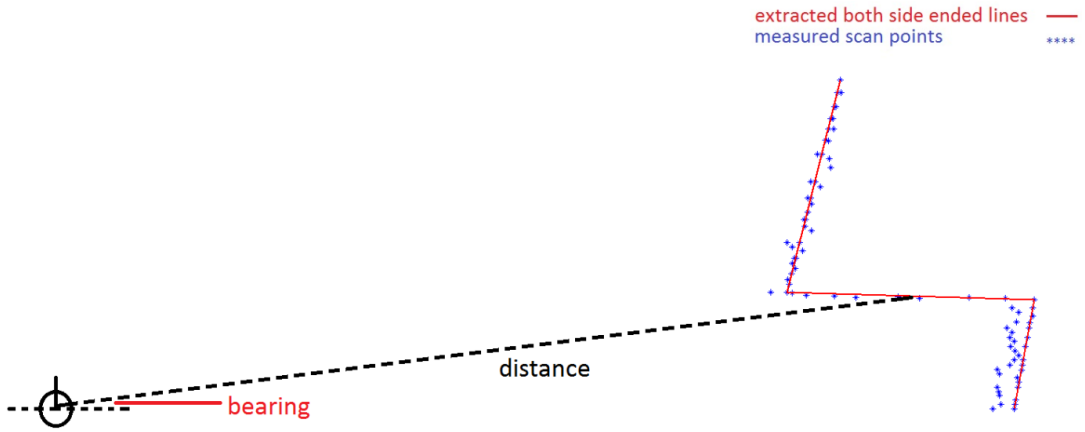


Figure 3-10 Landmark initialization by using distance and bearing information from robot to the center point of total feature.

Landmark state and covariance are calculated as follows;

State:

$$ls_t^{i,n} = \begin{pmatrix} lx_t^{i,n} \\ ly_t^{i,n} \end{pmatrix} \tag{3-41}$$

$$lx_t^{i,n} = px_t^i + r_t * \cos(p\theta_t^i + \theta_t) \tag{3-42}$$

$$ly_t^{i,n} = py_t^i + r_t * \sin(p\theta_t^i + \theta_t) \tag{3-43}$$

Where;

$$ps_t^i = \begin{pmatrix} px_t^i \\ py_t^i \\ p\theta_t^i \end{pmatrix} \tag{3-44}$$

$$z_t = \begin{bmatrix} r_t \\ \theta_t \end{bmatrix} \tag{3-45}$$

Covariance:

$$l\Sigma_{\mathrm{t}}^{\mathrm{i}} = H_{v,t}^i Q {H_{v,t}^i}^T \tag{3-46}$$

Where;

$$H_{v,t}^i = \begin{bmatrix} \frac{\partial d_t^i}{\partial px_t^i} & \frac{\partial d_t^i}{\partial py_t^i} & \frac{\partial d_t^i}{\partial p\theta_t^i} \\ \frac{\partial b_t^i}{\partial px_t^i} & \frac{\partial b_t^i}{\partial py_t^i} & \frac{\partial b_t^i}{\partial p\theta_t^i} \end{bmatrix} \tag{3-47}$$

$$d_t^i = \sqrt{(lx_t^{i,n} - px_t^i)^2 + (ly_t^{i,n} - py_t^i)^2} \tag{3-48}$$

$$b_t^i = \arctan\left(\frac{(ly_t^{i,n} - py_t^i)}{(lx_t^{i,n} - px_t^i)}\right) - p\theta_t^i \tag{3-49}$$

For the case that a landmark in the memory is re-observed, a matching table is constructed. This table carries information about matching landmarks and the parts

of the landmarks that are matched specifically. This table enables robust matching when only a limited part of the landmark is observed since even if there is no direct match between new and previously seen landmarks, a partial match can be established. Different parts of a landmark are tracked by the number of corners and both side ended lines since they are the base structures of a landmark.

When there is a matching case, in other words matching table is not empty, measurement update state of the FastSLAM algorithm is activated.

3.2.7. Jacobian Computation

In this part of the algorithm, following Jacobians are calculated:

- Predicted landmark position with respect to vehicle state

$$H_{v,t}^{i} = \begin{bmatrix} \frac{\partial d_t^i}{\partial px_t^i} & \frac{\partial d_t^i}{\partial py_t^i} & \frac{\partial d_t^i}{\partial p\theta_t^i} \\ \frac{\partial b_t^i}{\partial px_t^i} & \frac{\partial b_t^i}{\partial py_t^i} & \frac{\partial b_t^i}{\partial p\theta_t^i} \end{bmatrix} \tag{3-50}$$

  Where d_t^i is the distance b_t^i is the bearing information of the predicted landmark observation $zp = [d_t^i, b_t^i]$. This Jacobian is used while computing proposal distribution in particle update.

- Predicted landmark position with respect to landmark state

$$H_{l,t}^{i} = \begin{bmatrix} \frac{\partial d_t^i}{\partial lx_t^{i,n}} & \frac{\partial d_t^i}{\partial ly_t^{i,n}} \\ \frac{\partial b_t^i}{\partial lx_t^{i,n}} & \frac{\partial b_t^i}{\partial ly_t^{i,n}} \end{bmatrix} \tag{3-51}$$

This Jacobian is used while computing proposal distribution in both particle and landmark update.

3.2.8. Particle Update

In this part, states of the particles are updated according to observed landmarks which enable to overcome the incremental errors coming from platform noise. The pseudo code for particle update step is given at Table 3-8

Table 3-8 Pseudo code of implemented particle update algorithm

1:	**Algorithm** $\boldsymbol{particle\_update}(ps_t^i, p\Sigma_t^i, J_t^i)$
2:	$v = z_f^i - z_p^i$
3:	$S_t^i = \left(H_{l,t}^i \, l\Sigma_t^i \, {H_{l,t}^i}^T\right) + Q$
4:	$p\Sigma_t^i = ({H_{v,t}^i}^T {S_t^i}^{-1} H_{v,t}^i + {p\Sigma_t^i}^{-1})^{-1}$
5:	$ps_t^i = ps_t^i + (p\Sigma_t^i \, {H_{v,t}^i}^T {S_t^i}^{-1} \, v)$
6:	**return** $[ps_t^i, p\Sigma_t^i]$

3.2.9. Landmark Update

In this part, states of the landmarks are updated according to observed landmark which is actually known as measurement update step of the Kalman Filter. At first Kalman gain computed and it is used to determine the correctness of observed landmarks. Therefore state of the landmark is updated by using Kalman gain. The pseudo code of the landmark update step is given at Table 3-9

Table 3-9 Pseudo code of implemented landmark update algorithm

1:	**Algorithm** $\boldsymbol{landmark\_update}(ls_t^i, l\Sigma_t^i, mls_t^i, J_t^i)$
2:	$v = z_f^i - z_p^i$
3:	$K_t^i = (ls_t^i \, {H_{l,t}^i}^T \, (H_{l,t}^i \, ls_t^i \, {H_{l,t}^i}^T + R)^{-1})$
4:	$ls_t^i = ls_t^i + K_t^i \, v$
5:	$l\Sigma_t^i = (I - K_t^i \, H_{l,t}^i) l\Sigma_t^i$
6:	**return** $[ls_t^i, l\Sigma_t^i]$

3.2.10. Particle Weight

In order to obtain information about the credibility of each particle, importance weights are calculated for all particles. This weight actually represents the consistency of a particle state by using the difference between observed and predicted landmark states. Weights of particles are important for the re-sampling part of the algorithm where the undesired and corrupted particles are eliminated.

It is important to calculate particle weights appropriately so that problems like particle degeneracy can be avoided. Weights should be neither distributed along all particles in a uniform fashion nor collected by only one particle. In this both cases undesired particle diversity is achieved.

The pseudo code of the particle weight calculation is given at Table 3-10

Table 3-10 Pseudo code of the implemented particle weight algorithm

1:	**Algorithm** $\boldsymbol{particle\_weight}(p\Sigma_t^i, J_t^i)$
2:	$v = z_f^i - z_p^i$
3:	$N = size(v)$
4:	$V = v^T$
5:	$S_t^i = \left(H_{l,t}^i \; l\Sigma_t^i \; {H_{l,t}^i}^T\right) + Q$
6:	$T_t^i = \left(H_{v,t}^i \; p\Sigma_t^i \; {H_{v,t}^i}^T\right) + S_t^i$
7:	$w_t^i = \frac{\sqrt{(2\pi^N)\lvert T_t^i \rvert}}{e^{-1\frac{1}{2}V^T {T_t^i}^{-1} V}}$
8:	**return** w_t^i

3.2.11. Particle Re-Sampling

In this part, particle set is updated by using a re-sampling approach known as Roulette Wheel. The aim is basically to obtain particles with higher weights so that correction against noises and errors can be robust. Alongside with a good weight calculation, re-sampling is also important to have good particle diversity. In this aspect selecting particles only with the highest weights is not the solution. Re-sampling approach should provide an updated particle set which should have a various particles with weights high enough to be trusted.

The pseudo code of the particle re-sampling is given at Table 3-11

Table 3-11 Pseudo code of implemented particle resampling algorithm

1: **Algorithm** $\boldsymbol{particle\_resampling}(X_t)$

2: $N = particle\ number$

3: $w\_sum = size(w_t^i)$

4: $startPoint\ = rand(1,1) * 1/N$

5: $steps = 0{:}\frac{1}{N}{:}(1-\frac{1}{N})$

6: $randomPoints = startPoint + steps$

7: $boundary = 0$

8: $currParticleIndex = 1$

9: $particleSetIndex = 1$

10: $selectedParticles = zeros(N, 1)$

11: $\boldsymbol{while}\ currParticleIndex! = N + 1\ \boldsymbol{and}\ particleSetIndex\ ! = N + 1$

12: **if** $boundary \leq randomPoints(particleSetIndex)\boldsymbol{and}$ $randomPoints(particleSetIndex) < boundary + w_t^{particleSetIndex}$

13: $selectedParticles(particleSetIndex) = currParticleIndex$

14: $particleSetIndex = particleSetIndex + 1$

Table 3-11 (continued)

15:	**else**
16:	$boundary = boundary + w_t^{particleSetIndex}$
17:	$currParticleIndex = currParticleIndex + 1$
18:	**end if**
19:	**end while**
20:	$X_t = X_t(selectedParticles)$
21:	**return** X_t

3.2.12. Occupancy Grid Mapping

Reconstruction of the environment for each particle according to the sampled particle state and measurements is achieved in occupancy grid mapping. As the name implies environment is reconstructed by using grid cells in which probability for observing an obstacle in that particular cell is stored. This probability values in grid cells are updated whenever they are in the perceptual field of the robot, in other words selective updating, so there is no unnecessary computation of unseen grid cells.

Occupancy grid mapping works based on inverse sensor model which provides the occupancy likelihood of grid cells that are in the measured area. Model is described as follows:

$$inverseSensorModel(m_i, x_t, z_t) = log \frac{p(m_i|z_t, x_t)}{1 - p(m_i|z_t, x_t)} \quad (3\text{-}52)$$

Implementation of the occupancy grid mapping and inverse sensor model approaches are given with the pseudo codes in Table 3-12 and Table 3-13.

Table 3-12 Pseudo code of implemented occupancy grid mapping algorithm

1: **Algorithm** $\boldsymbol{occupancyGridMapping}(X_t, m_{t-1}, z_t)$

2: $\boldsymbol{for}$ *each cell* c_i

3: $\boldsymbol{if}$ c_i *is in perceptual field of* z_t

4: $m_{t,i} = m_{t-1,i} + \boldsymbol{inverseSensorModel}\left(c_i, ps_t^i, z_t\right) - m_0$

5: $\boldsymbol{else}$

6: $m_{t,i} = m_{t-1,i}$

7: $\boldsymbol{end\ if}$

8: $\boldsymbol{end\ for}$

9: $\boldsymbol{return}\ m_t$

Where;

$$m_0 = \log\frac{\mathrm{p}(c_i)}{1-\mathrm{p}(c_i)} \tag{3-53}$$

Table 3-13 Pseudo code of implemented inverse sensor model

1: **Algorithm** $\boldsymbol{inverseSenorModel}(c_i, ps_t^i, z_t)$

2: $[x_i, y_i] = \ center\ of\ mass\ of\ c_i$

3: $r = \sqrt{(x_i - x)^2 + (y_i - y)^2}$

4: $\phi = atan2(y_i - y, x_i - x) - \theta$

5: $k = argmin_j |\phi - \theta_{j,sens}|$

6: $\boldsymbol{if}\ r > \min\left(z_{max}, z_t^k + \frac{\alpha}{2}\right) or\ |\phi - \theta_{k,sens}| > \frac{\beta}{2}$

7: $\boldsymbol{return}\ m_0$

8: $\boldsymbol{end\ if}$

9: $\boldsymbol{if}\ z_t^k < z_{max}\ and\ |r - z_{max}| < \frac{\alpha}{2}$

10: $\boldsymbol{return}\ m_{occ}$

11: $\boldsymbol{end\ if}$

12: $\boldsymbol{if}\ r < z_t^k$

13: $\boldsymbol{return}\ m_{free}$

14: $\boldsymbol{end\ if}$

Where; α is thickness of the obstacles and β is the width of the sensor beam.

3.3. Chapter Summary

In this chapter, implementation of FastSLAM and feature based scan matching algorithm are presented. Measurements taken from the environment are first used in feature extraction algorithm to obtain line and corner features. Afterwards these features are fed to the scan matching approach to obtain the pose difference of the robot. Pose difference is used to calculate possible robot poses from the current robot ones by using proposal distributions of each particle. Then, from the extracted distinct features landmarks are constructed and at each step they are compared to the ones stored previous steps. When a match occurs between current landmarks and previously observed ones, both particle and landmark states are updated by using the difference between predicted and observed landmark positions. This difference is also used to establish the weights of each particle which are used for re-sampling the particles to work with more credible ones. Afterwards occupancy grid map of each particle is updated according to new states of particles and measurements obtained from the environment.

CHAPTER 4

MAP GROWING

The previous chapter focused on the first step of the study that is exploration of the environment by obtaining its 2D maps using SLAM algorithm. From this point on we will utilize the obtained 2D maps to reconstruct the environment in 3D. Towards this aim, the obtained maps should be matched and aligned with respect to each other in order to obtain correct correspondences and proper alignment between them. There are various ways to achieve these goals, but the problem is to find a solution which is robust that has less computational cost. The obtained maps are grid maps as mentioned in the previous chapter where each cell is labeled by a probability value. In these maps, assignment of the probability values can have errors which cause noise in the probabilities of the cells which eventually demolishes the similarities between the maps and makes it difficult to match them. In order to overcome the effect of noises and properly match the maps, we prefer to use map matching that is based on key features, lines and corners, extracted from maps and generated transformation and rotation matrices between the maps. Line features are first used to obtain an initial matching then corner features are used iteratively to optimize the matching.

Thereafter, the aligned maps are ready to create plane surfaces for the reconstruction of structures in the environment. Here, one important point is to determine whether or not to utilize prior knowledge about the environment since using prior knowledge can bring ease to the solution. Models can be obtained according to prior knowledge about the environment and reconstruction of the structures can be achieved with respect to these models. However, in this thesis, our environment is unknown and there is no prior knowledge available. In this case reconstruction of the structures is achieved with a novel approach called smooth map growing. In this novel technique

correspondence points are created for the structures at each level where local maps are SLAM generated for that level and those maps show similar characteristics when 2D maps are compared. Correspondence points are then generated according to these similarities boundaries of the similar regions are used. Dissimilarities in the maps are utilized as correspondence points in the form of vertices of tree structures constructed inside these regions. Then these points are tiled, which is connected, to each other according to the adopted triangulation method and 3D surfaces are reconstructed.

In the first part of this chapter, the theoretical background about tree construction is provided. Then, details of the steps that are described above, namely; creating correct correspondences, generating connection points and tiling of connection points are presented.

It is important to note that utilized data in this chapter is mostly taken from experimental results.

4.1.Theoretical Background

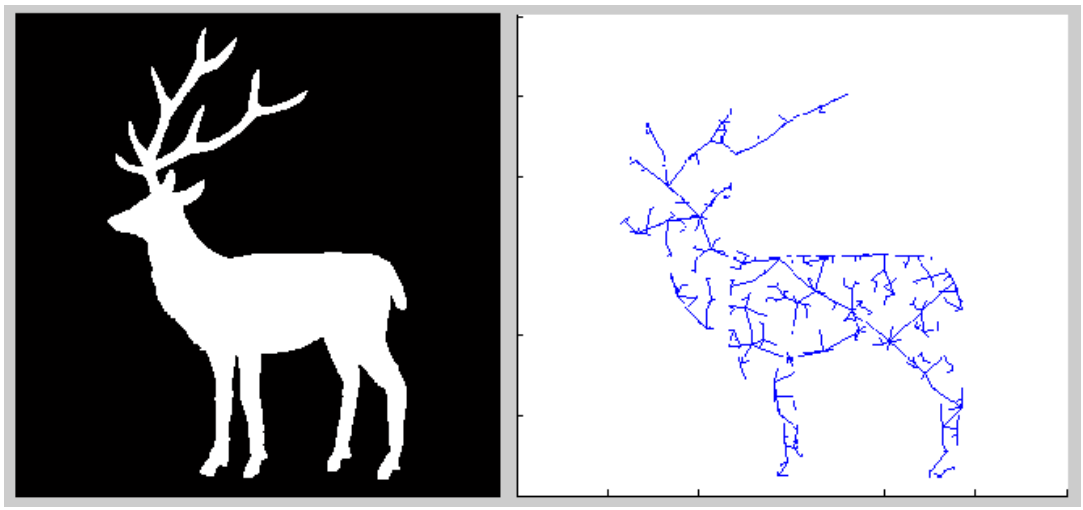
Tree structures provide solutions for various applications such as visualizing and organizing information hierarchically, path planning and area exploration methods for mobile robots. Tree construction methodologies can also differ in terms of being regular or non-regular i.e. randomized. Nevertheless the idea is common for utilizing the tree structures that is establishing a root and defined-length branches connected to the root. Branches can be used either to fill an area or present a hierarchy.

In this part of the present thesis, theoretical background for constructing randomized tree structures inside a region or shape in 2D is explained. Constructed trees consist of set of points which are sampled from a corresponding shape and their connections, i.e. branches, between each other are established such that obtained structures are used to define the region adequately.

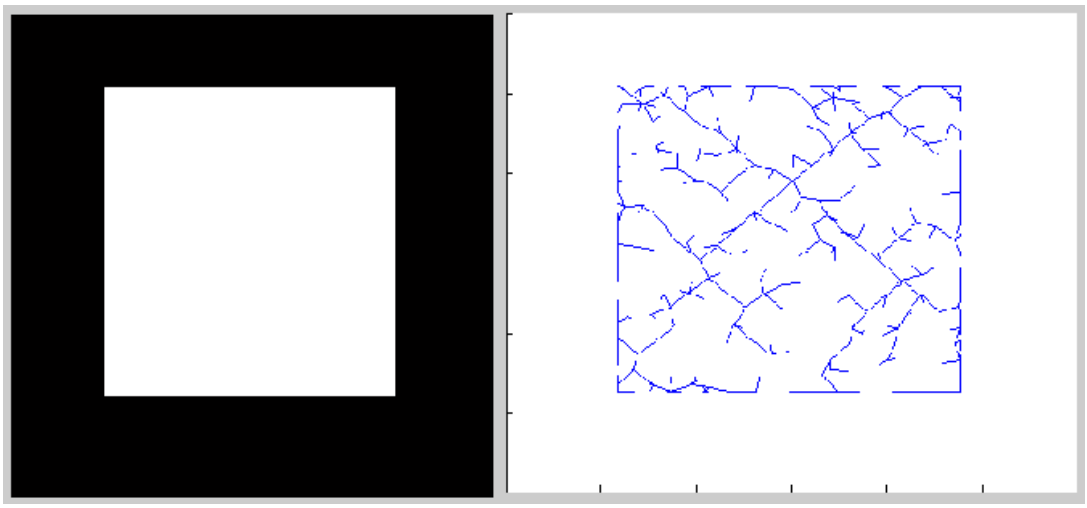
Important point is to have enough sample points so that these points can be used to define the 2D structure or shape. Using few points may not represent a shape definitely while using excessive amount of points can be inefficient to represent a shape. Connections between these points are also defined in a skeleton in order to avoid connections cross passing the boundary of the shape.

As previously mentioned, in this study randomized trees are utilized and they are generated by using Rapidly-exploring Random Trees (RRT) introduced by LaValle [41] and initially designed for generating solutions for wide range of path planning problems. Normally RRT has start and goal points and it builds connection between these points by randomly generating points at each step and go towards to that randomly generated point with a defined step size. Iteratively it builds a space-filling tree and generates a path on the tree which connects defined start point to goal point.

In this study, however, there is no defined goal point but instead the aim is to explore a structure with finite number of iteration and step size which are dynamically adjusted with respect to the size of the structure in the method. Structures or regions in which trees are constructed are extracted from the 2D grid maps represented with m_t obtained as result of SLAM algorithm. Grid maps are like grey scale images and can be converted to binary image to work on them easily. Therefore, in this study RRT implementation takes binary images as inputs and generates tree structures inside them. In Figure 4-1 RRT implementation results for covering different binary shapes are presented. As mentioned previously, the aim is to generate adequate sample points (which are going to be used as connection points later on) that cover inside of the structure properly.



(a)



(b)

Figure 4-1 RRT implementation results on different shapes

Pseudo code of RRT implementation is given in Table 4-1.

Table 4-1 Pseudo code of RRT implementation

1: **Algorithm** $\boldsymbol{RRT}(m_t, iterationNumber, stepSize)$

2: $[x, y] = [round(rand(\,)\;\; round(rand(\,)\,)]$

3: $\boldsymbol{while}\ [x, y] is\ not\ inside\ m_t$

4: $[x, y] = [round(rand(\,)\;\; round(rand(\,)\,)]$

5: $\boldsymbol{end\ while}$

6: $tree = [x, y]$

7: $treeSize = 1$

8: $edgeConnection = []$

9: $\boldsymbol{for}\ i < iterationNumber$

10: $[x, y] = [round(rand(\,)\;\; round(rand(\,)\,)]$

11: $[x_{dist}, y_{dist}] = tree - [x, y]$

12: $closestPointIndex = \min(x_{dist} + y_{dist})$

13: $currentPoint = tree(closestPointIndex)$

14: $\boldsymbol{for}\ j < stepSize$

15: $[x_{temp}, y_{temp}] = \boldsymbol{take\ a\ step\ from}\ currentPoint\ \boldsymbol{to}\ [x, y]$

Table 4-1 (continued)

16: $\boldsymbol{if}\ \left[x_{temp}, y_{temp}\right]\ is\ inside\ m_t$

17: $currentPoint = \left[x_{temp}, y_{temp}\right]$

18: $\boldsymbol{end\ if}$

19: $j = j + 1$

20: $\boldsymbol{end\ for}$

21: $tree = [tree; currentPoint]$

22: $treeSize = treeSize + 1$

23: $edgeConnection = [edgeConnection; [closestPointIndex\ treeSize]]$

24: $i = i + 1$

25: $\boldsymbol{end\ for}$

26: $\boldsymbol{return}\ tree, edgeConnection$

4.2. Creating Correct Correspondences

There are two main goals presented in this part of the thesis; one is segmentation of meaningful regions from maps and the other one is alignment of maps which are obtained from different height levels of the environment. In Figure 4-2 two maps from different height levels are presented. The left one is taken from lower height

level than the right one and from here on they are regarded as below map and above map.

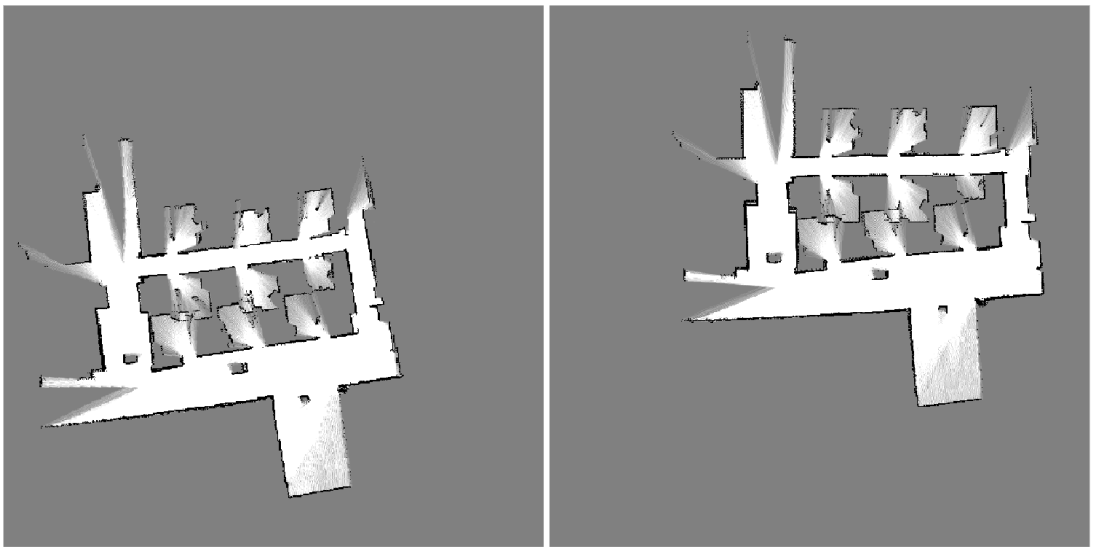


Figure 4-2 Two grid maps from different height levels. The map on the left is taken from lower height. The map on the right is taken from higher height.

Segmentation process aims to divide the map into semantically meaningful regions which is achieved by Connected Component Labeling (CCL) method utilizes spatial coherence between the grid cells and their values. However, in order to use CCL grid values should be binary or at least a threshold should be set to decide if two grid is connected or not. In this aspect grid maps are regarded as grey scale images and converted to binary images by using a threshold of initial probability value of a grid. In Figure 4-3 conversion results of the maps given in Figure 4-3 to binary format is presented.

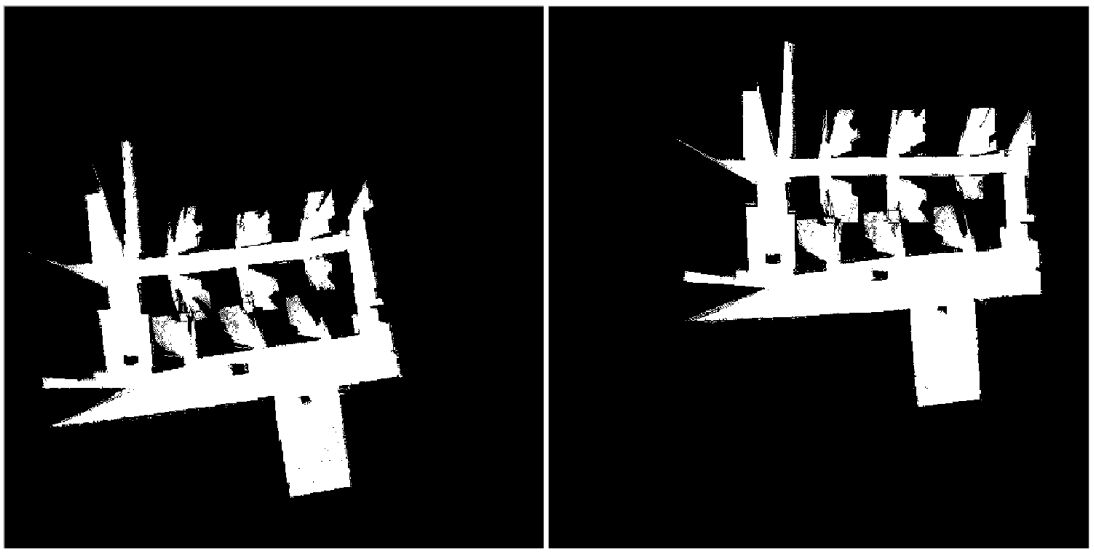


Figure 4-3 Binary image version of the below map (left) and above map (right)

Afterwards CCL is applied to those binary images and various regions from the grid maps are obtained. While applying CCL, small regions, possibly created because of noises, are eliminated. CCL results are given in Figure 4-4. In the top left image in Figure 4-4, which is the original image, all of the segmented regions are observed together. In the other images, segmented regions are presented separately. In the top right image the walls of the environment and empty spaces, cubical rooms, at the top of the environment are seen. In the remaining images, regions which are inside the environment can be observed.

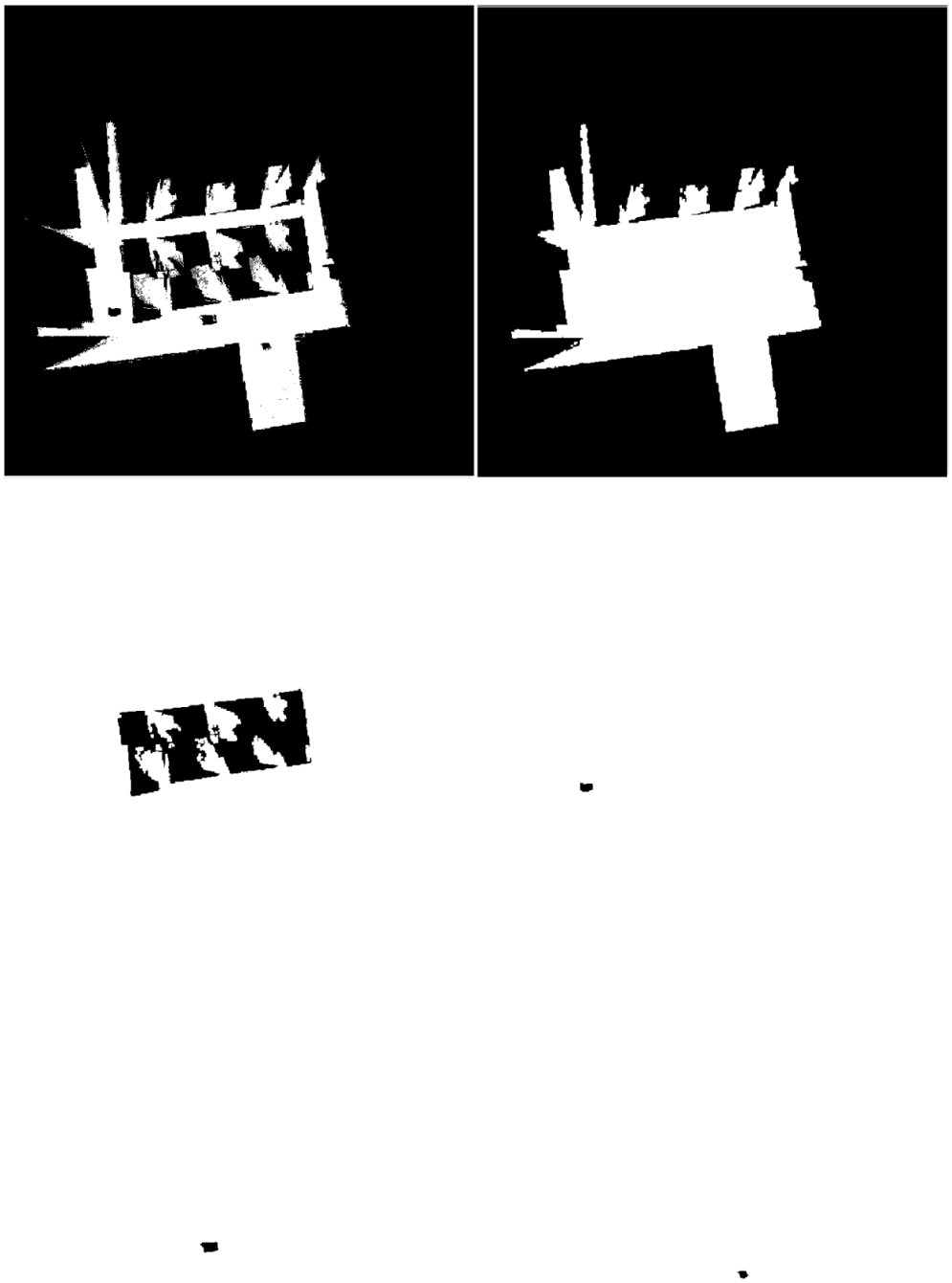


Figure 4-4 CCL results. Original image (top left), segmented regions

Pseudo code of Connected Component Labeling is given in Table 4-2.

Table 4-2 Pseudo code of connected component labeling

```
1:  Algorithm CCL(BW)
2:      labeledImage = zeros(size(BW))
3:      mergedLabels = [ ]
4:      nextLabel = 1
5:      for each grid in BW
6:      if BW_{i,j} = 1
7:          if BW_{i-1,j} = 1
8:              labeledImage(i, j) = labeledImage(i − 1, j)
9:              if BW_{i,j-1} = 1
10:                 if labeledImage(i, j − 1) != labeledImage(i − 1, j)
11:                 mergedLabels =
                    [mergedLabels; labeledImage(i, j − 1), labeledImage(i − 1, j) ]
12:                 end if
13:             end if
14:         elseif BW_{i,j-1} = 1
15:             labeledImage(i, j) = labeledImage(i, j − 1)
```

Table 4-2 (continued)

```
16:                 elseif BW_{i+1,j} = 0 and BW_{i,j+1} = 0
17:                     BW_{i,j} = 0
18:                 else
19:                     labeledImage(i,j) = nextLabel
20:                     nextLabel = nextLabel + 1
21:                 end if
22:             end if
23:             relabel labeledImage w.r.t. mergedLabels
24:         return BW, labeledImage
```

Segmentation is an important step for aligning different maps since it eliminates undesired features caused by mostly noise. Also out of the segmented regions key features are extracted and matched. These regions are also used as different layers of same structure while growing the maps into 3D.

After segmentation, next step is aligning the maps with respect to each other. At first, key features which are chosen to be lines and corners are extracted from each map. Then, line features are used with feature matching method for an initial alignment between the maps after which corner features are used with ICP method to optimize the alignment. Additionally line and corner features are used while determining similarity of regions on the maps whose details are explained later in this part.

Line features are extracted by applying split and merge method from the boundaries of regions. This method is introduced previously at “3.2.1.2. Split and Merge Algorithm” part of present thesis. Before applying split and merge algorithm, boundary points of regions are extracted and then obtained point sets are used in the algorithm. As output of split and merge algorithm line structures along with their properties whose details are provided at “3.2.1.3 Feature Property Extraction” part of the present thesis are obtained. In Figure 4-5 and Figure 4-6 results of the steps through line feature extraction are presented.

In the top left image in Figure 4-5, binary format of the below map is presented. In the top right image in Figure 4-5, result of the boundary extraction for each region located inside the binary map is shown. In the bottom left image in Figure 4-5, boundary point sets are obtained from the previously extracted boundaries of each region. In the bottom right image in Figure 4-5, line features are extracted by applying split and merge method to the obtained boundary point sets.

In the top left image in Figure 4-6, binary format of the above map is presented. In the top right image in Figure 4-6, result of the boundary extraction for each region located inside the binary map is shown. In the bottom left image in Figure 4-6, boundary point sets are obtained from the previously extracted boundaries of each region. In the bottom right image in Figure 4-6, line features are extracted by applying split and merge method to the obtained boundary point sets.

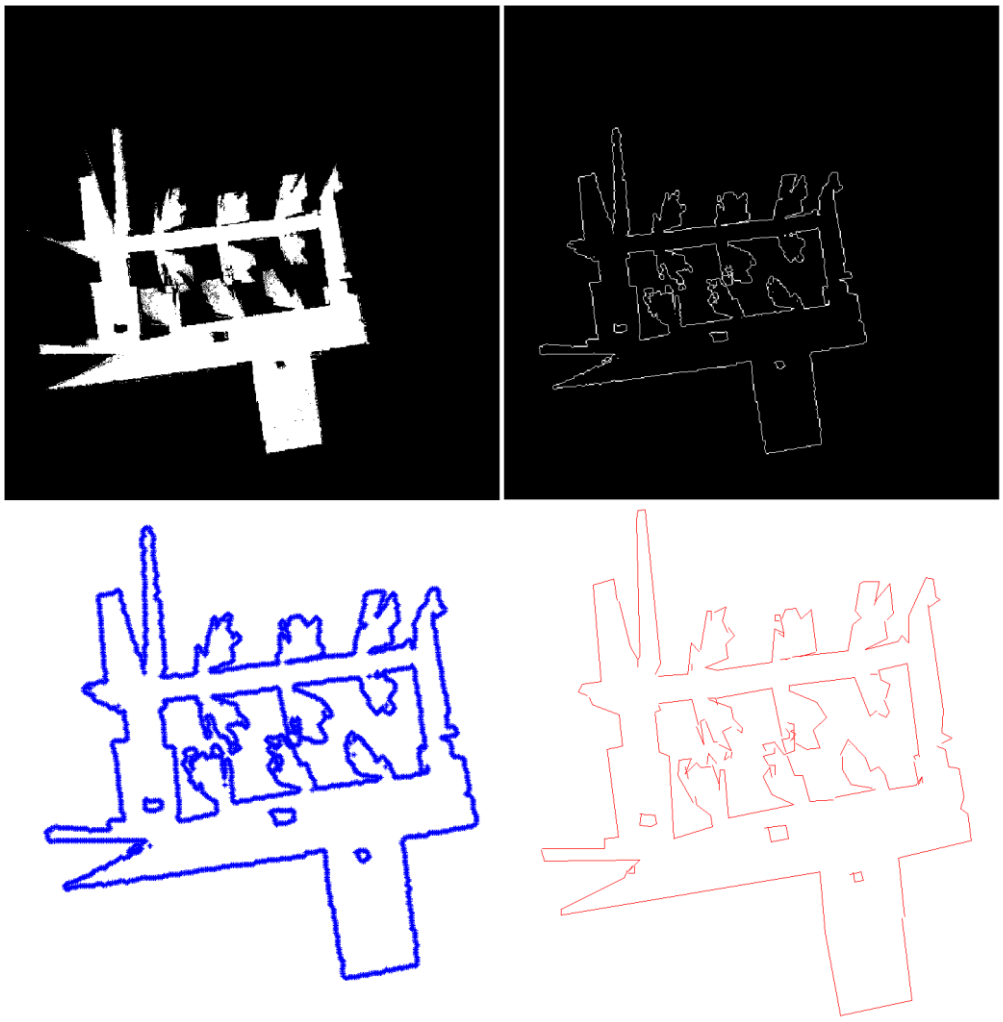


Figure 4-5 Results of the steps through line feature extraction for below map. Original image (top left), boundary extraction result (top right), boundary points extraction result (bottom left), line extraction result (bottom right)

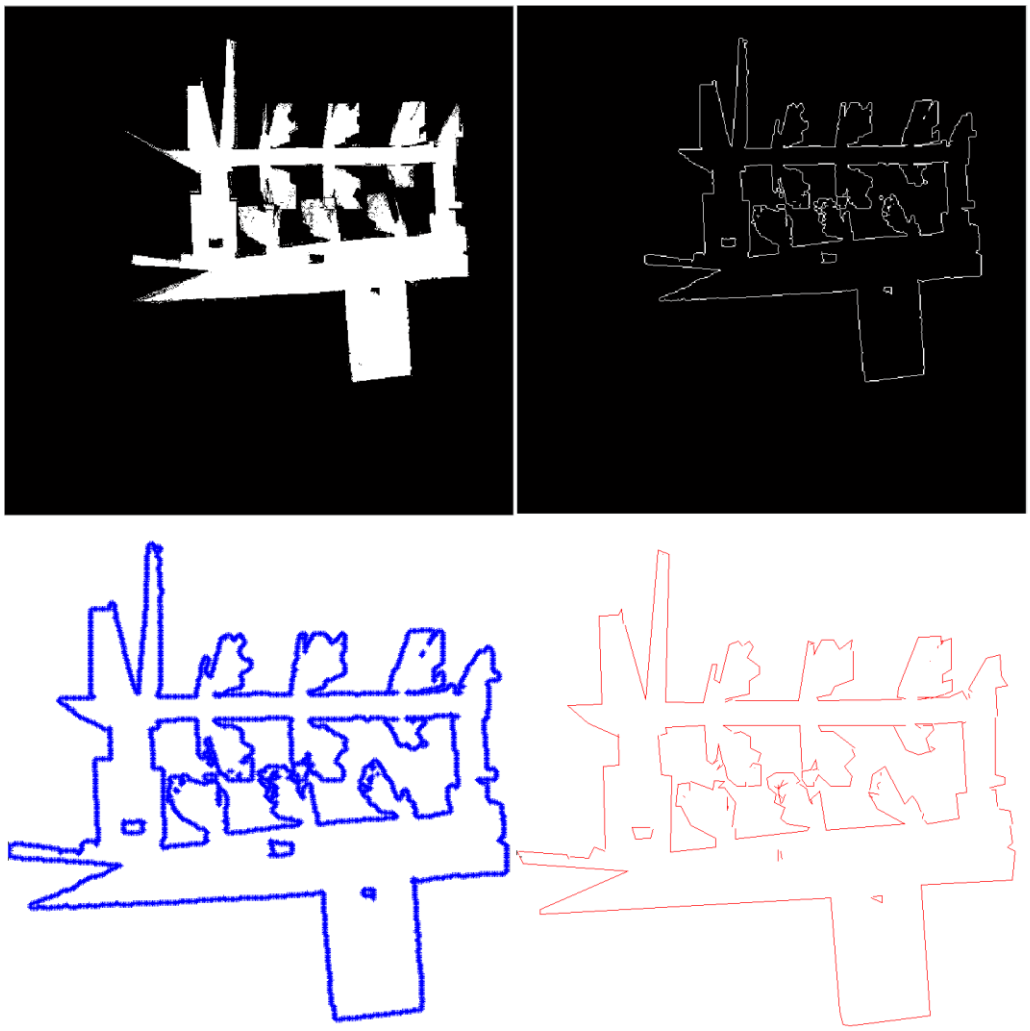


Figure 4-6 Results of the steps through line feature extraction for above map. Original image (top left), boundary extraction result (top right), boundary points extraction result (bottom left), line extraction result (bottom right)

Even all the lines at boundary of the regions are extracted successfully some of them are credible for matching and alignment purposes. In order to select those credible ones properties of the lines are utilized. To begin with, all line are both sided ended lines, so this property which is used earlier in the scan matching algorithm as a property for reliability does not provide desired information. On the other hand, length of the lines can be used here for credibility measure since the possibility of obtaining a long and disrupted line is lower than having a short and disrupted line as a result of noisy data. Therefore, lines longer than the mean length of the total line features are selected to be used in line matching. In Figure 4-7 extracted lines for

both maps (below and above) and the selected long lines (red lines) are presented. In this figure one can understand the reason for selecting and using longer lines in scan matching by observing the abrupt changes in the directions of short lines which do not provide healthy information.

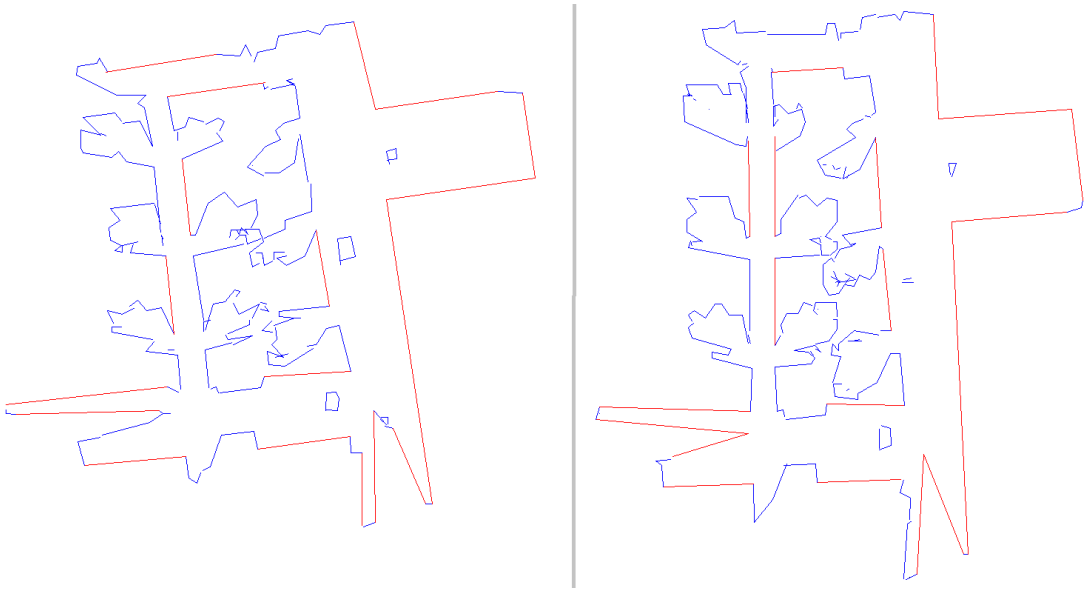
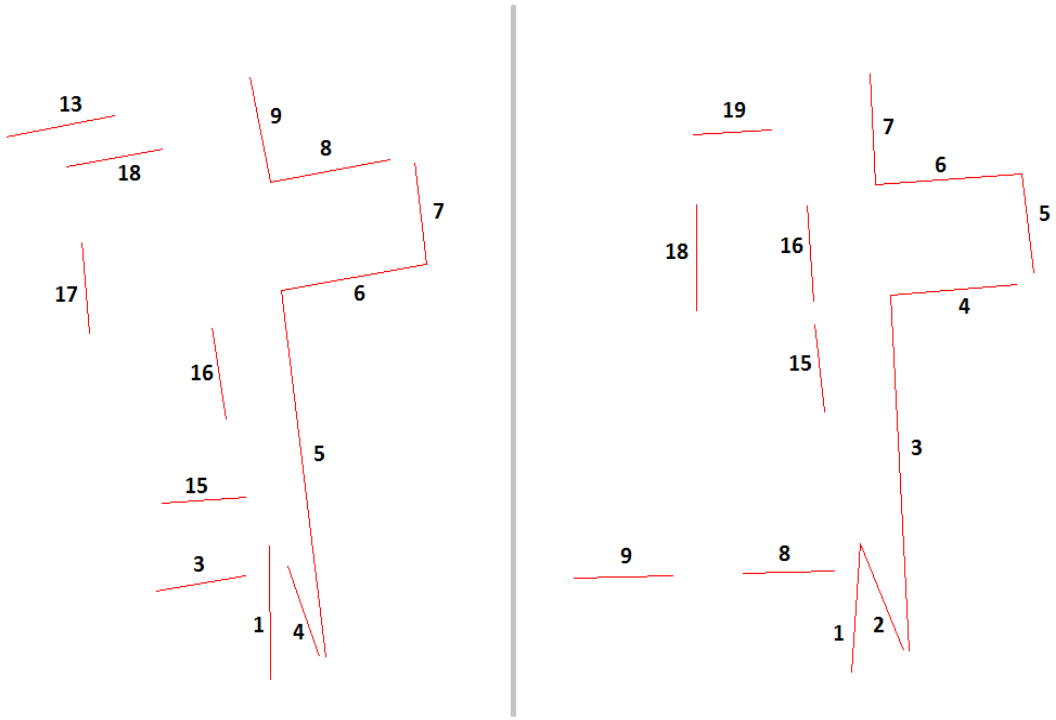


Figure 4-7 Longer line features that are selected to be used in line matching

After selecting reliable lines, angle and center position differences between the line features in each map are utilized for matching. It is important to point out that these properties are differential for the maps so that they do not change with rotation or translation motions. In Figure 4-8 matched lines are presented along with their line numbers and matches from the other map.



line_Matches =

1	1
3	8
4	2
5	3
6	4
7	5
8	6
9	7
13	9
15	16
16	15
17	18
18	19

Figure 4-8 Matched lines with their line numbers and corresponding matches

When the lines from different maps are matched, differences between slope angles and center positions are used to determine the rotation and translation matrices that can change the position and orientation of one map to align with the other one. In Figure 4-9 initial positions of the maps are presented while in Figure 4-10 positions of the maps after line matching are presented. It can be seen that, for the maps which are not aligned and distant from each other, line matching can be used to get maps closer and obtain an initial alignment. This alignment may not be a final result but it is easy to optimize the alignment from this point.

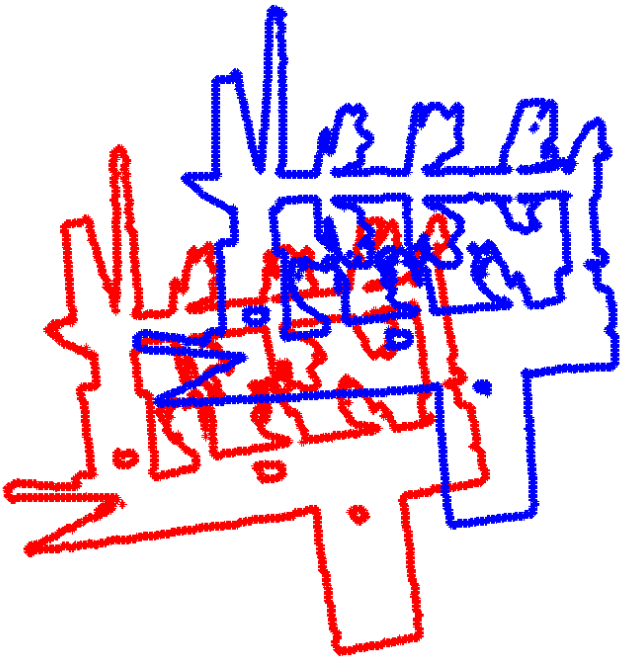


Figure 4-9 Initial positions of the maps with respect to each other (red one is below map and blue one is the above map)

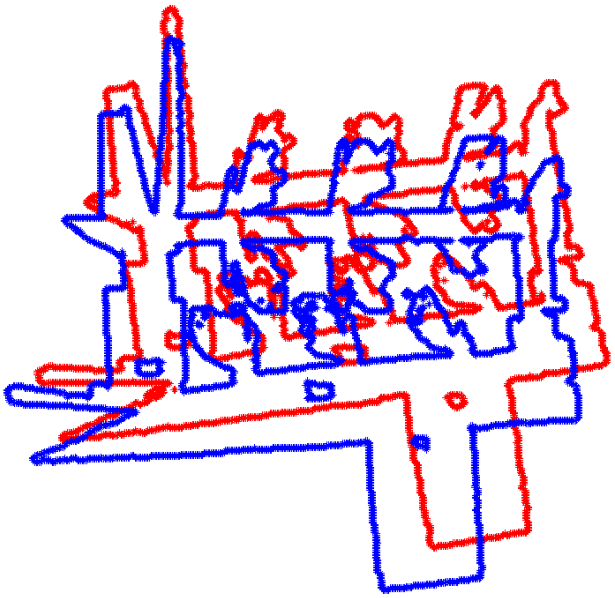


Figure 4-10 Positions of the maps after line matching result (red one is below map and blue one is the above map)

After achieving an initial alignment between maps, corner features are extracted from the boundaries of the segmented regions and used in the second alignment

process which optimizes the alignment. At this point, corner feature extraction is achieved by using Harris-Stephens corner detection algorithm [42]. Detected corner points by the mentioned corner detection method are shown on the binary maps in Figure 4-11. Afterwards extracted key points are matched by utilizing Iterative Closest Point (ICP) method. This method does not only match the point sets but also finds the transformation and rotation matrices that define the pose difference between two point sets. ICP method is explained in detail through three main steps in [43], namely projection of laser scan, data association and position estimation.

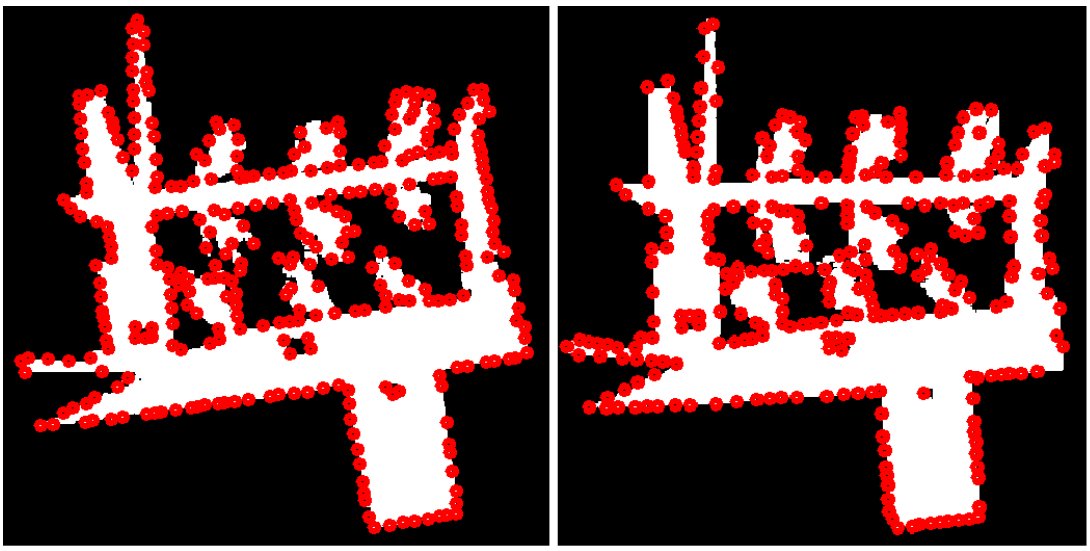


Figure 4-11 Detected corner points shown over binary grid maps.

Result of the ICP algorithm for corner feature sets of the maps is shown in Figure 4-12. It can be seen from Figure 4-12 that even most of the parts are aligned with respect to each other, there are still some regions which are not aligned. This region of misalignment is encircled in Figure 4-13. The main reason for obtaining such results is using SLAM based maps which can have localization errors and the use of probabilistic values while generating occupancy grid maps. Moreover, irregularities between 2D SLAM maps obtained from different heights are encountered frequently.

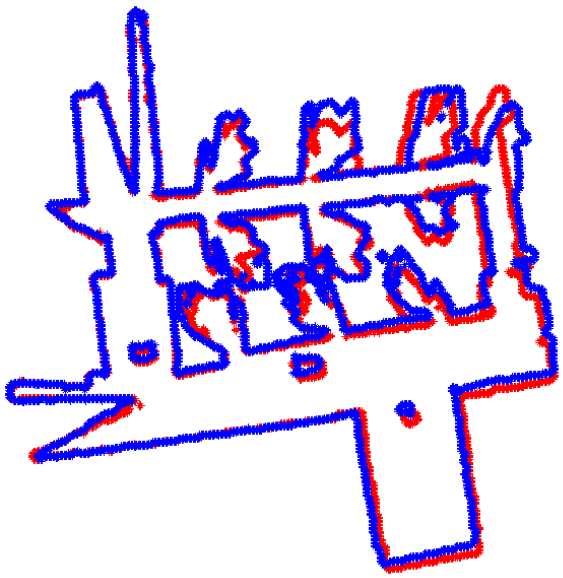


Figure 4-12 Result of ICP algorithm for corner point sets of the maps

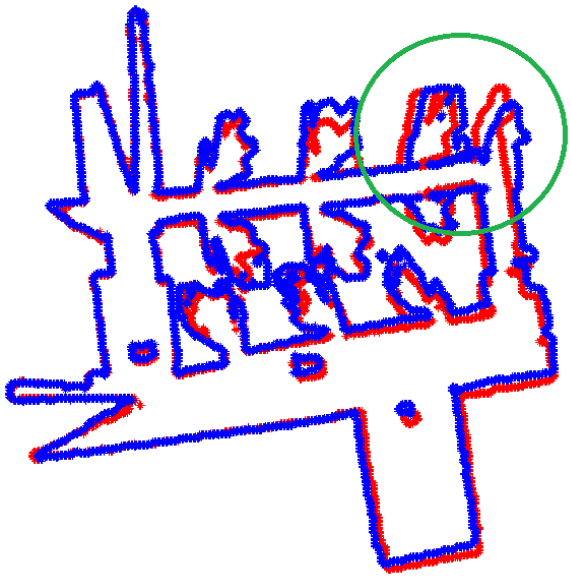


Figure 4-13 Regions that are not completely aligned in the result of ICP

Consequently, the ICP algorithm generates translation and rotation matrices that are used to evaluate the pose differences between two maps. Figure 4-14 illustrates the alignment of maps using translation and rotation matrices.

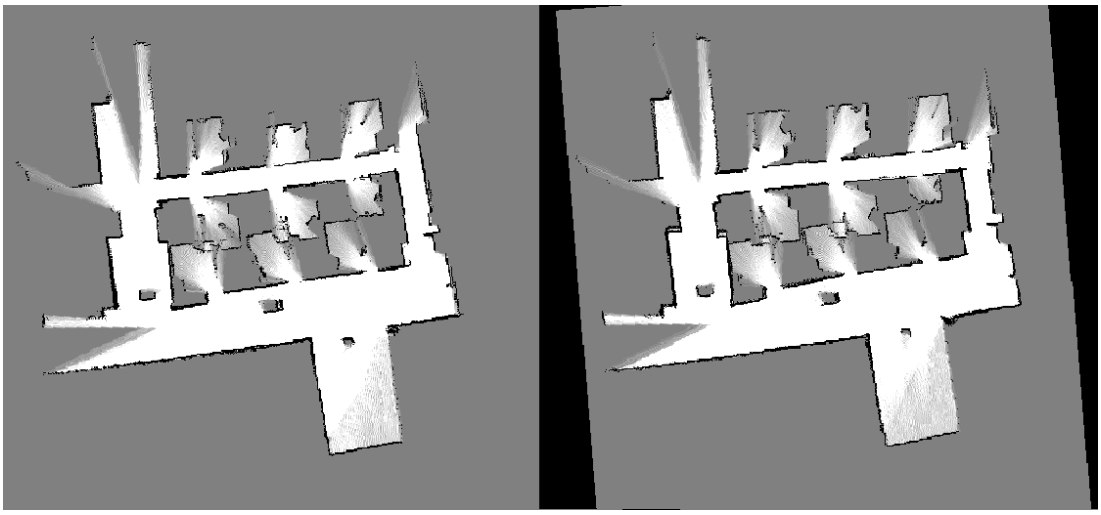


Figure 4-14 Alignment result of the maps (scale of the maps does not change)

4.2.1. Separation of Regions

Aligned maps include regions segmented earlier that reveal similar or dissimilar characteristics when compared to each other. Determination of such similarities between the regions is important since it effects the selection of appropriate approach to be used while establishing connections between maps.

First of all, it is necessary to define similarity measure used for determining similar and dissimilar regions. Previously alignment between grid maps that are constructed for different height levels of the environment is achieved which also means that regions determined on these grid maps are aligned. In the light of this information, regions from different height levels are available to be used in grid by grid comparison. At this point, it is important to point out that there are always mismatches between regions, so perfect matches between different regions cannot be obtained. However, the aim is to obtain a matching, or overlapping, over a certain percentage, which is configured as a parameter, with respect to total area size to claim that two regions are similar. The reason for using percentage value and not a defined number for grid matching is working independently from area size of the regions. Also, it should be noted that while comparing two regions computation of matching percentage is done according to larger regions so that if larger one

encapsulating the smaller one this can be still regarded as a dissimilarity if matching percentage is small.

Comparison between the regions that belong to previously aligned 2D maps taken at subsequent slices (below and above maps) is achieved and result is presented in Table 4-3 and Table 4-4. In Table 4-3 number of matching grids is presented, while in Table 4-4 the ratio between number of matching grids and total number of grids that belong to larger region is presented. When given two tables are examined together it can be seen that, for some regions the number of matching grids are high and this gives high matching percentage result for these regions. The aim of using these numbers is to find such regions by the help of a defined threshold, designated as a parameter, and take these regions as similar and the others as dissimilar regions.

Table 4-3 Result of the comparison between regions from below map and above map in number of grids

		Above Map Region Sizes															
		288999	**8**	**6**	**14332**	**9**	**5**	**2**	**4**	**8**	**89**	**4**	**2**	**7**	**167**	**188**	**61**
Below Map Region Sizes	**290085**	286234	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	14	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	15852	0	0	0	12919	0	5	2	3	0	69	0	0	0	0	0	0
	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	148	0	0	0	0	0	0	0	0	0	0	0	0	0	112	0	0
	232	0	0	0	0	0	0	0	0	0	0	0	0	0	0	128	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12

Table 4-4 Percentage of matching grid number to larger region

		Above Map Region Sizes															
		288999	8	6	14332	9	5	2	4	8	89	4	2	7	167	188	61
Below Map Region Sizes	290085	98.672	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	14	0.001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	15852	0	0	0	81.5	0	0.032	0.013	0.019	0	0.435	0	0	0	0	0	0
	3	0.0003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0.014	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	148	0	0	0	0	0	0	0	0	0	0	0	0	0	67.07	0	0
	232	0	0	0	0	0	0	0	0	0	0	0	0	0	0	55.17	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12.9

As a parameter threshold can be set to different values, yet for this process it is set to 75% matching percentage and that determination of similar regions is achieved as shown in Table 4-5.

Table 4-5 Detected similar and dissimilar regions

		Above Map Region Sizes															
		288999	8	6	14332	9	5	2	4	8	89	4	2	7	167	188	61
Below Map Region Sizes	290085	98.672	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	14	0.001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	15852	0	0	0	81.5	0	0.032	0.013	0.019	0	0.435	0	0	0	0	0	0
	3	0.0003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0.014	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	148	0	0	0	0	0	0	0	0	0	0	0	0	0	67.07	0	0
	232	0	0	0	0	0	0	0	0	0	0	0	0	0	0	55.17	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12.9

Similar	(green)
Dissimilar	

In Figure 4-15 similar regions that are detected according to the given matching percentages in Table 4-4, and threshold, 75%, for both maps are grouped together and presented.

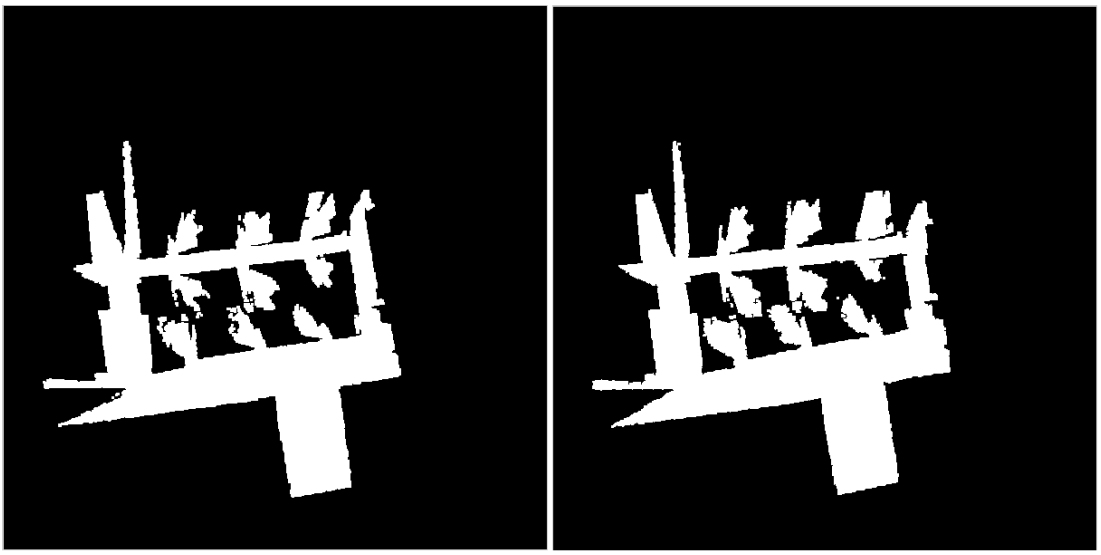


Figure 4-15 Detected similar regions for aligned maps.

In Figure 4-16 dissimilar regions that are detected according to the given matching percentages in Table 4-4, and threshold, 75%, for both maps are grouped together and presented.

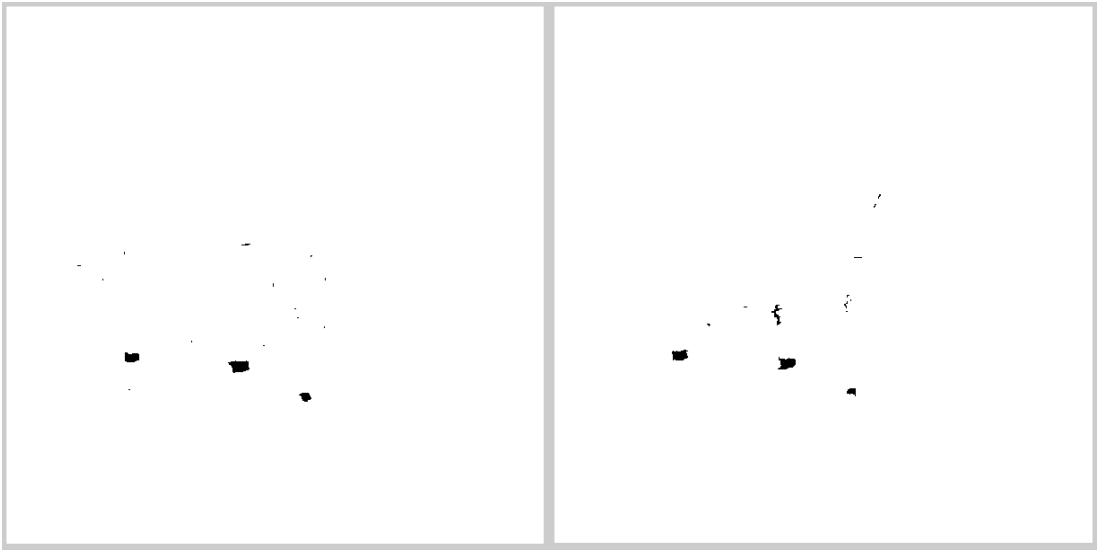


Figure 4-16 Detected dissimilar regions for aligned maps

4.2.2. System Feedback

After separation of regions, two types of feedbacks are provided at system level by using dissimilar regions:

- **Structures to focus on:** dissimilar regions that are seen approximately at the same position on both maps obtained from different height levels are the ones chosen to be focused on. While comparing positions of the regions, center points of them are utilized. If the distance between center points are lower than a defined threshold then positions of the center points are regarded as same. In Figure 4-17 these structures are presented for the example we carry all along in this chapter.

Figure 4-17 Structures to be focused upon for below map (left) and above map (right)

- **Region height:** Region height is provided as feedback so that vehicle that has appropriate sensor height for exploring the structure can be used.

4.3. Generating Connection Points

Generating connection points on the maps from different height levels is an important step while growing 3D map from 2D level maps since these points being connected actually lead to the creation of the 3D structures. Therefore, achieving proper connection of those point sets is the main objective in this subsection of the thesis.

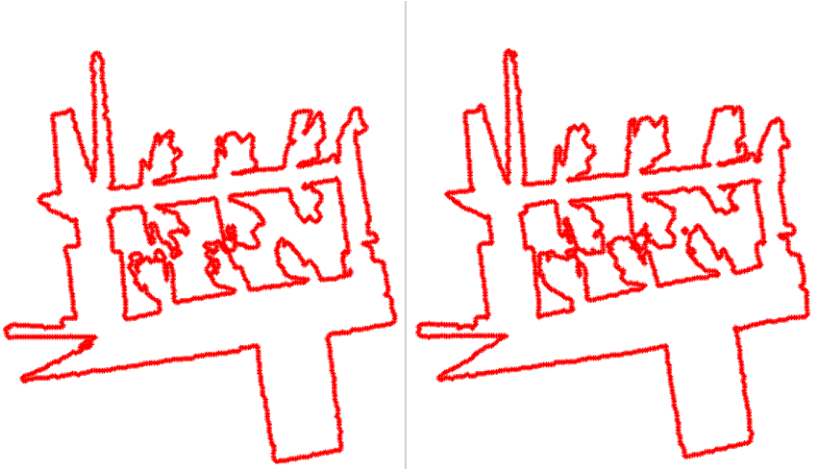
Approaches adopted for construction of structures can be divided into two types according to the usage of prior information. The ones that use prior information can fit a point set into surfaces, which eventually are utilized for constructing structures, by using defined mathematical models. The second type however requires generation of point sets and their connections without any model, but rather using different methodologies with admissible assumptions.

Methodology used in the present thesis belongs to the second type, hence there are no defined mathematical models used to construct the 3D structures. It consists of an initial generation of connection points followed by the tiling of these points according to the assumption adopted for the study. It should be noted that formalizing the assumption has a great importance for determining the techniques utilized for generation of connection points and tiling. While deciding on the assumption, physical characteristics of the structures in the environment which provide suitable information is considered. For instance, cross sections obtained from different height levels of the continuous structures reveal low variances when compared to each other. Therefore assuming that there are mostly continuous objects in the environment can lead to an approach which uses only boundary regions of the cross sections at different height levels to create the connections between them.

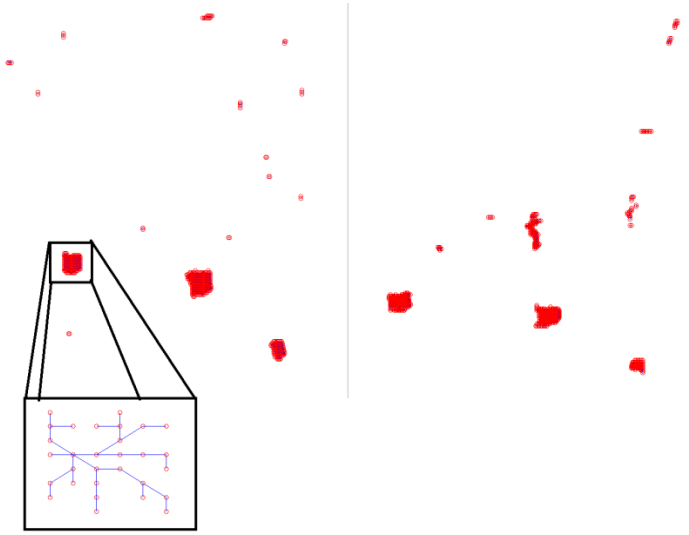
Assumption adopted in this thesis is that most of the structures in an indoor environment have discontinuities so that significant changes can be observed when the maps from different height levels are compared. Therefore, using always the boundaries of the regions for connecting maps at different height level may not provide proper results for map growing. In order to achieve the desired map growing connections should also be established inside of the regions as well as on their boundaries so as to provide a smooth growing of the structures even under discontinuities. However, creating connections inside the regions cannot be done directly, so proper connection points should be generated inside the regions which is achieved by constructing tree structures within regions.

At this point, two different approaches can be adopted in order to generate connection points; one is using boundary points and their connection between each other while the other one is generating tree structures inside regions and uses the vertices and edges as connection points and their connections. The selection of one of the approaches is done according to the similarities between regions of the maps obtained from different height levels of the environment. When regions are found similar with respect to each other, which means small changes are expected for regions between different heights levels that can be regarded as continuity, then boundary points are utilized as connection points. When regions are found dissimilar with respect to each other, which mean big changes are expected for the regions between different height levels that can be regarded as discontinuity, then tree structures are constructed within regions and their vertices are utilized as connection points.

In Figure 4-18, correspondence point generation results for similar (a) and dissimilar (b) regions of each map are presented.



(a)



(b)

Figure 4-18 Correspondence point generation results for; a) similar regions b) dissimilar regions

4.4.Tiling Connection Points

Tiling connection points is the final step for map growing. In this part, the aim is to reconstruct 3D surfaces of the structures in the environment by connecting the connection points appropriately via triangulation.

The method used for triangulation purpose is called Delaunay Triangulation (DT) which is also utilized in various studies related to biomedical imaging systems [14][15][44]. DT generates triangulation for a point set on a plane such that none of the points in the set remains inside the circumcircle of any triangle. Additionally, DT creates triangles while trying to maximize the angle values inside the triangles in order to avoid skinny triangles. An application of DT to set of points from different contours can be seen in Figure 4-19 [15] DT is applied to two point sets obtained from different contours is presented.

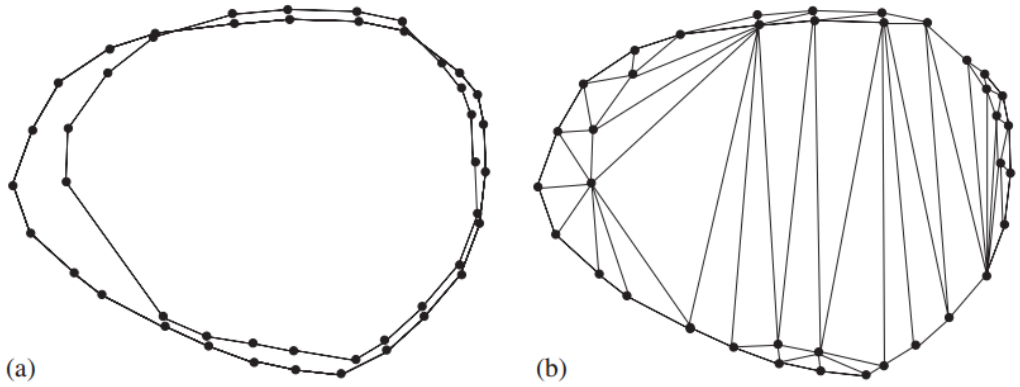


Figure 4-19 DT is applied to set of points from different contours; a) set of points b) Delaunay Triangulations [15]

Applying DT to connection points results in triangular meshes that form the surfaces to be reconstructed. However, edges of the obtained triangular meshes can have various lengths, as shown in Figure 4-20 which gives the constructed edges of the triangle connected set of points of Figure 4-19, which must be handled in order to avoid undesired connections between distant regions. Limited length edges are used to avoid undesired connections by computing the lengths of each edge on the triangles that are constructed by DT and if there is an edge longer than a defined value, which is a parameter in the algorithm, then the corresponding triangle is not formed.

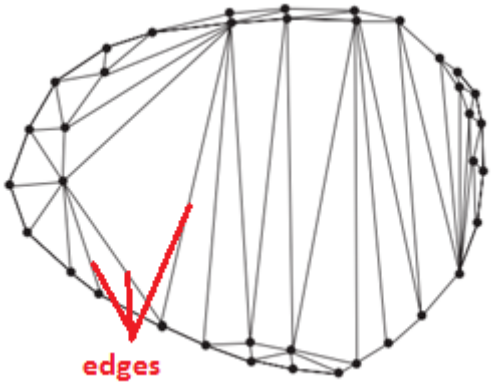


Figure 4-20 Illustration of edges for DT

This way, undesired connections that are not suitable for handling discontinuities in the environment can be avoided. If the length of the edges are not controlled then in discontinuity situations of the environment, boundaries of the regions are directly connected to each other which prevents smooth growing of the map.

When DT is applied to the point sets known as correspondence points obtained from aligned maps used in this chapter, the overall constructed 3D structure is obtained as in Figure 4-21. This is the result of map growing with two 2D maps obtained from two different height levels of the environment as given initially in Figure 4-2. As it can be seen from Figure 4-21, walls of the environment are created as surfaces and the 3D structures in the environment are constructed in volumetric forms. It should be also noticed that there are gaps between the walls of the environment. These gaps are the results of mismatches of aligned maps at the boundary parts.

Figure 4-21 Map growing result for two maps obtained from different height levels of the environment.

CHAPTER 5

SENSITIVITY ANALYSIS

In this chapter, sensitivity analysis is conducted on the methods which are grouped into two main titles namely; Fast-SLAM and map growing used in the present thesis.

5.1. Fast-SLAM

The main aim of sensitivity analysis done for the Fast-SLAM method is to analyze how values of the parameters effect results of the system and which values yield the best results. Since the most important output of the Fast-SLAM algorithm for this study is a good quality 2D map, parameters that have significant effects on the quality of the map are chosen. Selected parameters and motivations to choose them for this purpose are provided as follows:

- **Grid Size**: Grid size has a direct effect on resolution of the grid map which is an important property while working on the grid maps for map growing. When the resolution is high, reconstructed structures contain more details.
- **Particle Number**: Particle number has an effect on correct localization of the mobile robot which is achieved by incremental error correction of pose estimates. As the particle number increases, localization error decreases.
- **Probability Changes:** Probability changes are utilized while assigning probability values to a grid being occupied when generating occupancy grid maps. During the assignment of the probability value, current probability value is changed for the different cases of probability changes

for cells being occupied (m_{occ}) or probability changes for being in free space (m_{free})

5.1.1. Analysis Results of Grid Size

In Fast-SLAM algorithm 2D maps are generated with the use of grid cells which contain probability values for observing the obstacle they contain. Probability values of the grid cells are updated whenever they are in the perceptual field of the robot which means a selective updating is utilized for reducing computational burden. Selective updating approach leaves unseen grids non-updated (which are the grids that laser beams do not cross). This leads frequency of updates for the grids that are apart from the mobile robot to be low, since distance between top points of two consecutive laser beams get wider as their length increases. This leaves most of the grids untouched by the laser beam. In this case it is important to keep the frequency for update of a grid high in order to obtain a proper map whose free and occupied areas are determined adequately.

On the other hand, it should be noted that grid sizes determine the resolution of the maps: obtaining detailed maps that carry more information about the structure while map growing depends on the selecting correct grid sizes.

Considering both of the cases mentioned above, grid sizes cannot be too large which reduces frequency of grid update and also too small which reduces the resolution of the map. In this part of the thesis pertaining to sensitivity, different grid sizes are used in Fast-SLAM algorithm and the resulting 2D maps are evaluated.

In Figure 5-1, the result of the occupancy grid map construction for grid size of 0.01m is presented. It can be seen that grids away from the mobile robot are not updated enough, therefore, there are many grids that have probability value close to the uncertainty. However, grids that are closer to the path of the mobile robot have low probability of being occupied.

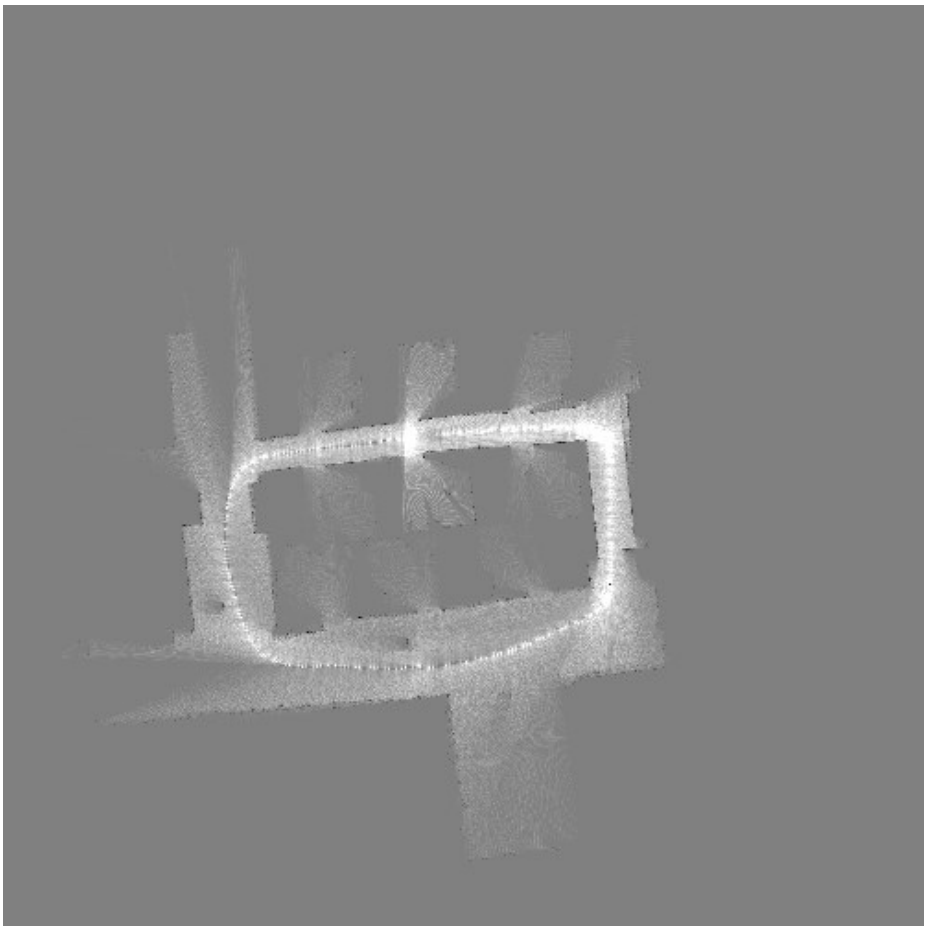


Figure 5-1 Occupancy grid map constructed with grid size 0.01m

In Figure 5-2, the result of the occupancy grid map construction for grid size of 0.02m is presented. Number of grids at the neighborhood of the path of mobile robot, which have low occupancy probability, is increased. However, there are still areas that have uncertainty probability values.

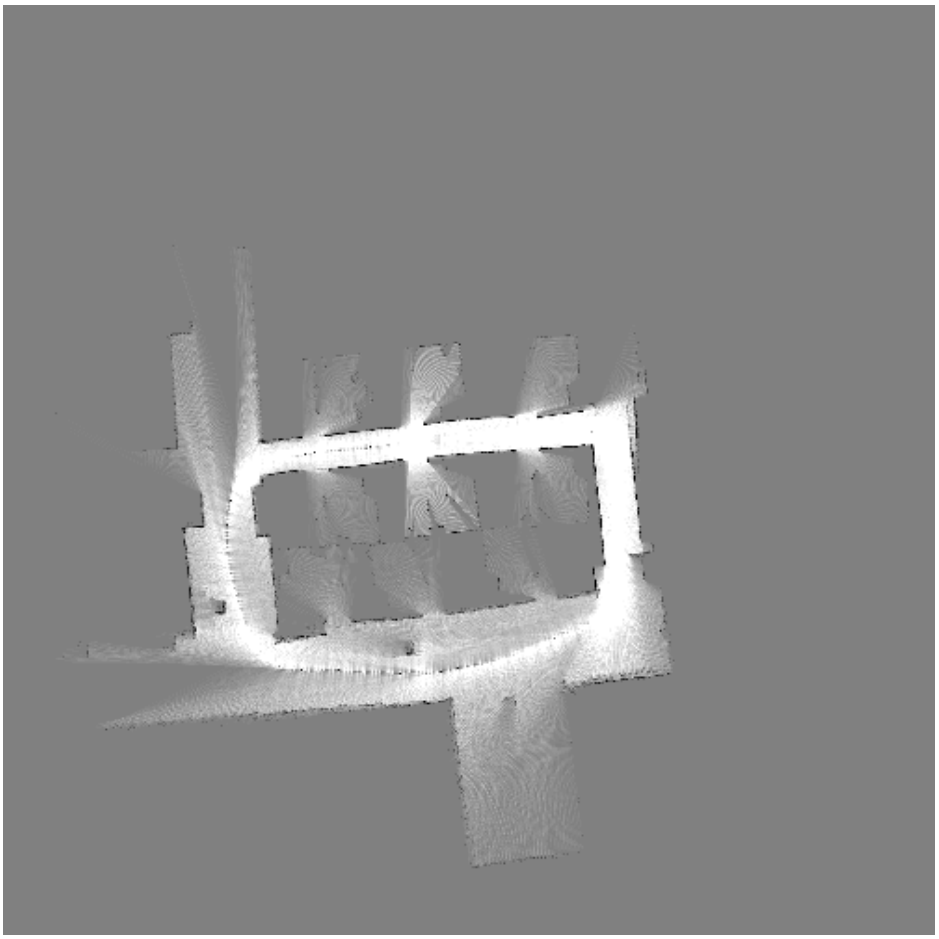


Figure 5-2 Occupancy grid map constructed with grid size 0.02m

InFigure 5-2 Figure 5-4, result of the occupancy grid map construction for grid size of 0.04m is presented. This time there are relatively less areas that have uncertainty probability values but still there are regions that remain uncertain, which are the grey areas seen in the figure.

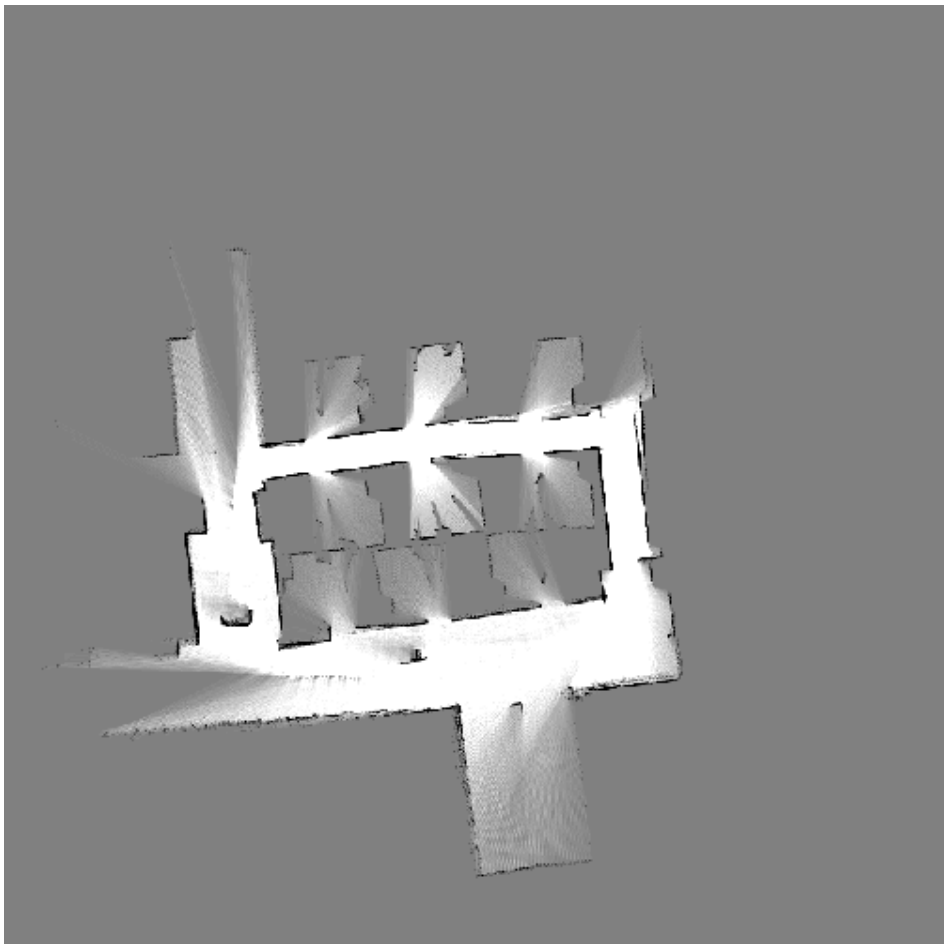


Figure 5-3 Occupancy grid map constructed with grid size 0.04m

InFigure 5-2 Figure 5-4, result of the occupancy grid map construction for grid size of 0.05m is presented. The result is better than the previous ones since uncertain regions, the grey regions, in the map get smaller, free and occupied regions, white regions, get larger.

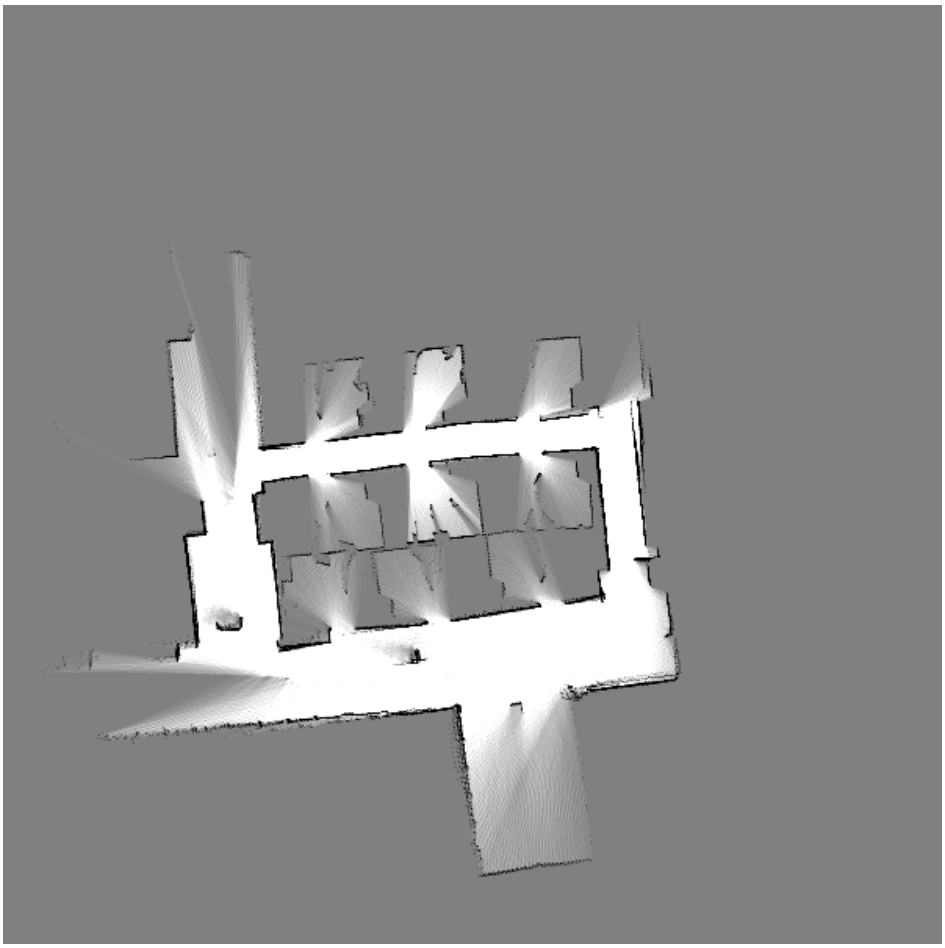


Figure 5-4 Occupancy grid map constructed with grid size 0.05m

InFigure 5-2 Figure 5-5, result of the occupancy grid map construction for grid size of 0.1m is presented. As expected, uncertainty areas are decreased even more when compared to previous results. However, this time, resolution is decreased drastically even compared to the nearest grid size value, which is 0.05m (resolution becomes one fourth of the nearest one). This is clearly seen at the boundary parts of the map which get thicker and more disordered.

Considering the results of different grid sizes, grid size of 0.05m provides an optimum result which has admissible geometric extension of uncertainty regions and good resolution.

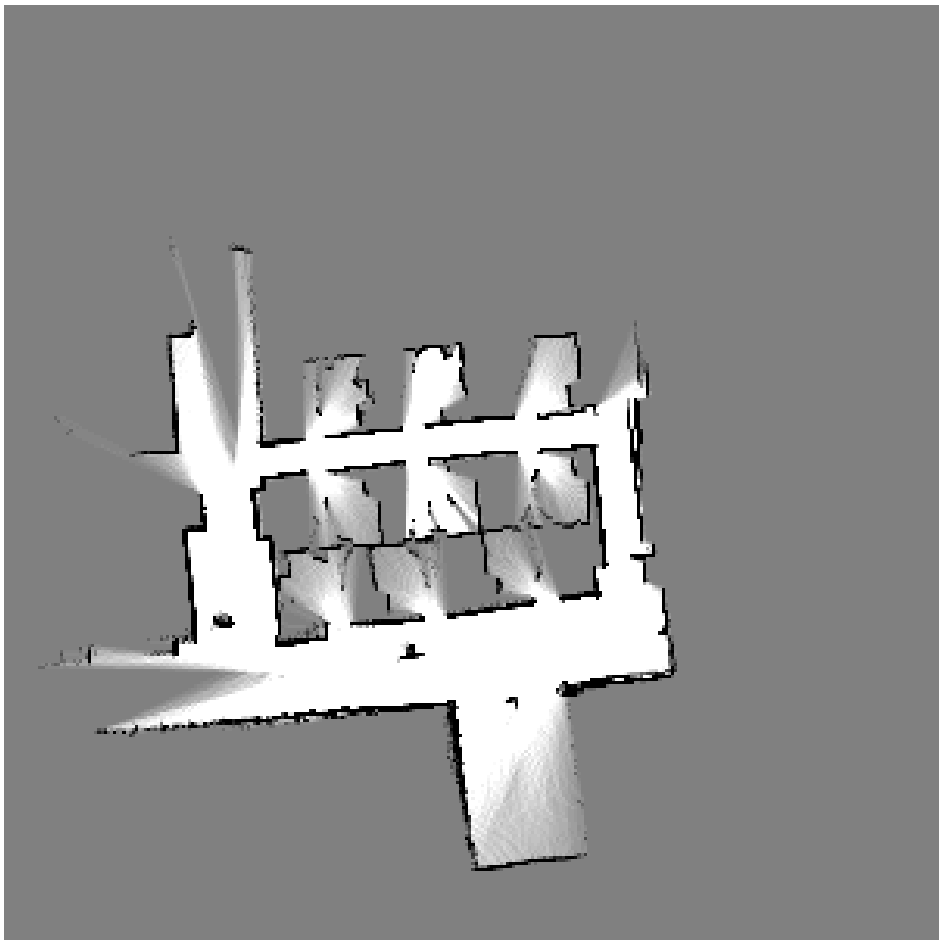


Figure 5-5 Occupancy grid map constructed with grid size 0.1m

5.1.2. Analysis Results of Particle Number

In Fast-SLAM algorithm localization and mapping work as complements of each other that is correct localization leads correct mapping of the environment and vice versa. Therefore, for more accurate map growing results achieving correct localization is important since constructed maps at different height levels of the environment can be aligned and grown properly by avoiding mismatches and holes in the 3D structure.

As mentioned earlier while presenting the implementation of the Fast-SLAM algorithm particles are utilized to represent possible pose estimates of the mobile robot. It is likely to obtain wrong pose estimates mostly caused by incremental error in the measurement of the motion, but measurement update and re-sampling of the

particles are effective to overcome this error by selecting the particles that have high probability of being at the estimated state considering the obtained measurement value. At this point it is important to note that if there are not enough particles to scatter and obtain correct localization with some of them so that they can have high probability to be chosen at re-sampling of the particles and wrong localization is inevitable. This leads to incorrect localization or particle degeneracy which defines ending up with only one particle that has much higher probability when compared to all other particles. One solution to this problem is to enhance the re-sampling method, but it is still not always sufficient to provide a solution by itself without considering the number of particles. The key element in this problem is using a proper number of particles, that prevents having particle degeneracy and thus decreases incremental error as much as possible.

In another aspect, using excessive number of particles can cause computational burden for the algorithm and use large memory since for each particle robot states, landmark states and occupancy grid map are stored. Therefore, it is significant to find the optimal number of particles so that results can both have good localization and avoid costly computations.

In this part of the thesis, different number of particles is used in Fast-SLAM algorithm considering the effect of particle number to the localization of the mobile robot in its own local map and results are presented. Initially, a ground truth that is needed for comparison of the results and computation of the distortions, errors, obtained while using different particle numbers is determined among the results. This is achieved by comparing the mean and variance of the errors obtained for x and y coordinates of the pose estimates of the mobile robot for different particle numbers. In Table 5-1 mean and variance values of the errors between pose estimates of the given particle numbers are presented. According to Table 5-1 calculated error for pose estimates of different particle numbers gets lower after the particle number of 500. In the optimum case it is expected that mean and variance values are zero, but since it is not practical, the aim here is to achieve such mean and variance values that are in a close neighborhood of the zero.

Table 5-1 Mean and variance values of the errors between pose estimates of different particle numbers

Number of Particles for compared results	Mean of pose error		Variance of pose error	
	x-coordinate	y-coordinate	x-coordinate	y-coordinate
10 - 50	-0.690441956	-0.286682125	0.530394447	0.174070091
50 - 100	0.255644435	0.120783675	0.077437373	0.016413482
100 - 250	0.125473372	-0.020475789	0.014771816	0.004983324
250 - 500	0.105037711	-0.007937913	0.016682663	0.005870776
500 - 1000	-0.043636374	0.05618888	0.001769732	0.001542423
1000 - 1500	-0.02405486	-0.055744801	0.002752184	0.000968058

Additionally, while searching for the ground truth, particle number should also be considered to be low in order to avoid computational burden as previously mentioned. In Table 5-2, computational time and memory utilization for the Fast-SLAM algorithm using different number of particles are provided. As it can be seen from the Table 5-2 as the particle number increases, elapsed time and memory utilization also increase.

Table 5-2 Computational time and memory utilization for Fast-SLAM algorithm using different number of particles.

Number of Particles	Elapsed Time (seconds)	Memory Utilization (bytes)
10	36.149	880947200
50	36.3153	880996352
100	39.9733	889262080
250	43.4673	901263360
500	47.4334	909180928
1000	50.8294	921042944
1500	58.1274	945152000

In the light of information obtained from Table 5-1 and Table 5-2 pose estimates obtained for particle number of 500 is found appropriate to be used as ground truth during the sensitivity analysis. This is because the pose error remains approximately the same after 500 particle numbers and it is adequate to choose a particle number as small as possible to avoid memory and time complexity.

In Figure 5-6 occupancy grid map constructed with 500 particles, which is the map of ground truth, is presented. In Figure 5-7 the path of the mobile robot, which consists of the pose estimates, for particle number of 500 is presented.

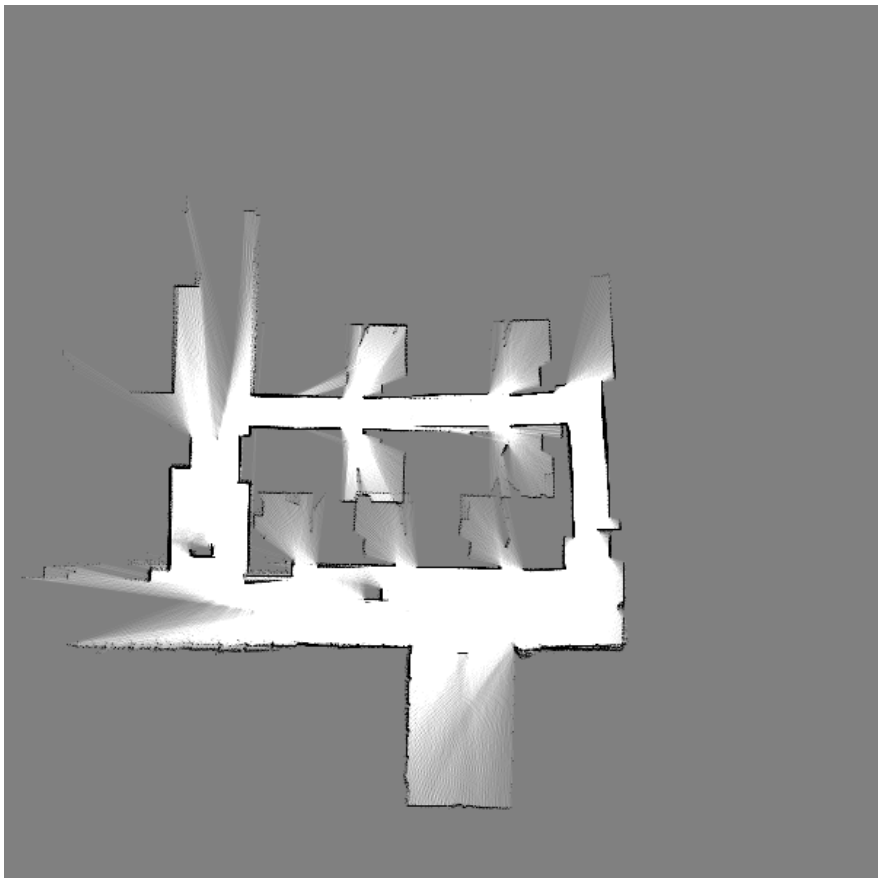


Figure 5-6 Occupancy grid map constructed with 500 particles

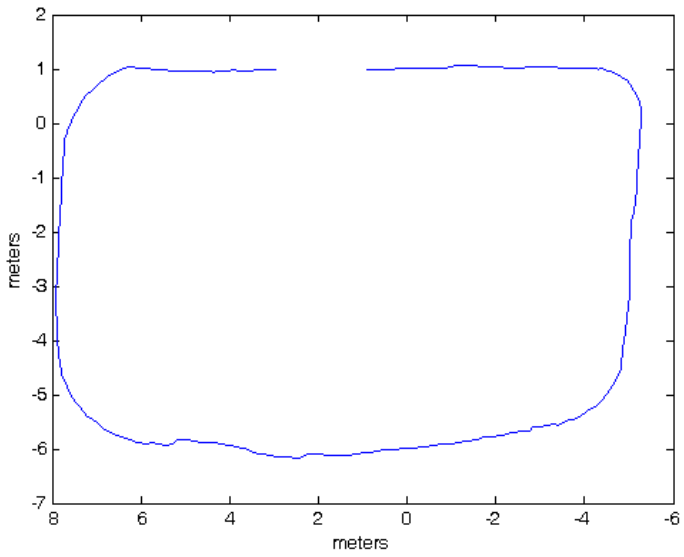


Figure 5-7 Path of mobile robot constructed with 500 particles

In Figure 5-8, occupancy grid map constructed with 10 particles is presented. Distorted areas which are caused by the errors in the localization of the mobile robot can be seen directly on the map. In Figure 5-9, path of the mobile robot constructed with 10 particles are presented along with the path of ground truth in order to illustrate the errors between the pose estimates for particle number of 10 and ground truth. It can be easily seen that there are large pose errors between the path of 10 particles and ground truth, and in total the mean and variance for the pose errors are large.

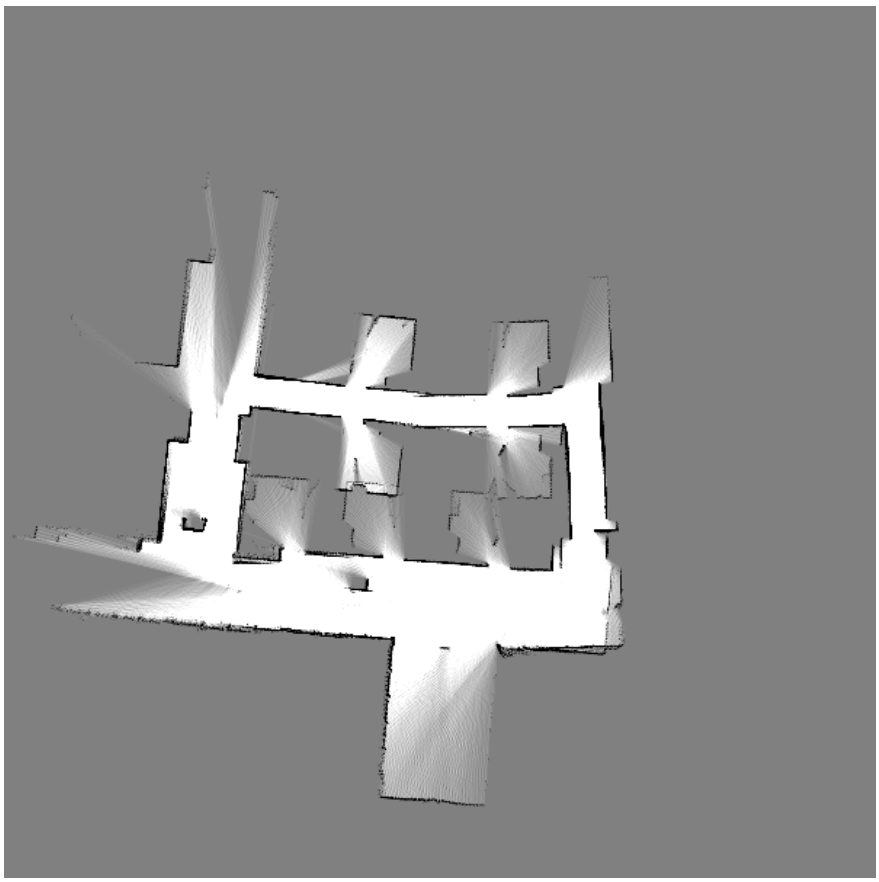


Figure 5-8 Occupancy grid map constructed with 10 particles

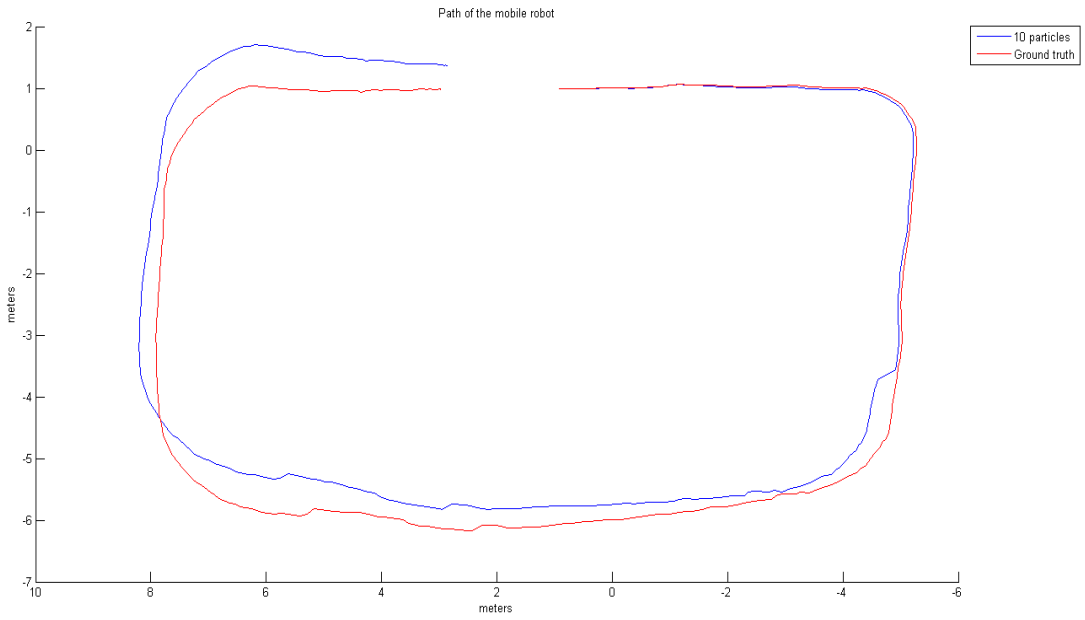


Figure 5-9 Path of mobile robot constructed with 10 particles (blue) and path of ground truth (red)

In Figure 5-10, the normal distribution of the errors in x and y coordinates of the pose estimates for 10 particles and ground truth are presented. It can be seen that for 10 particles variance (x: 0.110316, y: 0.053044) values are high and mean (x: -0.30932, y: -0.18637) values are not close enough to the zero.

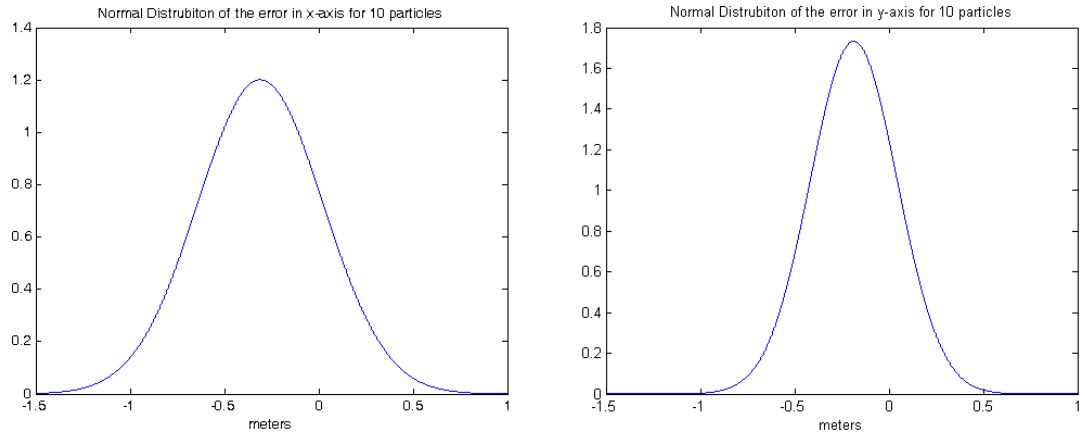


Figure 5-10 Normal distribution of the errors in x and y coordinates of the pose estimates for 10 particles and ground truth

In Figure 5-11, occupancy grid map constructed with 50 particles is presented. In Figure 5-11 it can be seen that there are still distorted areas at the point where loop closure occurs. In Figure 5-12, path of the mobile robot constructed with 50 particles are presented along with the path of ground truth in order to illustrate the errors between the pose estimates for particle number of 50 and ground truth. There are large errors between pose estimates of 50 particles and ground truth which can be seen in Figure 5-12. The reason for this result is that there are still not enough particles to obtain correct localization for the mobile robot.

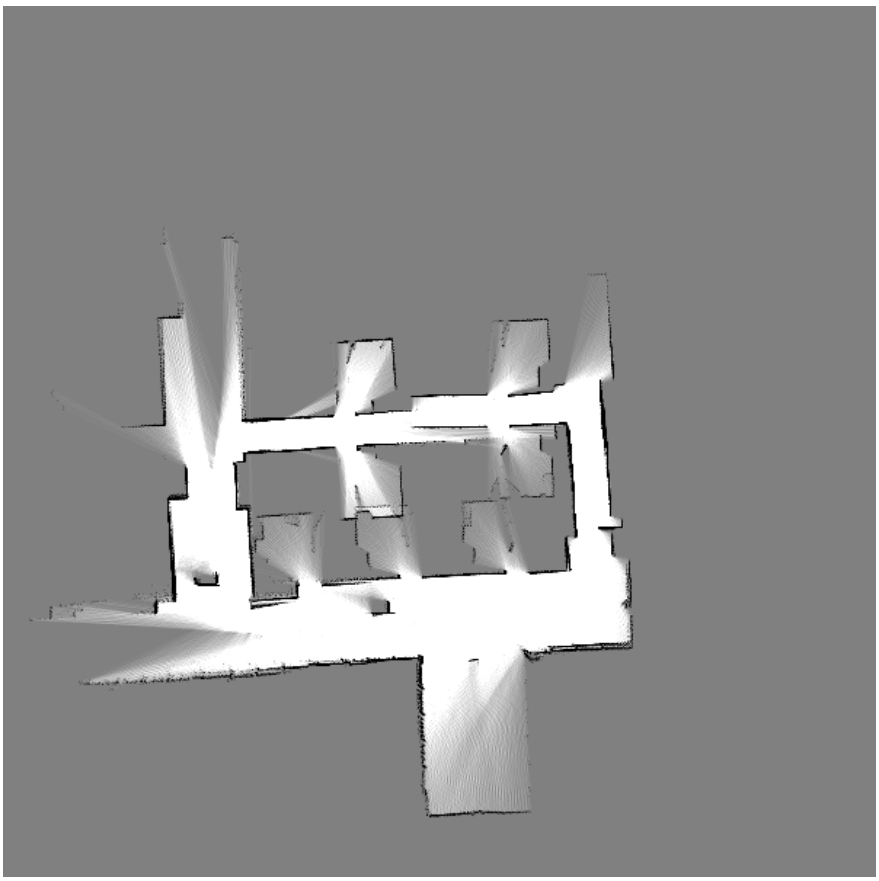


Figure 5-11 Occupancy grid map constructed with 50 particles

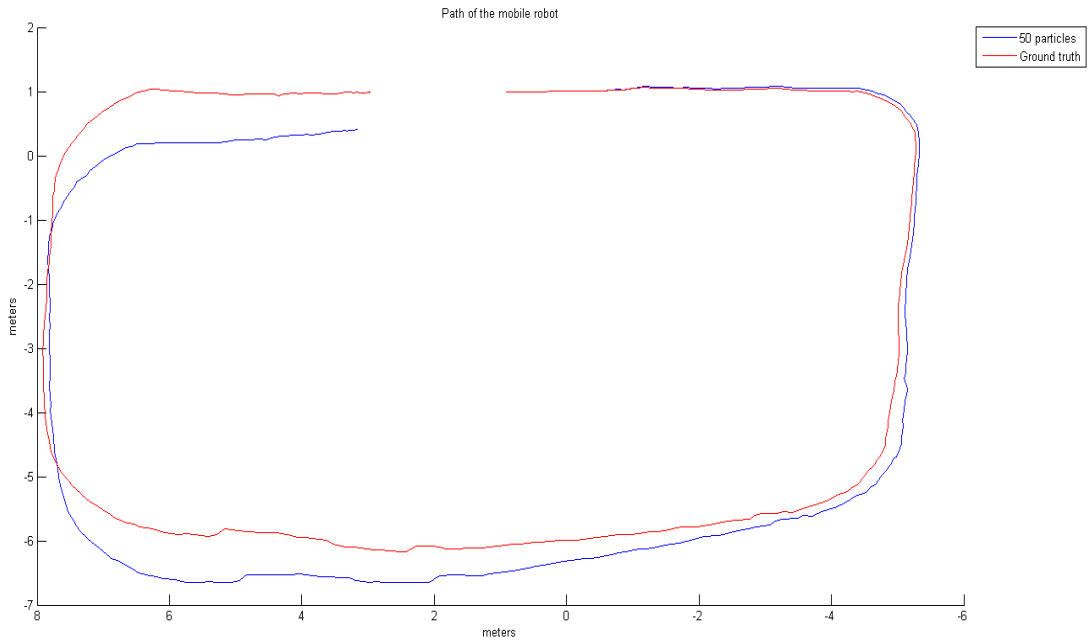


Figure 5-12 Path of mobile robot constructed with 50 particles (blue) and path of ground truth (red)

In Figure 5-13, the normal distribution of the errors in x and y coordinates of the pose estimates for 50 particles and ground truth are presented. It can be seen that variance (x: 0.157904, y: 0.036525) and mean (x: 0.381118, y: 0.100308) values are still not close enough to the zero which means that particle number of 50 is not enough to obtain correct enough localization.

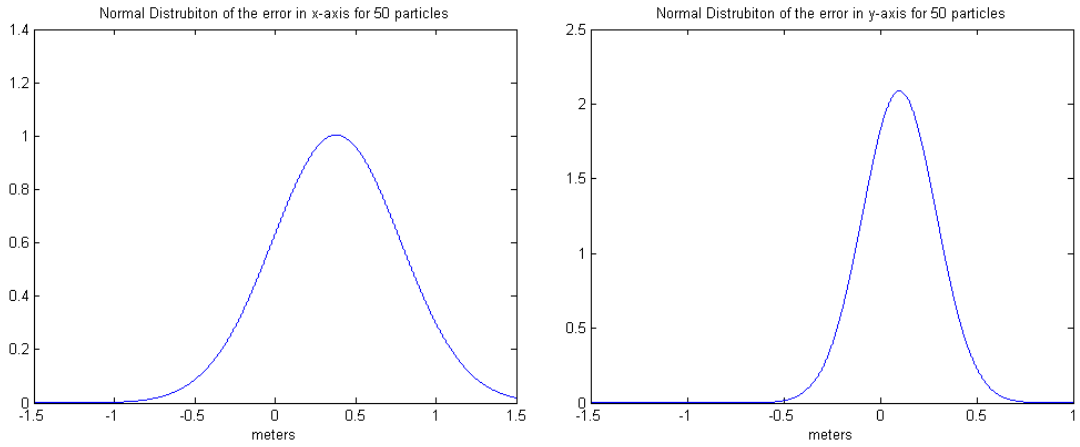


Figure 5-13 Normal distribution of the errors in x and y coordinates of the pose estimates for 50 particles and ground truth

In Figure 5-11, occupancy grid map constructed with 100 particles is presented. Distortion at the point, where loop closure occurs, gets smaller but does not totally disappear on the map when compared to previous results. In Figure 5-15, the path of the mobile robot constructed with 100 particles are presented along with the path of ground truth in order to illustrate the errors between the pose estimates for particle number of 100 and ground truth. As it can be seen errors between pose estimates are getting smaller as the number of particles increases.

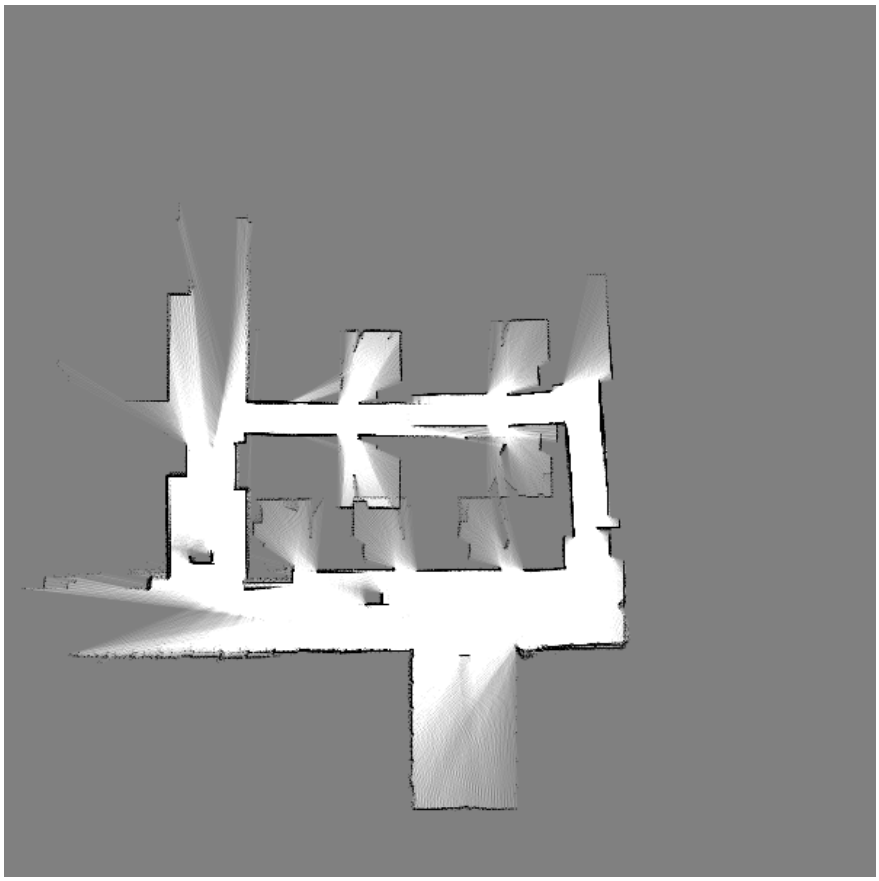


Figure 5-14 Occupancy grid map constructed with 100 particles

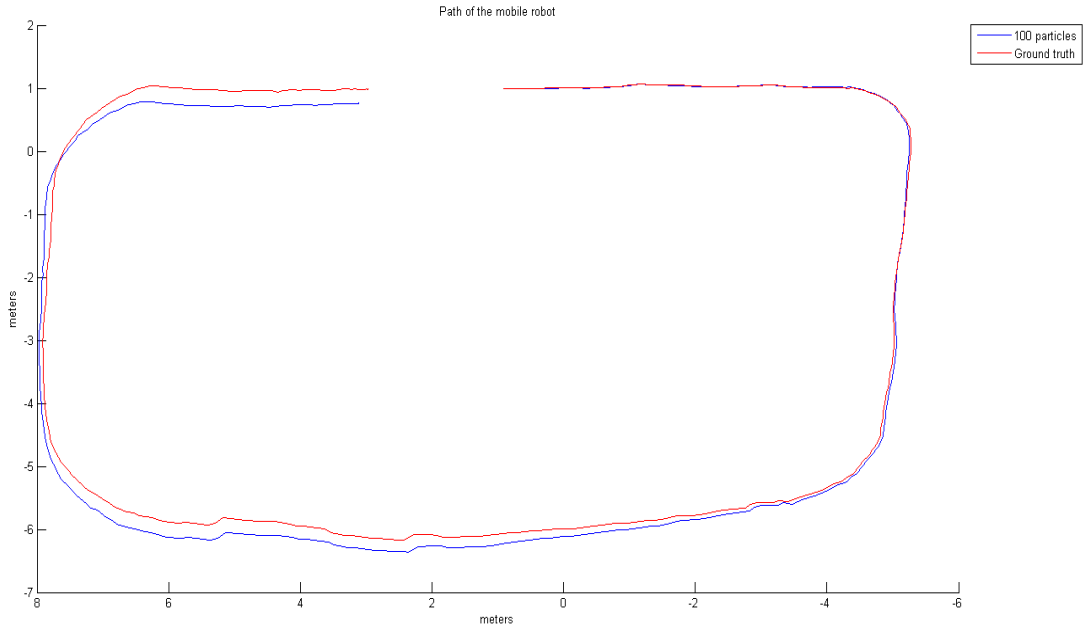


Figure 5-15 Path of mobile robot constructed with 100 particles (blue) and path of ground truth (red)

In Figure 5-16, the normal distribution of the errors in x and y coordinates of the pose estimates for 100 particles and ground truth are presented. When compared to previous results, variance (x: 0.014772, y: 0.004983) and mean (x: 0.125473, y: -0.02048) values get closer to the zero more swiftly since change in number of particle is also increased.

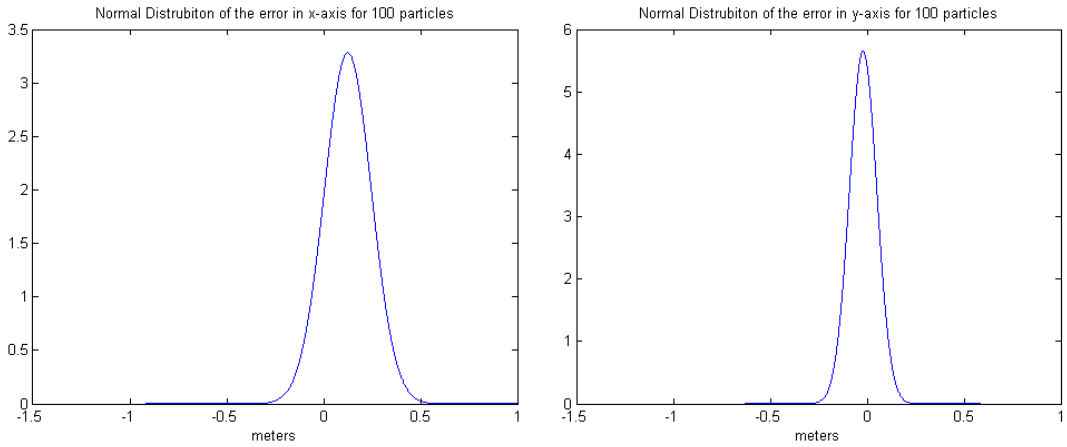


Figure 5-16 Normal distribution of the errors in x and y coordinates of the pose estimates for 100 particles and ground truth

In Figure 5-17, occupancy grid map constructed with 250 particles is presented. Distortion at the point, where loop closure occurs, can be still observed, but as can be seen in Figure 5-18, which shows the path of the mobile robot constructed with 250 particles along with the path of ground truth, errors obtained from the other parts apart from the point that loop closure occurs, become really small.

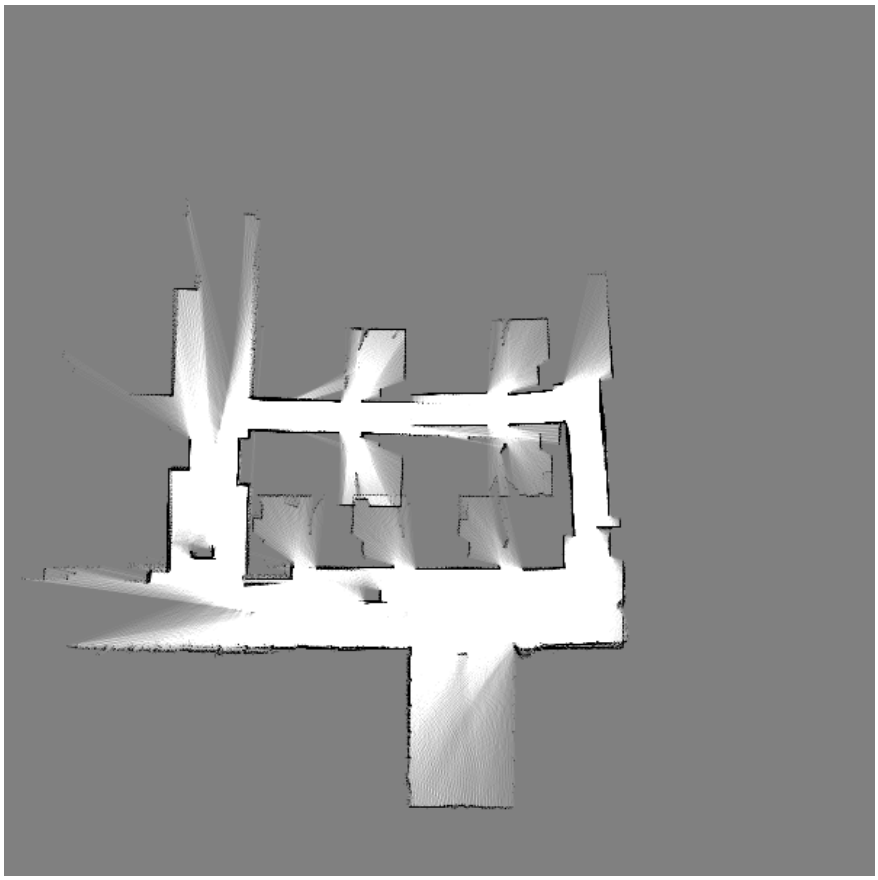


Figure 5-17 Occupancy grid map constructed with 250 particles

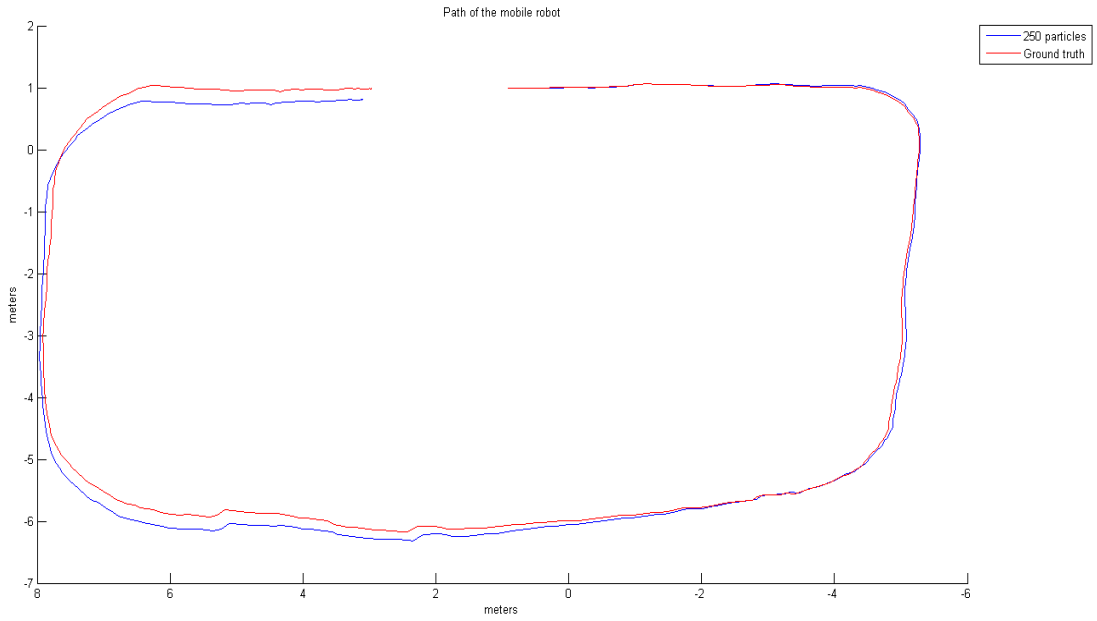


Figure 5-18 Path of mobile robot constructed with 250 particles (blue) and path of ground truth (red)

In Figure 5-19, the normal distribution of the errors in x and y coordinates of the pose estimates for 250 particles and ground truth are presented. When compared to previous results, variance (x: 0.016683, y: 0.005871) values do not change significantly but the mean (x: 0.105038, y: -0.00794) values get closer to the zero.

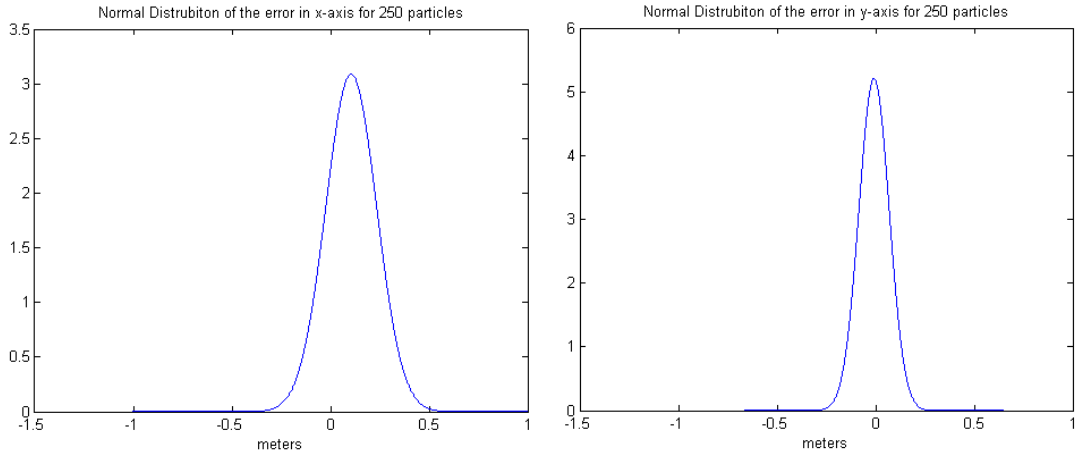


Figure 5-19 Normal distribution of the errors in x and y coordinates of the pose estimates for 250 particles and ground truth

Lastly, in order to illustrate the reason for choosing the result of 500 particles as ground truth and use this number for SLAM approach occupancy grid map constructed with 1000 particles is shown in Figure 5-20 from which it can be seen that distortion at the point where loop closure occurs gets really small. Additionally, in Figure 5-21, the path of the mobile robot constructed with 1000 particles are presented along with the path of ground truth. As one can see in Figure 5-21 there are only tiny errors between the pose estimations of particle number of 1000 and particle number of 500. Therefore, using pose estimation and mapping results of 500 particles as ground truth is acceptable.

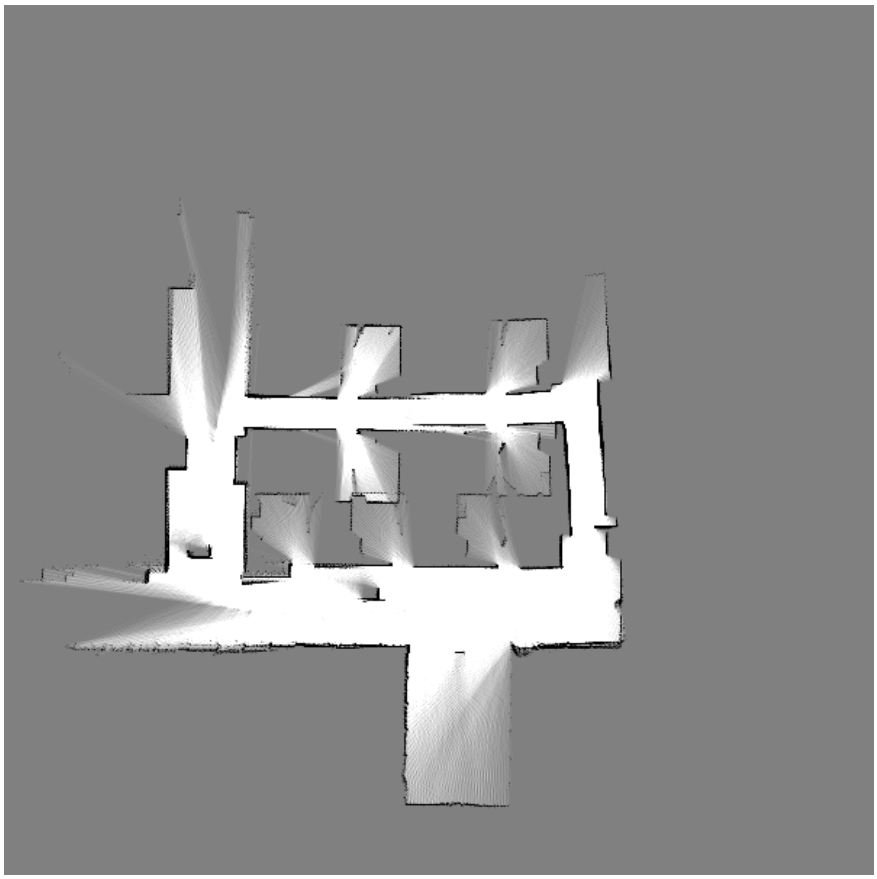


Figure 5-20 Occupancy grid map constructed with 1000 particles

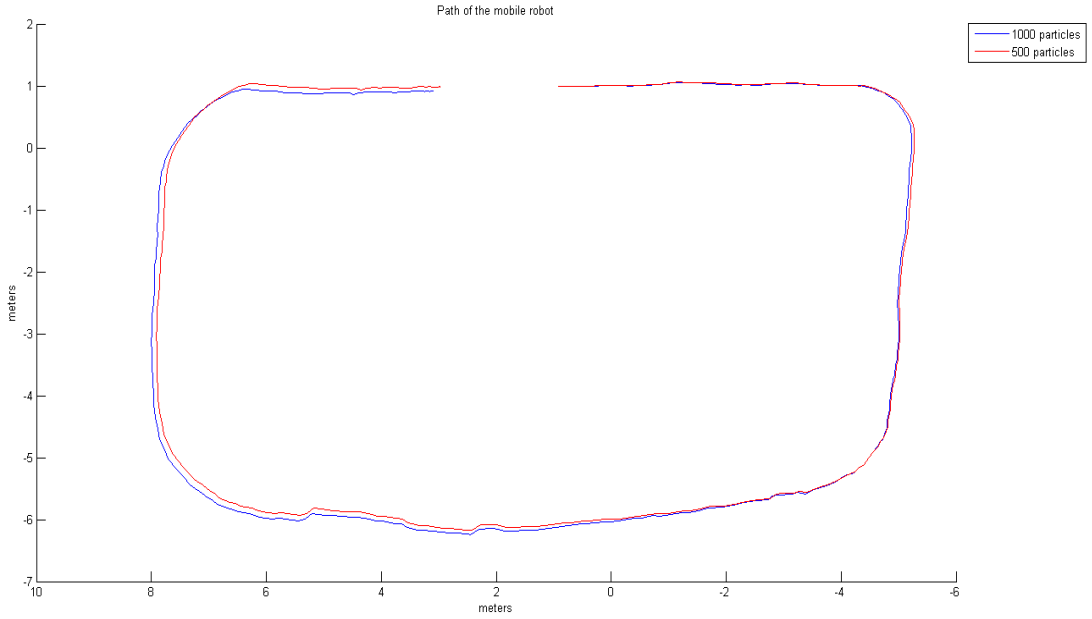


Figure 5-21 Path of mobile robot constructed with 1000 particles (blue) and path of ground truth (red)

In Figure 5-22, the normal distribution of the errors in x and y coordinates of the pose estimates for 1000 particles and ground truth are presented. It is clear that, variance (x: 0.00177, y: 0.001542) and mean (x: 0.043636, y: -0.05619) values are much more closer to the zero when compared to the previous results which actually shows that 1000 particles can provide correct enough localization.

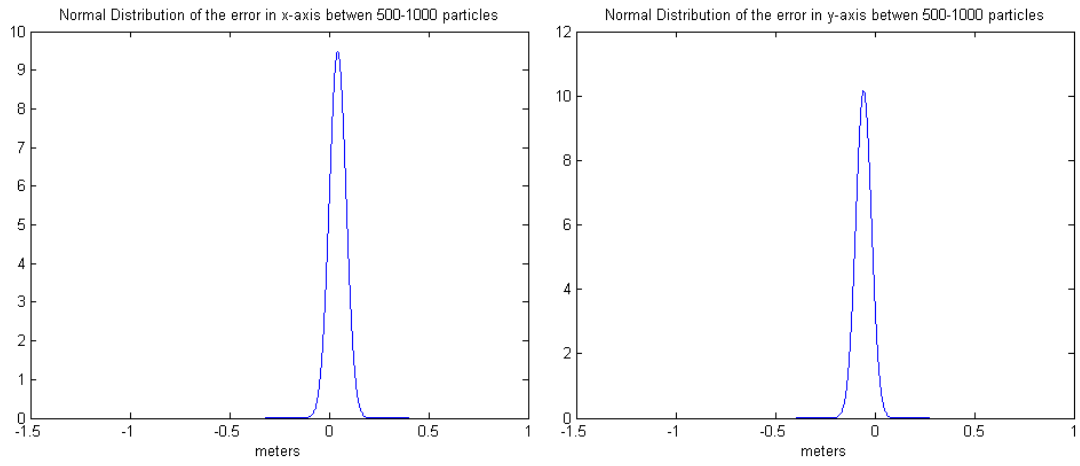


Figure 5-22 Normal distribution of the errors in x and y coordinates of the pose estimates for 1000 particles and ground truth

5.1.3. Analysis Results of Probability Changes

In Fast-SLAM algorithm while constructing occupancy grid map, the occupancy probability value of a grid is updated whenever it is in the perceptual field of the mobile robot which is mentioned as selective updating and used to reduce computational complexity. Selective updating approach does not update the grids that are unseen by the mobile robot, which are grid cells that the laser beam of the scanner do not touch.

For the grids that are in the perceptual field of the mobile robot, their occupancy probabilities are updated using the probability change for being occupied (m_{occ}) or else it is the probability change for being free space (m_{free}) that is used. In implementation of the occupancy grid mapping method of Fast-SLAM algorithm m_{occ} and m_{free} are fixed values and in this part of the thesis appropriate values are found. Probability change values directly affect the construction of areas that are either free or occupied during exploration of the environment. Therefore, the aim in probability change analysis is to determine m_{occ} and m_{free} values such that uncertainty regions are reduced and boundary parts of the environment are obtained adequately.

Before proceeding with further details, it can beneficial to note that m_{occ} is mostly used for the boundary regions which are naturally furthest points seen in a scan and update frequency of these regions are lower than the ones that are closer to the mobile robot. Therefore, assignment of the probability values at the boundary regions is more sensitive than the other ones which leads m_{occ} value to be chosen higher or equal to the m_{free} value so that probability assignment at the boundary regions can be more robust to the incorrect measurements coming from laser sensor.

In Figure 5-23, occupancy grid map results for (a) $m_{occ} = -0.1$ and $m_{free} = +0.02$, (b) $m_{occ} = -0.2$ and $m_{free} = +0.02$, (c) $m_{occ} = -0.5$ and $m_{free} = +0.02$ are presented. It can be seen that, as the m_{occ} increases boundaries or edges of the structures in the environment get thicker which is important for mapping small structures since their areas are directly affected from the width of the boundaries.

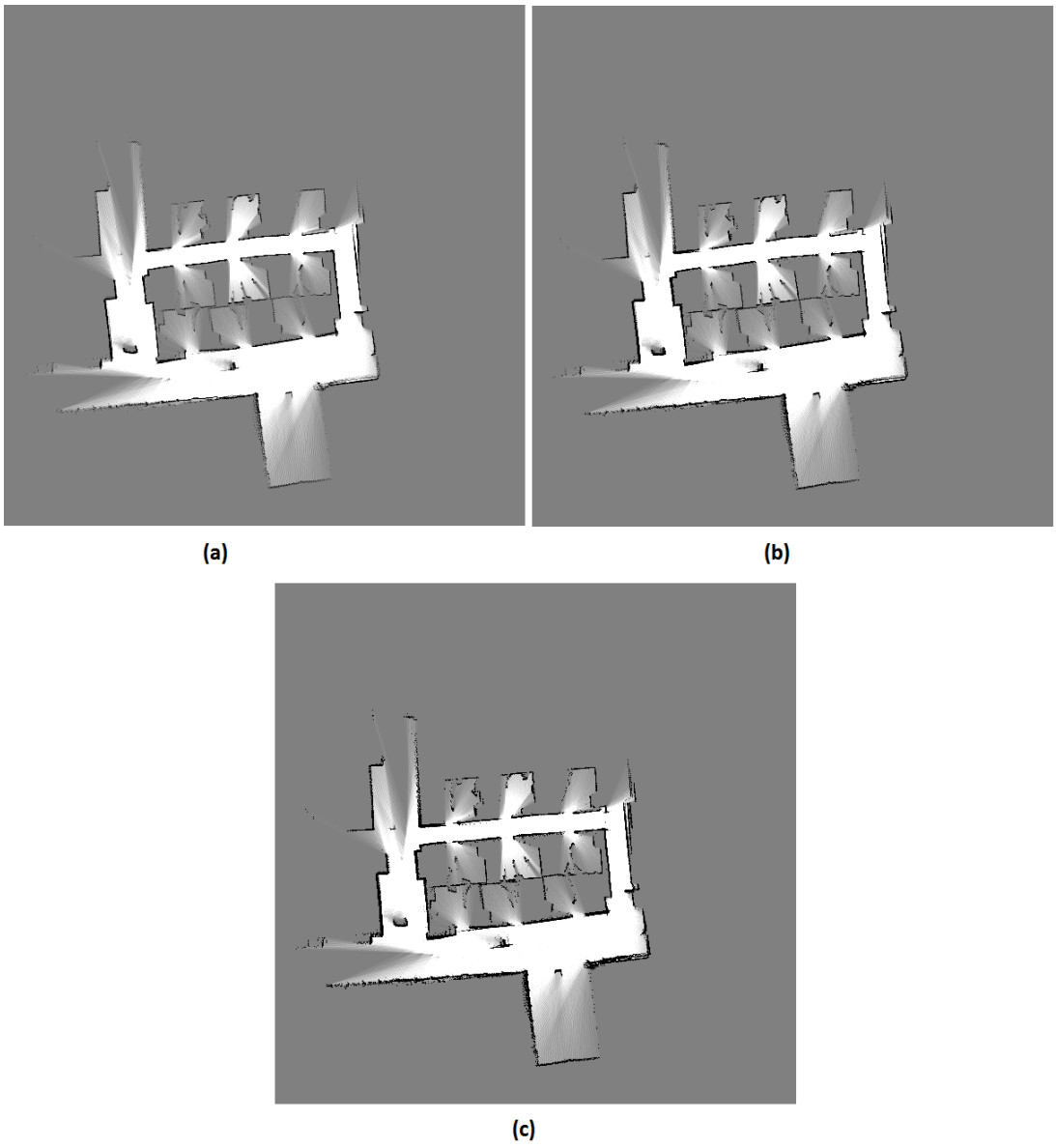


Figure 5-23 Occupancy grid map results for (a) $m_{occ} = -0.1$ and $m_{free} = +0.02$, (b) $m_{occ} = -0.2$ and $m_{free} = +0.02$, (c) $m_{occ} = -0.5$ and $m_{free} = +0.02$

In Figure 5-24, occupancy grid map results for (a) $m_{occ} = -0.1$ and $m_{fre} = +0.05$, (b) $m_{occ} = -0.2$ and $m_{free} = +0.05$, (c) $m_{occ} = -0.5$ and $m_{free} = +0.05$ are presented. The increase in the m_{free} value causes reduction of uncertainty regions which can be seen by comparing the grey regions in Figure 5-23 and Figure 5-24. However, this increase also affects areas of the structures in the environment,

especially for the small values of m_{occ} areas of the structures get smaller since number of occupied grid cells decreases as m_{occ} decreases.

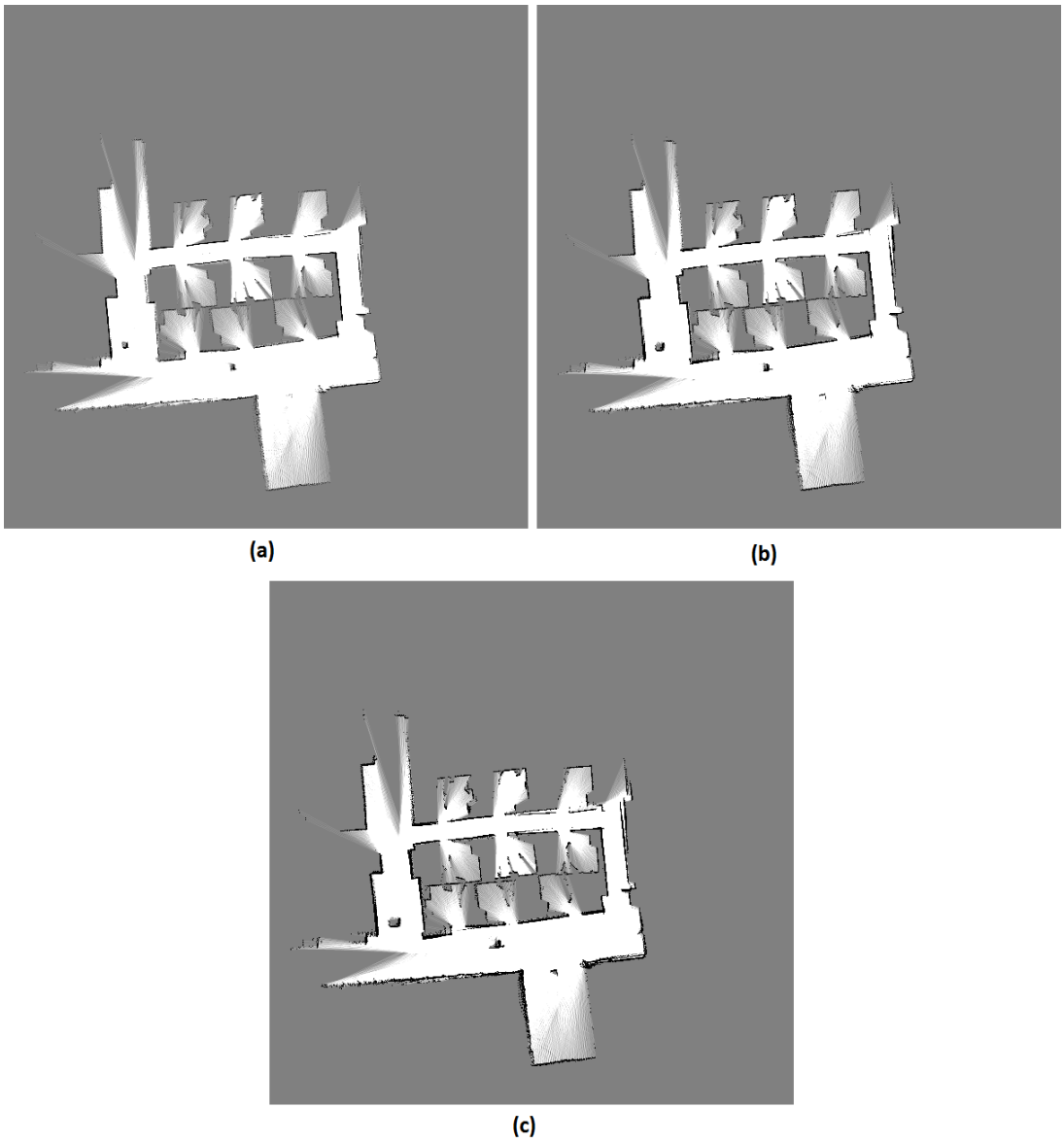


Figure 5-24 Occupancy grid map results for (a) $m_{occ} = -0.1$ and $m_{free} = +0.05$, (b) $m_{occ} = -0.2$ and $m_{free} = +0.05$, (c) $m_{occ} = -0.5$ and $m_{free} = +0.05$

In Figure 5-25, occupancy grid map results for (a) $m_{occ} = -0.1$ and $m_{free} = +0.1$, (b) $m_{occ} = -0.2$ and $m_{free} = +0.1$, (c) $m_{occ} = -0.5$ and $m_{free} = +0.1$ are presented. As one can see in Figure 5-25 (a) and (b), most of the boundary regions of the structures become very thin, thus areas of the regions get smaller, and even one

region is completely disappeared as m_{free} increases. Only in Figure 5-25 (c), as the m_{occ} value increases, boundaries get thicker and the disappeared region is constructed adequately. However, regions are small because of high m_{occ} value to be used in map growing properly.

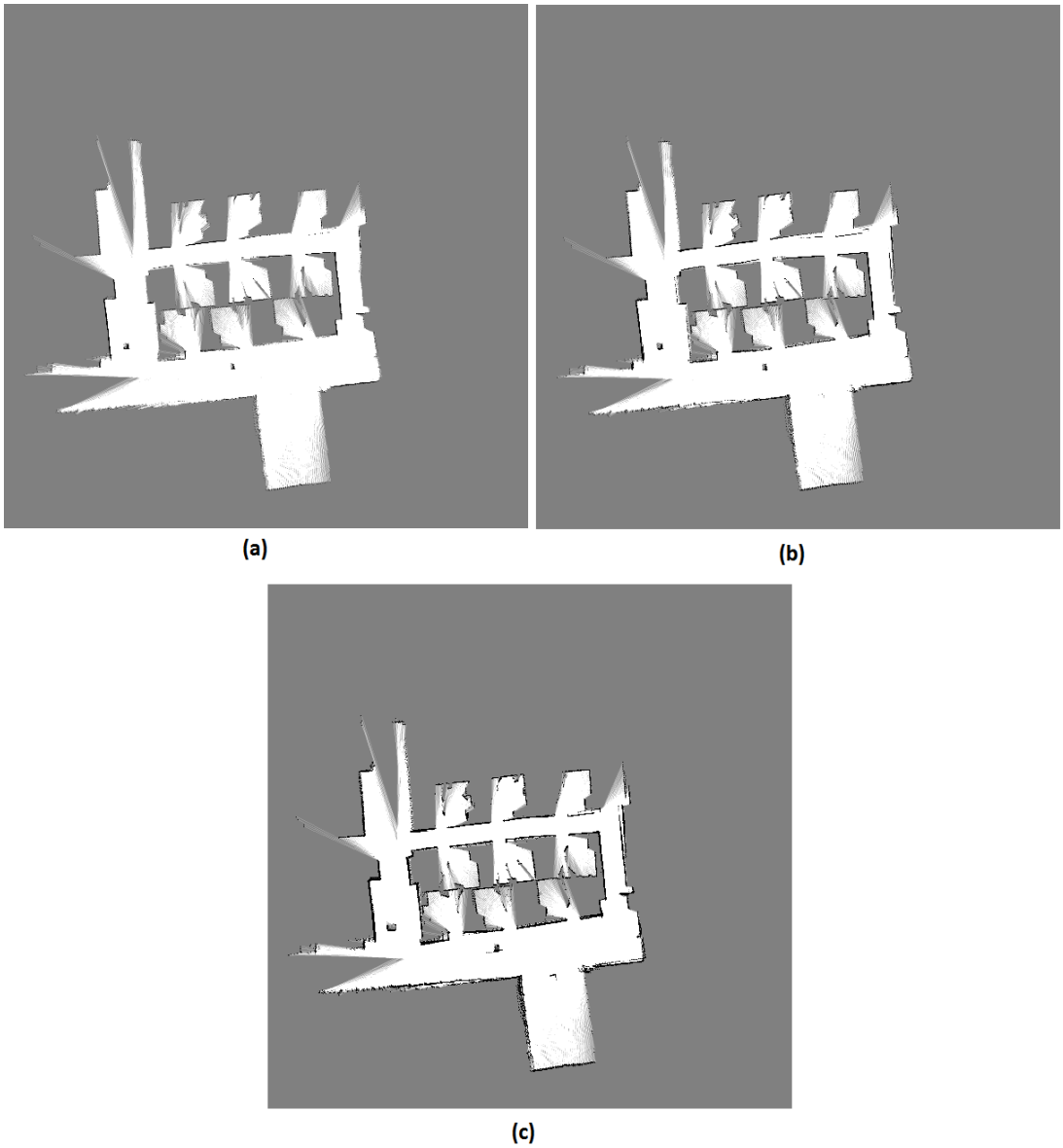


Figure 5-25 Occupancy grid map results for (a) $m_{occ} = -0.1$ and $m_{free} = +0.1$, (b) $m_{occ} = -0.2$ and $m_{free} = +0.1$, (c) $m_{occ} = -0.5$ and $m_{free} = +0.1$

In Figure 5-26, occupancy grid map results for (a) $m_{occ} = -0.2$ and $m_{free} = +0.2$, (b) $m_{occ} = -0.5$ and $m_{free} = +0.2$ are presented. This time there are only two

different constructed maps since m_{free} is taken as 0.2. As previously mentioned, increase in the m_{free} value causes boundaries to be thinner and small regions in the environment to disappear. Such a region that disappears is encircled (red) in Figure 5-26. For the case shown in Figure 5-26, increasing m_{occ} does not change the situation and results are still not suitable for utilization.

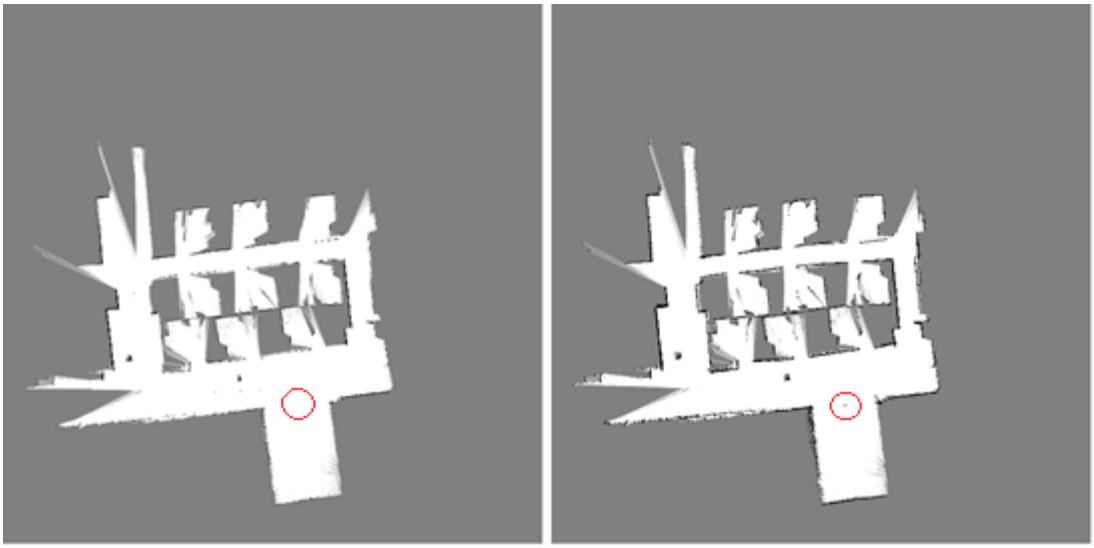


Figure 5-26 Occupancy grid map results for (a) $m_{occ} = -0.2$ and $m_{free} = +0.2$, (b) $m_{occ} = -0.5$ and $m_{free} = +0.2$

Considering the results obtained for different probability changes, high values of m_{free} reduces the width of the structures and uncertainty areas. Therefore, one should find a middle value such that there are less uncertain areas and structures in the environment have both thicker boundaries and cover large areas. When the m_{occ} value is considered, high values are useful since they yield robustness for boundaries against the problems such as low update frequency, which prevents repetitive update for boundary regions and boundary regions are not constructed adequately.

It is important to apply this analysis before achieving final values for both probability changes that are going to be used in the system.

5.2.Map Growing

The main aim of sensitivity analysis done for the map growing method is to designate the values for the parameters used in the method that reveal better results. There are two important parameters used in the map growing method, which are

- The percentage threshold used for separating similar-dissimilar regions
- The length limit used for tiling the correspondence points.

5.2.1. Analysis Results of Percentage Threshold for Region Separation

In the separation of the regions section of the map growing method, the similarity between the regions which belong to the aligned grid maps obtained from different height levels of the environment is measured using grid by grid comparison. Similarity is measured as the ratio between number of matching grids and total number of grids that belong to the larger region. In this region separation section which utilizes the similarity measure, a percentage threshold is used to evaluate the similarity measure and decides if compared regions are similar or dissimilar.

In this part of the thesis, the aim is to see the effect of percentage threshold to the map growing method. In order to achieve this purpose, three different percentage thresholds are utilized for region separation for which, their individual outcomes are examined separately along with the elapsed time and memory utilization.

Initially, 2D grid maps are generated for five different height levels and aligned with respect to each other. Then semantically meaningful regions are extracted by using CCL method and regions that belong to the successive height levels are compared grid by grid. Afterwards similarity is measured as explained previously, by computing the ratio between the number of matching grids and the total number of grids that belong to the wider region. In Table 5-3 matching percentage of regions belong to maps for the first (row) and second (column) height level are presented. It can be seen that for similar regions the matching percentage is high, such as regions 1-1 and 5-4, and for dissimilar regions matching percentage is low.

Table 5-3 Matching percentage of regions belong to the first height level, 81 cm, and second height level, 86 cm, maps

Region Numbers	Region Numbers											
	1	2	3	4	5	6	7	8	9	10	11	12
1	98.23474	0.001376	0	0	0	0	0	0	0	0	0	0
2	0.001384	0	0	0	0	0	0	0	0	0	0	0
3	0.000692	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	81.88158	0	0.012679	0.012679	0	0.012679	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0.013854	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0.013854	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	71.25749	0	0
15	0	0	0	0.006927	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	80.51948	0
17	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	63.15789

In Table 5-4 matching percentage of regions belong to the second (row) and third (column) height level maps are presented. From the table it can be seen that regions 1-1 and 4-5 have higher matching percentage than the other, thus their similarity is higher than the other matches.

Table 5-4 Matching percentage of regions belong to the second height level, 86 cm, and third height level, 91 cm, maps

Region Numbers	Region Numbers															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	98.079	0	0.0042	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0.0028	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	75.515	0	0.3325	0.0208	0	0.0831	0	0.0346	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0.0069	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0.0138	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0.062	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0.0413	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	47.305	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23.28	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24.59

In Table 5-5 matching percentage of regions belong to the third and fourth height level maps are presented. It can be seen that regions 1-1, 5-5 and 14-18 have higher matching percentage than the other matches which means they are more similar.

Table 5-5 Matching percentage of regions belong to the third height level, 91 cm, and fourth height level, 96 cm, maps

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	97.8102	0.00103	0.0172	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0.00412	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	71.6389	0.00688	0	0.01377	0.12391	0	0	0	0.05507	0	0	0	0.01377	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0.33677	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0.01321	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	71.4286	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9.85915	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

In Table 5-6 matching percentage of regions belong to the fourth and fifth height level maps are presented.

Table 5-6 Matching percentage of regions belong to the fourth height level, 96 cm, and fifth height level, 101 cm, maps

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	97.785	0	0.0027	0.0014	0.0055	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0014	0.0021	0.0014	0.0014	0.0014	0.0024	0
2	0.001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	72.278	0	0	0.1849	0.0264	0.0132	0.0132	0.0132	0.0198	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0.0071	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0.064	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0.2844	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0.128	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0.0213	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0.0213	0	0	0	0	0	0	0	0	7.4074	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0.0498	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30.519	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	50	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

At first, 90% is set as threshold for similarity which separates similar (green) and dissimilar regions as shown in Table 5-7 - Table 5-10. It can be seen that, selecting high percentage value as a threshold causes less region match due to similarity. As previously mentioned, creating correspondence points for dissimilar regions is achieved by using tree structures that are constructed inside the regions and thus most of the time tree structures are utilized for creating correspondence points while using high threshold value, since there are more dissimilar regions than the similar ones.

Table 5-7 Similarity percentage of regions between first height level, 81 cm, and second height level, 86 cm, and designated similar and dissimilar regions with respect to percentage threshold of 90%

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12
1	98.23474	0.001376	0	0	0	0	0	0	0	0	0	0
2	0.001384	0	0	0	0	0	0	0	0	0	0	0
3	0.000692	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	81.88158	0	0.012679	0.012679	0	0.012679	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0.013854	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0.013854	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	71.25749	0	0
15	0	0	0	0.006927	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	80.51948	0
17	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	63.15789

SIMILAR	
DISSIMILAR	

Table 5-8 Similarity percentage of regions between second height level, 86 cm, and third height level, 91 cm, and designated similar and dissimilar regions with respect to percentage threshold of 90%

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	98.07859	0	0.004152	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0.002753	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	75.51456	0	0.332502	0.020781	0	0.083126	0	0.034636	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0.006884	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0.013767	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0.061954	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0.041302	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	47.30539	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23.28042	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24.59016

SIMILAR	
DISSIMILAR	

Table 5-9 Similarity percentage of regions between third height level, 91 cm, and fourth height level, 96 cm, and designated similar and dissimilar regions with respect to percentage threshold of 90%

Region Numbers	Region Numbers																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	97.81017	0.001032	0.017205	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0.004122	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	71.63893	0.006884	0	0.013767	0.123907	0	0	0	0.05507	0	0	0	0.013767	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0.336767	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0.013207	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	71.42857	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9.859155	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

Table 5-10 Similarity percentage of regions between fourth height level, 96 cm, and fifth height level, 101 cm, and designated similar and dissimilar regions with respect to percentage threshold of 90%

Region Numbers	Region Numbers																								
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	97.78509	0	0.002748	0.001374	0.005496	0	0	0	0	0	0	0	0	0	0	0	0	0	0.001374	0.002061	0.001374	0.001374	0.001374	0.002405	0
2	0.001035	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	72.27787	0	0	0.184892	0.026413	0.013207	0.013207	0.013207	0.01981	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0.00711	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0.063989	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0.284394	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0.127977	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0.02133	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0.02133	0	0	0	0	0	0	0	0	7.407407	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0.049769	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30.51948	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	50	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

In Figure 5-27 map growing result for threshold value of 90% is presented. As one can notice that regions concentrated in the middle part of the area, which are inside the red circle, are reconstructed by using connection points generated by tree structures since they are regarded as dissimilar. These regions are relatively large regions considering the whole environment, thus growing these regions by using tree structures are computationally costly.

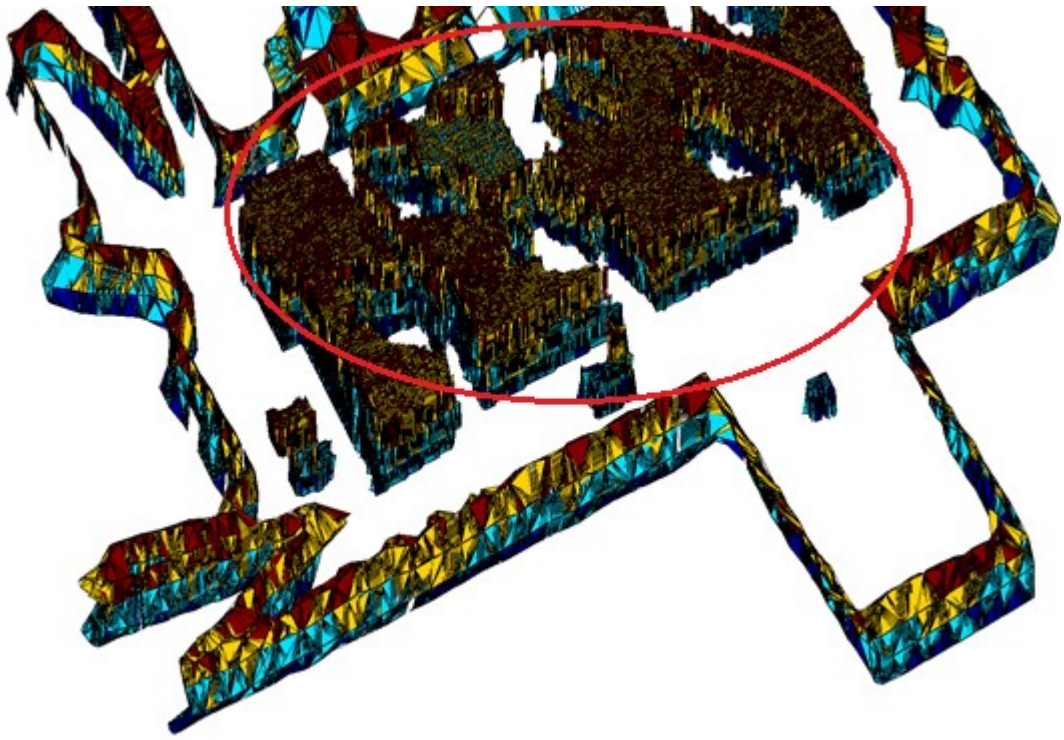


Figure 5-27 Map growing result for 90% threshold value

Secondly, 70% is set as threshold for similarity which separates similar (green) and dissimilar regions as shown in Table 5-11 - Table 5-14. This time a medium value is selected as threshold which enables large regions, such as the ones encircled in Figure 5-27, to be considered as similar when compared to the previous case where those regions are considered as dissimilar. Therefore, number of similar regions increase which leads to lower computational cost generating correspondence points and growing the regions.

Table 5-11 Similarity percentage of regions between first height level, 81 cm, and second height level, 86 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70%

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12
1	98.23474	0.001376	0	0	0	0	0	0	0	0	0	0
2	0.001384	0	0	0	0	0	0	0	0	0	0	0
3	0.000692	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	81.88158	0	0.012679	0.012679	0	0.012679	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0.013854	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0.013854	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	71.25749	0	0
15	0	0	0	0.006927	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	80.51948	0
17	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	63.15789

SIMILAR	
DISSIMILAR	

Table 5-12 Similarity percentage of regions between second height level, 86 cm, and third height level, 91 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70%

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	98.07859	0	0.004152	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0.002753	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	75.51456	0	0.332502	0.020781	0	0.083126	0	0.034636	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0.006884	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0.013767	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0.061954	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0.041302	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	47.30539	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23.28042	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24.59016

SIMILAR	
DISSIMILAR	

Table 5-13 Similarity percentage of regions between third height level, 91 cm, and fourth height level, 96 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70%

Region Numbers	Region Numbers																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	97.81017	0.001032	0.017205	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0.004122	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	71.63893	0.006884	0	0.013767	0.123907	0	0	0	0.05507	0	0	0	0.013767	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0.336767	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0.013207	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	71.42857	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9.859155	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

Table 5-14 Similarity percentage of regions between fourth height level, 96 cm, and fifth height level, 101 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70%

Region Numbers	Region Numbers																							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	97.78509	0	0.002748	0.001374	0.005496	0	0	0	0	0	0	0	0	0	0	0	0	0	0.001374	0.002061	0.001374	0.001374	0.001374	0.002405	0
2	0.001035	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	72.27787	0	0	0.184892	0.026413	0.013207	0.013207	0.013207	0.01981	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0.00711	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0.063989	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0.284394	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0.127977	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0.02133	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0.02133	0	0	0	0	0	0	0	0	7.407407	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0.049769	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30.51948	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	50	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

In Figure 5-28 map growing result for threshold value of 70% is presented. The difference from the previous case can be seen clearly for the middle part of the area, encircled in Figure 5-27, which is reconstructed with the connection of only boundary points since they are regarded as similar. Additionally, using only boundary points while growing the regions decrease the computational burden, as expected.

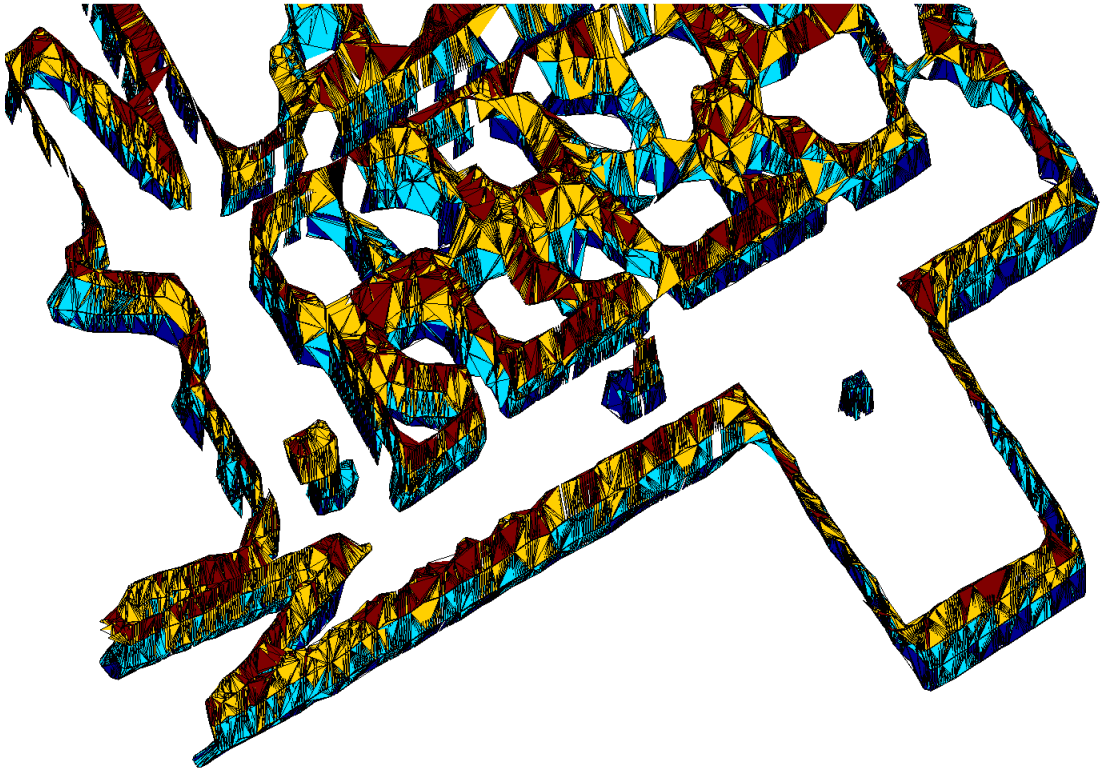


Figure 5-28 Map growing result for 70% threshold value

Lastly, 40% is set as threshold for similarity which separates similar (green) and dissimilar regions as shown in Table 5-15 - Table 5-18. Selection of a low threshold value increases the number of similar regions in total by including the regions that have low matching percentages, such as 63,158% and 47.305%, as can be seen in Table 5-15. Therefore, low similarity threshold means less computational cost since for similar regions boundary points are used in growing.

Table 5-15 Similarity percentage of regions between first height level, 81 cm, and second height level, 86 cm, and designated similar and dissimilar regions with respect to percentage threshold of 40%

Region Numbers	Region Numbers											
	1	2	3	4	5	6	7	8	9	10	11	12
1	98.23474	0.001376	0	0	0	0	0	0	0	0	0	0
2	0.001384	0	0	0	0	0	0	0	0	0	0	0
3	0.000692	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	81.88158	0	0.012679	0.012679	0	0.012679	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0.013854	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0.013854	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	71.25749	0	0
15	0	0	0	0.006927	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	80.51948	0
17	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	63.15789

SIMILAR	
DISSIMILAR	

Table 5-16 Similarity percentage of regions between second height level, 86 cm, and third height level, 91 cm, and designated similar and dissimilar regions with respect to percentage threshold of 40%

Region Numbers	Region Numbers															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	98.07859	0	0.004152	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0.002753	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	75.51456	0	0.332502	0.020781	0	0.083126	0	0.034636	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0.006884	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0.013767	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0.061954	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0.041302	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	47.30539	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23.28042	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24.59016

SIMILAR	
DISSIMILAR	

Table 5-17 Similarity percentage of regions between third height level, 91 cm, and fourth height level, 96 cm, and designated similar and dissimilar regions with respect to percentage threshold of 40%

Region Numbers	Region Numbers																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	97.81017	0.001032	0.017205	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0.004122	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	71.63893	0.006884	0	0.013767	0.123907	0	0	0	0.05507	0	0	0	0.013767	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0.336767	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0.013207	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	71.42857	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9.859155	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

Table 5-18 Similarity percentage of regions between fourth height level, 96 cm, and fifth height level, 101 cm, and designated similar and dissimilar regions with respect to percentage threshold of 40%

Region Numbers	Region Numbers																								
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	97.78509	0	0.002748	0.001374	0.005496	0	0	0	0	0	0	0	0	0	0	0	0	0	0.001374	0.002061	0.001374	0.001374	0.001374	0.002405	0
2	0.001035	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	72.27787	0	0	0.184892	0.026413	0.013207	0.013207	0.013207	0.01981	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0.00711	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0.063989	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0.284394	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0.127977	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0.02133	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0.02133	0	0	0	0	0	0	0	0	7.407407	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0.049769	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30.51948	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	50	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

In Figure 5-29 map growing result for threshold value of 40% is presented. In this case difference from previous case is small and seen for a smaller region which is encircled (red) in the figure. In order to illustrate the effect of smaller threshold a different perspective is utilized for presenting map growing result. In Figure 5-29 it can be seen that there is a hole inside the small region next to the larger region in the middle part of the area which shows that with the selection of the low threshold value, regions that have less similarity can also be seen as similar, such as the one encircled in Figure 5-29, consequently the map growing proceeds on the boundary parts which actually prevents the smooth growing and causes a decrease in the detail of the structure.

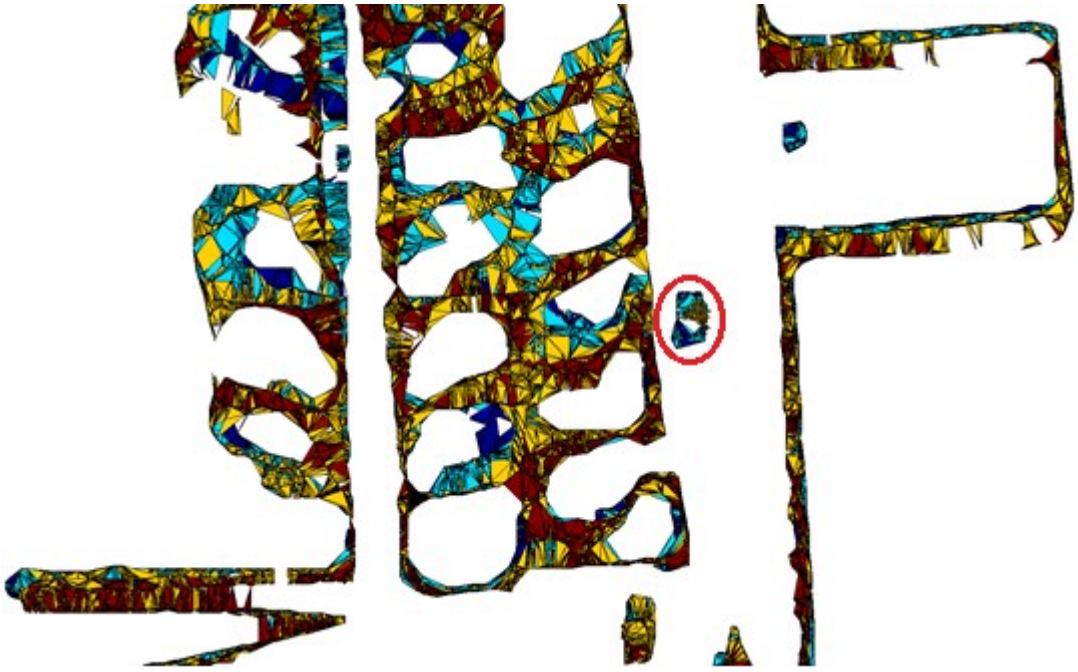


Figure 5-29 Map growing result for 40% threshold value

Additionally, for three different threshold values, the elapsed time and memory utilization values are computed and presented in Table 5-19. According to Table 5-19 connecting correspondence points which are extracted from the constructed tree structures inside a region seems to be the most effective element since time complexity increases drastically for threshold value of 90%. Another point that can be extracted from the Table 5-19 is that decreasing the threshold does not change the

complexity significantly, moreover from the previous analysis, it is experienced that low threshold value decreases the details of the reconstructed structures.

Table 5-19 Elapsed time and memory utilization for different threshold values

Percentage threshold (%)	Elapsed Time (sec)	Utilization of Memory (byte)
90	132,4563	972898304
70	5,3798	958791680
40	4,9841	952487936

In conclusion, in order to avoid time and memory complexity, and obtain results with proper details, this analysis should be conducted and proper values for the threshold parameter should be determined. In the light of the obtained analysis results, using medium, 70% in this analysis, threshold values are found more appropriate for this study.

5.2.2. Analysis Results of Length Limit for Tiling Correspondence Points

Tiling of correspondence points is used in order to reconstruct surfaces of the structures in the environment with the use of triangulation between correspondence points. As explained in Chapter 4, the method used for triangulation is the Delaunay Triangulation (DT) which generates triangulation for a point set on a plane such that none of the points in the set remains inside the circum-circle of any triangle.

Generated triangles can have various sizes, as long as they do not encircle any points, in order to avoid undesired connections between distant regions. Generating connections with limited length is used as a solution to the mentioned problem. Actually limitation of length is chosen as an assumption that should be used to achieve smooth growing. It is important to note that length limit cannot be used as a fixed value since different systems have different parameters that can affect this limit. Therefore, determination of the length limit should be done by considering the system and its parameters, such as height difference between successive maps, resolution of the maps (grid size) and number of grids that can be regarded as neighbors.

In this part of the thesis, an analysis on length limit for tiling correspondence points is conducted and results obtained by using different limits are examined. As previously mentioned, different limits are determined by considering the height differences between successive maps (which is 5 cm), grid size of the maps (which is 5 cm (0.05 m coming from the grid size analysis)), and number of grids that can be regarded as neighbors which is changing, thus adjusting the length limit. As it can be deducted from Figure 5-30 the length limit is determined by using a circle shaped neighborhood region whose radius is computed as a certain number of grids multiplicative of the grid size, and height difference between maps. The reason for using multiplicities of grid size is to determine a proper range in which at least one point can exist. Therefore the formulation of length limit becomes:

$$length_{Limit} = \sqrt{(height\_difference)^2 + (n_{grids} * grid\_size)^2} \quad (5\text{-}1)$$

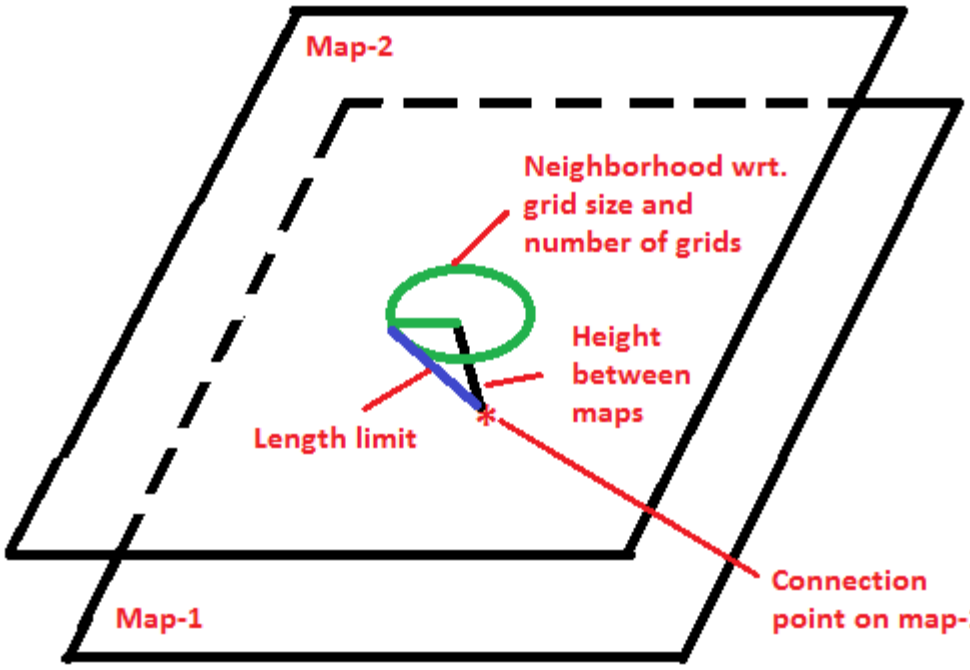


Figure 5-30 Determination of length limit using circle shaped neighborhood region and height difference between maps

Different length limits are computed by using the formula given in (5-1) with the values from Table 5-20 and the results are provided. These length limits are then utilized throughout the analysis and results of each is discussed in order to determine a proper value for length limit.

Table 5-20 Computation of different length limits

Height (cm)	GridSize (cm)	NumberOfGrid	Length Limit (cm)
5	5	1	7,071067812
5	5	2	11,18033989
5	5	3	15,8113883
5	5	4	20,61552813
5	5	5	25,49509757
5	5	6	30,41381265
5	5	7	35,35533906
5	5	8	40,31128874
5	5	9	45,27692569
5	5	10	50,24937811

Firstly 7cm is taken as length limit for tiling correspondence (connection) points and result of map growing is shown in Figure 5-31 and Figure 5-32. In Figure 5-31 top view of the reconstructed area with length limit of 7 cm is presented. At the top part of the figure, misalignments caused by the errors which are encountered during map construction and alignment of the maps are emphasized in red circles. However, these misaligned parts are not connected to each other with the utilized length limit (7 cm) which can also be seen in Figure 5-32.

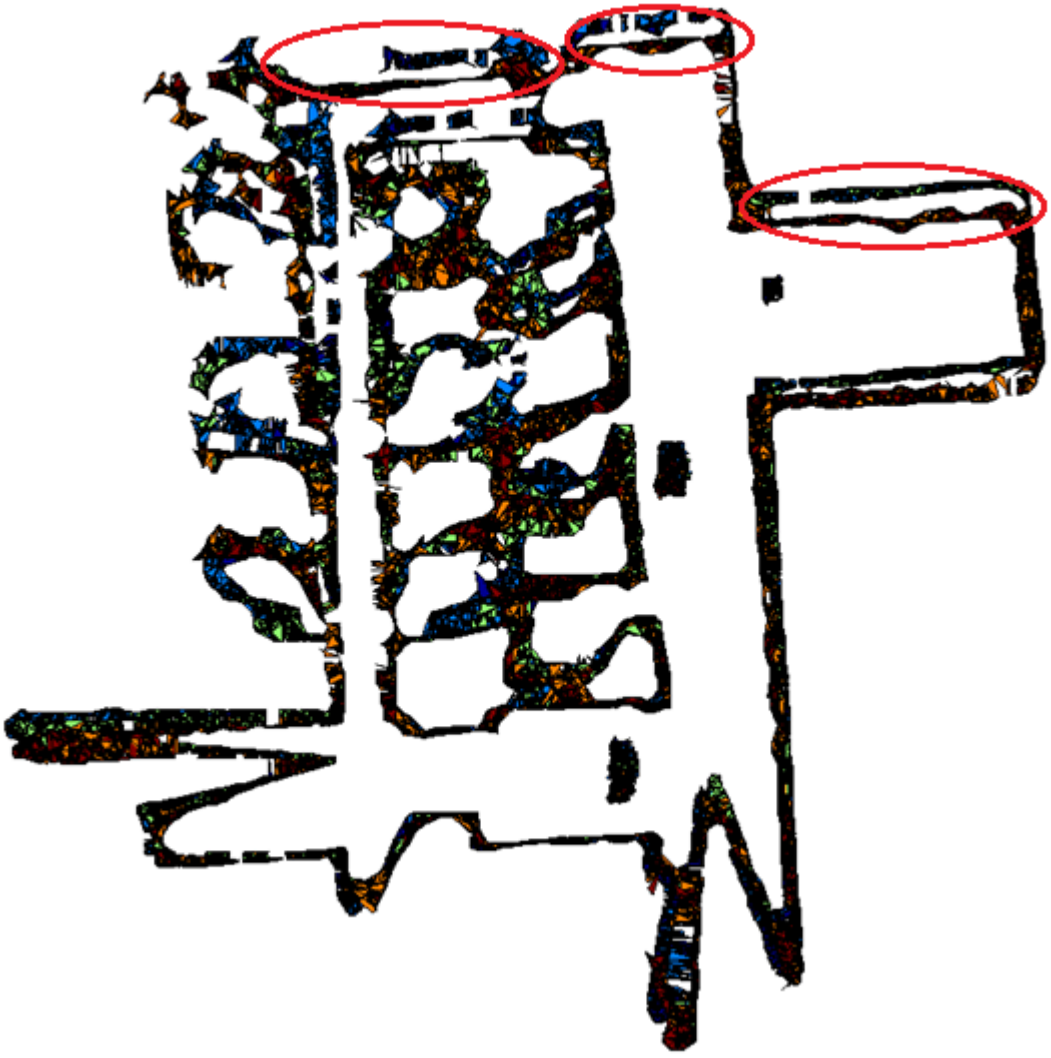


Figure 5-31Top view of the reconstructed area with length limit of 7 cm

In Figure 5-32 a side view of the reconstructed area with length limit of 7 cm is presented. Misaligned areas which are not connected at the top view can be seen in the side view clearly. As it can be seen there are holes, which are caused by the errors in alignment cannot be recovered at the tiling part with the given length limit, in the walls of the reconstructed area.

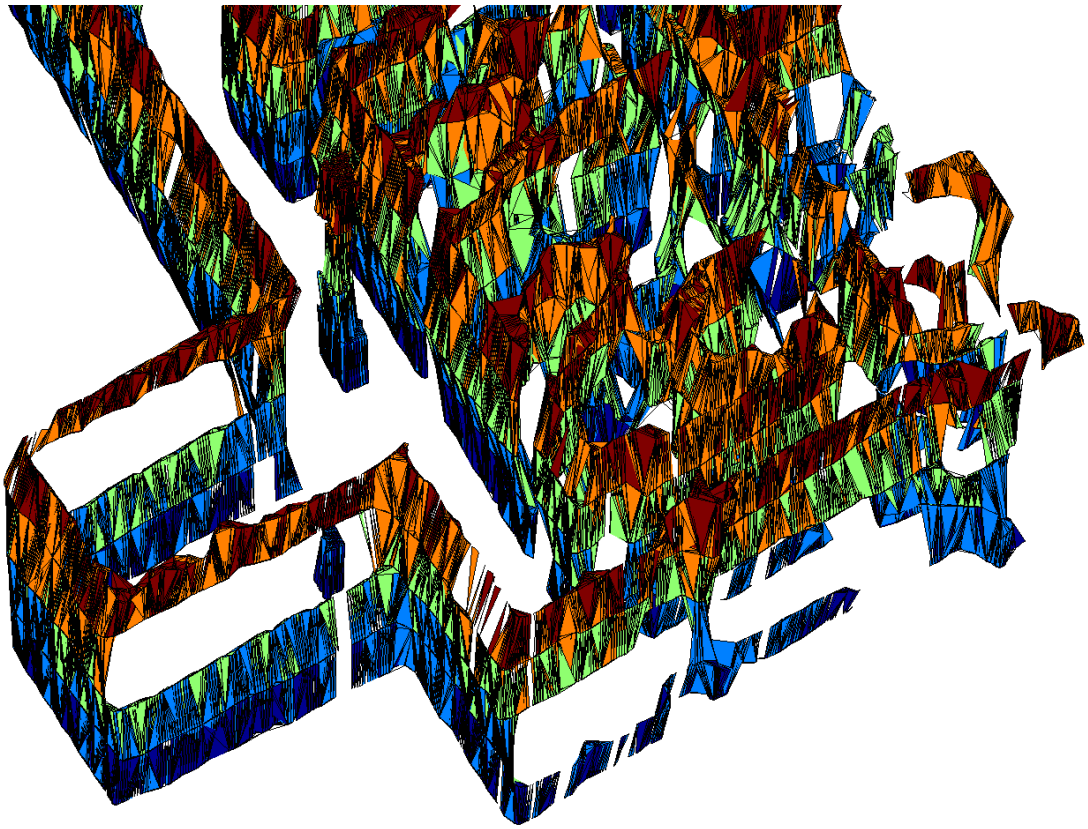


Figure 5-32 Side view of the reconstructed area with length limit of 7 cm

In the second step 11 cm is taken as length limit for tiling correspondence (connection) points and result of map growing is shown in Figure 5-33 and Figure 5-34. In Figure 5-33 top view of the reconstructed area with length limit of 11 cm is presented. At the top part of the figure, misalignments caused by the errors which are encountered during map construction and alignment of the maps are emphasized in red circles. When compared to the Figure 5-31, there are few connections established between misaligned parts, shown in red circles, since the length limit is increased and creating connections become possible.

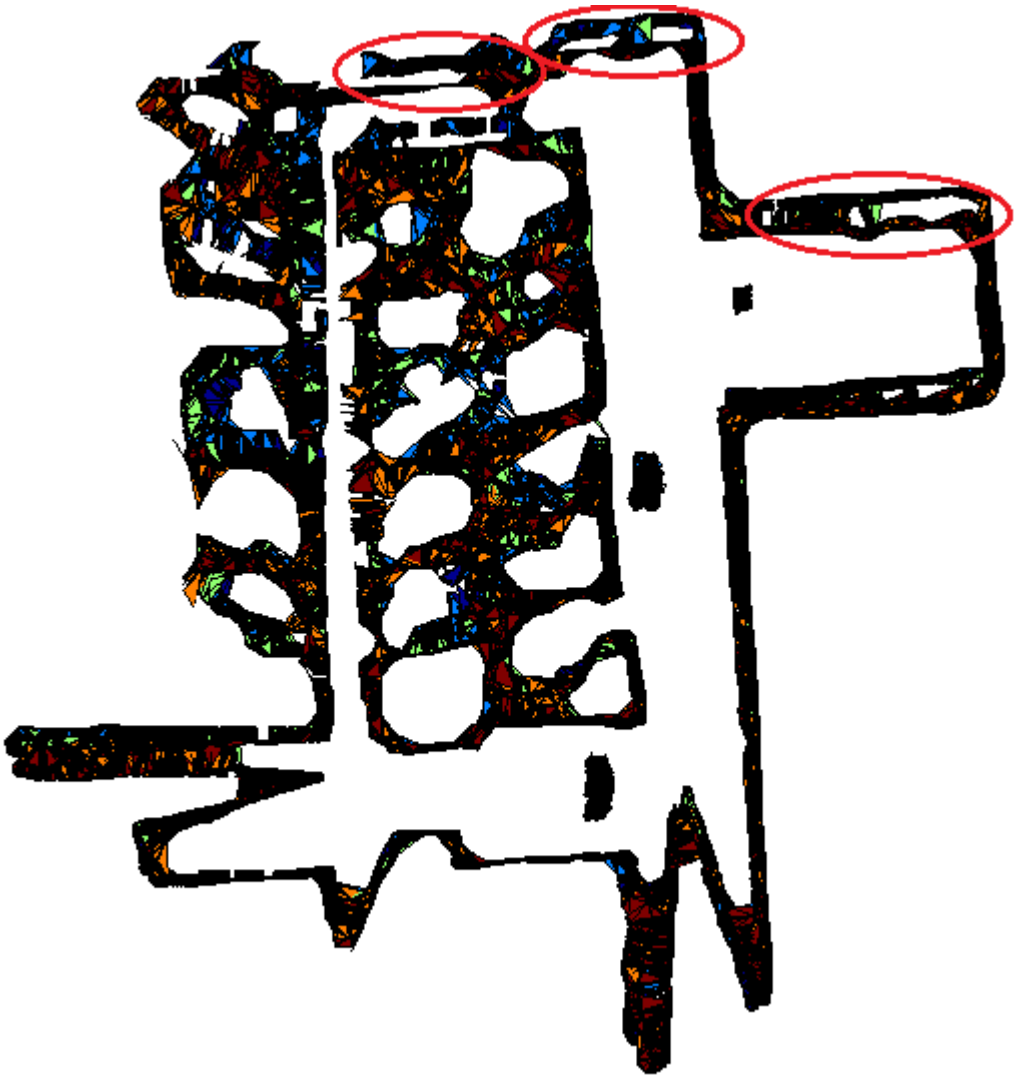


Figure 5-33 Top view of the reconstructed area with length limit of 11 cm

In Figure 5-34 a side view of the reconstructed area with length limit of 11 cm is presented. From this perspective, connections between misaligned parts which do not exist in Figure 5-32 can be seen clearly (emphasized with red circles). This is because as the length limit increases misaligned parts which are close to each other can establish connections between each other. However, holes at the walls are not completely recovered with this length limit.

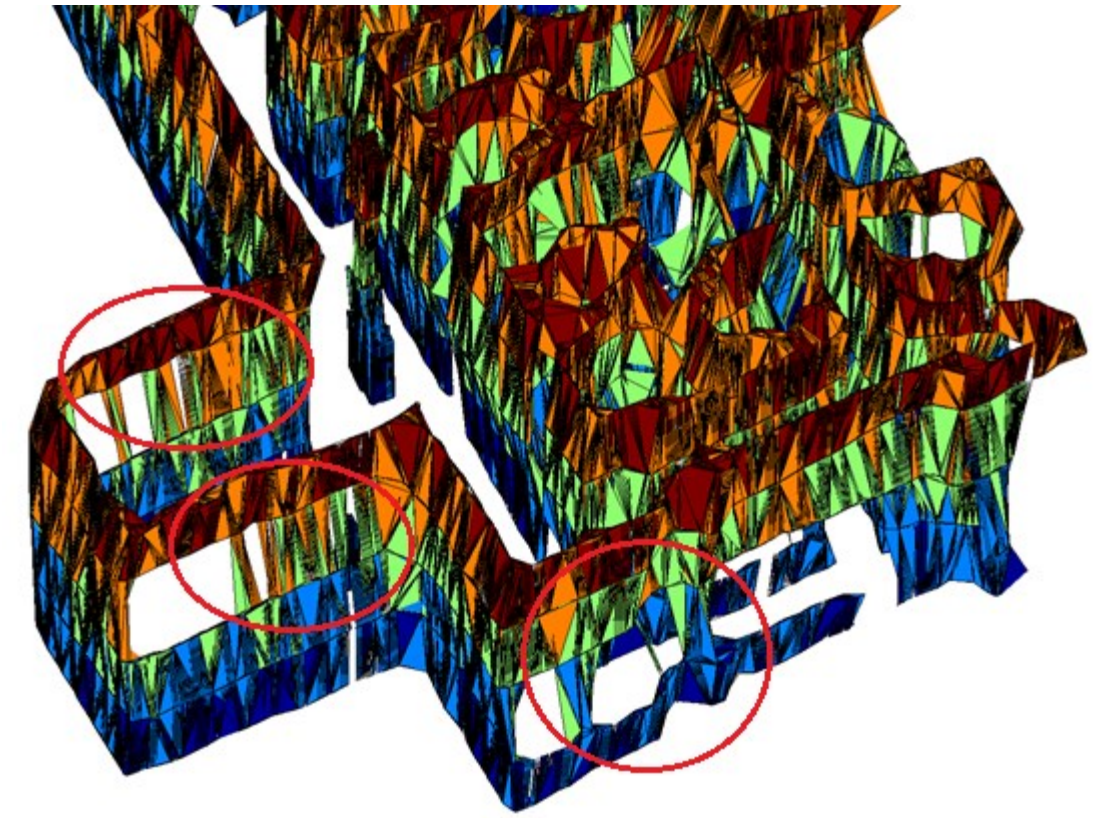


Figure 5-34 Side view of the reconstructed area with length limit of 11 cm

In the third step 15 cm is taken as length limit for tiling correspondence (connection) points and result of map growing is shown in Figure 5-35 and Figure 5-36. In Figure 5-35 top view of the reconstructed area with length limit of 15 cm is presented and it can be seen that approximately all misaligned parts have connections. However, as the length limit increases, close regions from different height levels which are not actually matched with each other (mostly seen in corridors), start to generate connection among themselves, which are shown in the figure with red circles, because of errors in alignment of these regions. When errors are encountered in the alignment of the maps, regions can get close to each other more than they actually are. Hence, during the tiling part they can be matched incorrectly and establish connections and for high length limits these connections cannot be avoided.

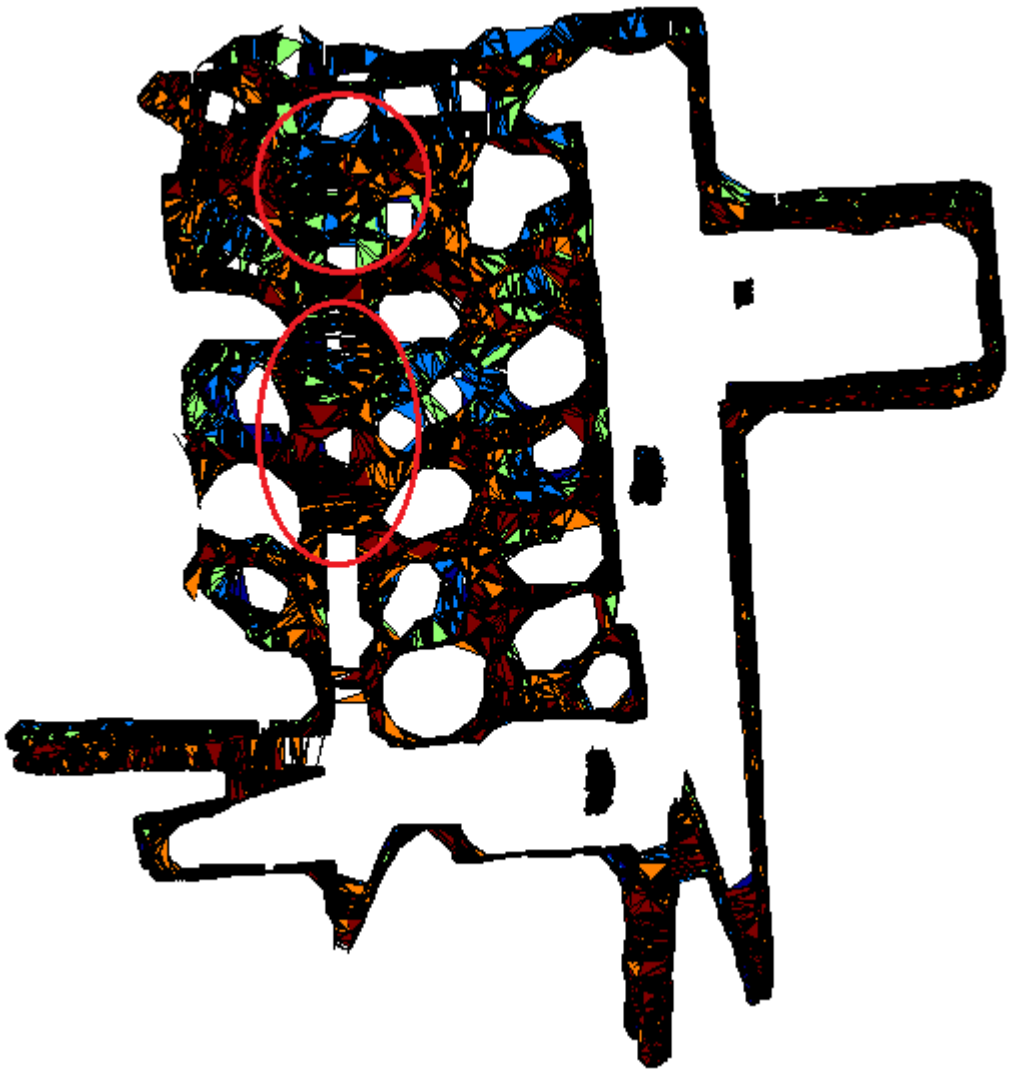


Figure 5-35 Top view of the reconstructed area with length limit of 15 cm

In Figure 5-36 a side view of the reconstructed area with length limit of 15 cm is presented. When compared with the result given in Figure 5-34, in Figure 5-36 regions from different height levels are almost completely connected to each other such that approximately all of the holes on the walls are recovered, which can be seen on the figure emphasized with red circles.

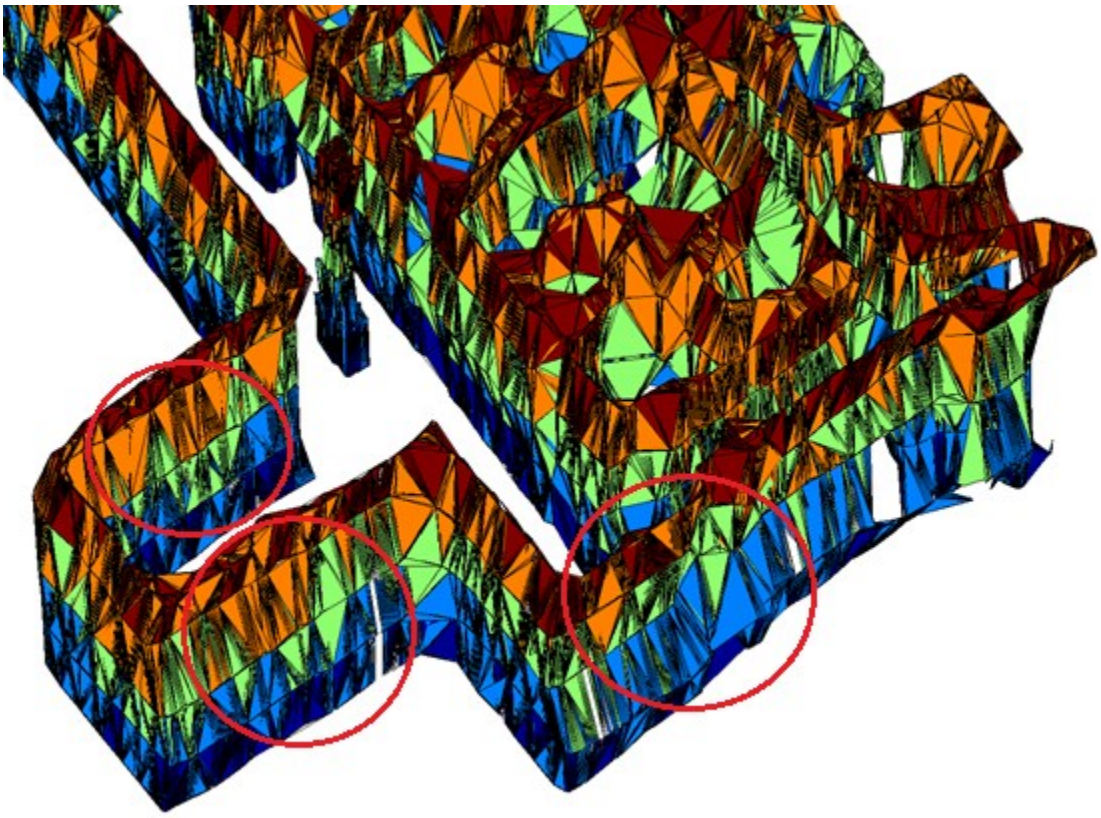


Figure 5-36 Side view of the reconstructed area with length limit of 15 cm

In the fourth step 20 cm is taken as length limit for tiling correspondence (connection) points and result of map growing is shown in Figure 5-37 and Figure 5-38. In Figure 5-37 top view of the reconstructed area with length limit of 20 cm is presented. Connections between misaligned parts get better with the increase in the length limit, but the density of the undesired connections is also increasing at the close parts in the area, such that reconstructed corridor, which is shown with red circle, is occupied with this connections, when compared to the result given in Figure 5-35.

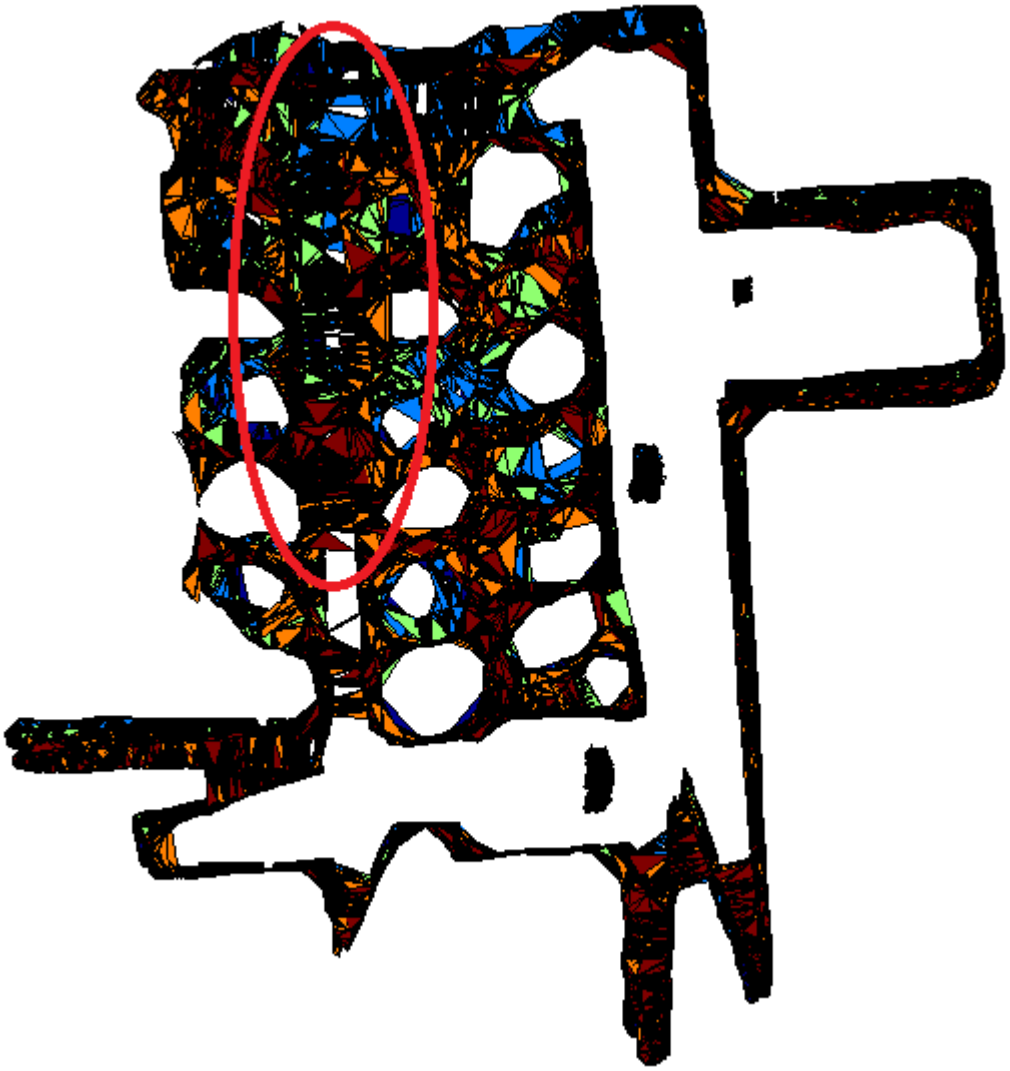


Figure 5-37 Top view of the reconstructed area with length limit of 20 cm

In Figure 5-38 a side view of the reconstructed area with length limit of 20 cm is presented. It can be seen that connections on the walls are approximately the same with the result given in Figure 5-36. Therefore, it can be said that no significant change is observed when compared to the previous case, so no improvement is achieved for this part with the increase in the length limit.

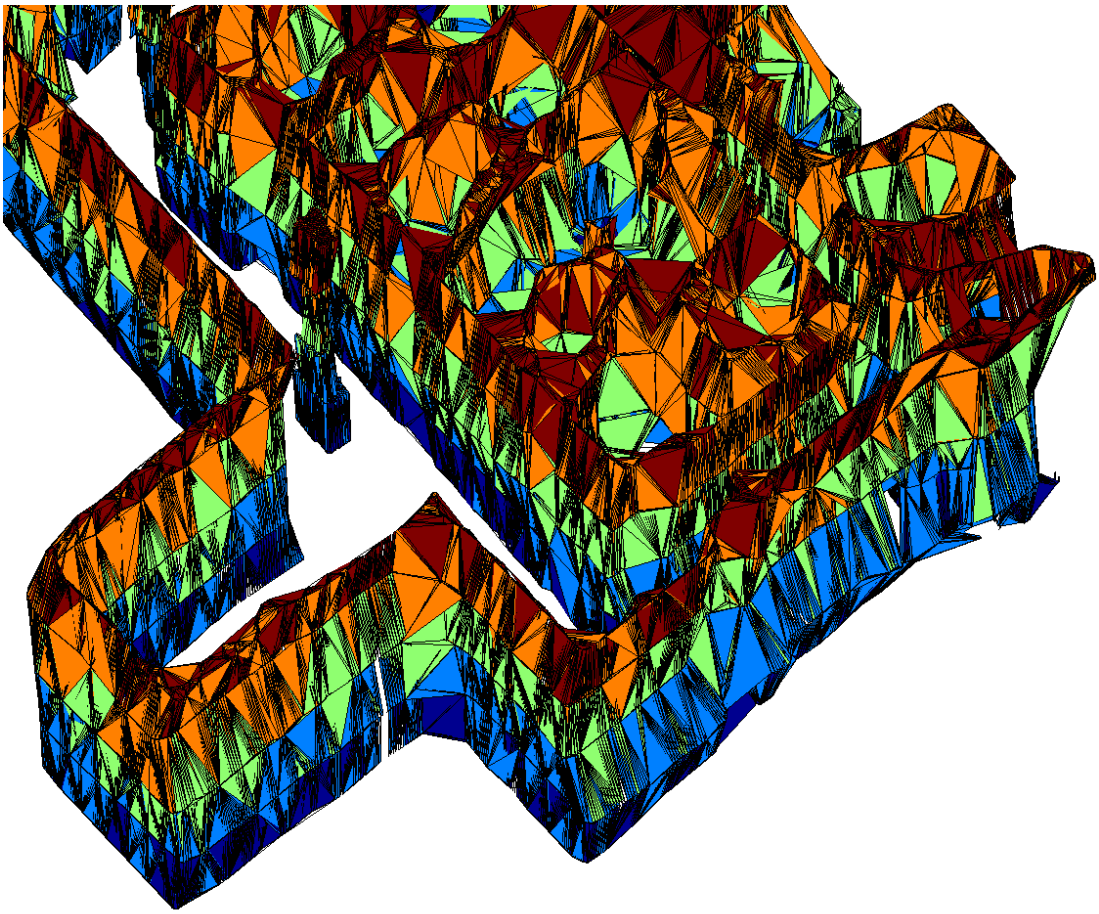


Figure 5-38 Side view of the reconstructed area with length limit of 20 cm

After the fourth step it does not seem necessary to continue the analysis by increasing length limits since there are no improvements, and on the contrary more undesired connections are obtained.

In conclusion, length limit used in the tiling part has two major effects on map growing results which are both related to the misalignments between the maps. First one is about the misaligned but matching regions of the maps which are desired to have connections among each other. It is evaluated that for low values of length limit enough connections may not be created between the matching regions so that holes are observed inside the structures. As the length limit increases, the number of connections also increases, so that all holes are recovered.

The second one is about the misaligned and mismatching regions of the maps for which connections between each other are not desired. It is experienced that for low values of length limit, connections between misaligned regions are not observed. However, as the length limit increases, undesired connections between mismatching regions from different height levels are created. These yields that length limit should not be too low so that connections between misaligned regions can be established and should not be too high so that connections between mismatching regions can be avoided. Proper value for the length limit should be determined by considering these two criterions.

In the light of these results, it is important to apply the sensitivity analysis to length limit for tiling correspondence points to the system to work with, in order to properly determine the parameter of length limit before obtaining the final results from map growing.

5.3. Chapter Summary

In this chapter, the sensitivity analysis conducted on Fast-SLAM and map growing methods are presented. The main aim is to observe the effects of parameters and designate proper values for the parameters used in both methods. For Fast-SLAM method analyzed parameters and motivations for the analysis can be summarized as follows:

- **Grid Size**: Grid size has a direct effect on resolution of the grid map which is an important property while working on the grid maps for map growing. When the resolution is high, reconstructed structures contains more details.
- **Particle Number**: Particle number has an effect on correct localization of the mobile robot which is achieved by incremental error correction of pose estimates. As the particle number increases, rate of localization error decreases.
- **Probability Change Rates:** Probability change rates are utilized while assigning probability value to a grid for being occupied when generating

occupancy grid maps. During the assignment of the probability value, current probability value is changed with probability change rate for being occupied (m_{occ}) or probability change rate for being free space (m_{free}).

For map growing method analyzed parameters and motivations for the analysis can be summarized as follows:

- **Percentage Threshold for Separation of Regions**: Percentage threshold directly affects the time and memory complexity, and details of the map construction. Therefore, this analysis should be conducted and proper values for the threshold parameter should be determined.
- **Length Limit for Tiling**: Length limit affects the quality of connection of the correspondence points and reconstructed structure by preventing undesired connections between incorrect matches and connecting misaligned parts up to the allowed limit. Therefore, it is important to apply analysis of length limit for tiling correspondence points to the system to work with, in order to determine the parameter of length limit properly before obtaining final results from map growing.

CHAPTER 6

EXPERIMENTAL RESULTS

Methodologies proposed and presented in the present thesis are implemented on hardware and results are presented in this chapter. Experiments done with the proposed methodologies are based on three different settings:

1- Indoor area exploration from different height levels.
2- Scanning an object-1
3- Scanning an object-2

The steps followed for each experiment can be explained as given in Figure 6-1 which is the block diagram of the whole system that explains the inputs-outputs and functionality of each block or method.

In the first part of this chapter, heterogeneous vehicles used in the thesis are presented. Afterwards, conducted experiments are explained in detail according to the order provided by the block diagram given in Figure 6-1.

6.1.General System Architecture

The system flow diagram presented in Figure 6-1, summarizes steps of each experiment. At first, the indoor environment is explored with heterogeneous vehicles and obtained measurement data is used in FastSLAM algorithm to generate 2D grid maps. Afterwards, meaningful regions are segmented from 2D maps and these regions are used to align consecutive maps with respect to each other by using line matching and ICP methods. Then, similar and dissimilar regions between consecutive maps are separated and for dissimilar regions system feedbacks are generated. Successively, correspondence points are generated for similar and dissimilar regions by using boundary points of similar regions and vertices on the

tree structures constructed inside the dissimilar regions. Then, these correspondence points are connected together by using triangular meshing method, the Delaunay Triangulation, and surfaces are reconstructed in 3D.

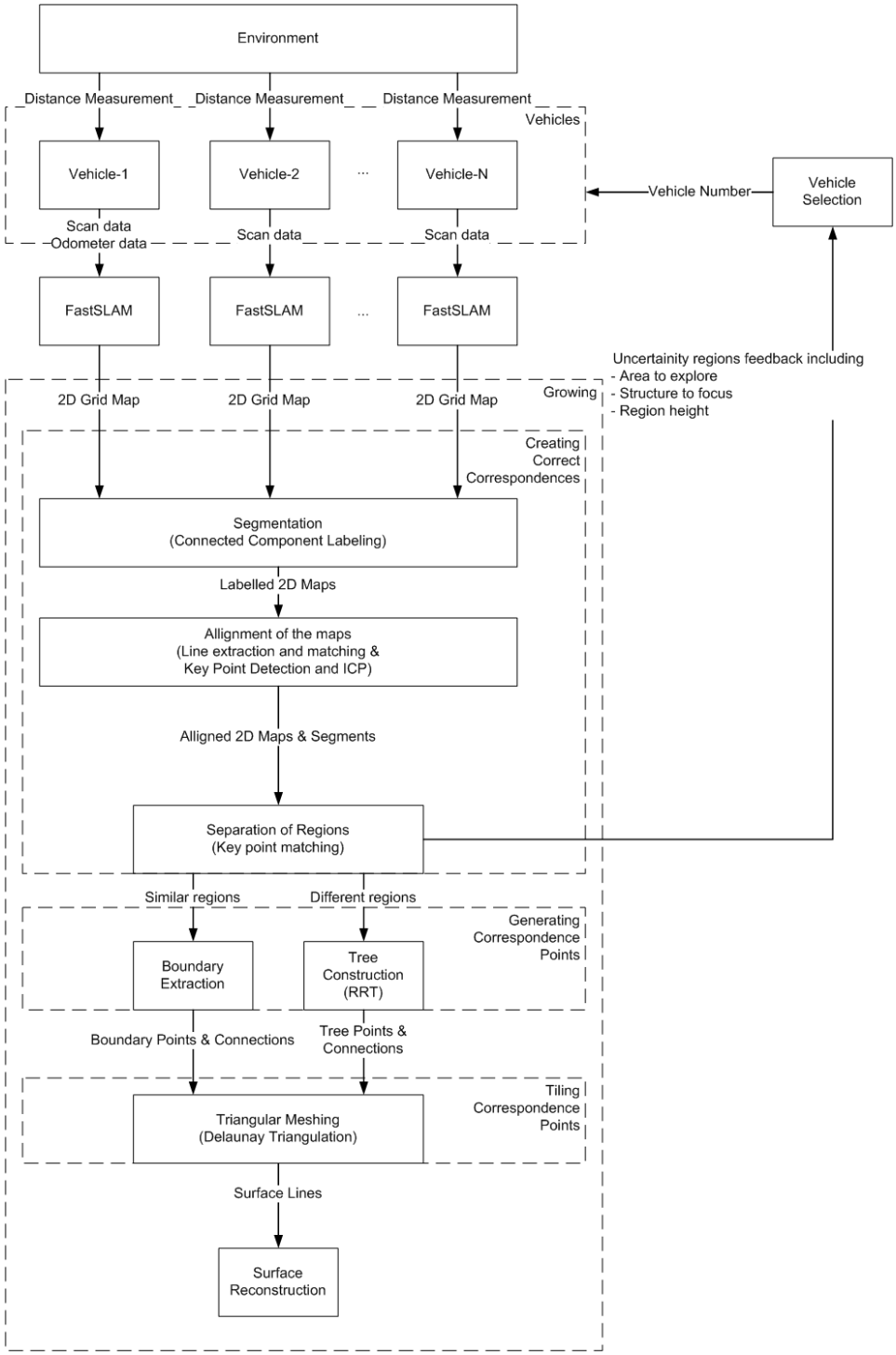


Figure 6-1 Block diagram of the system

6.2.Vehicles

6.2.1. Vehicle 1

The first vehicle used for the experiment is a Pioneer 3AT robot with a differential drive capability. The hardware components on the vehicle, shown in Figure 6-2 can be listed as follows:

1. SICK OEM1000 laser scanner, 360 degree scan with maximum range of 80 meters.
2. Embedded PC
3. Motor controllers
4. Battery, 12V 7Ah, lead acid
5. Encoders at each motor and IMU
6. Bullet M2 Titanium WiFi & Wireless Bridge
7. Antenna

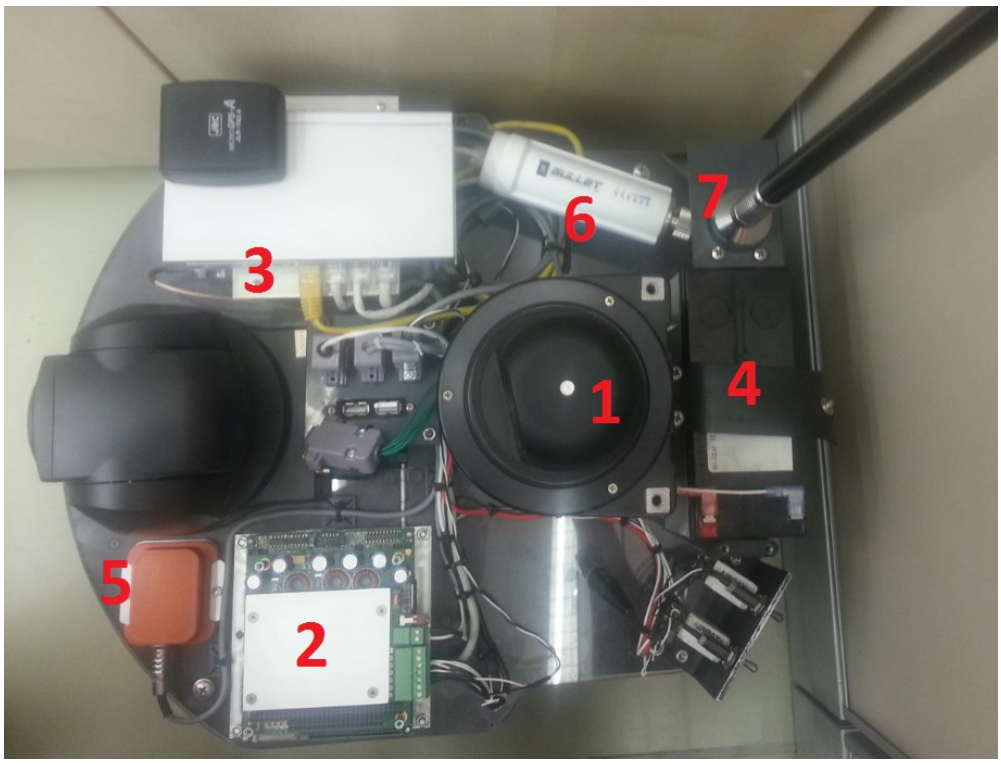


Figure 6-2 Vehicle-1 and its hardware components

Vehicle-1 is controlled with commands coming from joystick attached to the computer and the given commands are transferred to the vehicle via WiFi module of the computer. Software that handles incoming commands and data packaging is downloaded to the embedded pc on the vehicle.

6.2.2. Vehicle 2

The second vehicle used for the experiments is a manually driven carrier, which has a platform whose height is adjustable between 28cm and 53cm, has the ability to carry vehicle 1 on top of it. Vehicle 2 makes it possible to work at different height levels in the environment and has different dynamics when compared to Vehicle 1 which affects the motion model used in the SLAM algorithm. Additionally, there is no odometer or IMU sensor on the vehicle in order to measure the motion directly. Instead laser scans obtained at different times are used for scan matching and measurement of the motion is estimated. Hardware components on the vehicle, shown in Figure 6-3 can be listed as follows:

1. Vehicle 1 (Pioneer 3AT)
2. Manually driven carrier with height adjustable platform

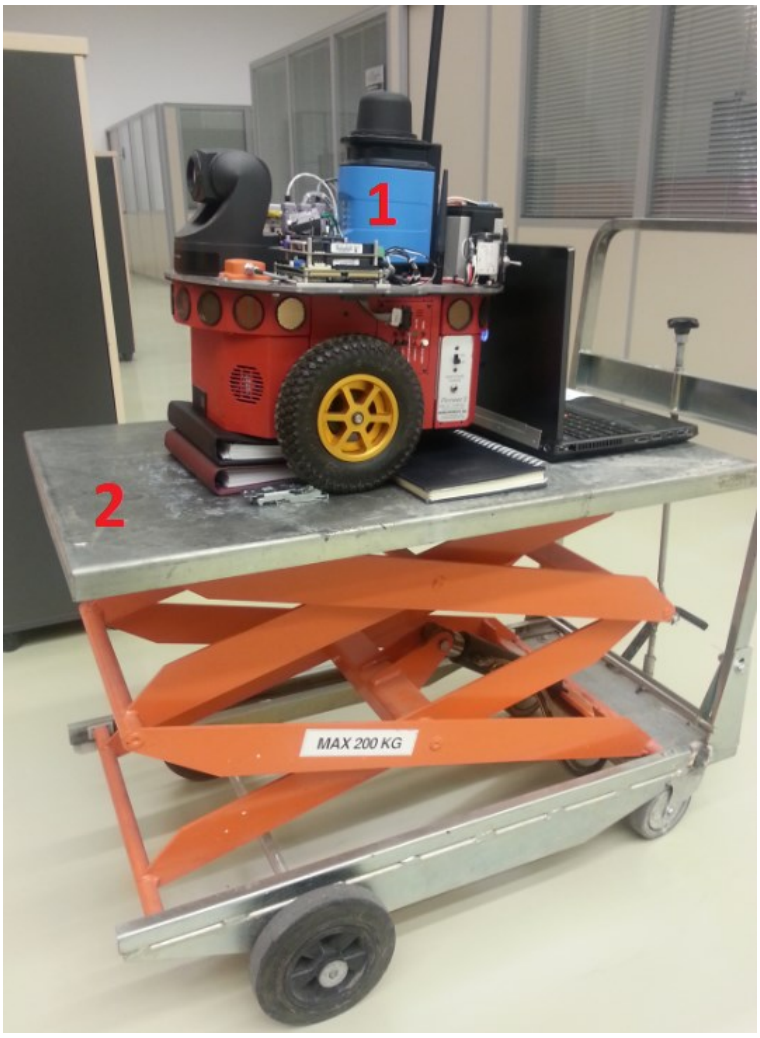


Figure 6-3 Vehicle-2 and its hardware components

6.2.3. Vehicle 3

The third vehicle used for the experiments is a manually driven carrier that has a platform with height fixed at a height value of 13cm unlike the case for Vehicle-2 where height is adjustable. Vehicle-3 makes it possible to achieve an intermediate height level between the height achieved by Vehicle-1 and minimum height achieved by Vehicle-2 and has different dynamical characteristics when compared to other two vehicles. Similar to the Vehicle-2 there is no odometer or IMU sensors that can provide direct motion measurement, rather its motion is estimated using the scan matching method. Hardware components on the vehicle, shown in Figure 6-4 can be listed as follows:

1. Vehicle 1 (Pioneer 3AT)
2. Manually driven carrier with fixed height platform

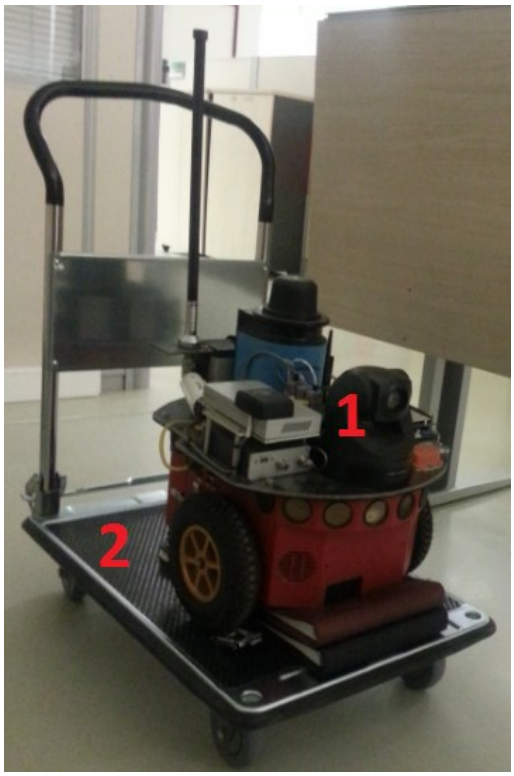


Figure 6-4 Vehicle-3 and its hardware components

6.3. Experiment 1: Indoor Area Exploration from Different Height Levels

6.3.1. Environment of the Experiment 1

The environment used for Experiment-1 is an indoor environment in ASELSAN which is explored at 5 different height levels which are 81cm, 86cm, 91cm, 96cm, 101cm.Images from the environment are provided in Figure 6-5 to Figure 6-8 in order to visualize the environment in which experiment is conducted. In Figure 6-5, a view from the explored environment is given which provides information about the relative locations of the two structures in the environment and a corner point on the wall. In Figure 6-6, Figure 6-7 and Figure 6-8, three different structures inside the environment, which are localized and grown at the end of the experiment, is presented.

Figure 6-5 A view from the explored environment

Figure 6-6 Image of the object number 1

Figure 6-7 Image of the object number 2

Figure 6-8 Image of the object number 3

In Experiment-1, Vehicle-2 is utilized to explore the environment from 5 different height levels with height difference of 5 cm beginning from 81cm as the minimum height value.

6.3.2. Fast-SLAM Exploration Results in Experiment 1

The obtained laser scans are used in the Fast-SLAM algorithm based on the following parameters:

- Grid size: 0.05 m
- Particle number: 500
- Probability change rates:
  - $m_{occ} = -0.5$
  - $m_{free} = +0.05$

2D grid maps for all different height levels are obtained and presented in Figure 6-9 – Figure 6-13. In Figure 6-9, 2D mapping result for the exploration of the indoor environment at 81cm height is provided. Observed objects (encircled by red) and rooms (encircled by blue) in the environment can be seen in the constructed map. Regions that are out of the range limit of the scanner are left uncertain.

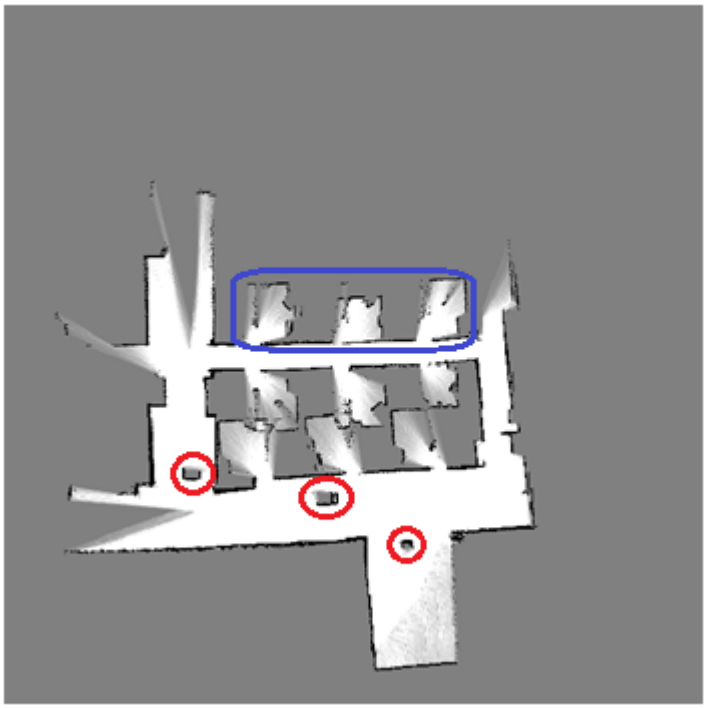


Figure 6-9 2D grid map obtained from Fast-SLAM at first height level, 81 cm, for Experiment-1

In Figure 6-9, 2D mapping result for the exploration of the indoor environment at 86 cm height is provided. Previously observed objects, emphasized in Figure 6-9, are also observed in Figure 6-9, but there are slight differences between the mapping results such as the orientation of the maps, since each map is localized in their local frames.

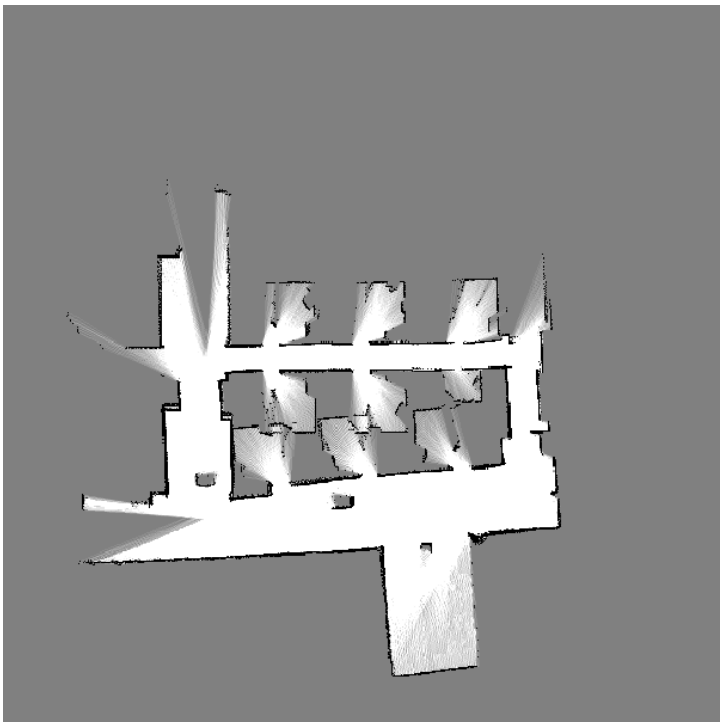


Figure 6-10 2D grid map obtained from Fast-SLAM at second height level, 86 cm, for Experiment-1

In Figure 6-11, 2D mapping result for the exploration of the indoor environment at 91cm height is provided. It can be seen that orientation of the 2D map changes considerably which is handled in the map growing part during alignment of the maps.

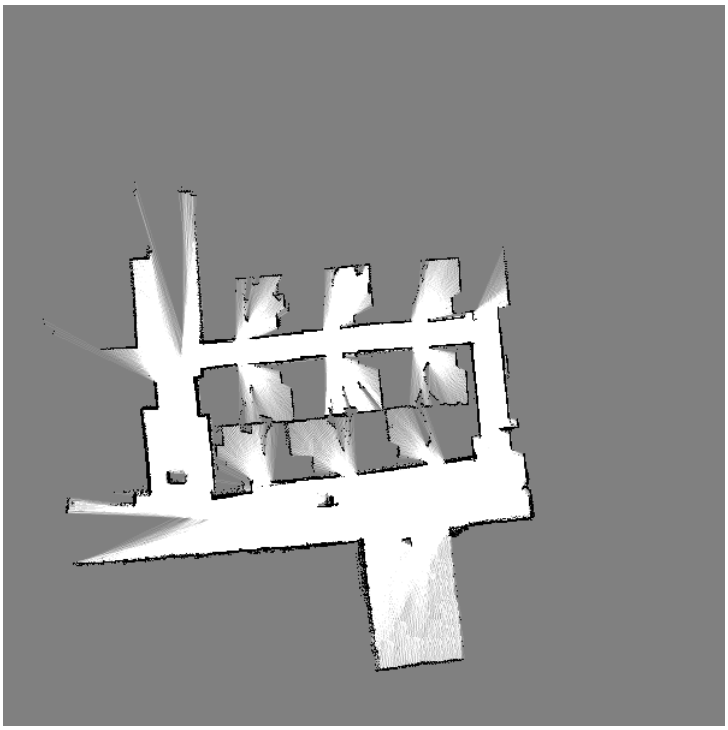


Figure 6-11 2D grid map obtained from Fast-SLAM at third height level, 91 cm, for Experiment-1

In Figure 6-12, 2D mapping result for the exploration of the indoor environment at 96cm height is provided. One can see that one of the objects in the environment, emphasized by red circle, becomes smaller when compared to previous results. Additionally, there is a mapping error for the rooms at the top of the map such that two rooms are missing (locations are shown by blue circles) and the rooms next to them shift to the left when compared to the previous results

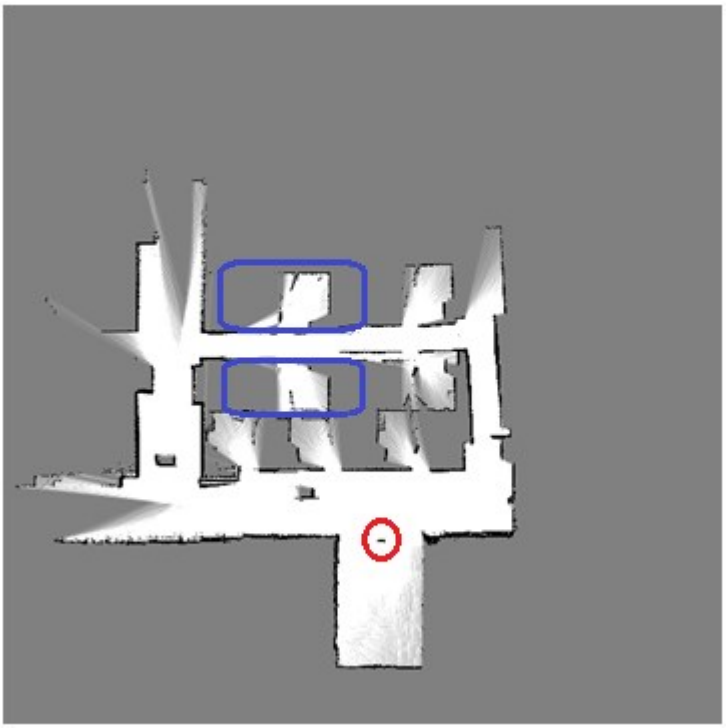


Figure 6-12 2D grid map obtained from Fast-SLAM at fourth height level, 96 cm, for Experiment-1

In Figure 6-13, 2D mapping result for the exploration of the indoor environment at 101 cm height is provided. It can be seen that the object which becomes smaller in Figure 6-12, disappears in this figure (position of the object is shown with red circle). Moreover, area of an object gets smaller which is emphasized in the figure by blue circle.

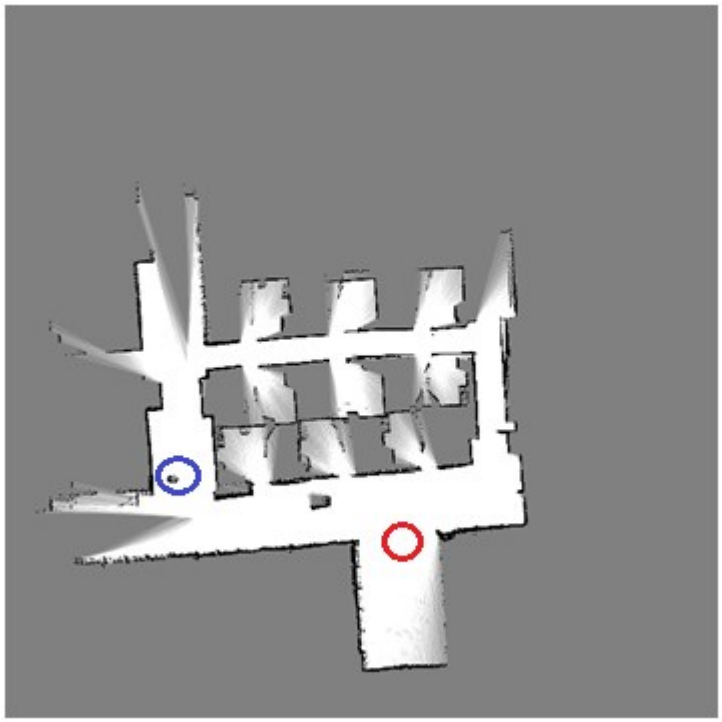


Figure 6-13 2D grid map obtained from Fast-SLAM at fifth height level, 101 cm, for Experiment-1

6.3.3. Map Growing

After obtaining 2D grid maps for different height levels, map growing method is applied in the experiment. Initially, correct correspondences are created between successive maps by first segmenting meaningful regions, then aligning maps by using features extracted from these regions and finally, separating the regions as similar and dissimilar according to similarity measure between them. Afterwards, correspondence points are generated in order to be used while connecting regions on the maps for reconstruction purpose which is defined in the final step namely; tiling of the correspondence points. In map growing method following values for the parameters are utilized.

- Percentage Threshold for Separation of Regions: 70%
- Length Limit for Tiling: 11 cm

The subsequent sections illustrate each of the aforementioned steps.

6.3.3.1. Creating Correct Correspondences

In order to visualize the achieved alignments between successive maps, boundaries of the segmented regions are utilized. In Figure 6-14 alignment between first height level (red), 81cm, and second height level (blue), 86cm, is presented. The aim is to illustrate the alignment results of the maps with respect to each other. The walls of the environment and the middle area, rooms, are aligned properly as it can be seen in the figure. There are some parts at the top of the map, shown with green circle, which are not aligned because of large position and orientations differences with respect to each other.

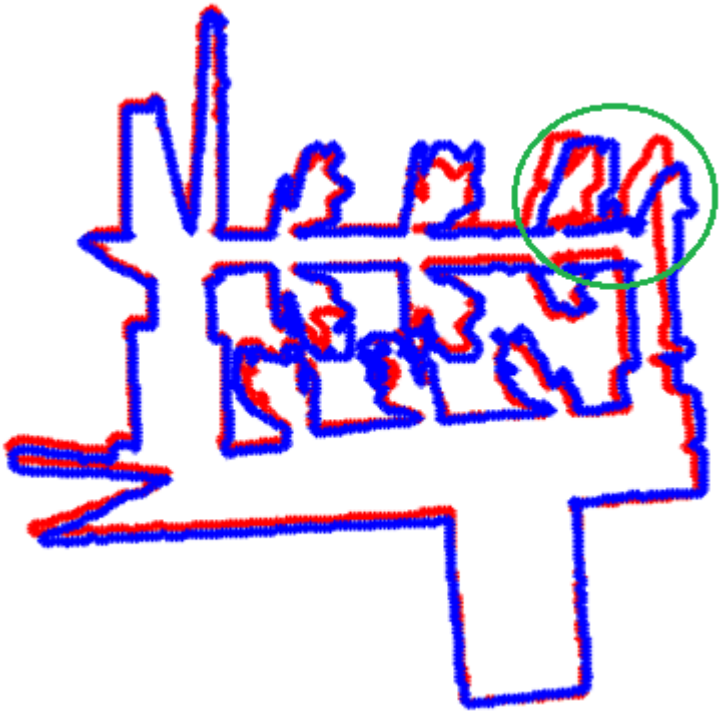


Figure 6-14 Achieved alignment between first height level (red), 81 cm, and second height level (blue), 86 cm, for Experiment-1

In Figure 6-15 alignment between second height level (red), 86cm, and third height level (blue), 91cm, is presented. When compared to the result in Figure 6-14, there is a larger alignment error at the right part of the map, shown with green circle, caused

mostly by the errors in mapping process since remaining parts of the maps are aligned properly.

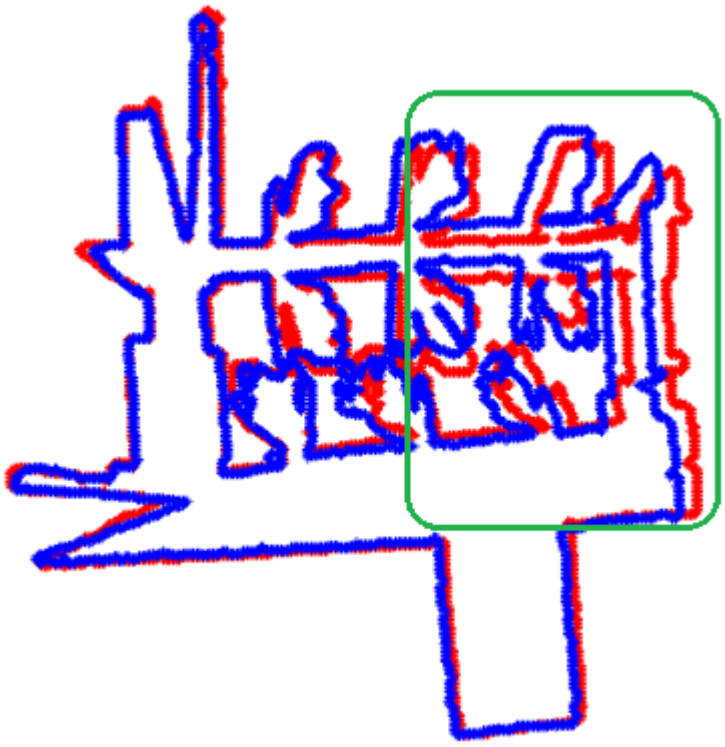


Figure 6-15 Achieved alignment between second height level (red), 86 cm, and third height level (blue), 91 cm, for Experiment-1

In Figure 6-16 alignment between third height level (red), 91cm, and fourth height level (blue), 96 cm, is presented. In this result, the effect of the mapping error, which is about missing rooms at the top of the map that is observed previously in Figure 6-12, to the alignment of the maps is seen clearly which is shown with green circle.

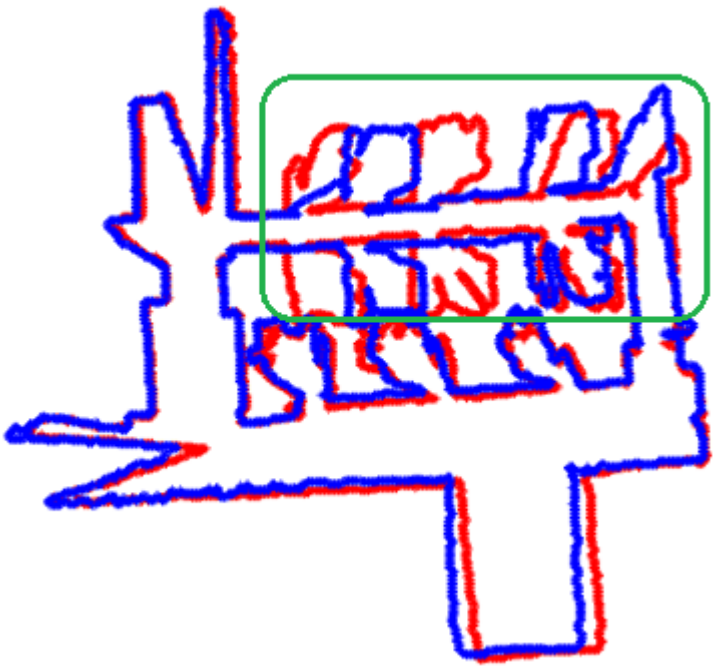


Figure 6-16 Achieved alignment between third height level (red), 91 cm, and fourth height level (blue), 96 cm, for Experiment -1

In Figure 6-17 alignment between fourth height level (red), 96 cm, and fifth height level (blue), 101 cm, is presented. The same alignment error that is previously seen in Figure 6-16, caused by the mapping error observed in Figure 6-12 is also seen in Figure 6-17 (emphasized with the green circle).

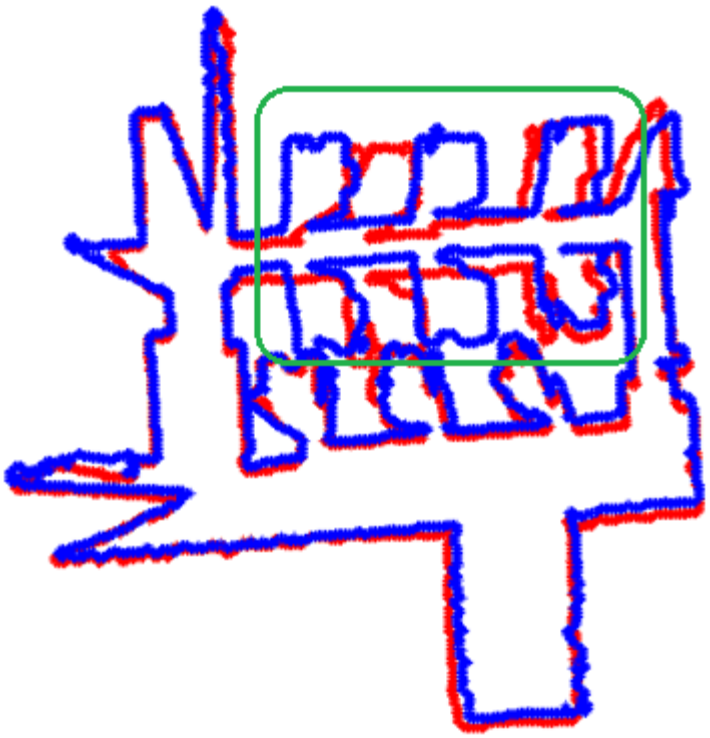


Figure 6-17 Achieved alignment between fourth height level (red), 96 cm, and fifth height level (blue), 101 cm, for Experiment-1

After the alignment of the maps, similarities between regions is measured in percentage by applying grid by grid comparison and the percentage threshold of 70% is used to designate the similar and dissimilar regions, results of which are presented in Table 5-11 - Table 5-14. It can be seen that similarity percentage of a region varies for comparison at different height levels, which means a region can be similar with a height level and dissimilar with other one or visa versa.

Additionally, feedbacks are generated to the system level by using the dissimilar regions. These are at approximately the same positions for both maps and regions and are provided along with their height information, which are given in the explanation of each figure. While comparing positions of the regions, their center points are utilized. If the distance between center points are lower than a defined threshold then positions of the center points are considered as same.

In Figure 6-18 dissimilar regions between first hight level, 81cm, and second height level, 86 cm, are presented. It can be seen that dissimilar regions between given height levels are small which means discontinuity between given height levels is low.

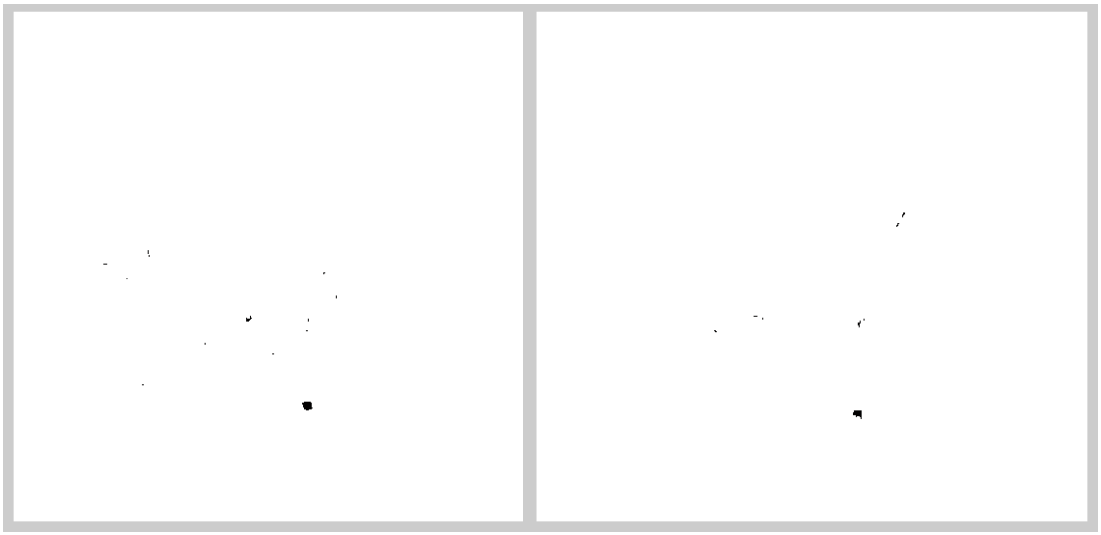


Figure 6-18 Dissimilar regions between first height level (left), 81 cm, and second height level (right), 86 cm for Experiment-1

In Figure 6-19 regions, which are among the dissimilar regions between first height level, 81cm, and second height level, 86 cm, whose positions are same on the maps of both height levels are presented and highlighted by circles with the same color. Positions of these regions are provided as feedbacks to the controller of the vehicles.

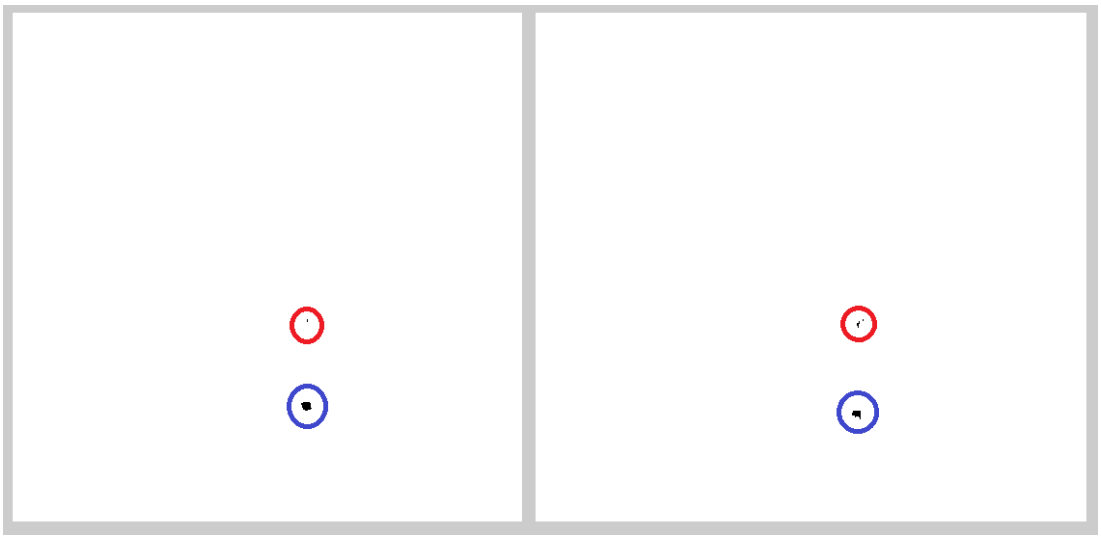


Figure 6-19 Regions to focus on for both first height level (left), 81 cm, and second height level (right), 86 cm, for Experiment-1

In Figure 6-20 dissimilar regions between second height level, 86 cm, and third height level, 91 cm, are presented. One can observe larger dissimilar regions between given height levels when compared to regions in Figure 6-18, which means discontinuity between given height levels is higher than previous case.

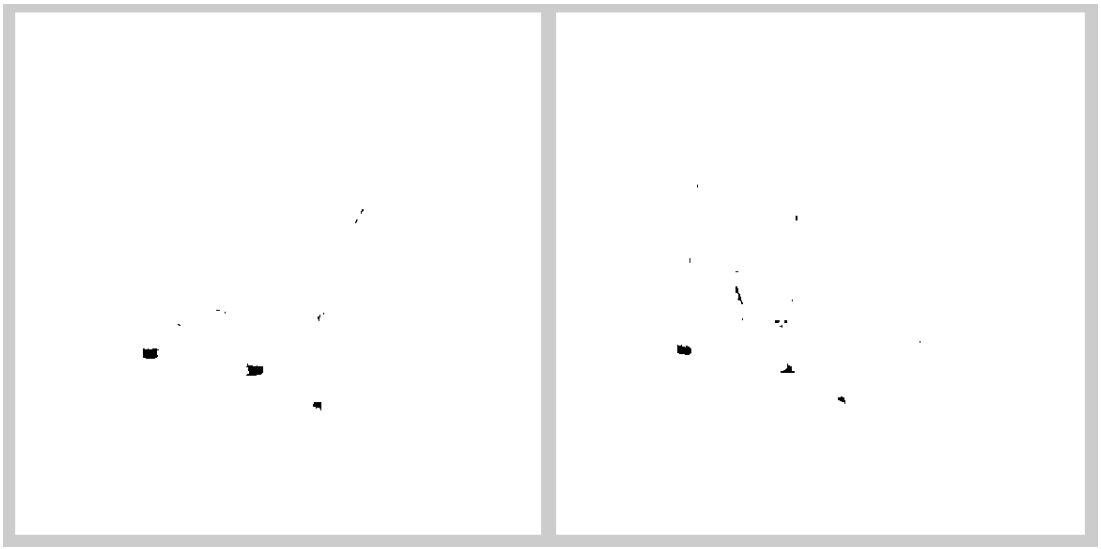


Figure 6-20 Dissimilar regions between second height level (left), 86 cm and third height level (right), 91 cm, for Experiment-1

In Figure 6-21 regions, which are among the dissimilar regions between second height level, 86 cm, and third height level, 91 cm, whose positions are same on the maps of both height levels are presented and highlighted by circles with same color. Positions of these regions are provided as feedbacks to the controller of the vehicles.

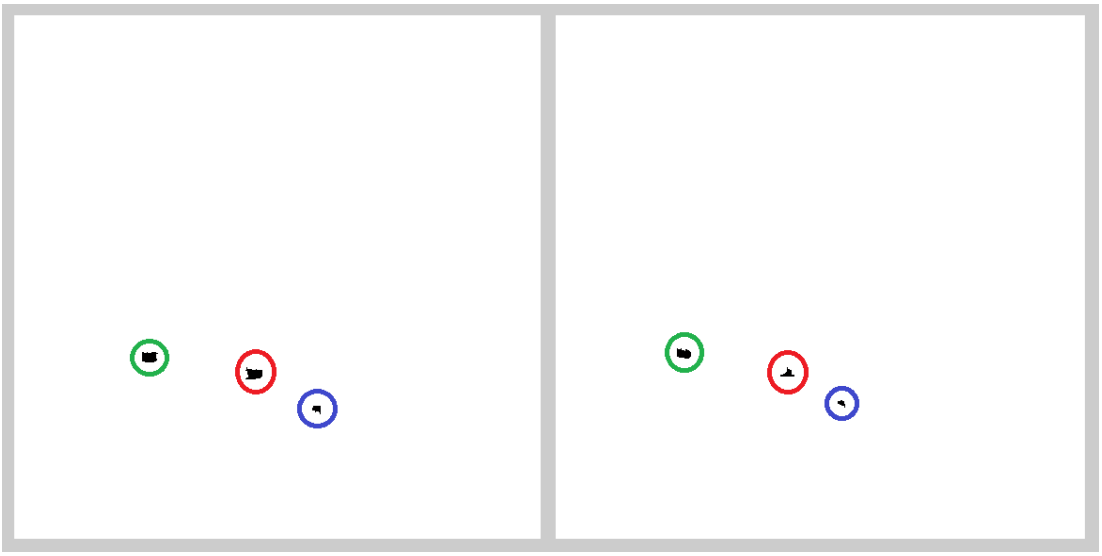


Figure 6-21 Regions to focus on for both second height level (left), 86 cm, and third height level (right), 91 cm, for Experiment-1

In Figure 6-22 dissimilar regions between third height level, 91 cm, and fourth height level, 96 cm, are presented. It can be observed that some dissimilar regions seen in Figure 6-20 at the third height level, 91 cm, cannot be seen in Figure 6-22 at the same height level. This shows that the discontinuities of the structures are changing between height levels.

Figure 6-22 Dissimilar regions between third heigh level (left), 91 cm, and fourth height level (right), 96 cm, for Experiment-1

In Figure 6-23 regions, which are among the dissimilar regions between third height level, 91 cm, and fourth height level, 96 cm, whose positions are same on the maps of both height levels are presented and highlighted by circles with same color. Positions of these regions are provided as feedbacks to the controller of the vehicles.

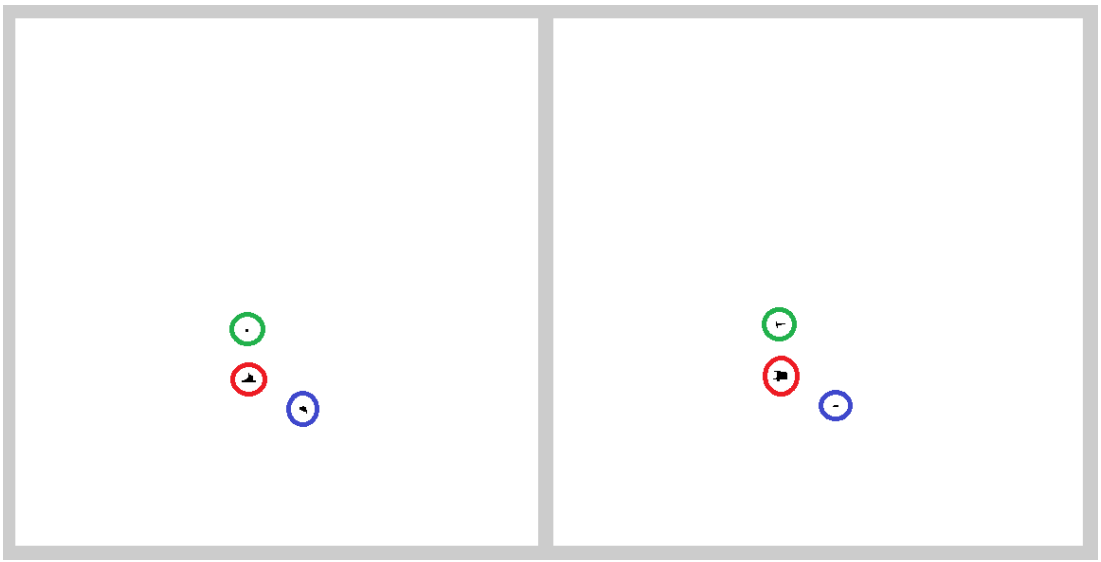


Figure 6-23 Regions to focus on for both third height level (left), 91 cm, and fourth height level (right), 96 cm, for Experiment-1

In Figure 6-24 dissimilar regions between fourth height level, 96 cm, and fifth height level, 101 cm, are presented. In this result one can see that there many discontinuous parts between the given height levels which are caused by real discontinuities on the structures or errors encountered in mapping and alignment processes.

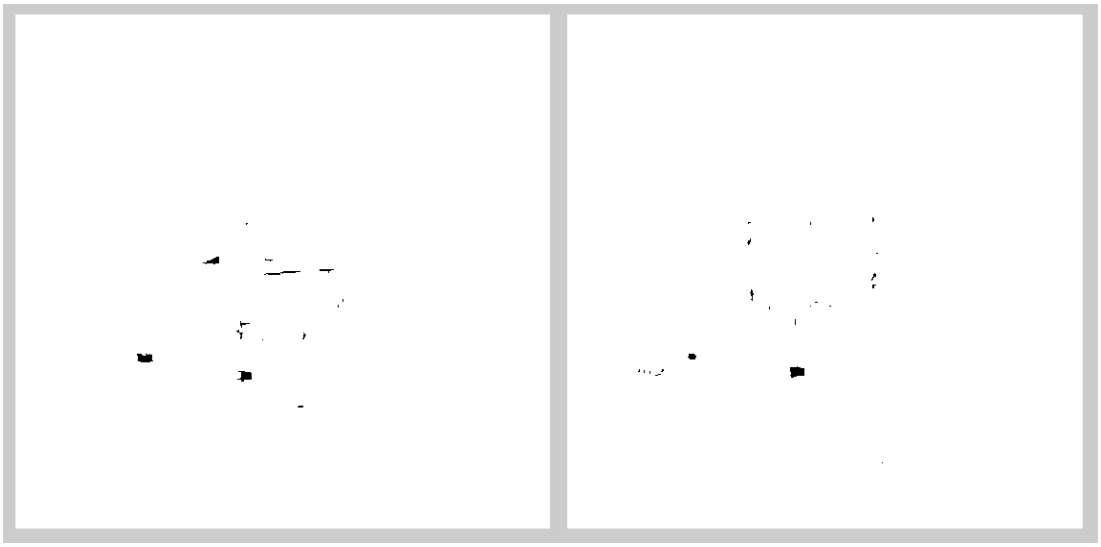


Figure 6-24 Dissimilar regions between fourth height level (left), 96 cm, and fifth height level (right), 101 cm, for Experiment-1

In Figure 6-25 regions, which are among the dissimilar regions between fourth height level, 96 cm, and fifth height level, 101 cm, whose positions are same on the maps of both height levels are presented and highlighted by circles with same color. Positions of these regions are provided as feedbacks to the controller of the vehicles.

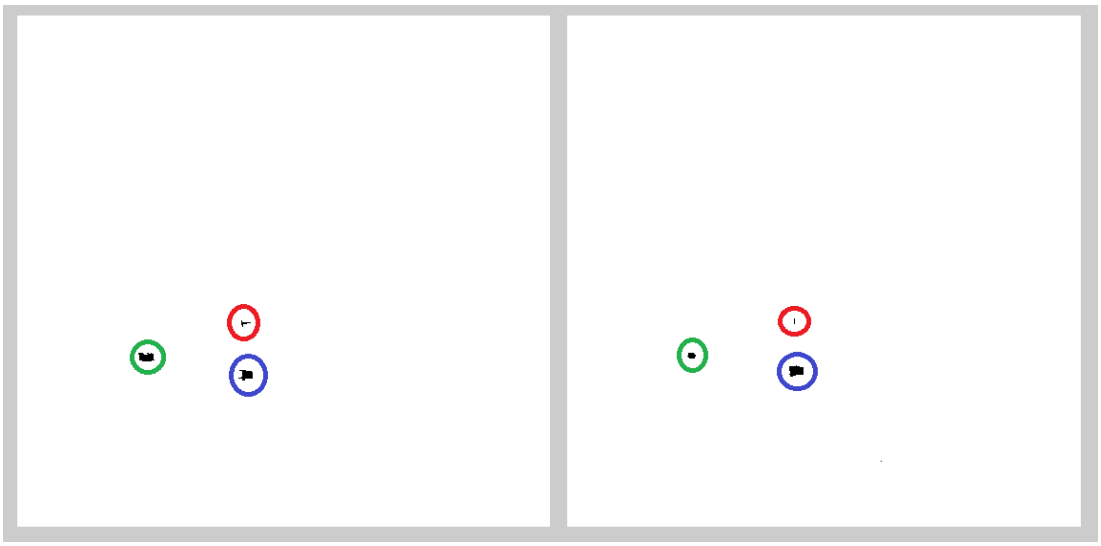


Figure 6-25 Regions to focus on for both fourth height level (left), 96 cm, and fifth height level (right), 101 cm, for Experiment-1

6.3.3.2. Generating Correspondence Point

Correspondence point sets are generated for similar and dissimilar regions by using different approaches, which are;

1) Using boundary points and
2) Using vertices of the tree structures constructed in the regions, as explained in Chapter 4.

In this part, results of correspondence point set generation are presented for Experiment-1. It should be carefully observed that for the regions on a map from middle height levels (height levels of 86, 91, 96 cm) which are coupled to two different maps for connecting, correspondence points according to similarity results.

- **Correspondence point sets generated for the connections of first height level, 81 cm, and second height level, 86 cm, :**

In Figure 6-26 similar regions of the first height level, 81 cm, and their generated correspondence point sets are presented. As previously mentioned for similar regions correspondence points are generated by using boundary parts of the regions.

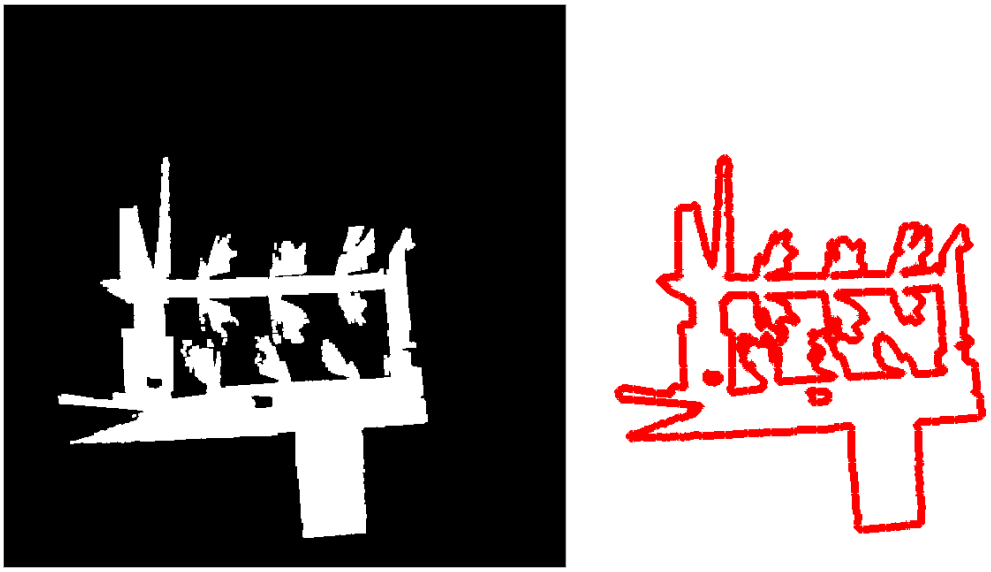


Figure 6-26 Similar regions of the first height level (left), 81 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-27 similar regions of the second height level, 86 cm, and their generated correspondence point sets are presented. One can see that boundary parts of the similar regions for this height level are used for correspondence point generation. When the results of Figure 6-26 and Figure 6-27 are analyzed one can see that, boundary points of the walls and two regions inside the environment are used for growing between height levels of 81 – 86 cm.

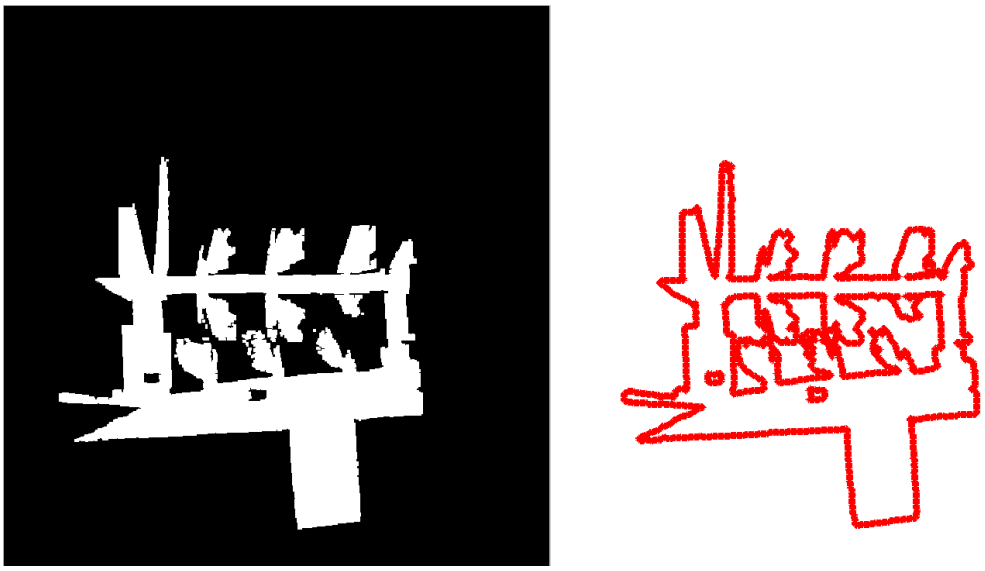


Figure 6-27 Similar regions of the second height level (left), 86 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-28 dissimilar regions of the first height level, 81 cm, and their generated correspondence point sets are presented. As previously mentioned, correspondence points are generated by constructing tree structures inside the dissimilar regions. Constructed tree structures for the dissimilar regions of first height level, 81 cm, and magnified versions of these structures are emphasized in black boxes. One can see the vertices, which are the correspondence points, and branches connecting the vertices which are the constraint connections that vertices have.

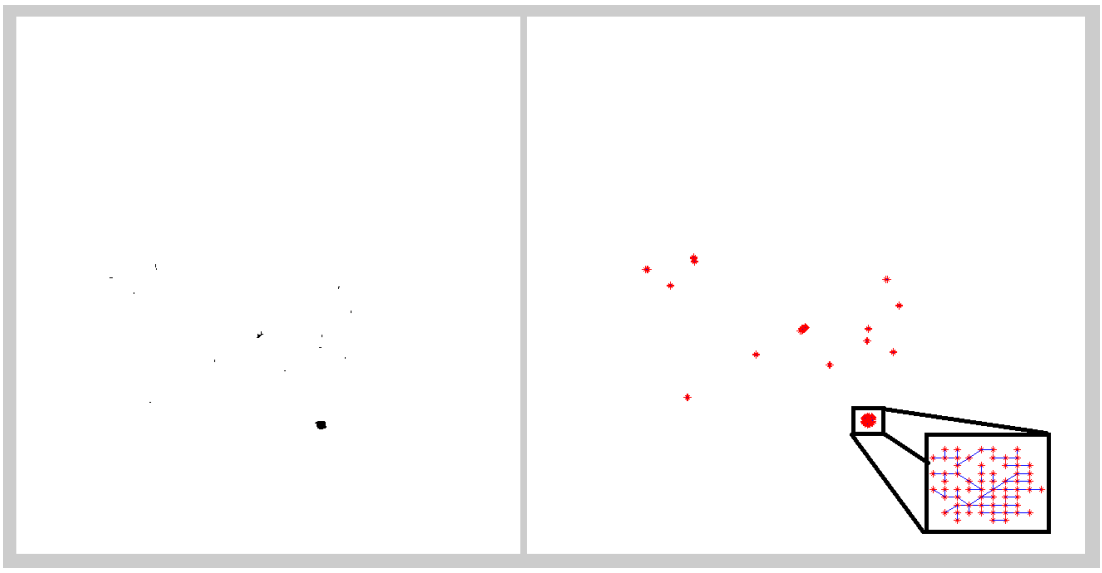


Figure 6-28 Dissimilar regions of the first height level (left), 81 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-29 dissimilar regions of the second height level, 86 cm, and their generated correspondence point sets are presented. Similar to the result obtained in Figure 6-28, tree structures are constructed inside the dissimilar regions of second height level, 86 cm, and magnified versions of these structures are emphasized in black boxes.

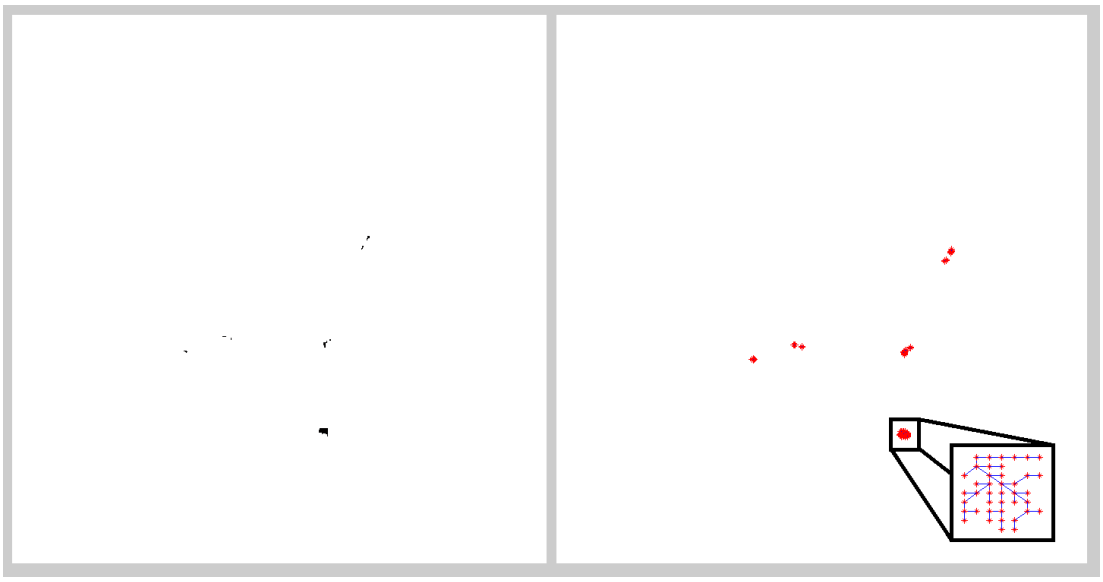


Figure 6-29 Dissimilar regions of the second height level (left), 86 cm, and their generated correspondence point sets (right) for Experiment-1

- **Correspondence point sets generated for the connections of second height level, 86 cm and third height level, 91 cm,**:

In Figure 6-30 similar regions of the second height level, 86 cm, and their generated correspondence point sets are presented. It can be seen that similar regions for the height level of 86 cm are changed when compared to result given in Figure 6-27. Two regions inside the environment, which are regarded as similar in Figure 6-27, are not regarded as similar for this case.

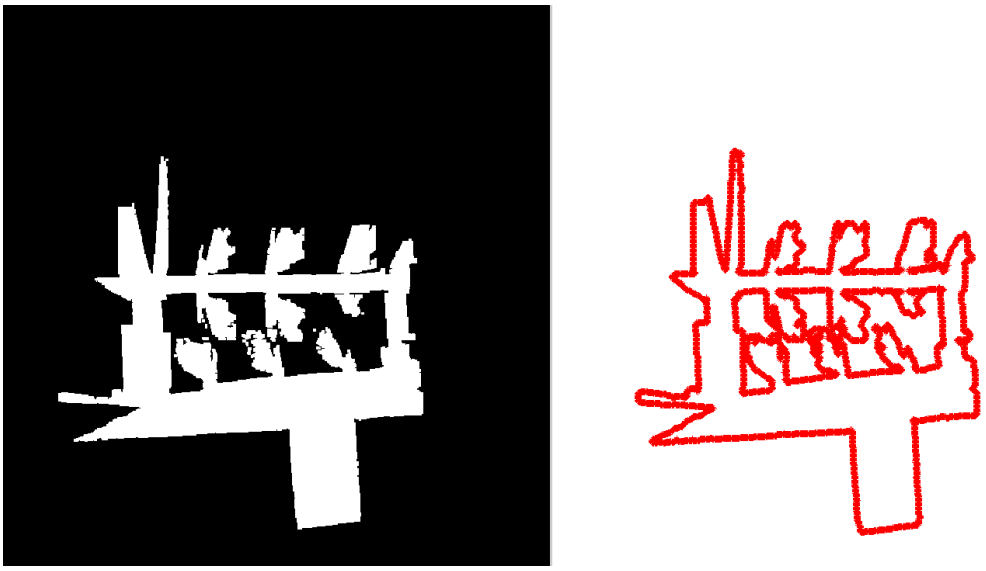


Figure 6-30 Similar regions of the second height level (left), 86 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-31 similar regions of the third height level, 91 cm, and their generated correspondence point sets are presented. Boundaries of the similar regions are used for correspondence point generation as expected. When the results of Figure 6-30 and Figure 6-31 are analyzed one can see that, boundary points of the walls of the environment are used for growing between height levels of 86 – 91 cm. This is different from the results obtained for the height levels of 81 – 86 cm, in which there are also two regions inside the environment whose boundary points are utilized for growing.

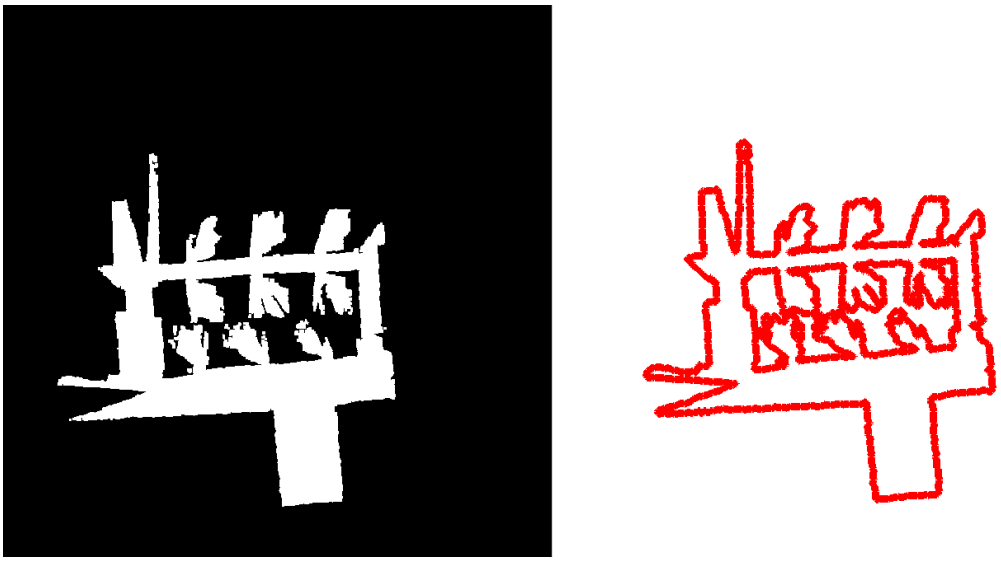


Figure 6-31 Similar regions of the third height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-32 dissimilar regions of the second height level, 86 cm, and their generated correspondence point sets are presented. As one can notice, two regions inside the environment that are regarded as similar in Figure 6-27 for the comparison of height levels of 81-86 cm, are found dissimilar for the comparison of height levels of 86-91 cm. Therefore, two regions that are previously seen in Figure 6-27 and absent in Figure 6-30, can be found in Figure 6-32. In Figure 6-32, tree structures are constructed inside the dissimilar regions, and these magnified versions of these structures are emphasized in black boxes.

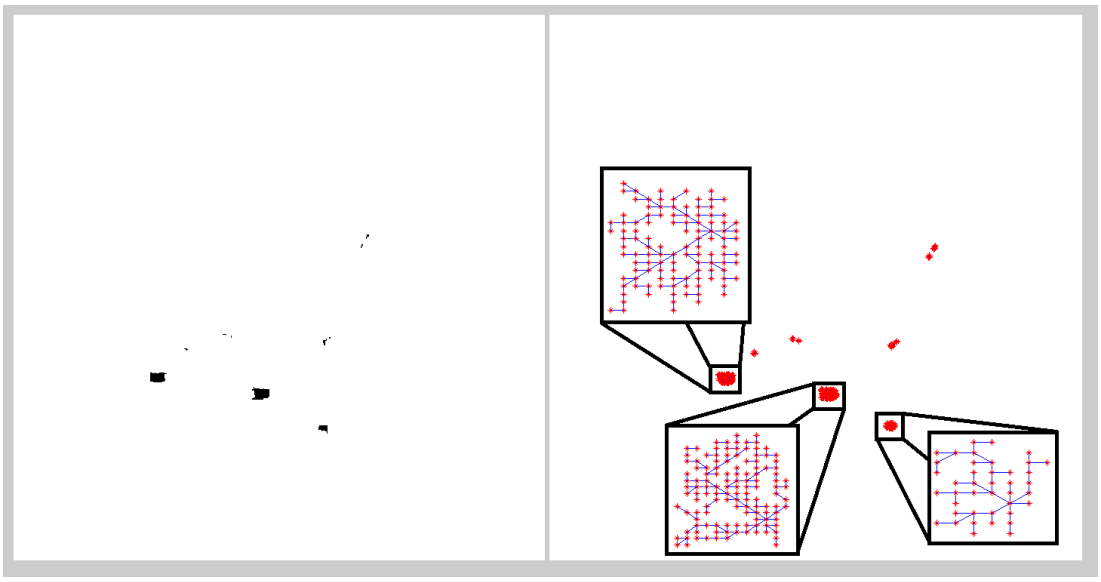


Figure 6-32 Dissimilar regions of the second height level (left), 86 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-33 dissimilar regions of the third height level, 91 cm, and their generated correspondence point sets are presented. Similar to the result obtained in Figure 6-32, tree structures are constructed inside of dissimilar regions of third height level, 91 cm, and magnified versions of these structures are emphasized in black boxes.

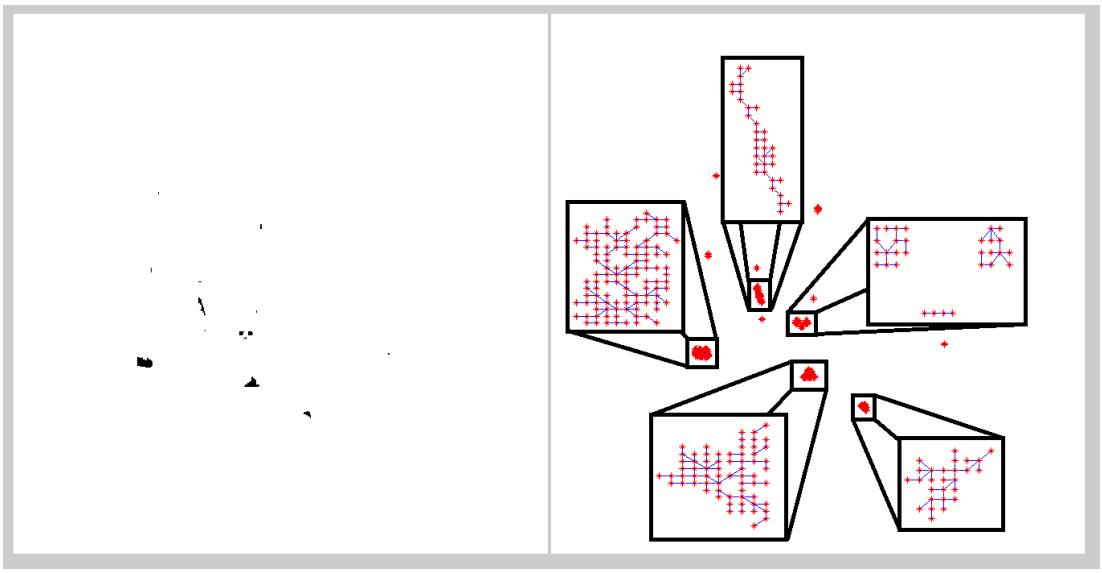


Figure 6-33 Dissimilar regions of the third height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-1

- **Correspondence point sets generated for the connections of third height level, 91 cm, and fourth height level, 96cm,:**

In Figure 6-34 similar regions of the third height level, 91 cm, and their generated correspondence point sets are presented. It can be seen that similar regions for the height level of 91 cm are changed when compared to result given in Figure 6-31. There are no regions inside the environment in Figure 6-31, but in Figure 6-34, there is one region inside the environment that is regarded as similar.

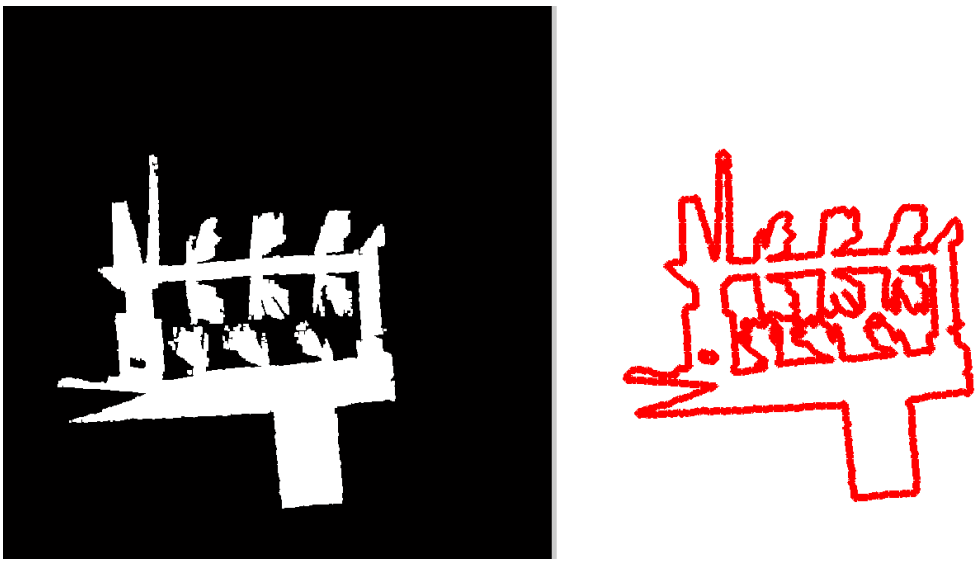


Figure 6-34 Similar regions of the third height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-35 similar regions of the fourth height level, 96 cm, and their generated correspondence point sets are presented. When the results of Figure 6-34 and Figure 6-35 are analyzed one can see that, boundary points of the walls and one region in the environment are used for growing between height levels of 91 – 96 cm. This is different from the results obtained for the height levels of 86 – 91 cm, in which there are only walls, but no region inside the environment, whose boundary points are utilized for growing. Additionally, at the top part of the Figure 6-35, correspondence points are affected from the error seen in the mapping process, which is about missing one empty room in the environment, and generated accordingly in wrong locations.

Figure 6-35 Similar regions of the fourth height level (left), 96 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-36 dissimilar regions of the third height level, 91 cm, and their generated correspondence point sets are presented. As one can notice, there is one dissimilar region missing when Figure 6-33 and Figure 6-36 are compared. This region is the one that is seen in Figure 6-34 as similar region, for the comparison of height levels of 91-96 cm. In Figure 6-36 tree structures are constructed inside the dissimilar regions, and magnified versions of these structures are emphasized in black boxes.

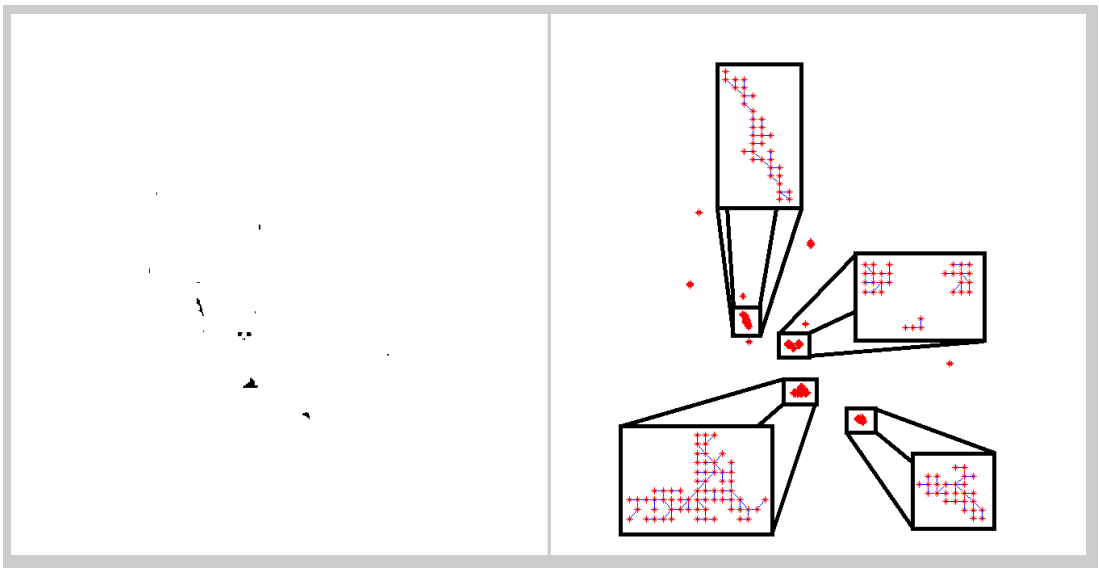


Figure 6-36 Dissimilar regions of the third height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-37 dissimilar regions of the fourth height level, 96 cm, and their generated correspondence point sets are presented. Similar to the result obtained in Figure 6-36Figure 6-32, tree structures are constructed inside the dissimilar regions of fourth height level, 96 cm, and magnified versions of these are emphasized in black boxes.

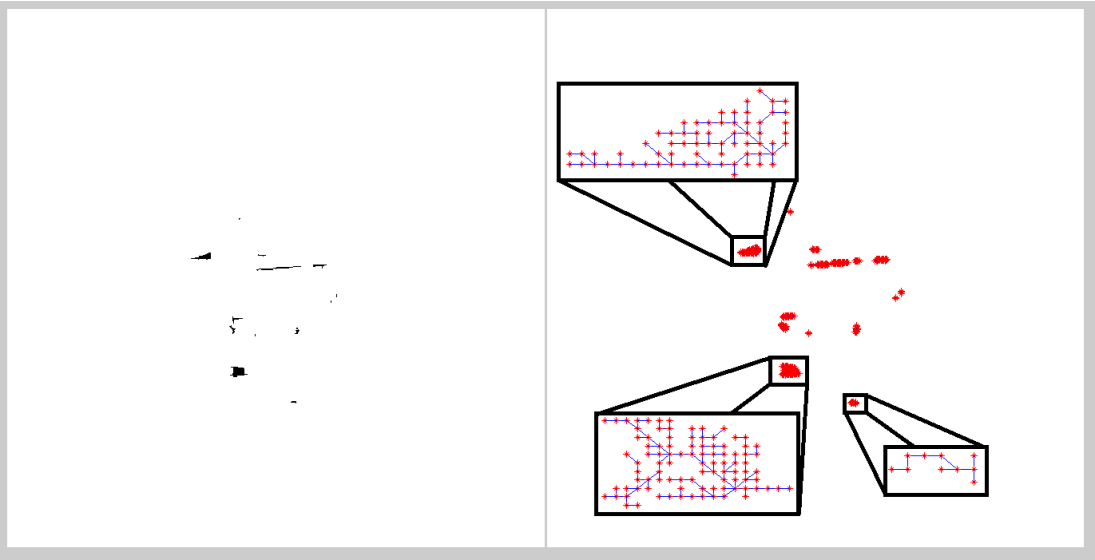


Figure 6-37 Dissimilar regions of the fourth height level (left), 96 cm, and their generated correspondence point sets (right) for Experiment-1

- **Correspondence point sets generated for the connections of fourth height level, 96 cm, and fifth height level, 101 cm,:**

In Figure 6-38 similar regions of the fourth height level, 96 cm, and their generated correspondence point sets are presented. It can be seen that boundaries of the similar regions are used for correspondence point generation as expected and similar regions for the height level of 96 cm are changed when compared to result given in Figure 6-35. The difference is that one region seen inside the environment in Figure 6-35 cannot be seen in Figure 6-38.

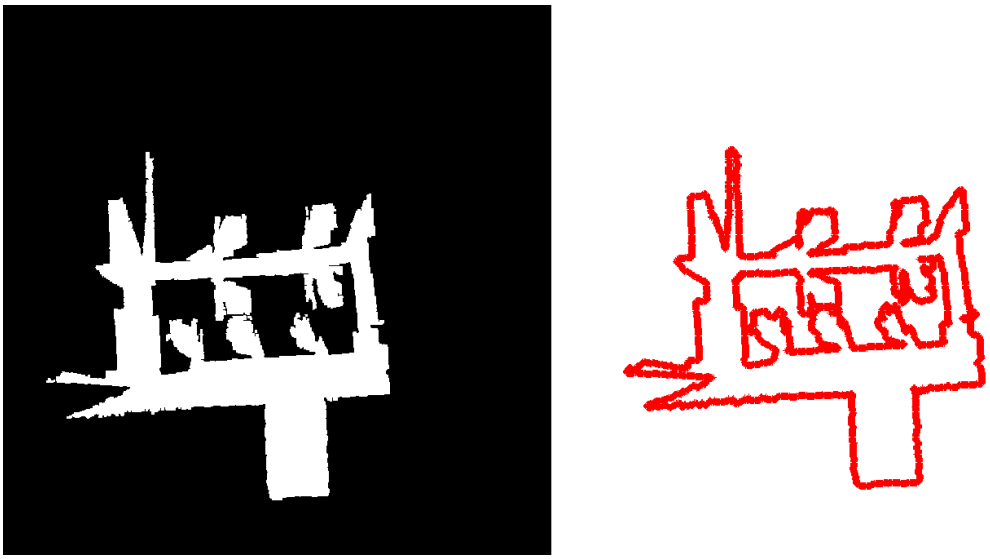


Figure 6-38 Similar regions of the fourth height level (left), 96 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-39 similar regions of the fifth height level, 101 cm, and their generated correspondence point sets are presented. When the results of Figure 6-38 and Figure 6-39 are analyzed one can see that, boundary points of the walls of the environment are used for growing between height levels of 96 – 101 cm. This is different from the results obtained for the height levels of 91 – 96 cm, in which there is one region inside the environment, whose boundary points are utilized for growing.

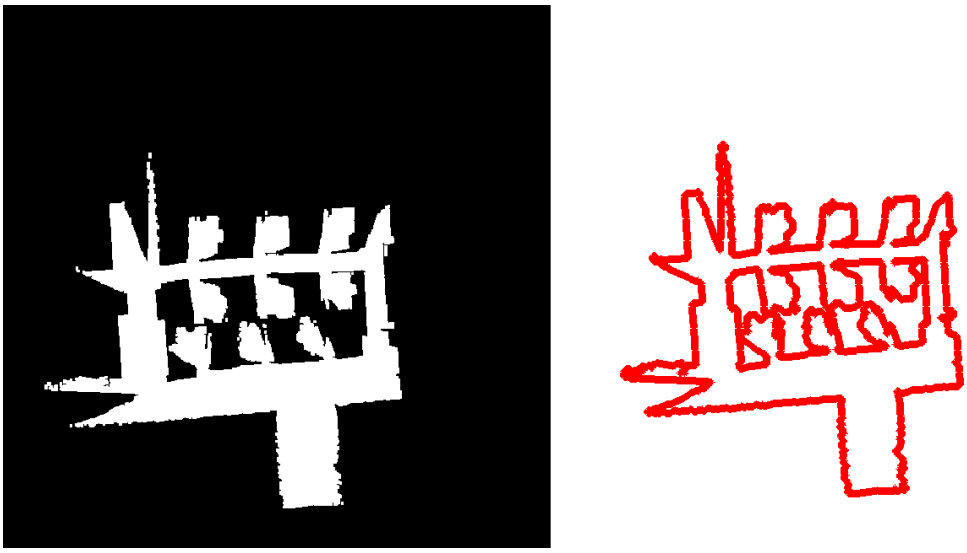


Figure 6-39 Similar regions of the fifth height level (left), 101 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-40 dissimilar regions of the fourth height level, 96 cm, and their generated correspondence point sets are presented. As one can notice, there is one additional dissimilar region in Figure 6-40 when compared to Figure 6-37. This region is the one that is seen in Figure 6-35 as similar region for the comparison of height levels of 91-96 cm, but missing in Figure 6-38. In Figure 6-36 tree structures are constructed inside the dissimilar regions, and magnified versions of these structures are emphasized in black boxes

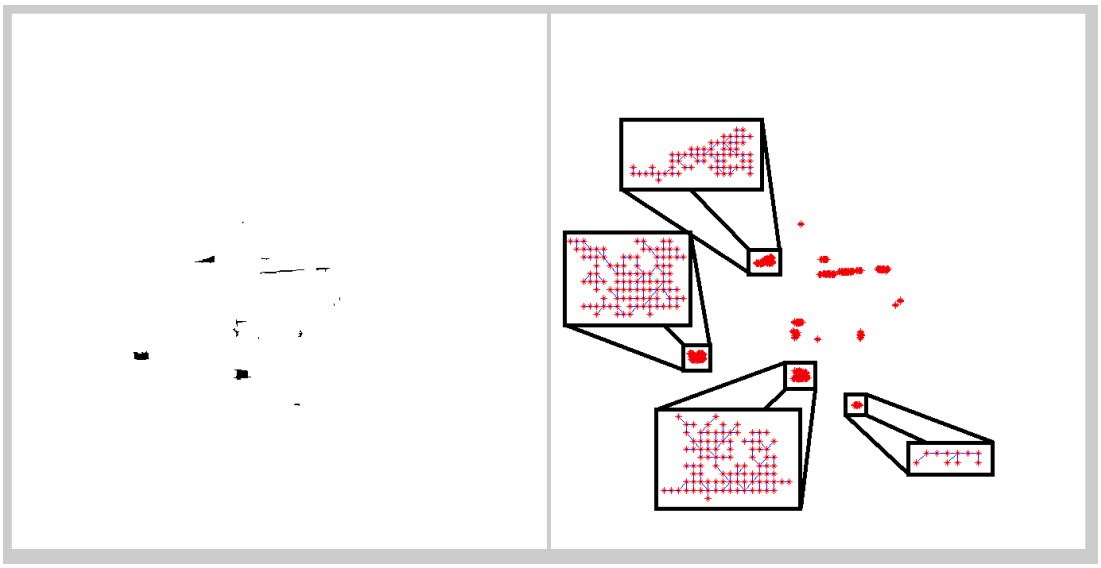


Figure 6-40 Dissimilar regions of the fourth height level (left), 96 cm, and their generated correspondence point sets (right) for Experiment-1

In Figure 6-41 dissimilar regions of the fifth height level, 101 cm, and their generated correspondence point sets are presented. Similar to the result obtained in Figure 6-40Figure 6-32, tree structures are constructed inside the dissimilar regions of fourth height level, 96 cm, and magnified versions of these are emphasized in black boxes.

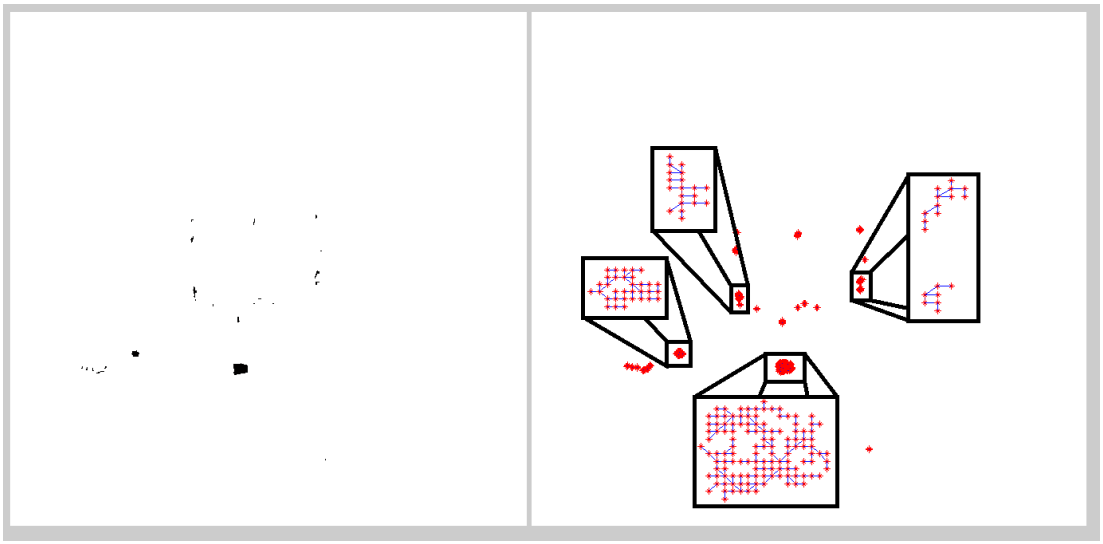


Figure 6-41 Dissimilar regions of the fifth height level (left), 101 cm, and their generated correspondence point sets (right) for Experiment-1

6.3.3.3.Tiling Correspondence Points

In this part, connections between correspondence point sets are established by using the Delaunay Triangulation with determined length limit for edges of the triangles which is 11cm.

In Figure 6-42 tiling result for correspondence point sets of first height level, 81 cm, and second height level, 86 cm, is presented. As it can be seen from the figure, there are holes on some of the surfaces which are caused by the errors seen in map construction and alignment processes.

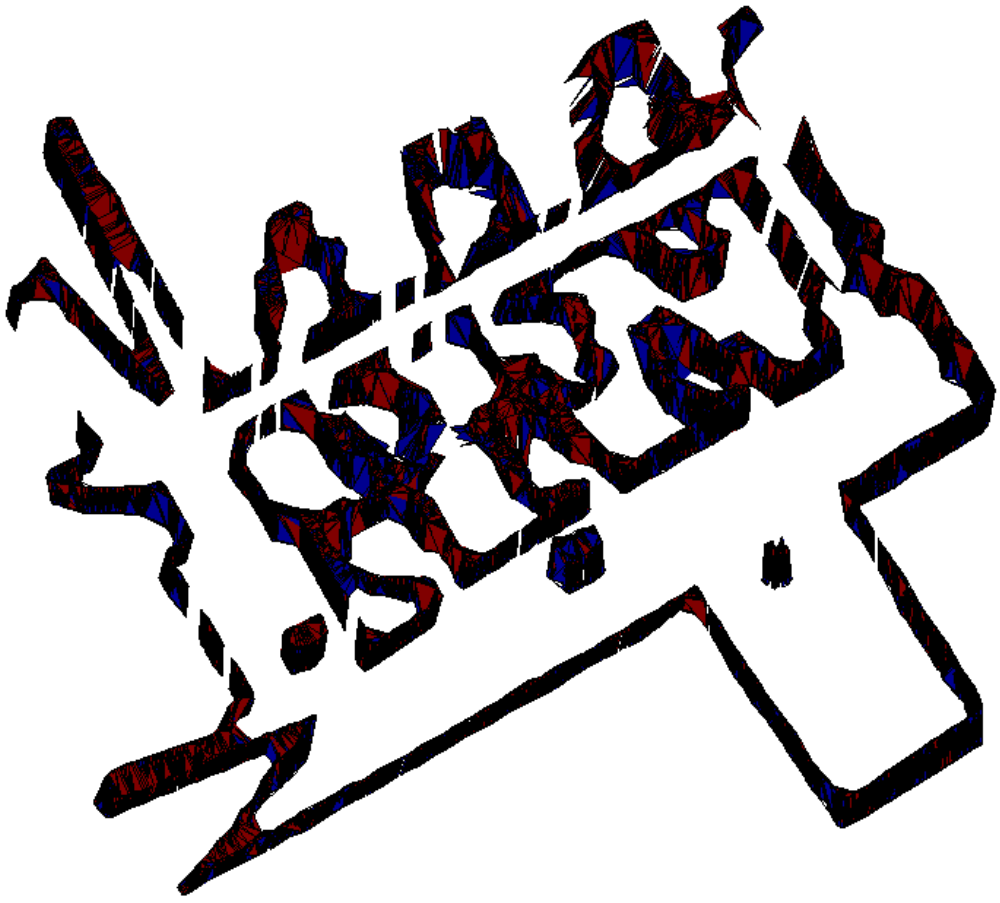


Figure 6-42 Tiling result for correspondence point sets of first height level, 81 cm, and second height level, 86 cm, for Experiment-1

In Figure 6-43 tiling result for correspondence point sets of second height level, 86 cm, and third height level, 91 cm, added to the previous result, given in Figure 6-42, is presented. When compared to the previous result presented in Figure 6-42, a wrong connection is detected in the result given in Figure 6-43, which is emphasized by red circle. The reason for this wrong connection is the alignment error previously seen between height levels of 86-91 cm provided in Figure 6-15 (emphasized with green circle).

Figure 6-43 Tiling result for correspondence point sets of second height level, 86 cm, and third height level, 91 cm, added to the previous result for Experiment-1

In Figure 6-44 tiling result for correspondence point sets of third height level, 91 cm, and fourth height level, 96 cm, added to the previous result, given in Figure 6-43, is presented. In the given result, it is observed that one of the constructed empty spaces is disrupted because of the alignment error seen between height levels of 91-96 cm, given in Figure 6-16 and emphasized with green circle, since there is an empty room missing and this shifts the locations of the correspondence points generated for the height level of 96 cm, given in Figure 6-35 and Figure 6-38.

Figure 6-44 Tiling result for correspondence point sets of third height level, 91 cm, and fourth height level, 96 cm, added to the previous result for Experiment-1

In Figure 6-45 tiling result for correspondence point sets of fourth height level, 96 cm, and fifth height level, 101 cm, added to the previous result, provided in Figure

6-44 is presented. It is important to clarify that there are no wrong connections inside the corridor, even if it seems so, which is located with the red box in Figure 6-45, and this fact is also seen clearly in Figure 6-50.

Figure 6-45 Tiling result for correspondence point sets of fourth height level, 96 cm, and fifth height level, 101 cm, added to the previous result for Experiment-1

6.3.4. Results

This subsection aims to evaluate the results of Experiment-1

Figure 6-46 shows the final form of the reconstructed environment in which objects, encircled by different colors and labeled with their individual numbers are observed.

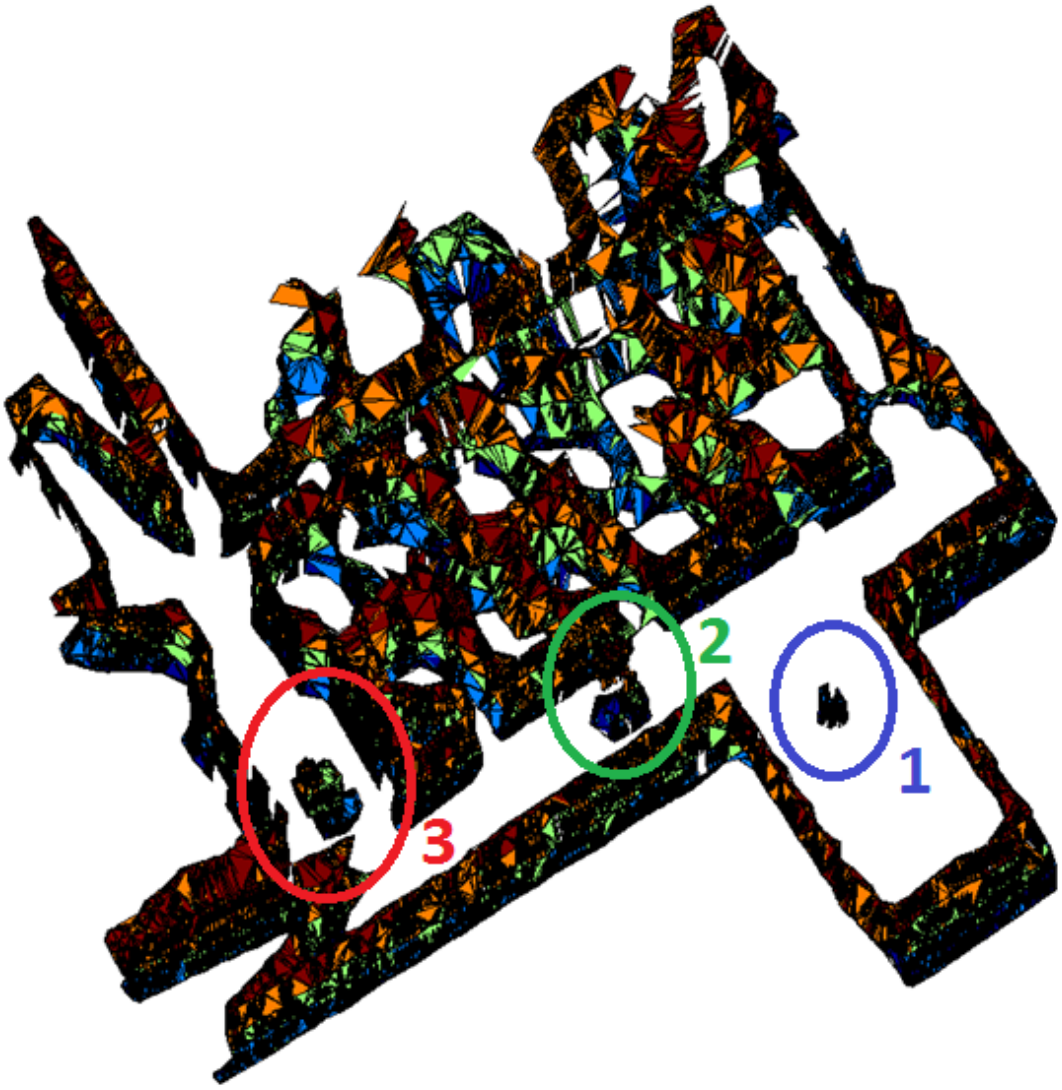


Figure 6-46 Reconstructed environment and objects inside it

Real objects are presented with the corresponding reconstructed versions in order to achieve a geometrical comparison between them. In Figure 6-47 Object-1 and its reconstructed version is presented. It can be seen that the reconstructed version does not resembles the real version completely. The bottom parts are reconstructed correctly, but the upper parts are not reconstructed adequately because data coming from the constructed 2D maps for the upper part of this object is inadequate, which

can be seen in Figure 6-12 and Figure 6-13. In Figure 6-12 the region that belongs to the 2D projection of the object gets smaller and in Figure 6-13 it disappears.

Figure 6-47 Object-1(left) and its reconstructed version (right) for Experiment-1

In Figure 6-48, Object-2 and its reconstructed version are presented. One can see that the geometry of the reconstructed version resembles the real object, but there are some points in which disruptions are observed such as the middle part of the reconstructed version. This is caused by the error encountered in map construction, which can be seen in Figure 6-11, so that the region that belongs to the 2D projection of the object is not constructed properly.

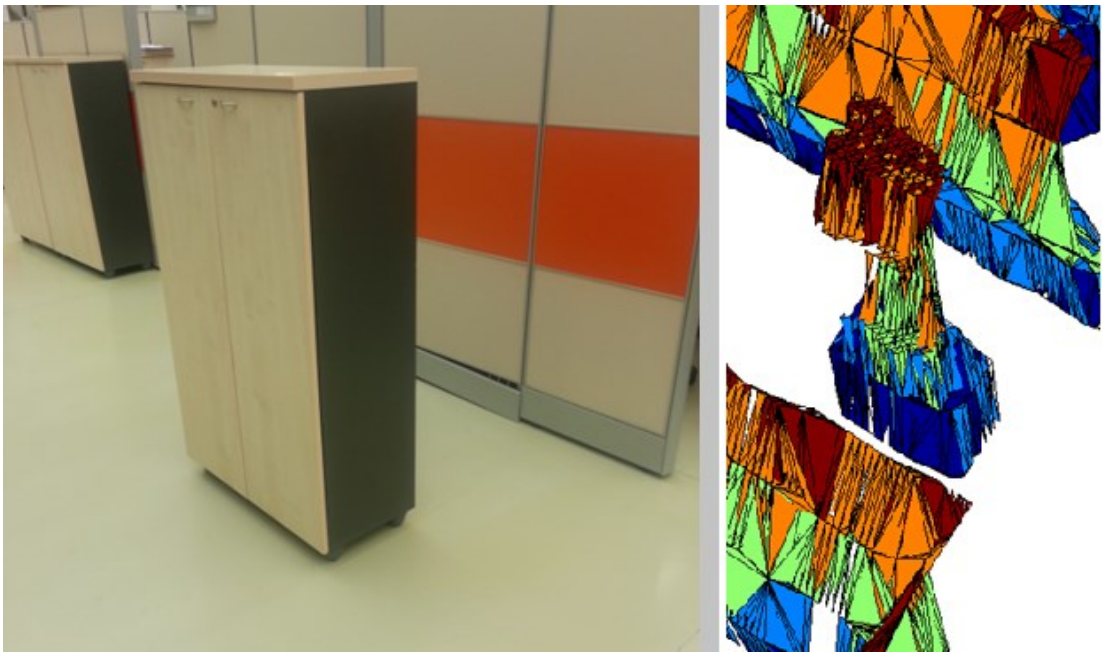


Figure 6-48 Object-2(left) and its reconstructed version (right) for Experiment-1

In Figure 6-49 Object-3 and its reconstructed version are presented. It can be observed that reconstructed version of Object-3 resembles its real version more successfully than the others. The stairway structure of the Object-3 is achieved in the reconstructed version properly.

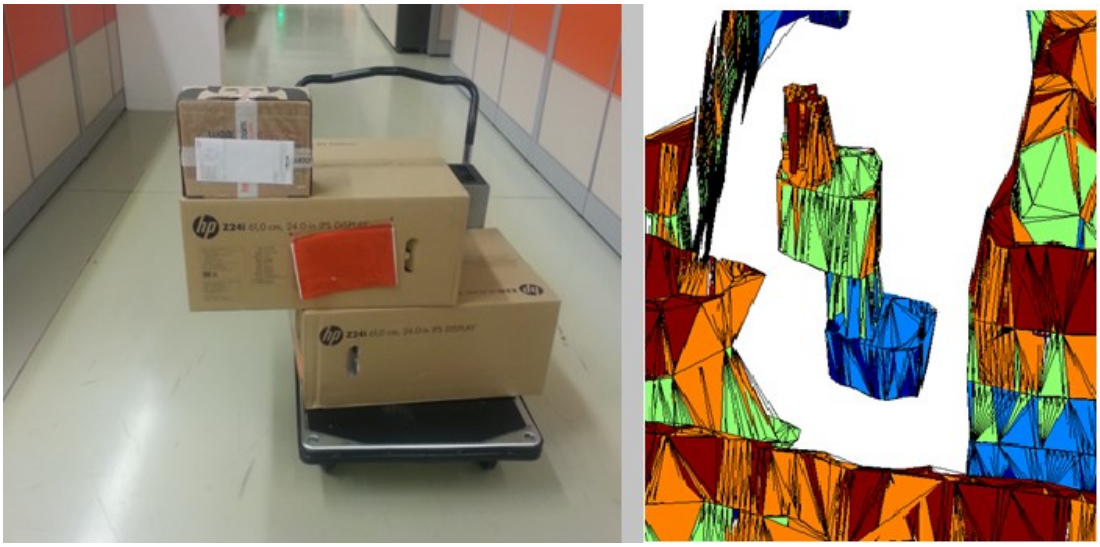


Figure 6-49 Object-3(left) and its reconstructed version (right) for Experiment-1

Additionally, in order to illustrate the errors in the reconstruction result, top view and side view of the final form is presented in Figure 6-50 and Figure 6-51 respectively.

In Figure 6-50 observed errors are categorized in three namely; wrong connections, disconnections (holes) and reconstruction of the empty spaces. Wrong connections (highlighted with red) and disconnections (highlighted with green) are triggered by errors encountered in map construction and alignment. In order to prevent these problems to occur, length limit for the edges of triangles is used, but adjusting length limit brings a tradeoff between wrong connections and disconnections and it can only provide error minimization instead of complete elimination of errors. Errors seen in the reconstruction of empty spaces are mostly triggered by the errors in the map construction. For this experiment the map constructed at the fourth height level has large errors in construction of the empty spaces (cubical rooms in the area).

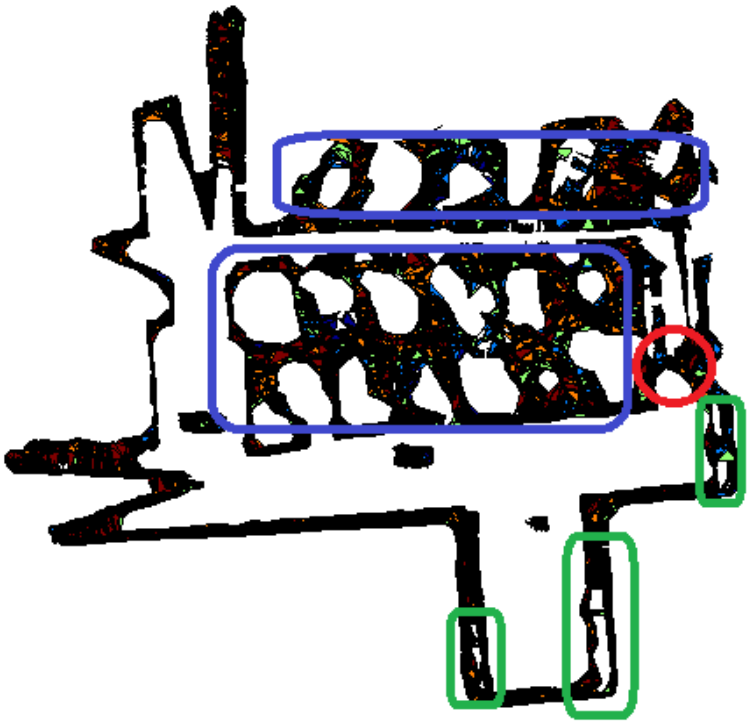


Figure 6-50 Top view of the reconstructed environment

In Figure 6-51 observed holes in the walls (highlighted with red circles) of the environment is shown in details with the side view of the reconstruction result. The

reasons for observing these holes are the alignment errors and length limitation, which is analyzed in the sensitivity analysis section, for triangles constructed in tiling.

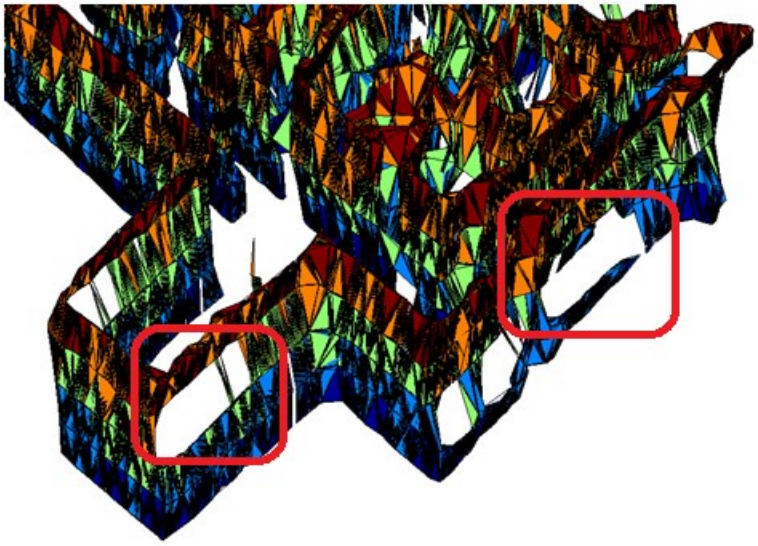


Figure 6-51 Side view of the reconstructed environment

6.4. Experiment 2: Scanning an Object-1

6.4.1. Environment of the Experiment 2

The environment used for Experiment-2 is an indoor environment in ASELSAN which contains a selected object, which has stairway geometry as given in Figure 6-52. This environment explored at 3 different height levels which are 81, 91 and 101 cm. Images taken from the environment during this experiment can be seen in Figure 6-52.

Figure 6-52 Environment used in Experiment-2

In Experiment-2, Vehicle-2 is utilized to explore the environment from 3 different height levels with height difference of 10 cm beginning from 81cm as the minimum height value.

6.4.2. Fast-SLAM Exploration Results in Experiment 2

The obtained laser scans are used in the Fast-SLAM algorithm whose parameters are chosen as:

- Grid size: 0.01 m
- Particle number: 500
- Probability change rates:
  - $m_{occ} = -0.5$
  - $m_{free} = +0.05$

2D grid maps for all different height levels are obtained and presented in Figure 6-53 - Figure 6-55. In Figure 6-53, 2D mapping result for the exploration of the indoor

environment and the object at height level of 81 cm is provided. One can easily detect the region that is the 2D projection of the scanned object at given height level.

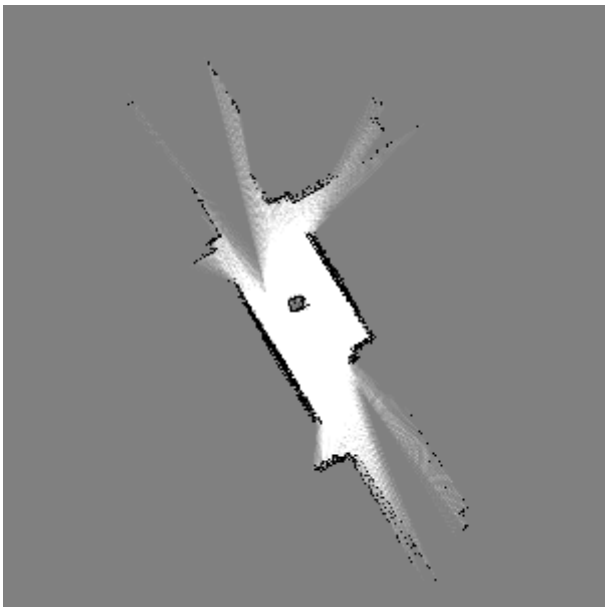


Figure 6-53 2D grid map obtained from Fast-SLAM for first height level, 81 cm, for Experiment-2

In Figure 6-54, 2D mapping result for the exploration of the indoor environment and the object at height level of 91 cm is presented. The main difference between the result given in Figure 6-54 and the previous result given in Figure 6-53 is the orientation of the maps which is handled in the map alignment part in map growing.

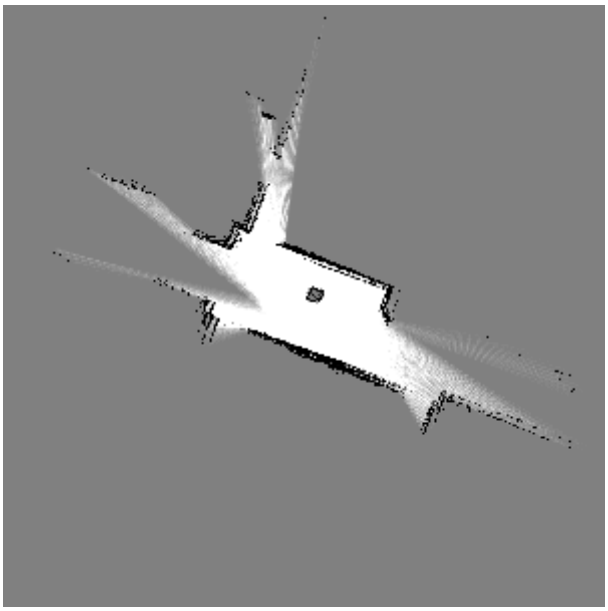


Figure 6-54 2D grid map obtained from Fast-SLAM for second height level, 91 cm, for Experiment-2

In Figure 6-55, 2D mapping result for the exploration of the indoor environment and the object at height level of 101 cm is presented. When compared to previous results, presented in Figure 6-53 and Figure 6-54, orientation of the result given in Figure 6-55 is different. Additionally, there are disruptions at the walls of the environment caused by the errors encountered during map construction for the given height level, 101 cm, which are emphasized with the red circles in Figure 6-55. These disruptions cause problems during the alignment of the maps and generating correspondence points for the given height level.

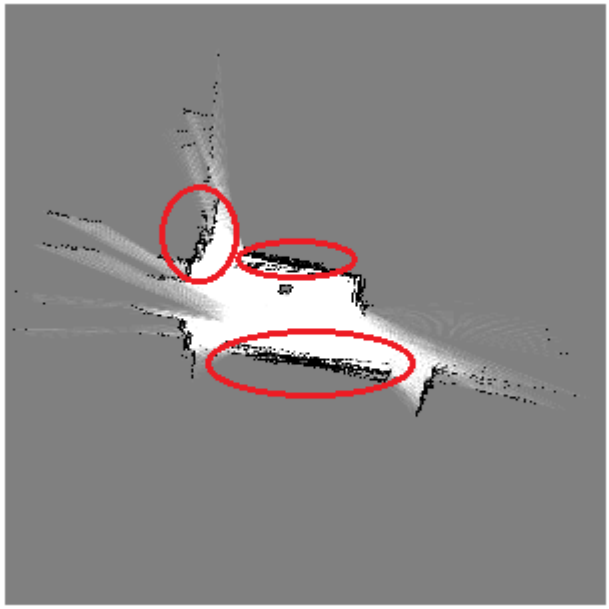


Figure 6-55 2D grid map obtained from Fast-SLAM for third height level, 101 cm, for Experiment-2

6.4.3. Map Growing

After obtaining 2D grid maps for different height levels, map growing method is applied in the experiment. Initially, correct correspondences are created between successive maps by first segmenting meaningful regions, then aligning maps by using features extracted from these regions and finally, separating the regions as similar and dissimilar according to similarity measure between them. Afterwards, correspondence points are generated in order to be used while connecting regions on the maps for reconstruction purpose which is defined in the final step namely; tiling of the correspondence points. In map growing method, the following values for the parameters are utilized.

- Percentage Threshold for Separation of Regions: 70%
- Length Limit for Tiling: 8 cm

6.4.3.1.Creating Correct Correspondences

In order to visualize the achieved alignments between successive maps, boundaries of the segmented regions are utilized. In Figure 6-56 alignment between first height level (red), 81 cm, and second height level (blue), 91 cm, is presented. The aim is to illustrate the alignment results of the maps with respect to each other. It can be seen that, there are some parts differ considerably in both maps because of the effect of uncertainty regions in the maps which are encircled by green. Also, there is a difference between the positions of the object inside the environment in both maps.

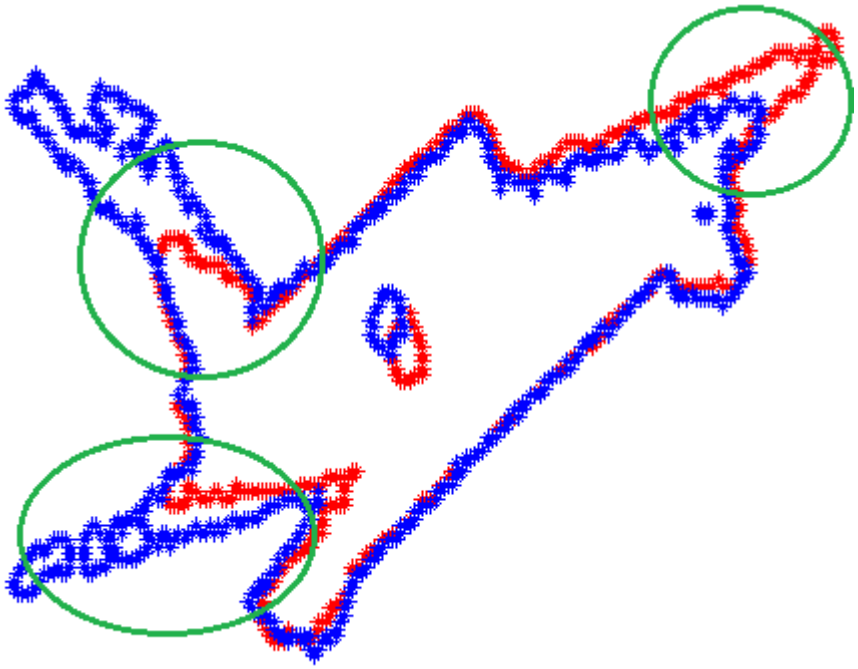


Figure 6-56 Achieved alignment between first height level (red), 81 cm, and second height level (blue), 91 cm, for Experiment-2

In Figure 6-57 alignment between second height level (red), 91 cm, and third height level (blue), 101 cm, is presented. The same effects previously seen in Figure 6-56, caused by the uncertainty regions, are also seen in Figure 6-57 which are emphasized by green circles. Additionally, negative effect of the disruptions, observed on the walls of the environment in Figure 6-55, can be seen in the alignment of the maps in Figure 6-57 which are shown with green arrows.

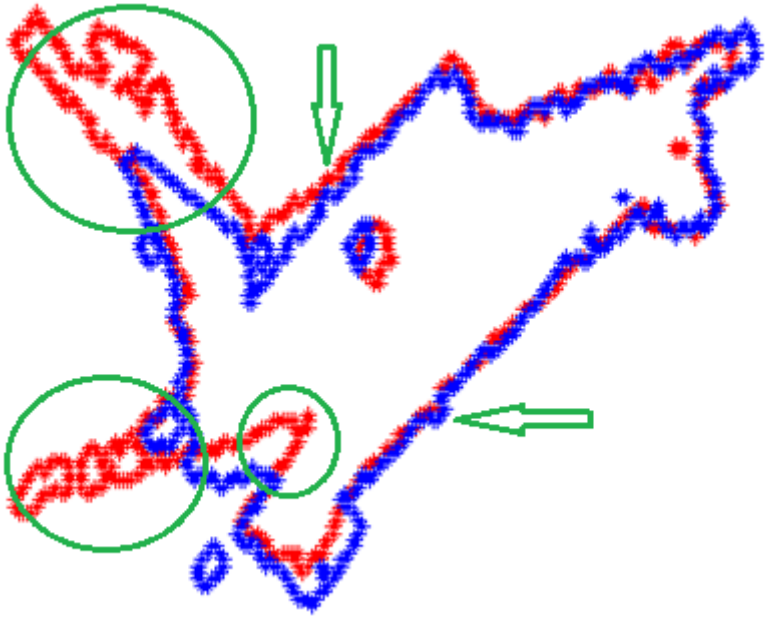


Figure 6-57 Achieved alignment between second height level (red), 91 cm, and third height level (blue), 101 cm, for Experiment-2

After alignment of the maps, similarities between regions is measured in percentage by applying grid by grid comparison, and percentage threshold of 70% is used to separate similar and dissimilar regions, results of which is presented in Table 6-1 - Table 6-2. One can see that there is one region regarded as similar for all cases, which is the boundaries, walls, of the environment. Other regions which have smaller similarity among themselves are regarded as dissimilar.

Table 6-1 Similarity percentage of regions between first height level, 81 cm, and second height level, 91 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70% for Experiment-2

		Region Numbers							
		1	2	3	4	5	6	7	8
RN	1	98.96751	0.002347	0.002347	0.00352	0.004693	0	0	0
	2	0	0	0	0	0	16.12903	0	0

SIMILAR	
DISSIMILAR	

Table 6-2 Similarity percentage of regions between second height level, 91 cm, and third height level, 101 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70% for Experiment-2

Region Numbers	1	2	3
1	98.96666	0	0
2	0.002348	0	0
3	0.002348	0	0
4	0.003523	0	0
5	0	0	0
6	0	24.5614	0
7	0	0	0
8	0	0	0

SIMILAR	
DISSIMILAR	

In the experiments system feedback is generated for detecting and focusing on the discontinuities in the environment. However, in this experiment the aim is to focus on a selected object by using the proposed method and reconstruct 3D structure of it. Therefore, it is not significant to evaluate the discontinuities in the environment and focus on them for this experiment.

6.4.3.2.Generating Correspondence Point

Correspondence point sets are generated for similar and dissimilar regions by using different approaches, which are using boundary points and using vertices of the tree structures constructed in the regions, as explained in Chapter 4. In this part, results of correspondence point set generation are presented for Experiment-2.

- **Correspondence point sets generated for the connections of first height level, 81 cm, and second height level, 91 cm,**:

In Figure 6-58 similar regions of the first height level, 81 cm, and their generated correspondence point sets are presented. As previously seen in Table 6-1, there is only one similar region and for this similar region correspondence points are generated by using boundary parts of the region, shown in Figure 6-58.

Figure 6-58 Similar regions of the first height level (left), 81 cm, and their generated correspondence point sets (right) for Experiment-2

In Figure 6-59 similar regions of the second height level, 91 cm, and their generated correspondence point sets are presented. When the results of Figure 6-58 and Figure 6-59 are analyzed it can be seen that, boundary points of the similar regions, which are the walls of the environment for both maps, are used for correspondence point generation for both maps, as expected.

Figure 6-59 Similar regions of the second height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-2

In Figure 6-60 dissimilar regions of the first height level, 81 cm, and their generated correspondence point sets are presented. As seen in the Table 6-1, there is only one dissimilar region for this height level and corresponding point generation is achieved by constructing tree structure inside it, magnified version of which is shown inside the black box in Figure 6-60. One can see that vertices, which are the correspondence points, and branches, which are the constraint connections between correspondence points, are generated while constructing tree structures

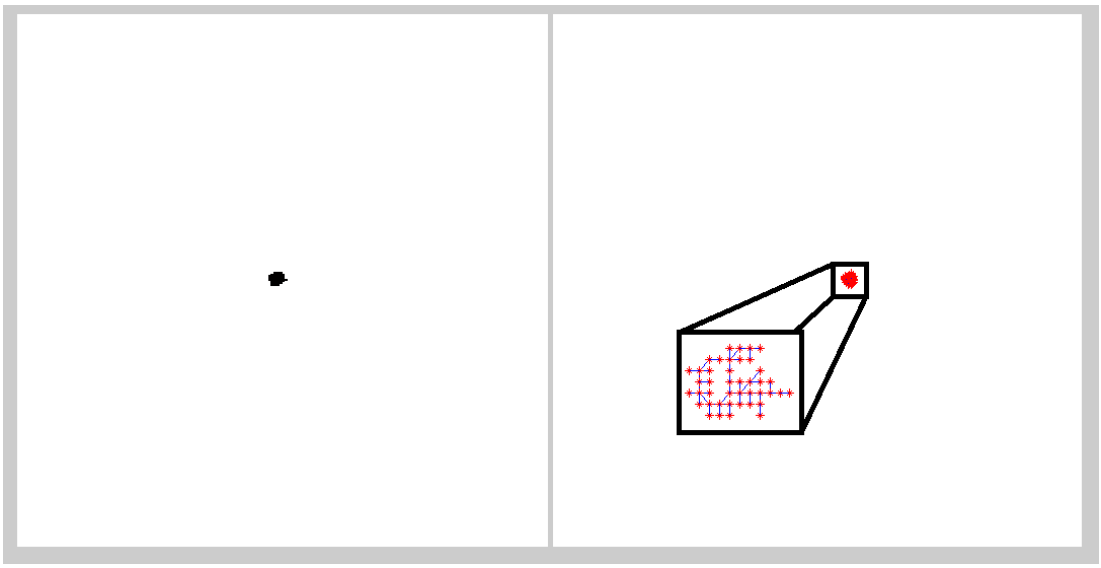


Figure 6-60 Dissimilar regions of the first height level (left), 81 cm, and their generated correspondence point sets (right) for Experiment-2

In Figure 6-61 dissimilar regions of the second height level, 91 cm, and their generated correspondence point sets are presented. Similar to the result obtained in Figure 6-60, tree structures are constructed inside the dissimilar regions. It can be seen that, there are many dissimilar regions which have small areas such that they consist of several grids. Therefore, for only the largest region, magnified version of corresponding point generation result is emphasized in black box in Figure 6-61.

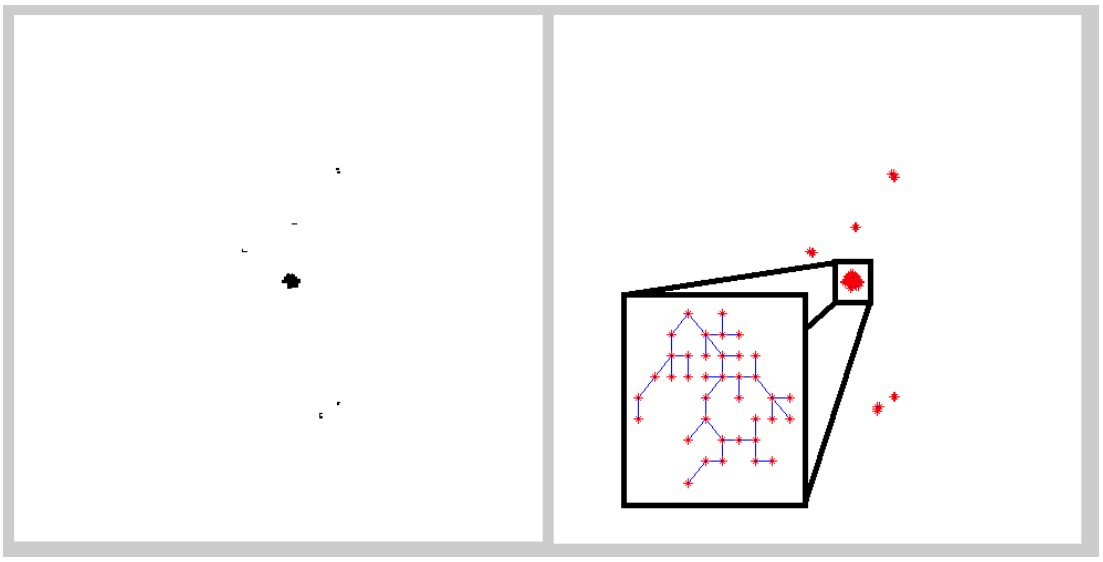


Figure 6-61 Dissimilar regions of the second height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-2

- **Correspondence point sets generated for the connections of second height level, 91 cm, and third height level, 101 cm,**:

In Figure 6-62 similar regions of the second height level, 91 cm, and their generated correspondence point sets are presented. It can be seen that there is no change for similar regions at the height level of 91 cm when compared to the result presented in Figure 6-59. The boundary of the similar region is used to generate correspondence points for this height level.

Figure 6-62 Similar regions of the second height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-2

In Figure 6-63 similar regions of the third height level, 101 cm, and their generated correspondence point sets are presented. Similar to the result given in Figure 6-62, boundary of the similar region is used to generate correspondence points for given height level.

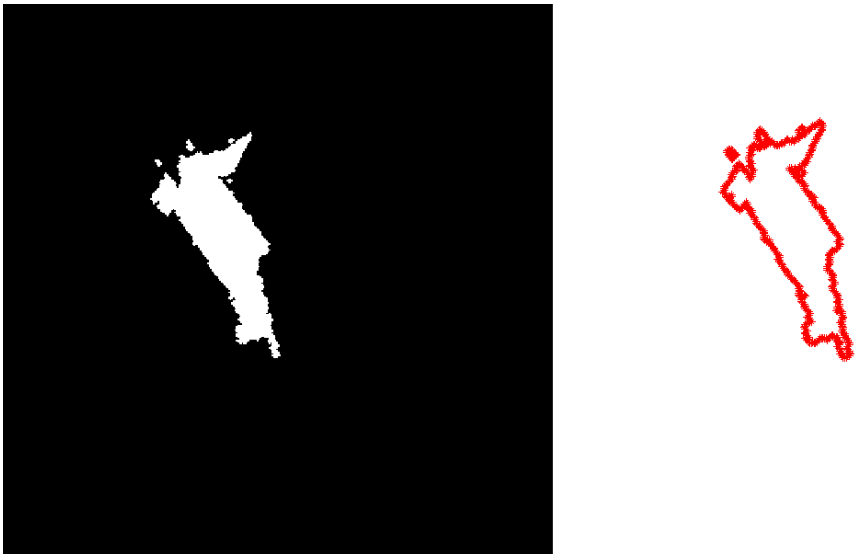


Figure 6-63 Similar regions of the third height level (left), 101 cm, and their generated correspondence point sets (right) for Experiment-2

In Figure 6-64 dissimilar regions of the second height level, 91 cm, and their generated correspondence point sets are presented. When compared to the result given in Figure 6-61, there is no change for the correspondence point generation for the dissimilar regions.

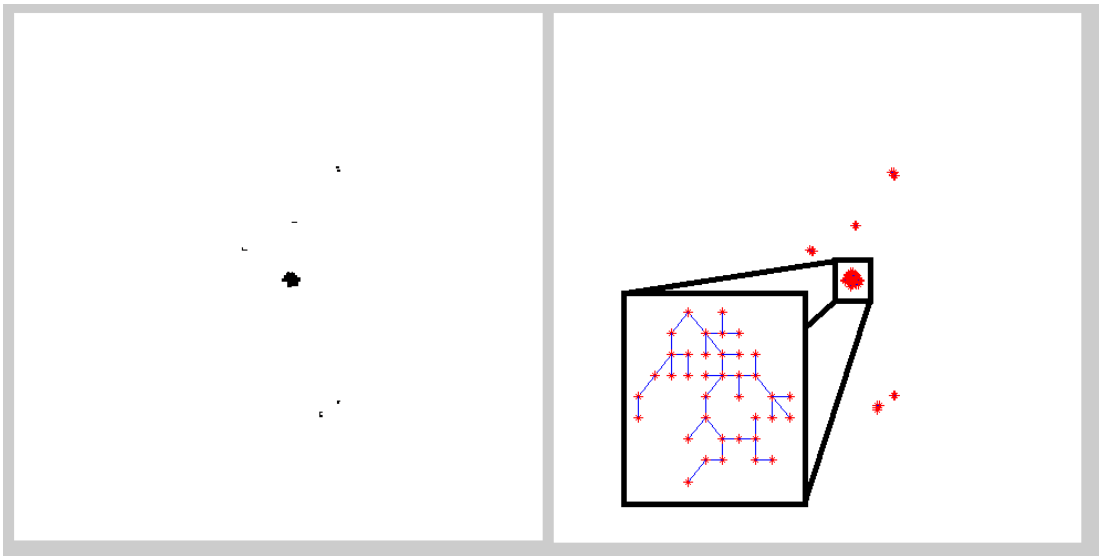


Figure 6-64 Dissimilar regions of the second height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-2

In Figure 6-65 dissimilar regions of the third height level, 101 cm, and their generated correspondence point sets are presented. One can notice that there are only two dissimilar regions in Figure 6-65, which can also be seen in Table 6-1, for which tree structures are constructed inside them to generate correspondence points. Since one of them is too small, magnified version of the generated correspondence points of larger one is emphasized in black box in Figure 6-65.

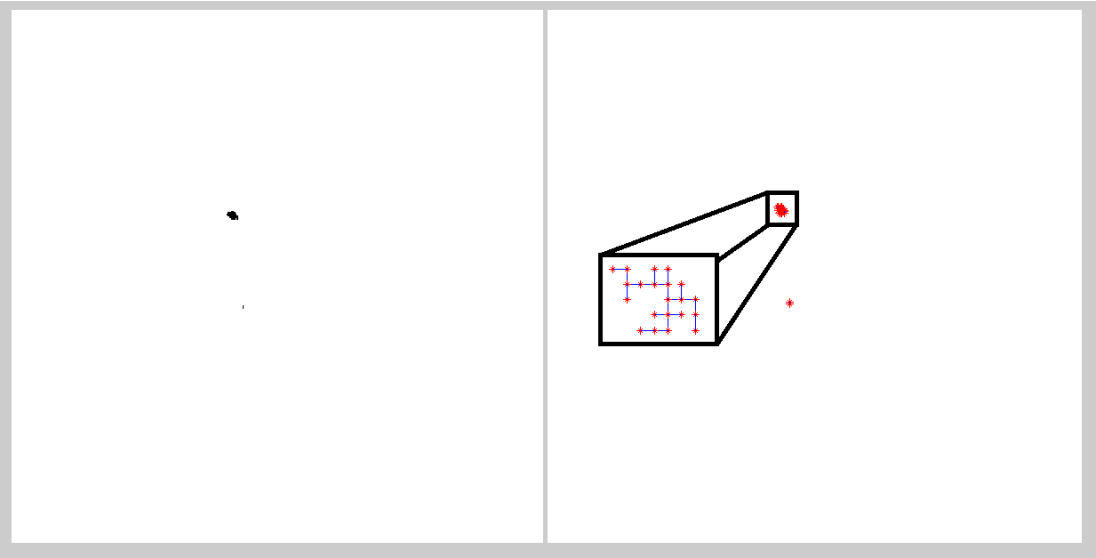


Figure 6-65 Dissimilar regions of the third height level (left), 101 cm, and their generated correspondence point sets (right) for Experiment-2

6.4.3.3.Tiling Correspondence Points

In this part, connections between correspondence point sets are established by using Delaunay Triangulation with determined length limit for edges of the triangles.

At this point, it is important to mention that in order to obtain a good visualization each height level is escalated by 10 cm and then connected to the upper layer.

In Figure 6-66 tiling result for correspondence point sets of first height level, 81 cm, and second height level, 91 cm, is presented. It can be seen that there are approximately no holes on the walls of the environment which shows that low error in map construction and proper alignment between maps are achieved for the given height levels. Additionally, tiling result for the dissimilar regions, correspondence points of which are generated by using tree structures, is encircled by red in Figure 6-66, which shows that smooth growing is achieved at discontinuous part of the object.

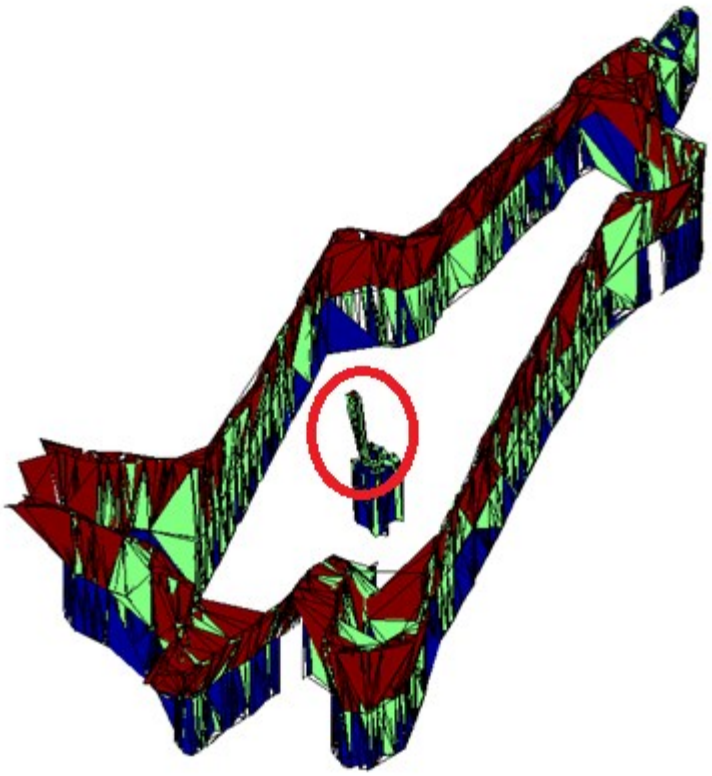


Figure 6-66 Tiling result for correspondence point sets of first height level, 81 cm, and second height level, 91 cm, for Experiment-2

In Figure 6-67 tiling result for correspondence point sets of second height level, 91 cm, and third height level, 101 cm, added to the previous result, given in Figure 6-66 is presented. In the given result, it is observed that there are no holes on the walls of the environment and the reconstructed structure has the similar stairway geometry when compared to the scanned object given in Figure 6-52. The discontinuous parts on the structure, the stairs, are handled by smooth growing applied by the proposed method and these parts are emphasized with the red arrows in Figure 6-67.

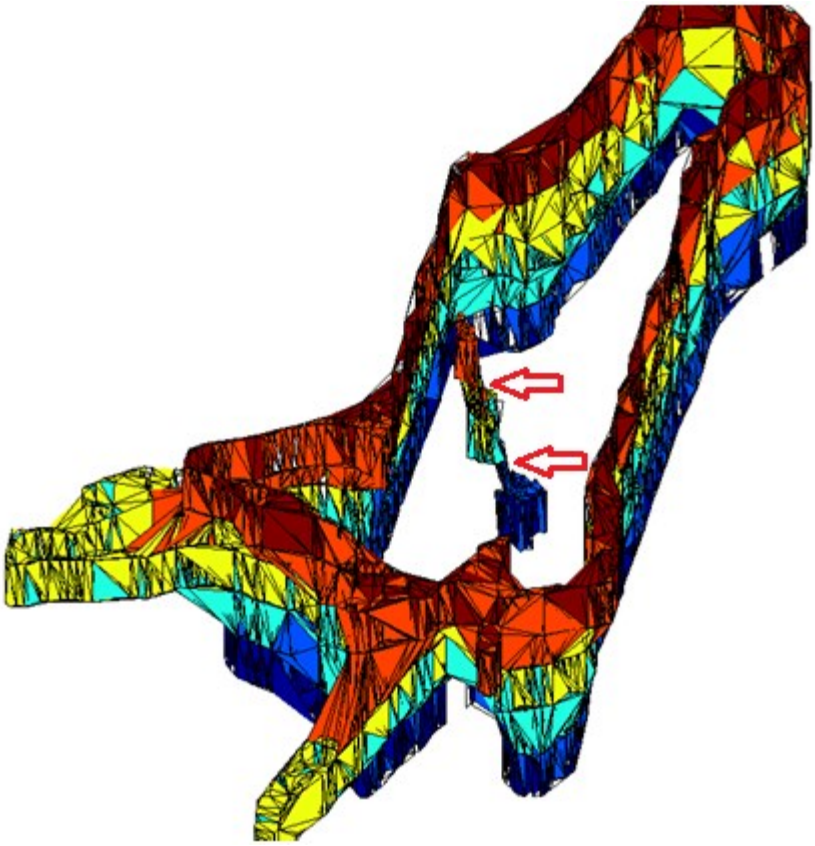


Figure 6-67 Tiling result for correspondence point sets of second height level, 91 cm, and third height level, 101 cm, added to the previous result for Experiment-2

6.4.4. Results

In this subsection, the aim is to evaluate the results of Experiment-2.

In this experiment an object in the indoor environment is scanned. The obtained scan data is used in the system block diagram of which is given in Figure 6-1. The main goal is to test the smooth growing property of the proposed approach. Therefore, an alternative result which does not contain smooth growing and connects different height levels from the boundaries is presented in Figure 6-68. As it can be seen in Figure 6-68 at the discontinuity points, boundaries are directly connected to each other which is not a correct way to reconstruct an object unless height difference between consecutive height levels are very small. Smooth growing allows two different layers to establish connection in an admissible range of overlapping parts for discontinuous, which is mentioned as dissimilar in the present work, regions.

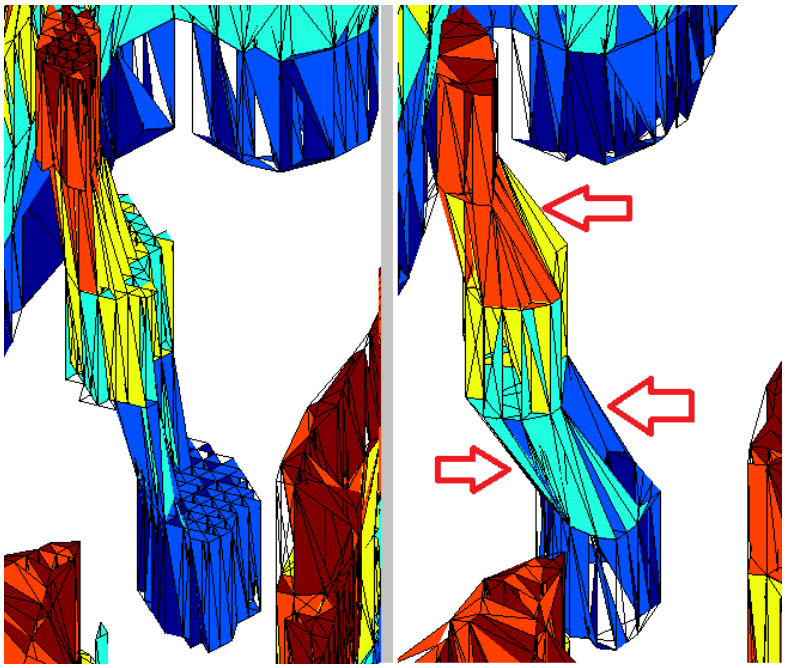


Figure 6-68 Alternative reconstruction result without smooth growing (right) and the original result (left)

In order to achieve a comparison, the real object and its reconstructed version are presented in Figure 6-69. As one can see that the reconstructed version has the same geometry, stairway geometry, with the real object and discontinuities on the real object are handled in the reconstructed version correctly by proposed method.

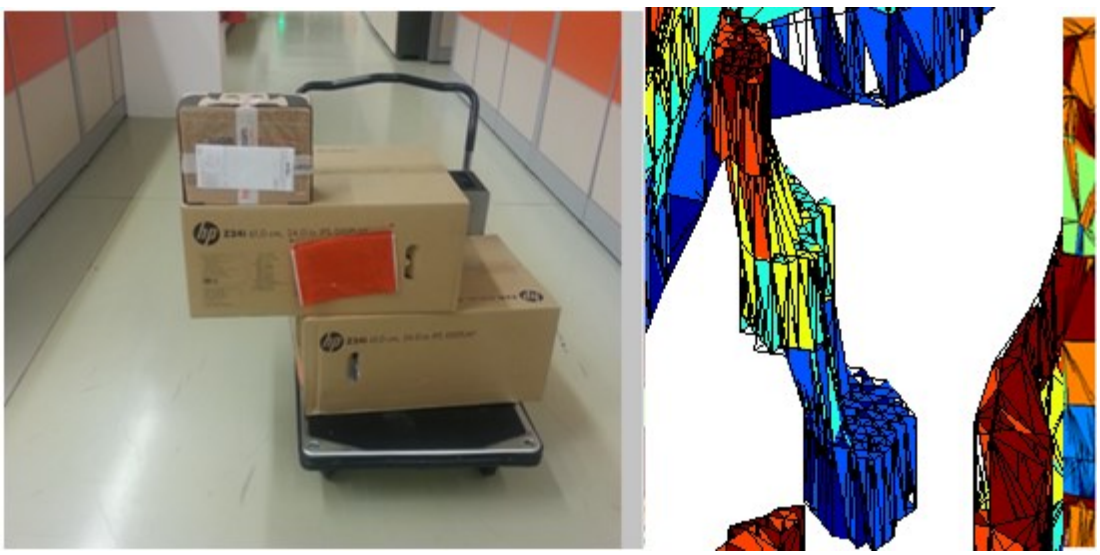


Figure 6-69 Real object (left) and its reconstructed version (right) for Experiment-2

6.5. Experiment 3: Scanning an Object-2

6.5.1. Environment of the Experiment 3

Environment used for Experiment-3 is an indoor environment in ASELSAN which contains an L-shaped table with a drawer and three boxes on it and explored at 8 different height levels which are 53, 66, 81, 86, 91, 96, 101 and 106 cm. Images taken from the environment during this experiment can be seen in Figure 6-70.

Figure 6-70 Environment used for Experiment-3

In Experiment-2, Vehicle-1, Vehicle-2 and Vehicle-3 are utilized to explore the environment from 8 different height levels with height differences of;

- 13cm between first height level, 53 cm and second height level, 66 cm.
- 15cm between second height level, 66 cm and third height level, 81 cm.
- 5cm between other height levels beginning from 81cm.

6.5.2. Fast-SLAM Exploration Results in Experiment 3

When exploration of the environment is completed, obtained laser scans are used in the Fast-SLAM algorithm whose parameters are chosen as follow:

- Grid size: 0.02 m
- Particle number: 500
- Probability change rates:
  - $m_{occ} = -0.5$
  - $m_{free} = +0.05$

As outputs of the Fast-SLAM algorithm 2D grid maps for all different height levels are achieved and presented in Figure 6-71– Figure 6-78. In Figure 6-71, 2D mapping result for the exploration of the indoor environment and the given object at height level of 53 cm is provided. In the 2D grid map result for given height level, regions which are cross sections of the legs of the table, encircled by red, and the bottom part of the drawer, encircled by blue, are observed.

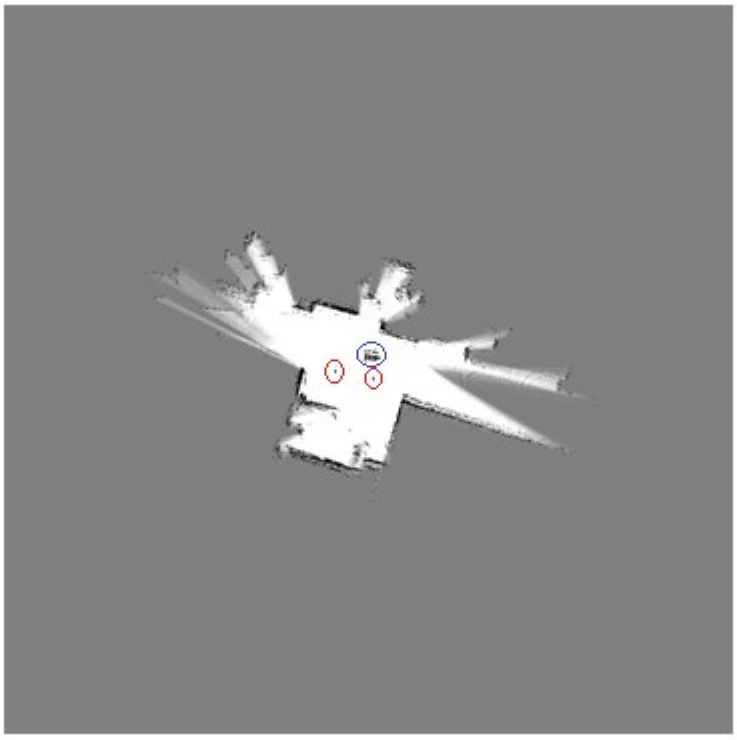


Figure 6-71 2D grid map obtained from Fast-SLAM for first height level, 53 cm, for Experiment-3

In Figure 6-72, 2D mapping result for the exploration of the indoor environment and the given object at height level of 66 cm is provided. When compared to the result given in Figure 6-71, there is no significant change in the 2D map and cross sections of the same parts of the table are seen in the given result.

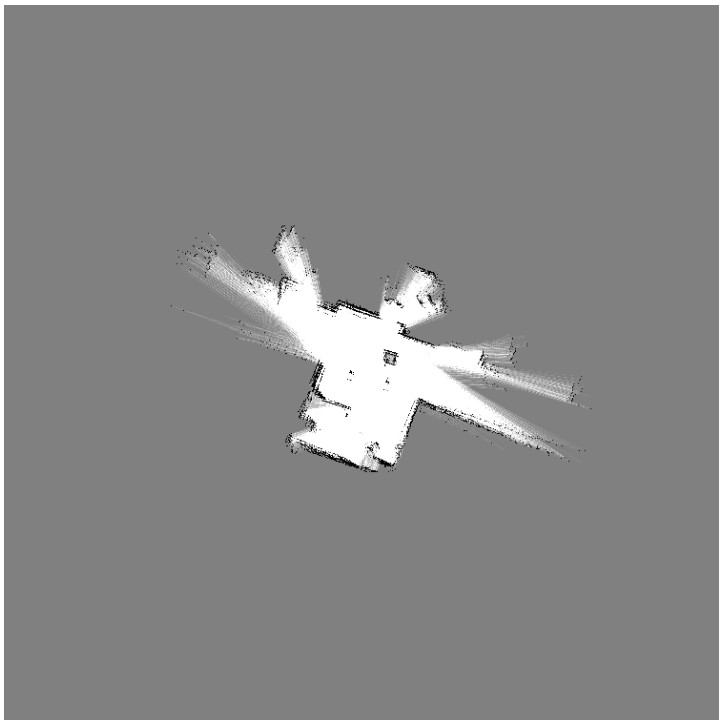


Figure 6-72 2D grid map obtained from Fast-SLAM for second height level, 66 cm, for Experiment-3

In Figure 6-73, 2D mapping result for the exploration of the indoor environment and the given object at height level of 81 cm is provided. It can be seen that there in an additional region observed in the 2D map result at the given height level which is emphasized with red circle. This is the back part of the table which can be observed in the top right image in Figure 6-70.

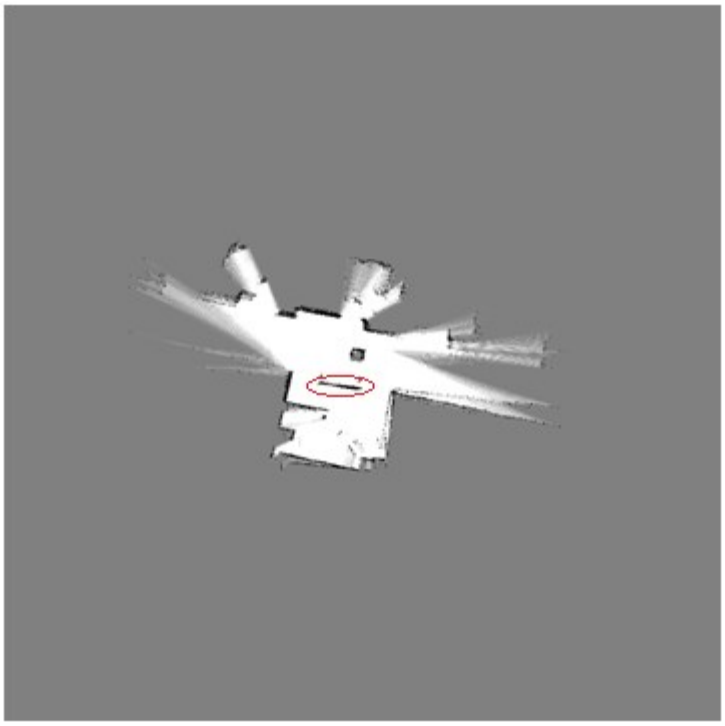


Figure 6-73 2D grid map obtained from Fast-SLAM for third height level, 81 cm, for Experiment-3

In Figure 6-74, 2D mapping result for the exploration of the indoor environment and the given object at height level of 86 cm is provided. One can see that there is no significant change when the previous result, given in Figure 6-73 and the result given in Figure 6-74 is compared. Same regions are obtained for the 2D map result at the given height level.

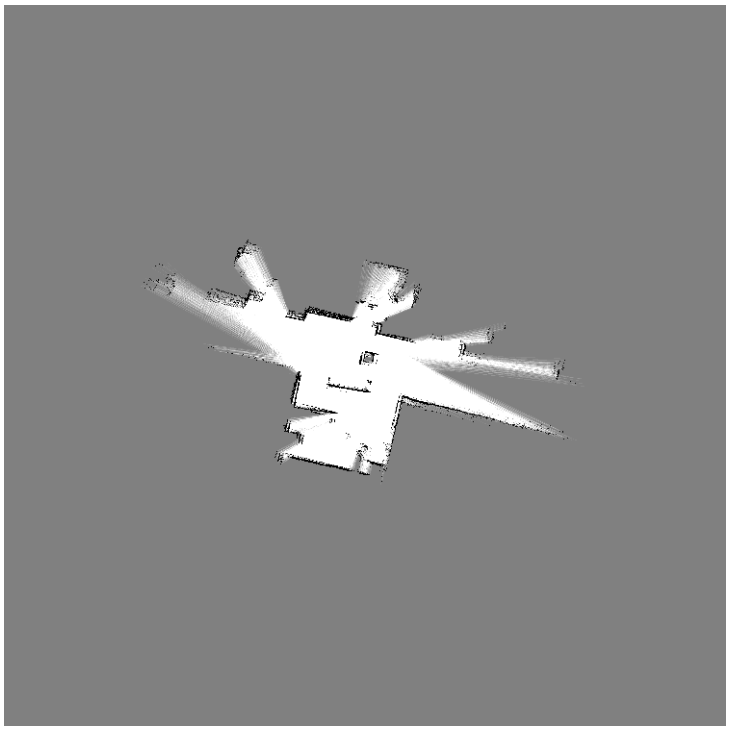


Figure 6-74 2D grid map obtained from Fast-SLAM for fourth height level, 86 cm, for Experiment-3

In Figure 6-75, 2D mapping result for the exploration of the indoor environment and the given object at height level of 91 cm is provided. In the obtained result given in Figure 6-75, the boxes on the L-shaped table, which can be seen in Figure 6-70, are observed and encircled by red.

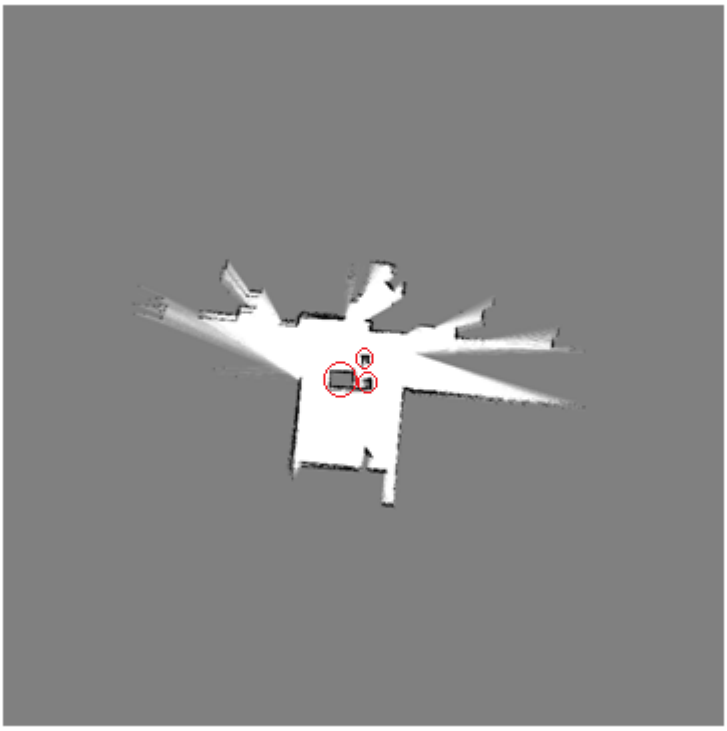


Figure 6-75 2D grid map obtained from Fast-SLAM for fifth height level, 91 cm, for Experiment-3

In Figure 6-76, 2D mapping result for the exploration of the indoor environment and the given object at height level of 96 cm is provided. One can see that there is no significant change when the previous result, given in Figure 6-75 and the result given in Figure 6-76 is compared. Same regions are obtained for the 2D map result at the given height level.

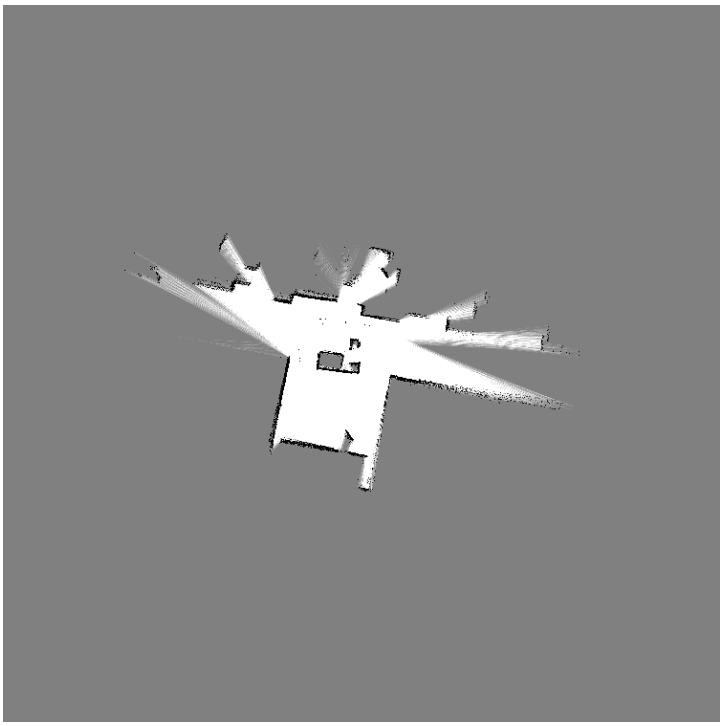


Figure 6-76 2D grid map obtained from Fast-SLAM for sixth height level, 96 cm, fore Experiment-3

In Figure 6-77, 2D mapping result for the exploration of the indoor environment and the given object at height level of 101 cm is provided. It can be observed that, only the top back part of the L-shaped table, which can be seen in the images of the environment given in Figure 6-70, is seen and encircled by red in Figure 6-77.

Figure 6-77 2D grid map obtained from Fast-SLAM for seventh height level, 101 cm, for Experiment-3

In Figure 6-78, 2D mapping result for the exploration of the indoor environment and the given object at height level of 106 cm is provided. When the result given in Figure 6-78 is compared to the previous result given in Figure 6-77, there is only orientation difference between the obtained 2D grid maps. Apart from this, there is no significant change when the two results are compared to each other such that the same region is obtained for the 2D map result at the given height level.

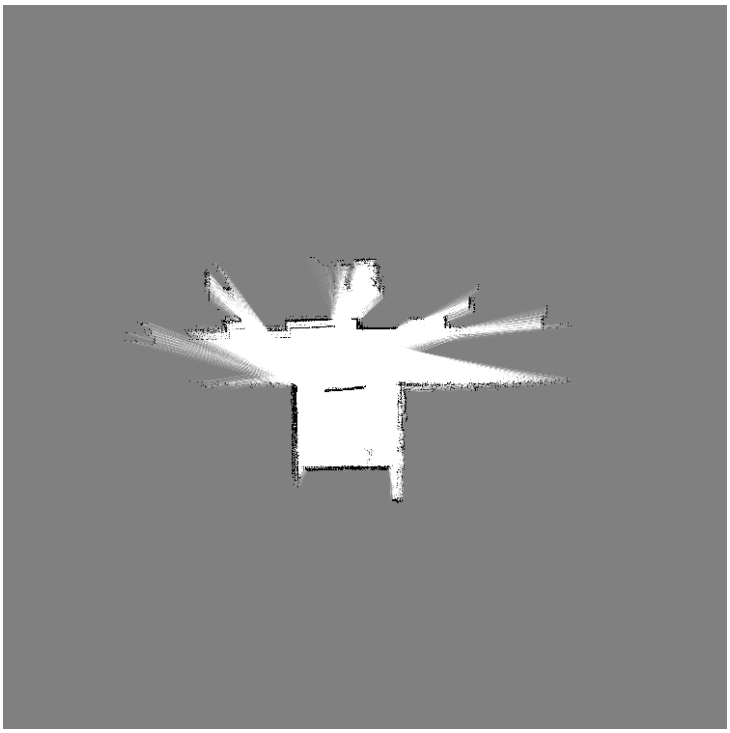


Figure 6-78 2D grid map obtained from Fast-SLAM for eighth height level, 106 cm, for Experiment-3

6.5.3. Map Growing

After obtaining 2D grid maps for different height levels, map growing method is applied in the experiment. Initially, correct correspondences are created between successive maps by first segmenting meaningful regions, then aligning maps by using features extracted from these regions and finally, separating the regions as similar and dissimilar according to similarity measure between them. Afterwards, correspondence points are generated in order to be used while connecting regions on the maps for reconstruction purpose which is defined in the final step namely; tiling of the correspondence points. In map growing method following values for the parameters are utilized.

- Percentage Threshold for Separation of Regions: 70%
- Length Limit for Tiling: 11 cm

6.5.3.1.Creating Correct Correspondences

In order to visualize the achieved alignments between successive maps, boundaries of the segmented regions are utilized. In Figure 6-79 alignment between first height level (red), 53 cm, and second height level (blue), 66 cm, is presented. The aim is to illustrate the alignment results for the maps with respect to each other. One can see that, a proper alignment between the two maps is achieved, apart from the several mismatches which are caused by the uncertainty regions in the maps, emphasized with green circles.

Figure 6-79 Achieved alignment between first height level (red), 53 cm, and second height level (blue), 66 cm, for Experiment-3

In Figure 6-80 alignment between second height level (red), 66 cm, and third height level (blue), 81 cm, is presented. When compared to the previous result given in Figure 6-79, there are slight alignment errors at the boundary of several regions (not including the errors in uncertainty regions), which are encircled by green. Apart from

that, there an additional region, back part of the table, at the height level of 81 cm, which is previously observed in Figure 6-73.

Figure 6-80 Achieved alignment between second height level (red), 66 cm, and third height level (blue), 81 cm, for Experiment-3

In Figure 6-81 alignment between third height level (red), 86 cm, and fourth height level (blue), 91 cm, is presented. Similar to the result given in Figure 6-80, there are small errors in the alignment result, which are encircled by green, but in overall a proper alignment is achieved.

Figure 6-81 Achieved alignment between third height level (red), 81 cm, and fourth height level (blue), 86 cm, for Experiment-3

In Figure 6-82 alignment between fourth height level (red), 86 cm, and fifth height level (blue), 91 cm, is presented. For the given result, alignment between the two maps is more successful than the previous results, since the walls of the environment for both height levels overlap with each other. However, there are some regions that are not overlapped with each other. These are not the alignment errors but discontinuities in the environment, which are shown with green circle in Figure 6-82.

Figure 6-82 Achieved alignment between fourth height level (red), 86 cm, and fifth height level (blue), 91 cm, for Experiment-3

In Figure 6-83 alignment between fifth height level (red), 91 cm, and sixth height level (blue), 96 cm, is presented. It can be seen that the obtained alignment result has approximately no error, apart from the ones caused by uncertainty regions.

Figure 6-83 Achieved alignment between fifth height level (red), 91 cm, and sixth height level (blue), 96 cm, for Experiment-3

In Figure 6-84 alignment between sixth height level (red), 96 cm, and seventh height level (blue), 101 cm, is presented. Similar to the result given in Figure 6-82, there are a few alignment errors caused by the uncertainty regions. Mismatches seen in the figure are discontinuities in the environment, especially on the scanned object, which are encircled by green.

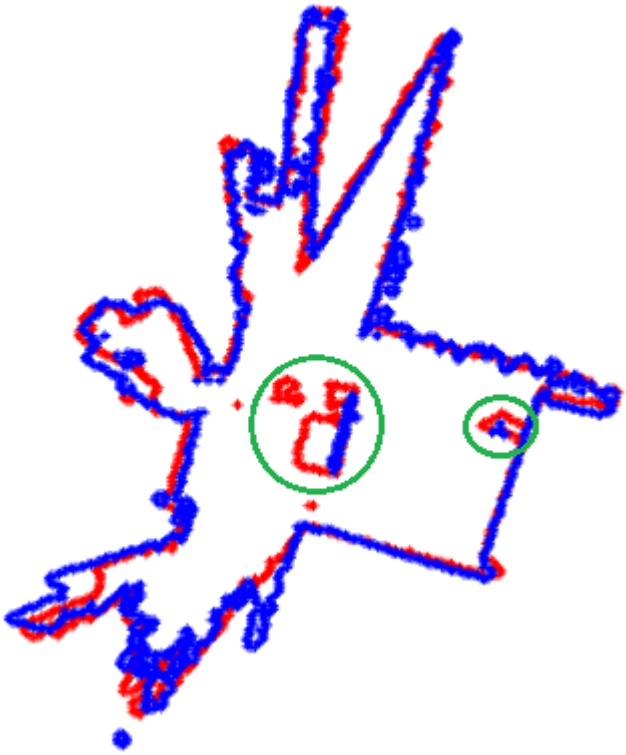


Figure 6-84 Achieved alignment between sixth height level (red), 96 cm, and seventh height level (blue), 101 cm, for Experiment-3

In Figure 6-85 alignment between seventh height level (red), 101 cm, and eighth height level (blue), 106 cm, is presented. In the given result there are only a few alignment errors caused by the uncertainty regions which are shown with green circles. Apart from them a proper alignment is obtained in the given result.

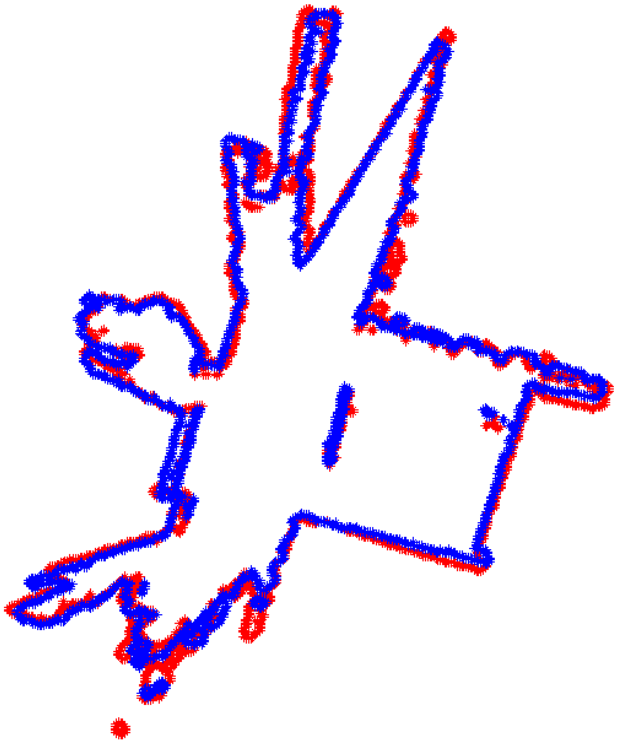


Figure 6-85 Achieved alignment between seventh height level (red), 101 cm, and eighth height level (blue), 106 cm, for Experiment-3

After alignment of the maps, similarities between regions is measured in percentage via applying grid by grid comparison and percentage threshold of 70% is used to designate similar and dissimilar regions, results of which are presented in Table 6-3 - Table 6-9. When each table is examined it can be seen that the largest similarity belongs to the environment itself. Apart from the environment, there are only two regions regarded as similar which are seen in Table 6-3 and Table 6-9. The other regions are regarded as dissimilar.

Table 6-3 Similarity percentage of regions between first height level, 53 cm, and second height level, 66 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70% for Experiment-3

Region Numbers	Region Numbers																																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
1	99.48	0.0002	0.0045	0	0.0013	0	0.0002	0	0	0	0	0.0007	0	0	0.0002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0105	0.0006	0.0006	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0.0013	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	55.738	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0.0004	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0.0019	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

Table 6-4 Similarity percentage of regions between second height level, 66 cm, and third height level, 81 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70% for Experiment-3

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
1	99.355	0	0	0.0002	0	0.0007	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0002	0	0	0	0	0.0004	0	0.0004	0	0	0.0004	0	0	0.0007	0
2	0.0004	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0.0011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	65.574	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0.0004	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	3.2051	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	5.8824	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4.5977	0	1.1494	1.1494	0	12.644	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

Table 6-5 Similarity percentage of regions between third height level, 81 cm, and fourth height level, 86 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70% for Experiment-3

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	99.423	0	0.0004	0.0002	0.0021	0	0	0	0	0	0	0	0	0	0.0004
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0.0013	0	0	0	0	4.0816	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	61.202	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	11.538	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0.0009	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0.0009	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0.0004	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	0.0004	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33	0.0013	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	0.0004	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

Table 6-6 Similarity percentage of regions between fourth height level, 86 cm, and fifth height level, 91 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70% for Experiment-3

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12
1	99.365	0	0	0	0.0015	0	0	0	0	0	0	0
2	0.0007	0	0	0	0	0	0	0	0	0	0	0
3	0.0006	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	25.556	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	15.707	0	0	0
12	0	0	0	0	0	0	0	0	1.0471	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

Table 6-7 Similarity percentage of regions between fifth height level, 91 cm, and sixth height level, 96 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70% for Experiment-3

Region Numbers	1	2	3	4	5	6	7	8	9
1	99.68	0	0	0	0	0.0009	0	0	0
2	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0
4	0.0004	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0
7	0	0	0	67.708	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0
9	0	0	0	0	79.058	0	0	0	0
10	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

Table 6-8 Similarity percentage of regions between sixth height level, 96 cm, and seventh height level, 101 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70% for Experiment-3

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	99.46	0	0	0.0002	0.0004	0.0015	0	0	0	0	0	0	0	0	0	0	0.0004	0.0007	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	16.134	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	3.7037	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMILAR	
DISSIMILAR	

Table 6-9 Similarity percentage of regions between seventh height level, 101 cm, and eighth height level, 106 cm, and designated similar and dissimilar regions with respect to percentage threshold of 70% for Experiment-3

Region Numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
1	99.484	0	0.0002	0	0	0	0	0.0038	0.0004	0.0006	0.0004	0.003	0.0004	0.0008	0.0017	0	0.0009	0	0	0	0	0.0002	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0.0004	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0.0026	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0.0002	0	0	0	9.5238	19.048	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0.0007	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	70.492	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Region Numbers

SIMILAR	
DISSIMILAR	

In the experiments system feedback is generated for detecting and focusing on the discontinuities in the environment. However, in this experiment the aim is to focus on a selected object by using the proposed method and reconstruct 3D structure of it. Therefore, it is not significant to evaluate the discontinuities in the environment and focus on them for this experiment.

6.5.3.2.Generating Correspondence Point

Correspondence point sets are generated for similar and dissimilar regions by using different approaches, which are using boundary points and using vertices of the tree structures constructed in the regions, as explained in Chapter 4. In this part, results of correspondence point set generation are presented for Experiment-3.

- **Correspondence point sets generated for the connections of first height level, 53 cm, and second height level, 66 cm,:**

In Figure 6-86 similar regions of the first height level, 53 cm, and their generated correspondence point sets are presented. As previously seen in Table 6-3, there is only one similar region and for this similar region correspondence points are generated by using boundary parts of the region, shown in Figure 6-86.

Figure 6-86 Similar regions of the first height level (left), 53 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-87 similar regions of the second height level, 66 cm, and their generated correspondence point sets are presented. According to the Table 6-3, there is only one similar region and for this similar region correspondence points are generated by using boundary parts of the region, shown in Figure 6-87.

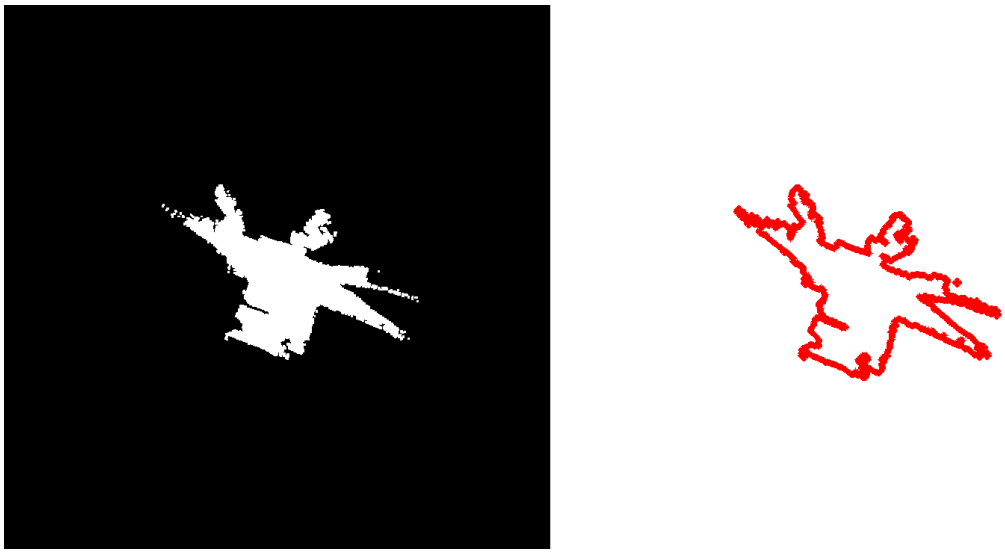


Figure 6-87 Similar regions of the second height level (left), 66 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-88 dissimilar regions of the first height level, 53 cm, and their generated correspondence point sets are presented. As seen in Table 6-3, there are many dissimilar regions and for these regions correspondence points are generated by constructing tree structures. Among these regions, magnified version of the correspondence points for the largest one is emphasized in black box in Figure 6-88.

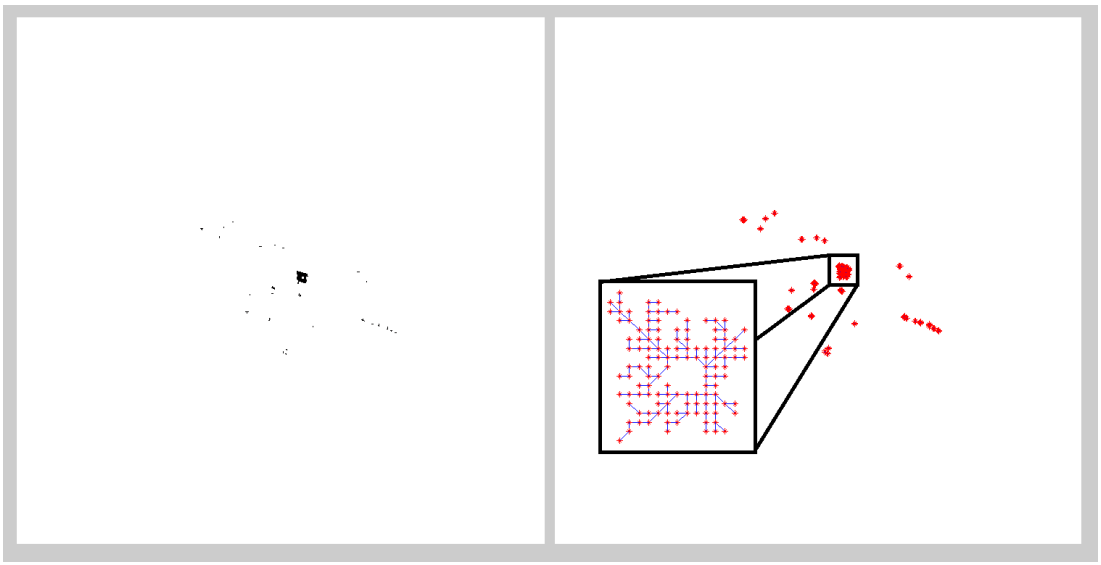


Figure 6-88 Dissimilar regions of the first height level (left), 53 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-89 dissimilar regions of the second height level, 66 cm, and their generated correspondence point sets are presented. Similar to the result given in Figure 6-88, tree structures are constructed inside regions for correspondence point generation. In the magnified version of the correspondence points for largest region, given in black box in Figure 6-89, one can see the vertices and the branches of tree structures which are respectively correspondence points and constraint connections of the correspondence points.

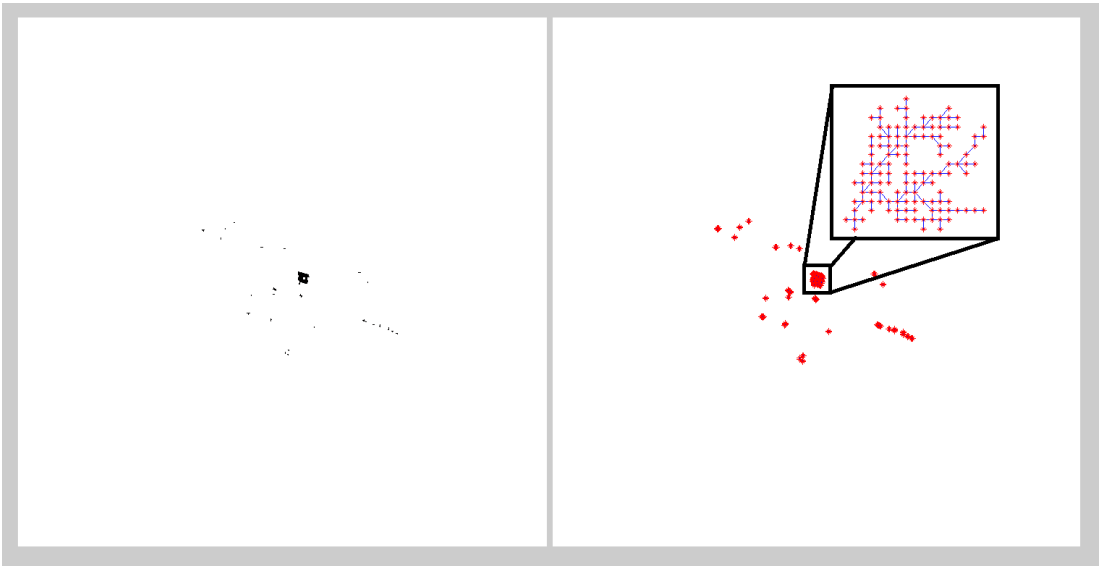


Figure 6-89 Dissimilar regions of the second height level (left), 66 cm, and their generated correspondence point sets (right) for Experiment-3

- **Correspondence point sets generated for the connections of second height level, 66 cm, and third height level, 81 cm,:**

In Figure 6-90 similar regions of the second height level, 66 cm, and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-87, there is no change in the correspondence point generation result for the height level of 66 cm.

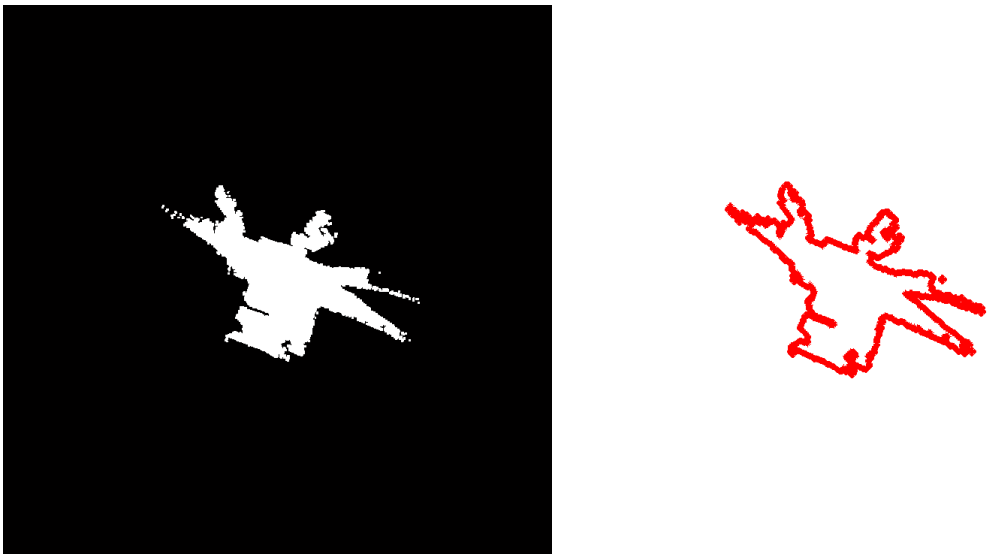


Figure 6-90 Similar regions of the second height level (left), 66 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-91 similar regions of the third height level, 81 cm, and their generated correspondence point sets are presented. As seen in Table 6-4, there is only one similar region for the given height level, and for this regions correspondence points are generated by using boundary parts of the region, shown in the Figure 6-91.

Figure 6-91 Similar regions of the third height level (left), 81 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-92 dissimilar regions of the second height level, 66 cm, and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-89, there is no change in the correspondence point generation result for the height level of 66 cm.

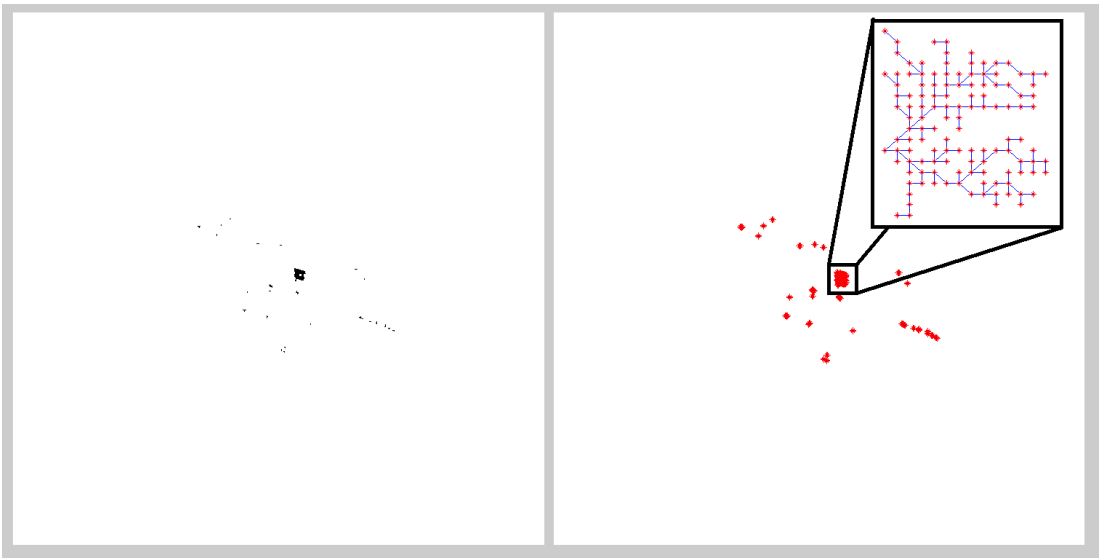


Figure 6-92 Dissimilar regions of the second height level (left), 66 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-93 dissimilar regions of the third height level, 81 cm, and their generated correspondence point sets are presented. It can be seen that there are three large regions among the dissimilar regions for the height level of 81 cm, and magnified version of the correspondence point generation for these regions are emphasized in black boxes presented in Figure 6-93.

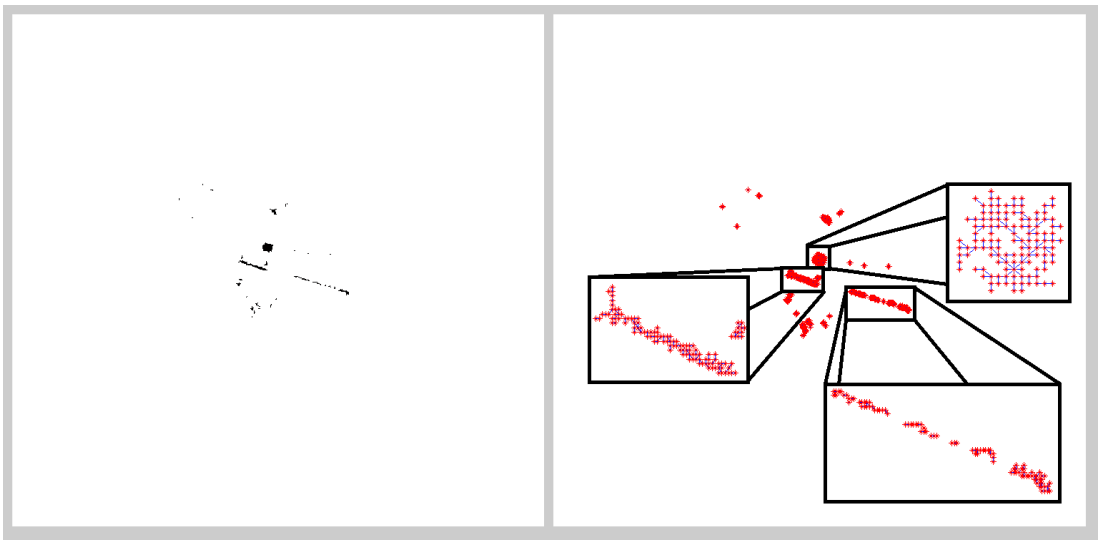


Figure 6-93 Dissimilar regions of the third height level (left), 81 cm and their generated correspondence point sets (right) for Experiment-3

- **Correspondence point sets generated for the connections of third height level, 81 cm and fourth height level, 86 cm,:**

In Figure 6-94 similar regions of the third height level, 81 cm, and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-91, there is no change in the correspondence point generation result for the height level of 81 cm.

Figure 6-94 Similar regions of the third height level (left), 81 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-95 similar regions of the fourth height level, 86 cm, and their generated correspondence point sets are presented. As seen in Table 6-5 there is only one similar region for the given height level, and for this regions correspondence points are generated by using boundary parts of the region, shown in the Figure 6-95.

Figure 6-95 Similar regions of the fourth height level (left), 86 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-96 dissimilar regions of the third height level, 81 cm, and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-93, there is no change in the correspondence point generation result for the height level of 81 cm.

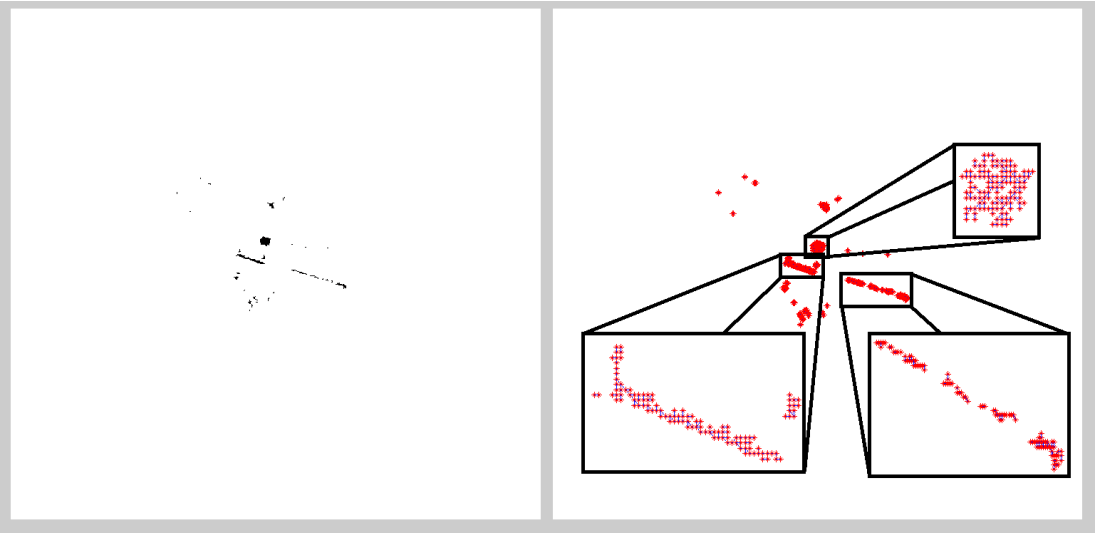


Figure 6-96 Dissimilar regions of the third height level (left), 81 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-97 dissimilar regions of the fourth height level, 86 cm, and their generated correspondence point sets are presented. It can be seen that similar to the result given in Figure 6-96, there are three large dissimilar regions and correspondence points for these regions are generated by constructing tree structures inside them.

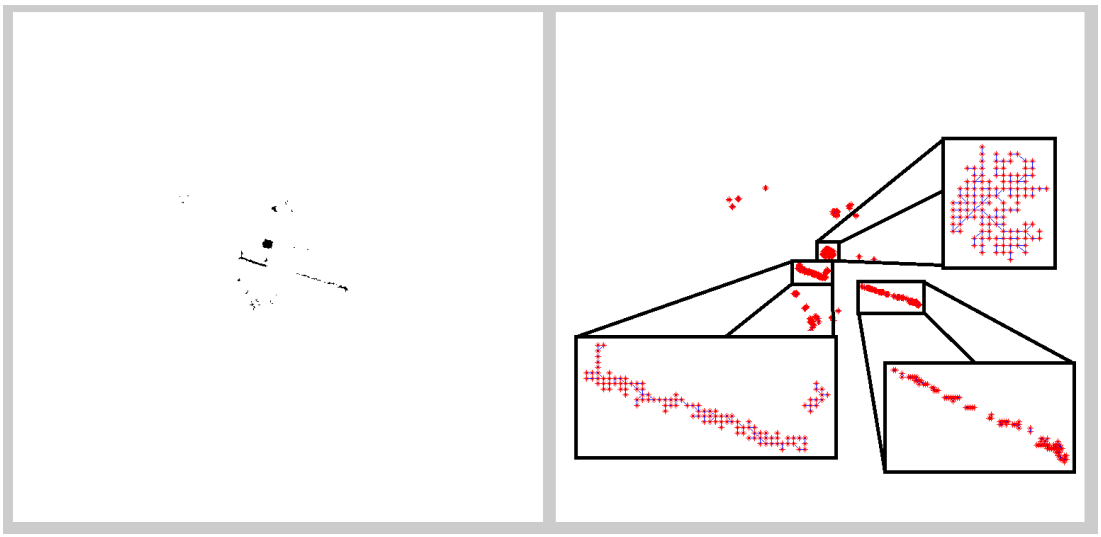


Figure 6-97 Dissimilar regions of the fourth height level (left), 86 cm, and their generated correspondence point sets (right) for Experiment-3

- **Correspondence point sets generated for the connections of fourth height level, 86 cm, and fifth height level, 91 cm,:**

In Figure 6-98 similar regions of the fourth height level, 86 cm, and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-95, there is no change in the correspondence point generation result for the height level of 86 cm.

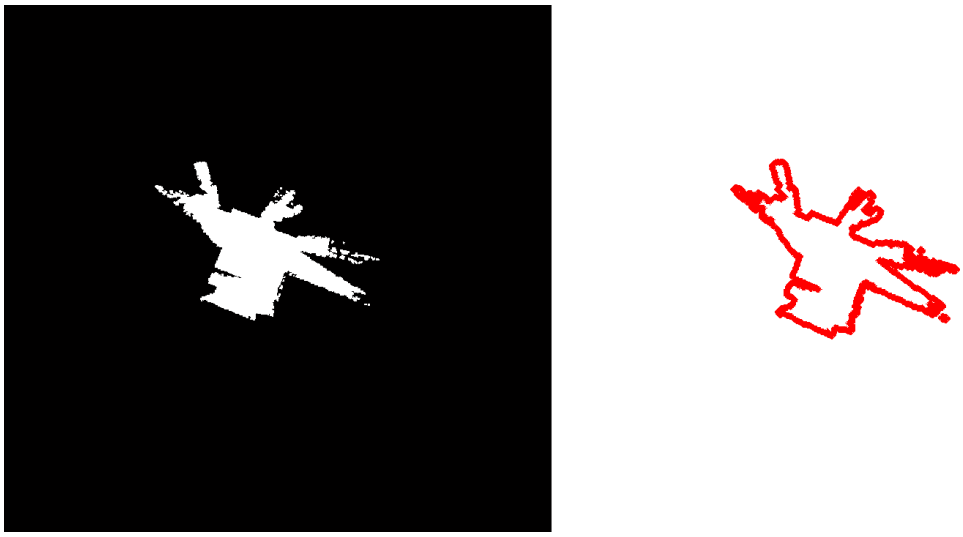


Figure 6-98 Similar regions of the fourth height level (left), 86 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-99 similar regions of the fifth height level, 91 cm, and their generated correspondence point sets are presented. As seen in Table 6-6 there is only one similar region for the given height level, and for this regions correspondence points are generated by using boundary parts of the region, shown in the Figure 6-99.

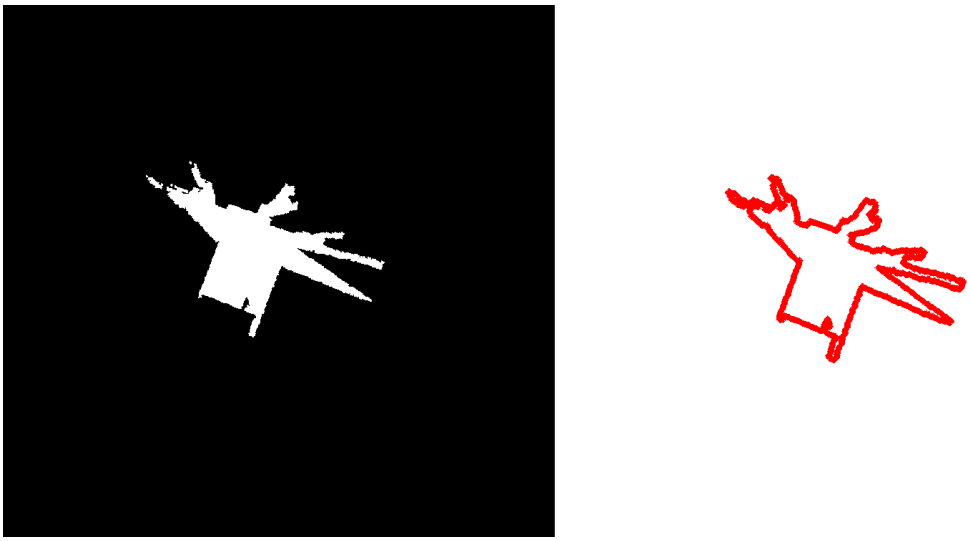


Figure 6-99 Similar regions of the fifth height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-100 dissimilar regions of the fourth height level, 81 cm, and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-97, there is no change in the correspondence point generation result for the height level of 86 cm.

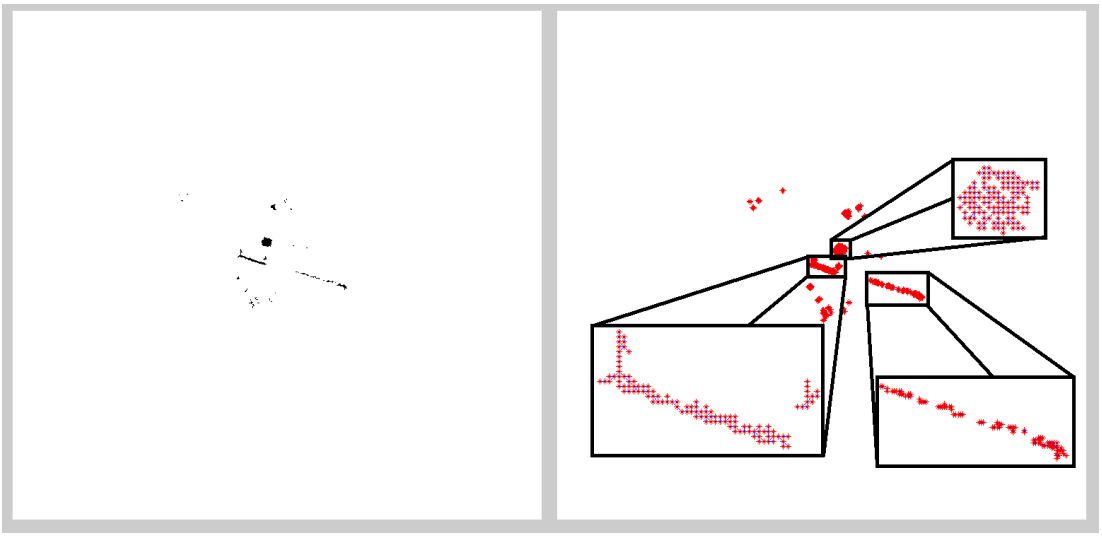


Figure 6-100 Dissimilar regions of the fourth height level (left), 86 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-101 dissimilar regions of the fifth height level, 91 cm, and their generated correspondence point sets are presented. From the result it can be seen that, there are two large dissimilar regions and correspondence points for these regions are generated by using tree structures, magnified version of which can be seen in the black box in Figure 6-101. Additionally, one can notice that, the shape and position of the dissimilar regions seen in both Figure 6-100 and Figure 6-101 are very different, which means there is high discontinuity between the given height levels.

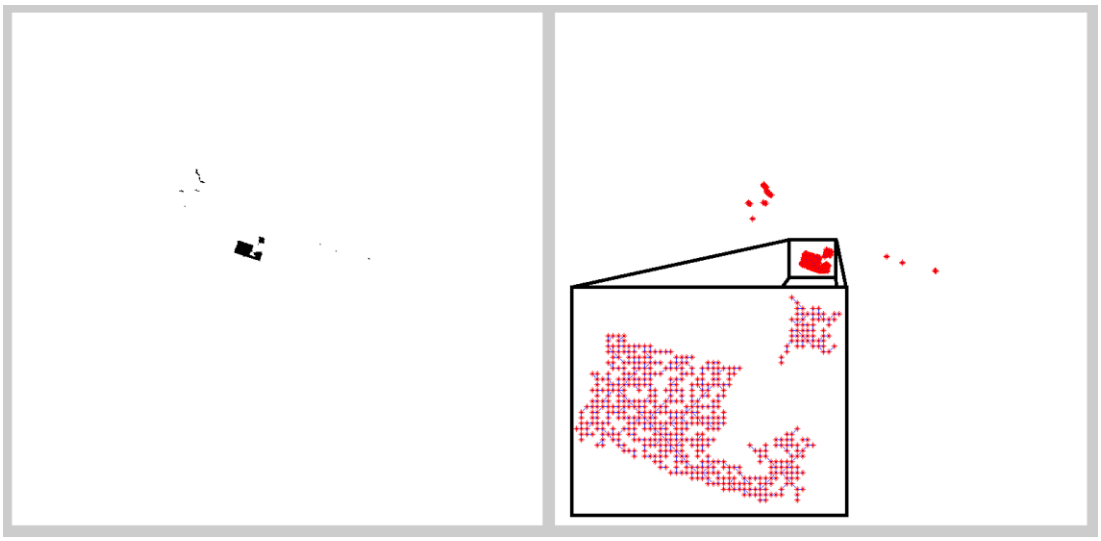


Figure 6-101 Dissimilar regions of the fifth height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-3

- **Correspondence point sets generated for the connections of fifth height level, 91 cm, and sixth height level, 96 cm,:**

In Figure 6-102 similar regions of the fifth height level, 91 cm, and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-99, there is an additional similar region, which can also be seen in Table 6-7, emphasized with green circle in Figure 6-102.

Figure 6-102 Similar regions of the fifth height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-103 similar regions of the sixth height level, 96 cm, and their generated correspondence point sets are presented. Similar to the result given in the Figure 6-102, there is another similar region apart from the environment which is shown with green circle in Figure 6-103. Correspondence points for this region are generated by using boundary regions as expected.

Figure 6-103 Similar regions of the sixth height level (left), 96 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-104 dissimilar regions of the fifth height level, 91 cm, and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-101, the region regarded as similar and shown in Figure 6-102 with green circle is subtracted among the dissimilar regions. Therefore correspondence point generation result for this region is not in the current result given in Figure 6-104.

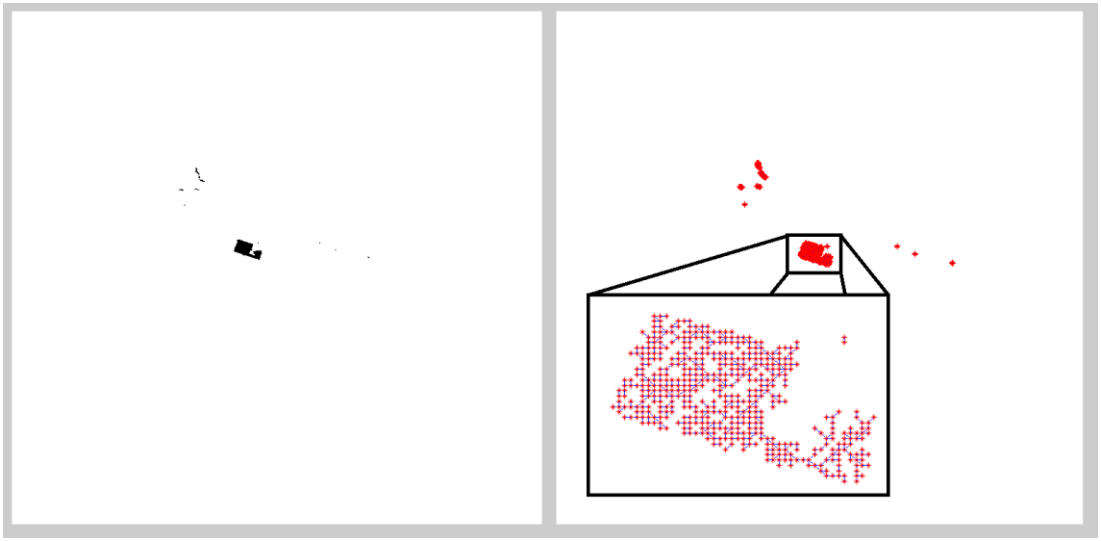


Figure 6-104 Dissimilar regions of the fifth height level (left), 91 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-105 dissimilar regions of the sixth height level, 96 cm and their generated correspondence point sets are presented. Similar to the result given in Figure 6-104, tree structures are constructed inside dissimilar regions and magnified version of the result for the largest region is shown in black box in Figure 6-105.

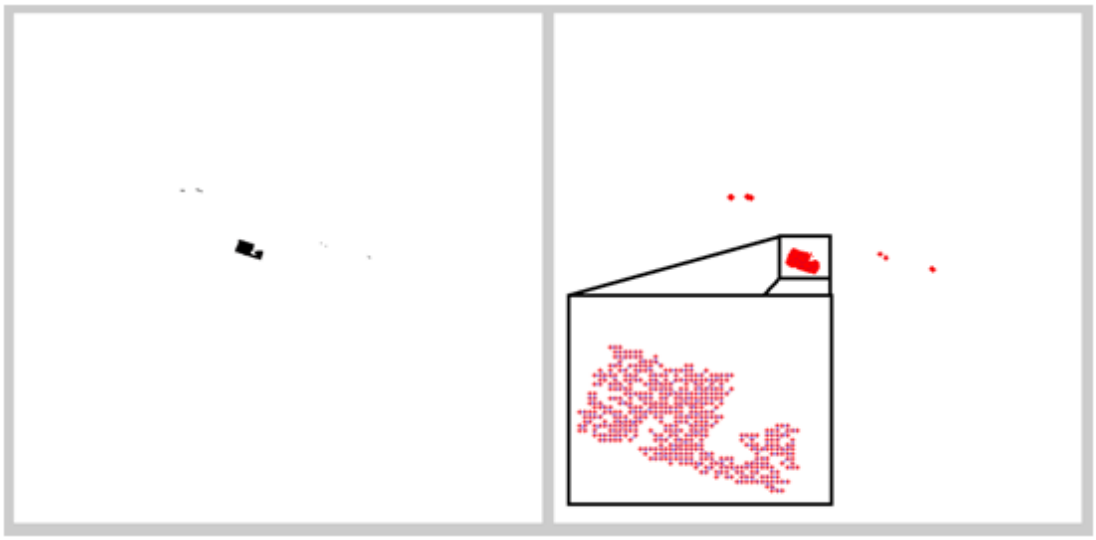


Figure 6-105 Dissimilar regions of the sixth height level (left), 96 cm, and their generated correspondence point sets (right) for Experiment-3

- **Correspondence point sets generated for the connections of sixth height level, 96 cm, and seventh height level, 101 cm,:**

In Figure 6-106 similar regions of the sixth height level, 96 cm, and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-103, the region, shown in green circle, which is regarded as similar is now considered as dissimilar, so that it is not among the similar regions in Figure 6-106, which can also be seen in Table 6-8.

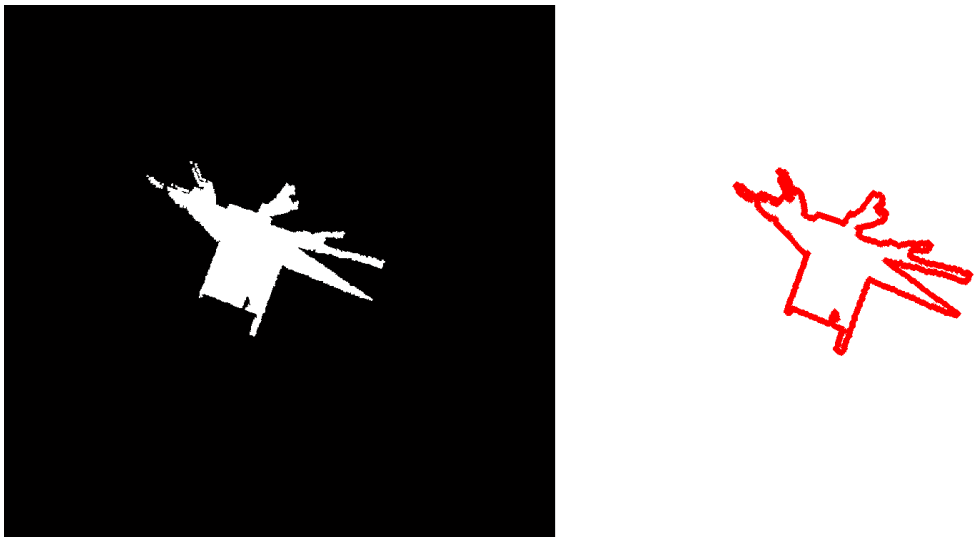


Figure 6-106 similar regions of the sixth height level (left), 96 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-107 similar regions of the seventh height level, 101 cm, and their generated correspondence point sets are presented. When the results of Figure 6-106 and Figure 6-107 are analyzed it can be seen that, boundary points of the similar regions, which are the walls of the environment for both maps, are used for correspondence point generation for both maps, as expected.

Figure 6-107 Similar regions of the seventh height level (left), 101 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-108 dissimilar regions of the sixth height level, 96 cm, and their generated correspondence point sets are presented. The region regarded as similar previously in Figure 6-103, is not seen in the result of corresponding point generation for dissimilar regions given in Figure 6-105. However, for this case, the region is regarded as dissimilar, so it is seen in Figure 6-108, emphasized with green circle.

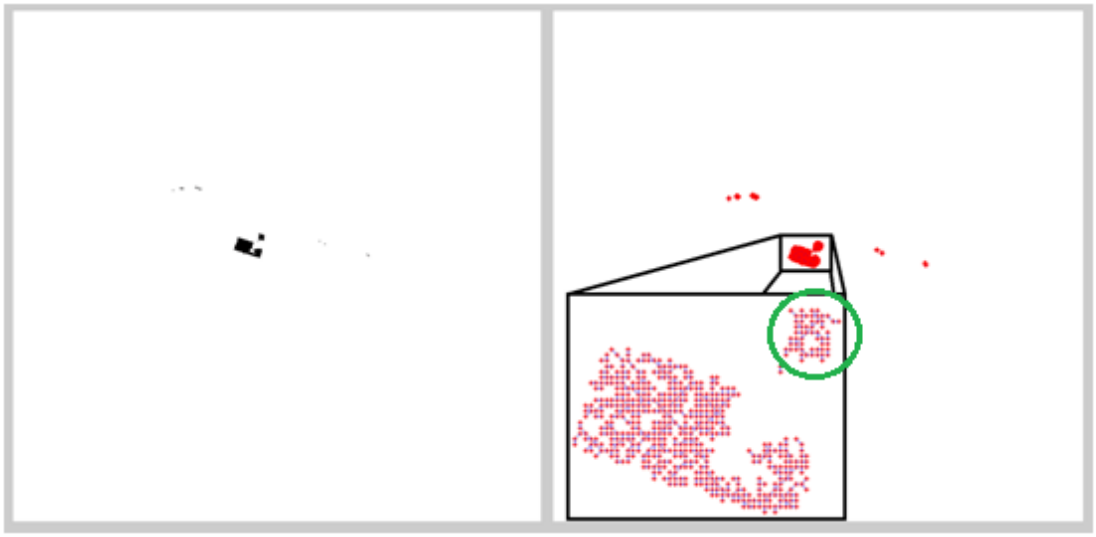


Figure 6-108 Dissimilar regions of the sixth height level (left), 96 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-109 dissimilar regions of the seventh height level, 101 cm, and their generated correspondence point sets are presented. When compared to the result given in Figure 6-108, shape and position of dissimilar regions, observed in the result given in Figure 6-109, are very different. This means there is high discontinuity between given height levels.

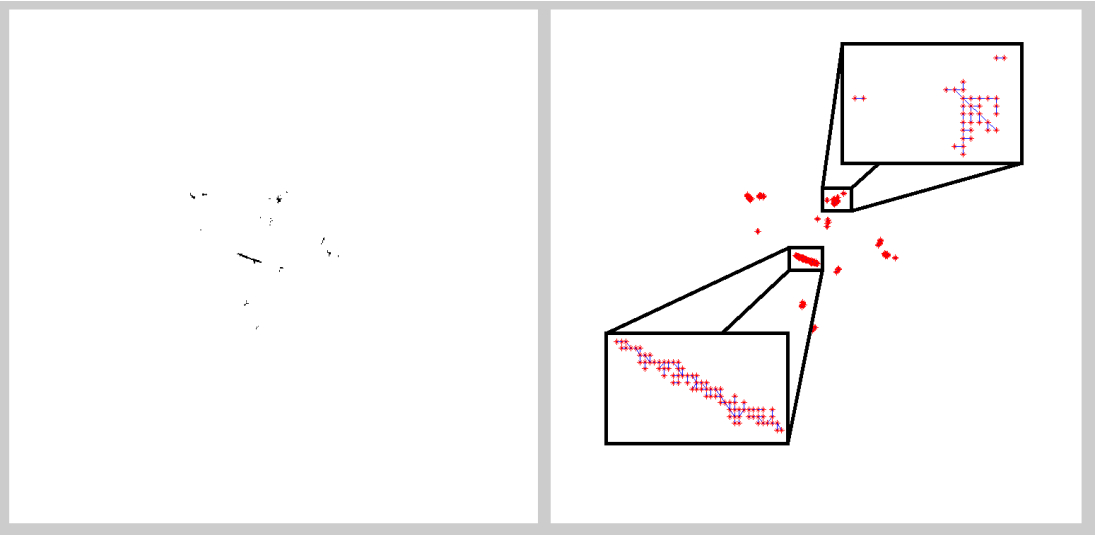


Figure 6-109 Dissimilar regions of the seventh height level (left), 101 cm, and their generated correspondence point sets (right) for Experiment-3

- **Correspondence point sets generated for the connections of seventh height level, 101 cm, and eighth height level, 106 cm,:**

In Figure 6-110 similar regions of the seventh height level, 101 cm, and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-106, there is no change in the correspondence point generation result for the height level of 101 cm.

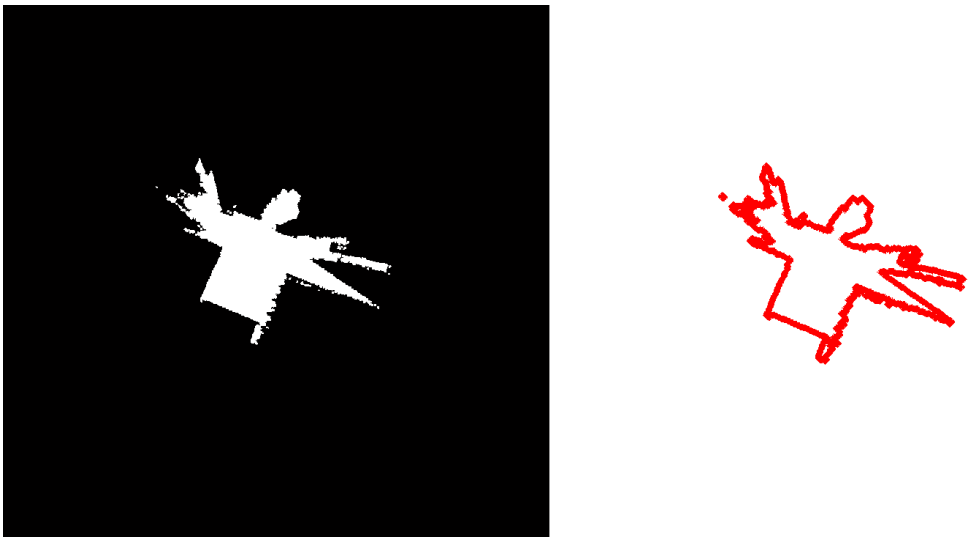


Figure 6-110 Similar regions of the seventh height level (left), 101 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-111 similar regions of the eighth height level, 106 cm, and their generated correspondence point sets are presented. Similar to the result obtained in Figure 6-110, boundary of the similar region is used to generate correspondence points for given height level.

Figure 6-111 Similar regions of the eighth height level (left), 106 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-112 dissimilar regions of the seventh height level and their generated correspondence point sets are presented. When compared to the previous result given in Figure 6-109, there is no change in the correspondence point generation result for the height level of 101 cm. There only an additional magnified result presented in black box at the bottom right in Figure 6-112.

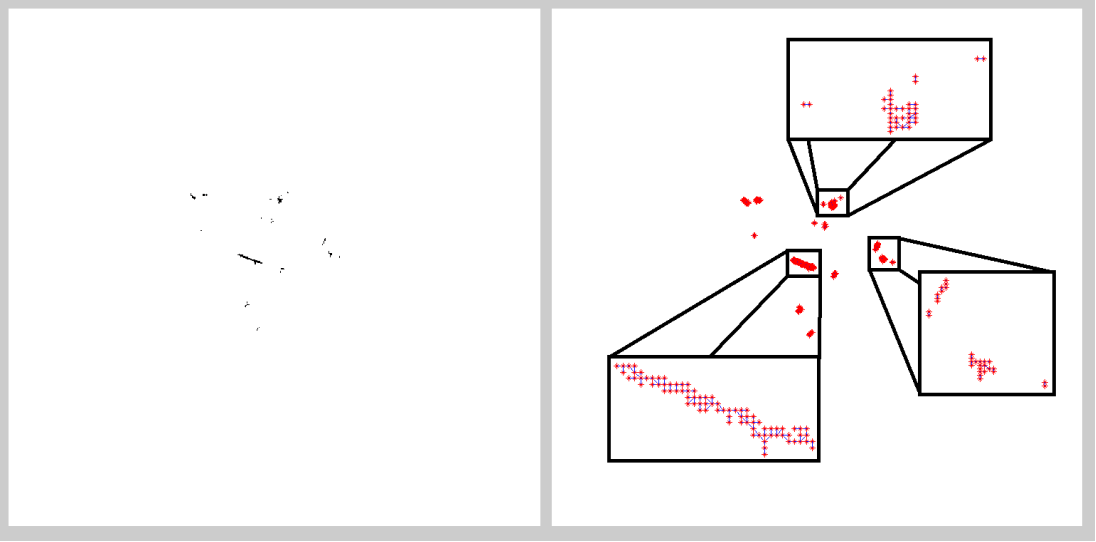


Figure 6-112 Dissimilar regions of the seventh height level (left), 101 cm, and their generated correspondence point sets (right) for Experiment-3

In Figure 6-113 dissimilar regions of the eighth height level, 106 cm, and their generated correspondence point sets are presented. The obtained correspondence point generation result is similar when compared to the previous result given in Figure 6-112. The shape and position of the dissimilar regions does not differ considerably, so it can be said that there is low discontinuity between the given height levels.

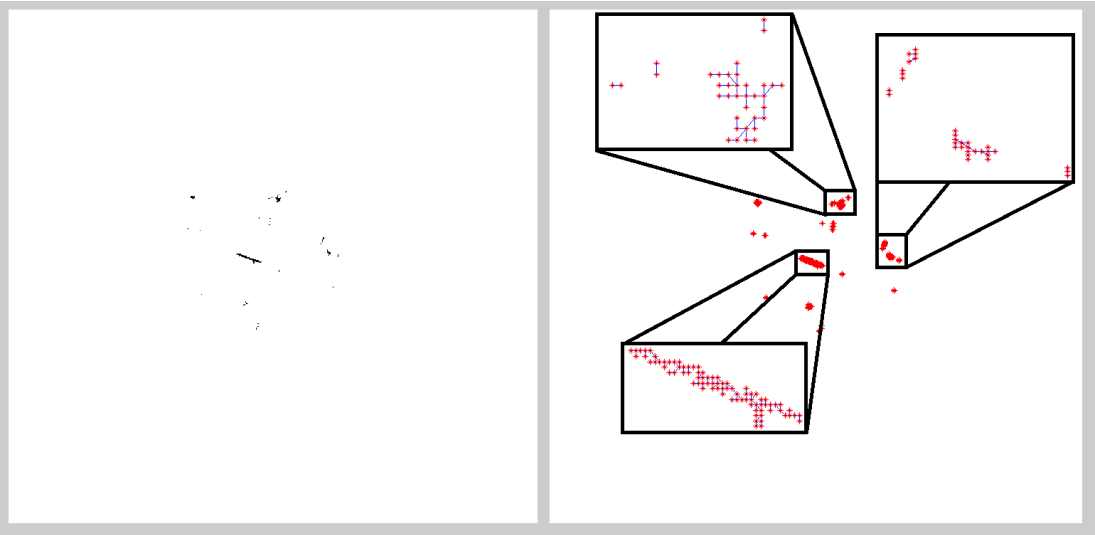


Figure 6-113 Dissimilar regions of the eighth height level (left), 106 cm, and their generated correspondence point sets (right) for Experiment-3

6.5.3.3.Tiling Correspondence Points

In this part, connections between correspondence point sets are established by using the Delaunay Triangulation with determined length limit for edges of the triangles.

In Figure 6-114 tiling result for correspondence point sets of first height level, 53 cm, and second height level, 66 cm, is presented. In the figure drawer, encircled by red, and legs of the L-shaped table, encircled by green, can be seen, which is an important result since it shows that a relatively narrow object such as legs of a table, whose width is 5cm, can be observed by using the proposed method. Additionally, there is approximately no holes on the walls of the environment, which indicates a proper growing result for the environment.

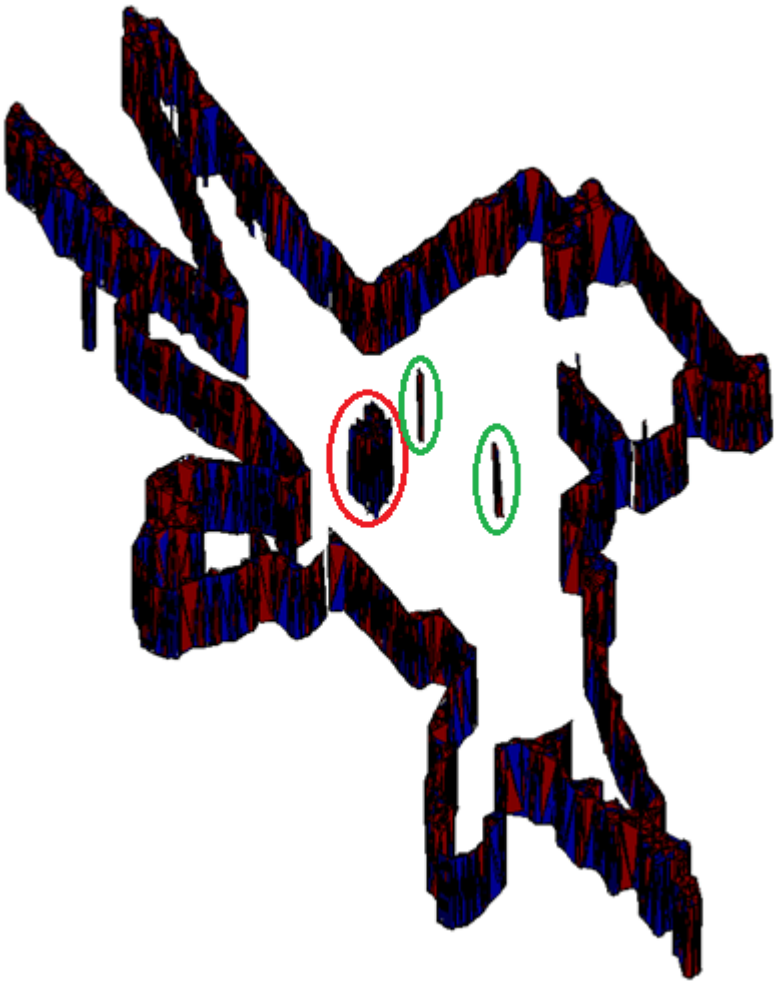


Figure 6-114 Tiling result for correspondence point sets of first height level, 53 cm and second height level, 66 cm, for Experiment-3

In Figure 6-115 tiling result for correspondence point sets of second height level, 66 cm, and third height level, 81 cm, added to the previous result given in Figure 6-114, is presented. Extensions of the drawer and the legs can be seen in the figure. At the right leg there is a little growing error which is emphasized with red circle. Another point is that up to now reconstructed walls of environment have no holes inside them, which shows proper mapping and alignment are achieved.

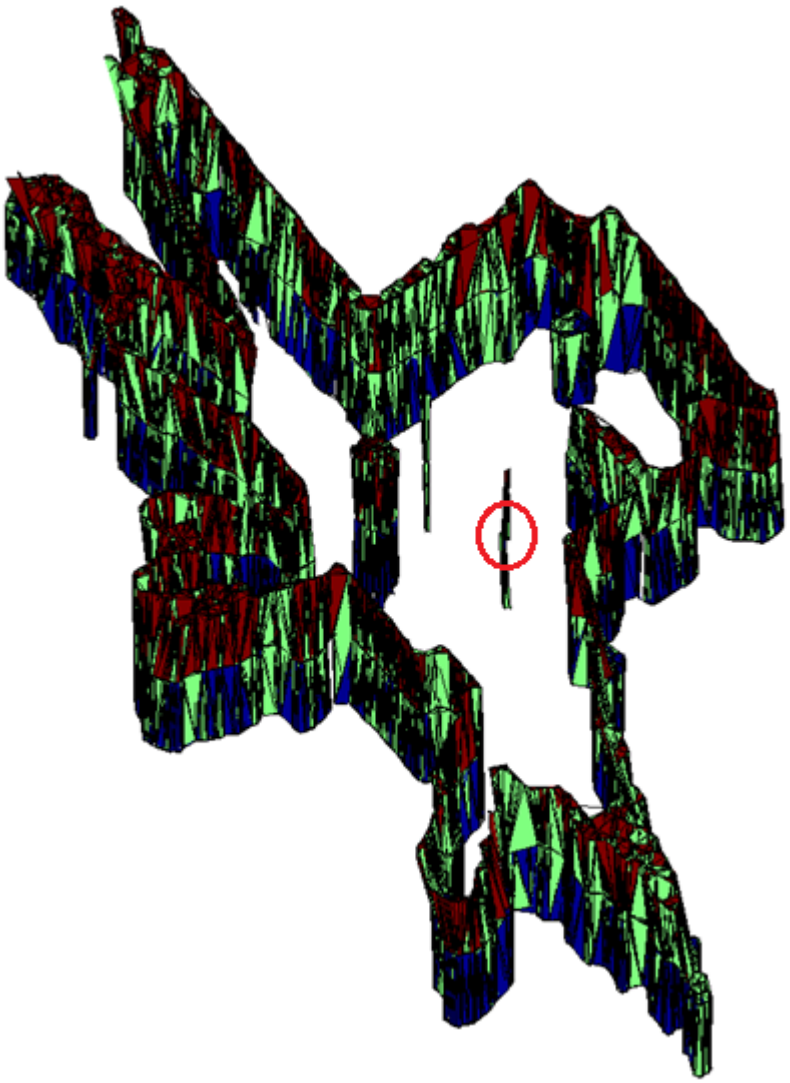


Figure 6-115 Tiling result for correspondence point sets of second height level, 66 cm, and third height level, 81 cm, added to the previous result for Experiment-3

In Figure 6-116 tiling result for correspondence point sets of third height level, 81 cm, and fourth height level, 86 cm, added to the previous result, given in Figure 6-115, is presented. In addition to the previous parts, bottom back part of the L-shaped table, encircled by red, can be observed from the figure which can also be seen in reality on the table in Figure 6-117.

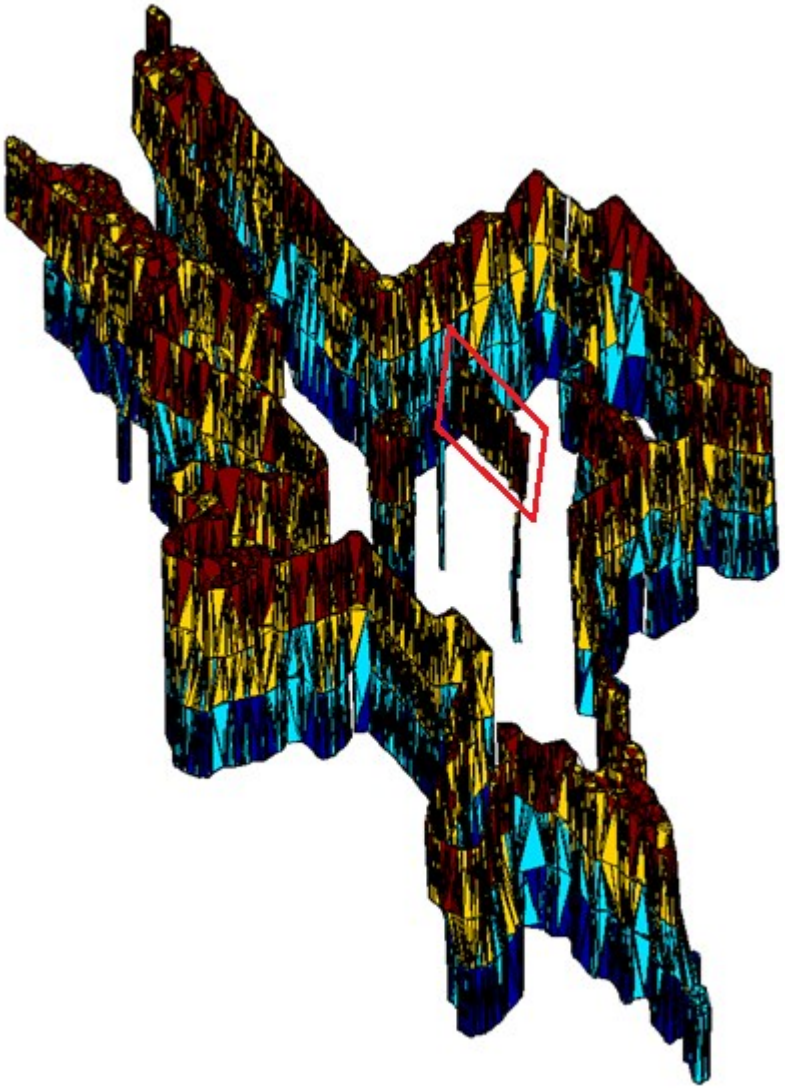


Figure 6-116 Tiling result for correspondence point sets of third height level, 81 cm, and fourth height level, 86 cm, added to the previous result for Experiment-3

Figure 6-117 A view of the L-shaped table without boxes

In Figure 6-118 tiling result for correspondence point sets of fourth and fifth height levels, which is added to the previous result, is presented. This is an important figure in which two black boxes on the L-shaped table can be observed, shown in red circles, but the white one cannot be observed yet. Additionally, the proposed method is unable to detect the surface of the L-shaped table, whose thickness is 3cm, due to limit of the implemented hardware. If the hardware has the higher precision for adjusting the height level, the surface can be seen by the hardware and the proposed method can use the observed data in growing process and in that case surface of the table can be seen. Thickness of the surface of the table can be seen in reality in Figure 6-117. Moreover, a hole can be seen in the wall at the top part of the figure, which is shown in green circle. The reason for this hole to be seen is not an error but there is another empty space at this part of the environment which can be seen in Figure 6-122.

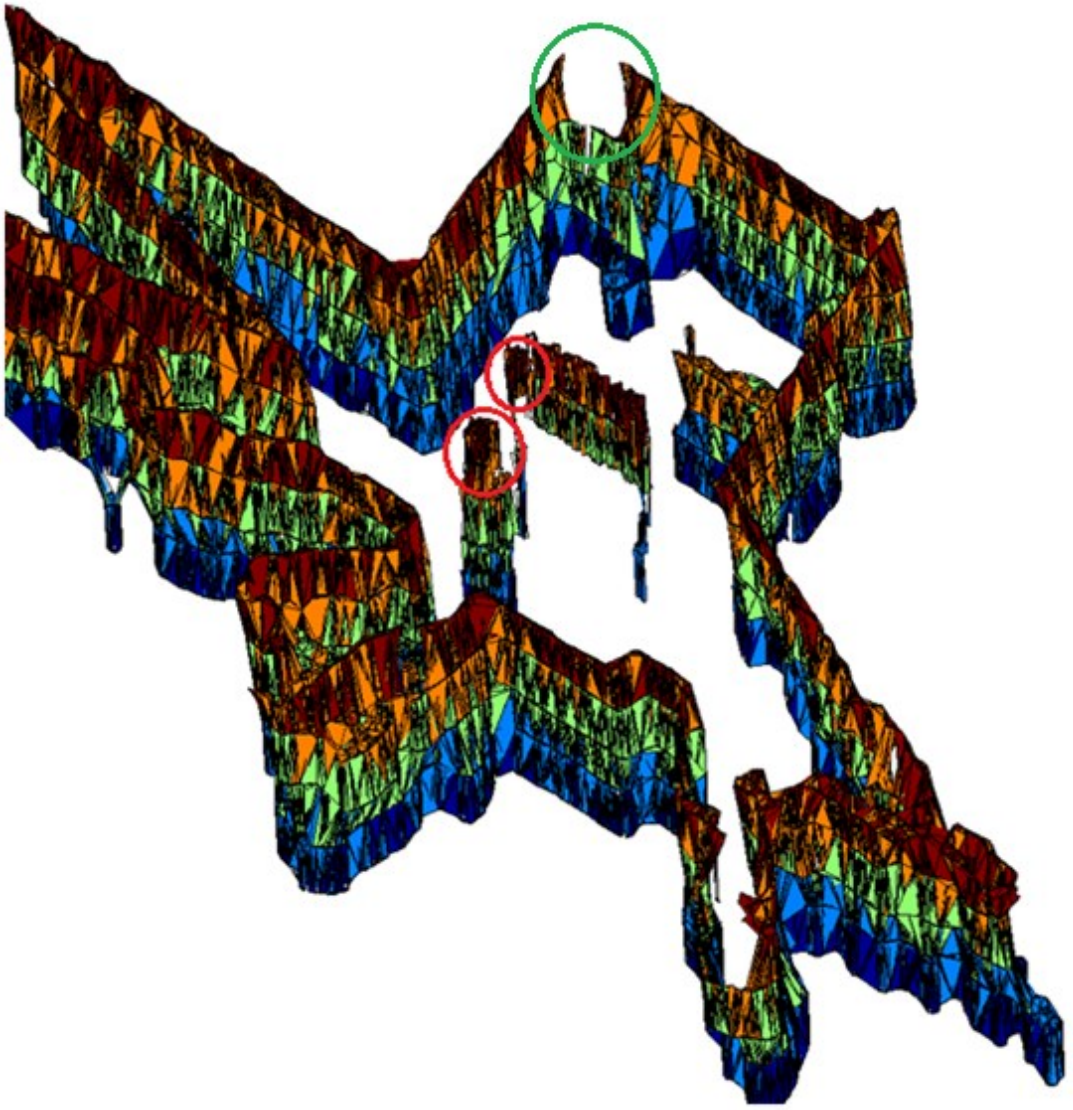


Figure 6-118 Tiling result for correspondence point sets of fourth height level, 86 cm, and fifth height level, 91 cm, added to the previous result for Experiement-3

In Figure 6-119 tiling result for correspondence point sets of fifth height level, 91 cm, and sixth height level, 96 cm, added to the previous result given in Figure 6-118, is presented. As it can be seen, this time white box, shown in green circle, is observed and grown by the proposed method. Moreover, an empty space, emphasized with blue circle in the figure, is reconstructed where a hole is seen previously.

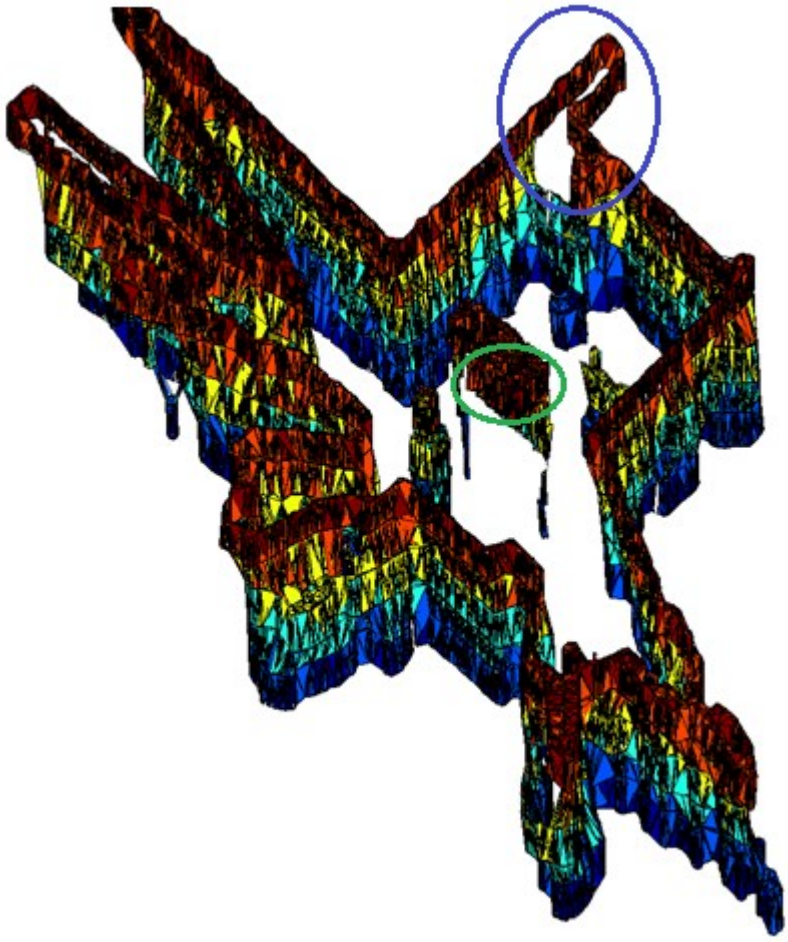


Figure 6-119 Tiling result for correspondence point sets of fifth height level, 91 cm, and sixth height level, 96 cm, added to the previous result for Experiment-3

In Figure 6-120 tiling result for correspondence point sets of sixth height level, 96 cm, and seventh height level, 101 cm, added to the previous result, given in Figure 6-119, is presented. In the given figure, upper back part of the L-shaped table is observed, which is encircled by red, in addition to the previously observed parts.

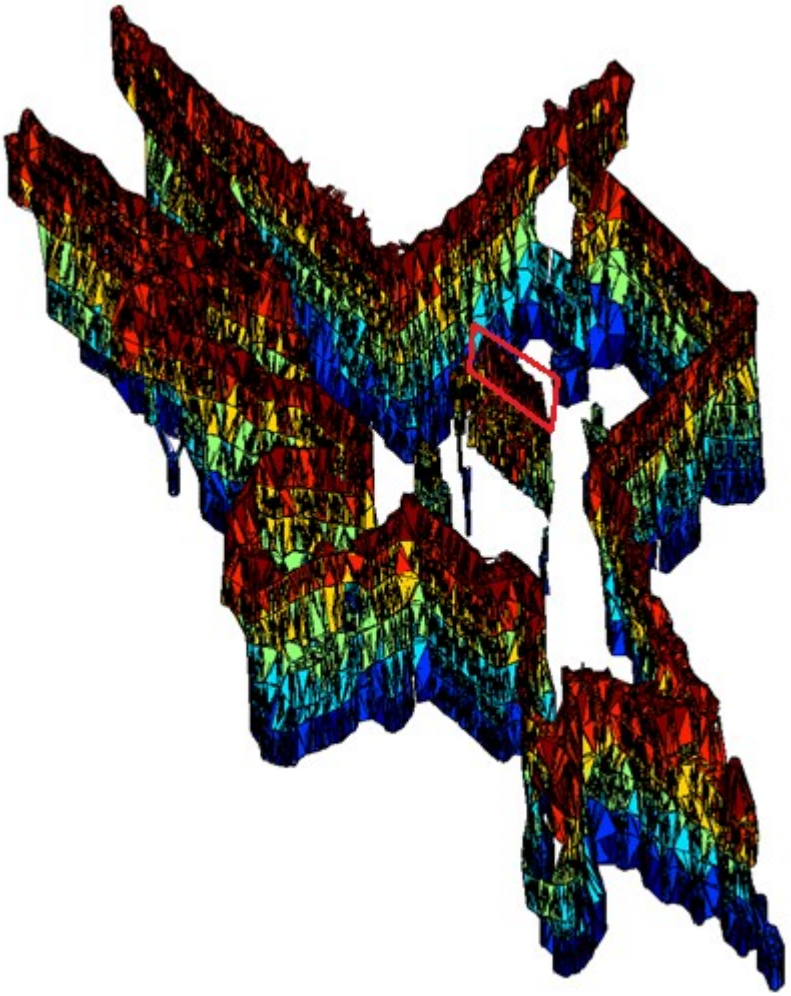


Figure 6-120 Tiling result for correspondence point sets of sixth and seventh height levels added to the previous result

In Figure 6-121 tiling result for correspondence point sets of seventh height level, 101 cm, and eighth height level, 106 cm, added to the previous result, given in Figure 6-120, is presented. This is the final form of the reconstruction result for Experiment-3. It can be seen that, apart from the surface of the table, all the parts seen in Figure 6-70 are reconstructed, which are explained in detail when connection between each height level is added to the current result.

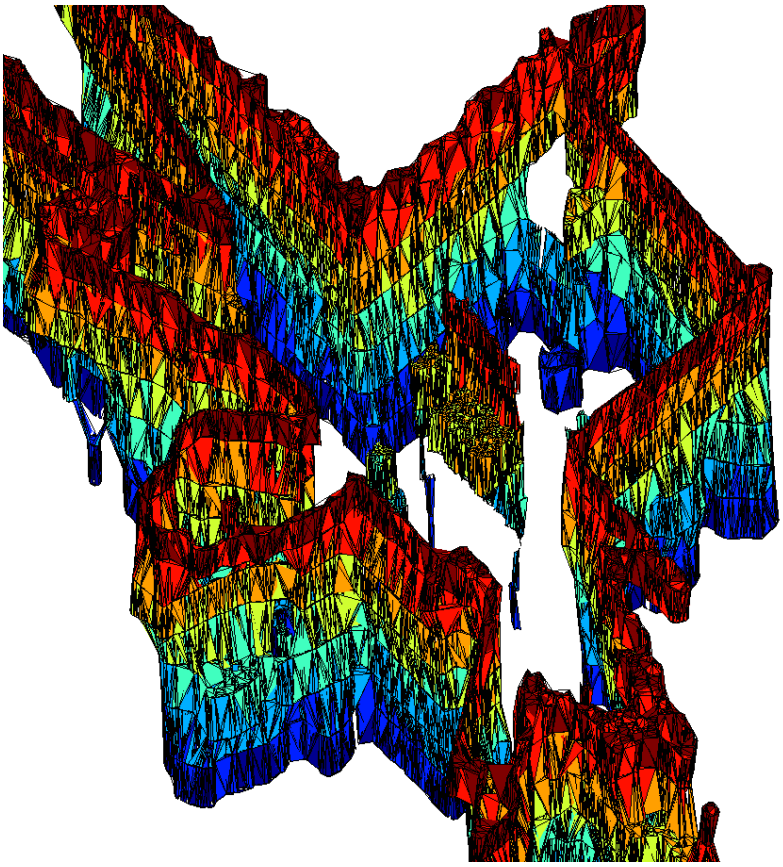


Figure 6-121 Tiling result for correspondence point sets of seventh and eighth height levels added to the previous result

6.5.4. Results

In this part, aim is to evaluate the results of Experiment-3.

In this experiment an L-shaped table with a drawer and three boxes on it is scanned from 8 different height levels and obtained scan data is used in the system block diagram of which is given in Figure 6-1. One aim for conducting this experiment is to increase the number of height levels to scan and see the performance of the proposed method. Moreover, the structure that is focused on provides a proper test environment for the proposed approach, since there are narrow legs, thin surface and many discontinuities on the structure. As mentioned previously legs of the table is observed but the surface cannot be observed by the approach which reveals the limits of the algorithm. One other important result is there are no holes but one which is

eventually grown as an empty space that previously is not seen because of another table blocking the sight and it is pointed out in Figure 6-122.

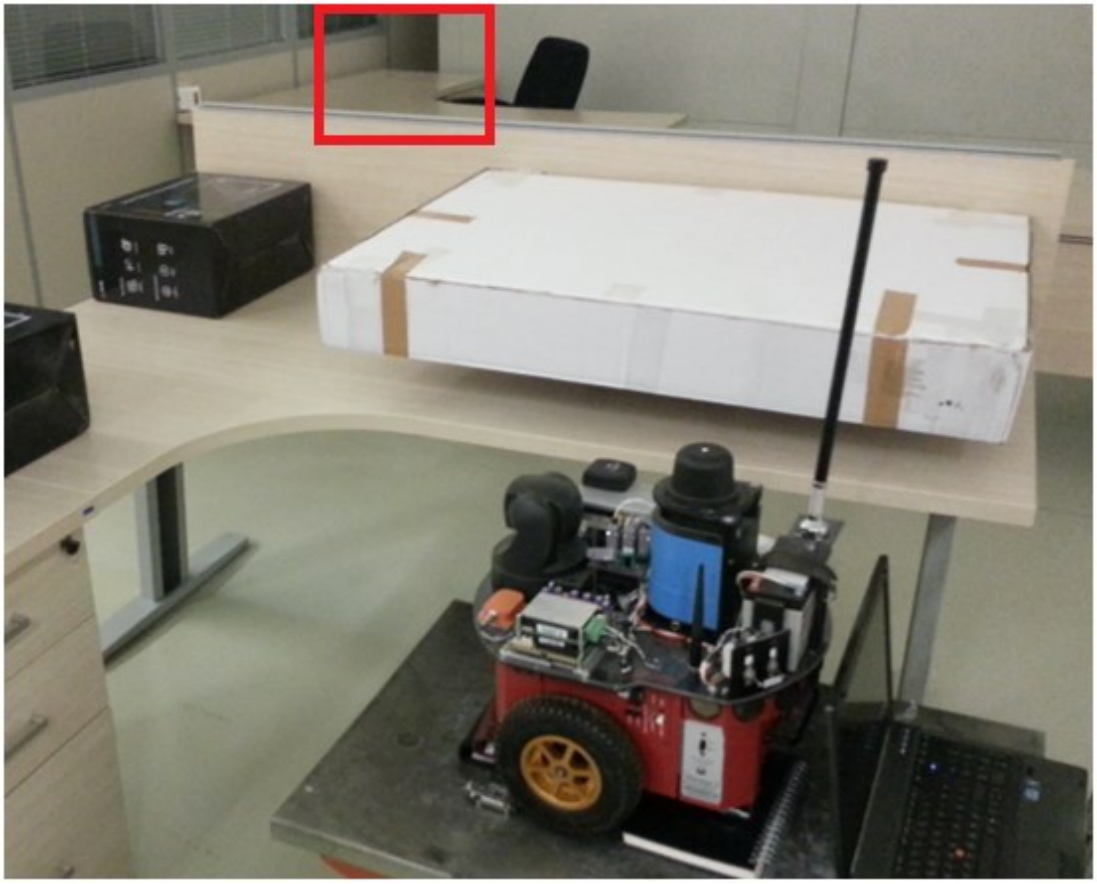


Figure 6-122 Empty space seen in the environment

In addition, reconstructed structure is extracted from the environment in order to compare with the real object, result of which is given in Figure 6-123. Different height levels can be understood with the help of color mappings in which as the height increases colors get hotter (cold (blue) → hot (red)). Also, it is clearly seen in the figure that legs are reconstructed successfully while surface of the table cannot be reconstructed. Missing surface of table is related to the limitation of the hardware that is used in Experiment-3, which does not have a high precision for adjusting the height level.

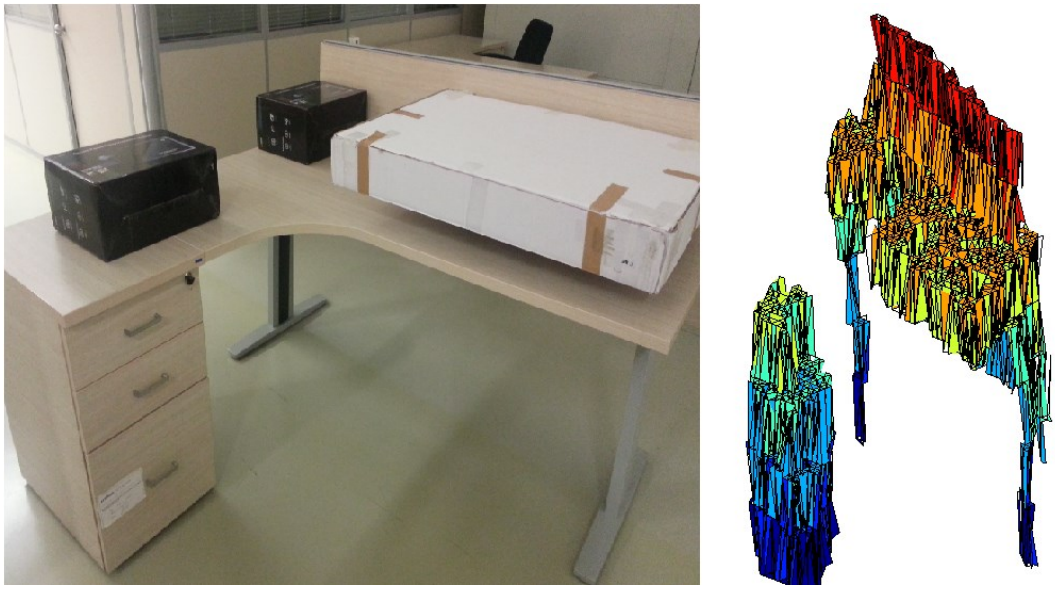


Figure 6-123 Real Object (left) and its reconstructed version (right) for Experiment-3

CHAPTER 7

CONCLUSION & FUTURE WORK

7.1. Conclusion

In the present thesis a new method for growing topological 3D maps of the environment from the 2D SLAM maps obtained at different height levels by heterogeneous vehicles is proposed. Main objective is to improve the utility of vehicles, which have different dynamics, while exploring and growing 3D structure of an indoor area and achieve this goal with a novel approach combines 2D mapping results coming from these vehicles while handling the discontinuities between different height levels in the environment by smooth growing.

Simultaneous Localization and Mapping is the first step for the proposed approach. In this step FastSLAM algorithm with feature based scan matching is applied. FastSLAM consists of particle and Kalman filters where each particle represents a possible state of robot and Kalman filter is used to estimate the states of particles and extracted landmarks. For each particle state changes are estimated by obtaining pose differences between consecutive scans with scan matching and using them in prediction step of the Kalman filter. In order to update the states of both particles and landmarks with respect to measurement data, data association over landmarks are utilized. When a landmark is re-observed, states of particles and landmarks are updated accordingly. In the case of a close loop incremental errors are eliminated and path of a particle is corrected which is also called Full SLAM. Mapping is handled with occupancy grid map approach which divides the environments into grids and assigns probabilities for occurrence of an obstacle in that particular grid.

After obtaining 2D occupancy grid maps from different height levels of the environment, maps are segmented into semantically meaningful regions. This is done by using spatial coherence between grids. Then using these regions key features,

which are lines and corners are extracted. Afterwards, key features are utilized for different purposes one of which is to align the maps by using line matching and Iterative Closest Point method. Secondly, they are used while separating similar and dissimilar regions. Moreover, feedbacks are provided for the vehicles like structures to focus on and height information of the structures at this step. After the separation of regions, correspondence point sets used for connection of regions from different height levels are generated according to similarity relationship of the regions. Eventually, generated correspondence points are tiled, i.e. connected, together, which is actually reconstruction of surfaces of the structures inside the environment, by using a well-known triangulation method called Delaunay Triangulation while obeying length constraints defined according to the system.

After implementing the proposed method in hardware and observe the obtained results, it is seen that critical points for achieving a proper map growing are;

- **Robust localization and mapping**: One of the reasons that most of the errors seen in the growing process. In sensitivity analysis the localization errors are examined, mean and variance of the pose errors are determined. Adjusting particle number is used as a solution to this issue in present thesis. There can be alternative solutions such as extracting more complex features to use as landmarks and obtain better localization results.
- **Precision in adjusting height levels**: In the experiments, 5 cm is the minimum height difference between two consecutive levels that can be adjusted properly by the hardware, so structures narrower than 5 cm is not observed such as surface of the L-shaped table. In order to prevent such undesired situations, vehicles which are more precise at adjusting the height level should be used.
- **Optimized alignment between the maps:** Alignment of the maps may be the most critical part in map growing since its results affect the subsequent parts such as separating similar and dissimilar regions, and generating correspondence points accordingly. Once there is an error in the alignment of the maps, reconstruction result is affected unfavorably.

7.2.Future Work

SLAM approach can be improved at feature extraction side where in this thesis line features are mostly used. In order to extract and use better landmarks and obtain better scan matching results more complex features can be extracted and used. This should minimize the errors seen in the 2D mapping part of the system which affects map growing results considerably.

3D map growing can be achieved by simpler hardware solutions such as one aerial vehicle equipped with laser scanner, IMU and altitude sensor. Different height levels can be explored easily and environment can be reconstructed by using the proposed method.

REFERENCES

[1] A. Nüchter, K. Lingemann, J. Hertzberg, and H. Surmann, "6D SLAM - 3D mapping outdoor environments," in *Journal of Field Robotics*, 2007, vol. 24, no. 8–9, pp. 699–722.

[2] P. Kohlbepp, P. Pozzo, M. Walther, and R. Dillmann, "Sequential 3D-SLAM for mobile action planning," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, 2004, vol. 1, pp. 722–729.

[3] D. M. Cole and P. M. Newman, "Using laser range data for 3D SLAM in outdoor environments," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, 2006, pp. 1556–1563.

[4] A. Nuchter, K. Lingemann, J. Hertzberg, and H. Surmann, "6D SLAM with approximate data association," in *ICAR '05. Proceedings., 12th International Conference on Advanced Robotics, 2005.*, 2005, pp. 242–249.

[5] A. J. B. Trevor, J. G. Rogers, and H. I. Christensen, "Planar surface SLAM with 3D and 2D sensors," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 3041–3048.

[6] J. Weingarten and R. Siegwart, "EKF-based 3D SLAM for structured environment reconstruction," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 3834–3839.

[7] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part II," *IEEE Robot. Autom. Mag.*, vol. 13, no. 3, pp. 108–117, 2006.

[8] H. Liu, X. Wang, and W. Qiang, "Implicit surface reconstruction from 3D scattered points based on variational level set method," in *2008 2nd International Symposium on Systems and Control in Aerospace and Astronautics*, 2008, pp. 1–5.

[9] L. Zhu, M. Lehtomäki, J. Hyyppä, E. Puttonen, A. Krooks, and H. Hyyppä, "Automated 3D Scene Reconstruction from Open Geospatial Data Sources: Airborne Laser Scanning and a 2D Topographic Database," *Remote Sens.*, vol. 7, no. 6, pp. 6710–6740, May 2015.

[10] P. Biber, S. Fleck, and T. Duckett, "3D Modeling of Indoor Environments for a Robotic Security Guard," in *2005 IEEE Computer Society Conference on*

Computer Vision and Pattern Recognition (CVPR'05) - Workshops, vol. 3, pp. 124–124.

[11] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," *ACM SIGGRAPH Computer Graphics*, vol. 26, no. 2. pp. 71–78, 1992.

[12] W. Wei, G. Wang, and H. Chen, "3D reconstruction of a femur shaft using a model and two 2D X-ray images," in *2009 4th International Conference on Computer Science & Education*, 2009, pp. 720–722.

[13] O. Nilsson, D. Breent, and K. Museth, "Surface reconstruction via contour metamorphosis: An eulerian approach with lagrangian particle tracking," *Proc. IEEE Vis. Conf.*, p. 52, 2005.

[14] D. Meyers, S. Skinner, and K. Sloan, "Surfaces from contours," *ACM Trans. Graph.*, vol. 11, no. 3, pp. 228–258, 1992.

[15] D. Wang, O. Hassan, K. Morgan, and N. P. Weatherill, "Efficient surface reconstruction from contours based on two-dimensional Delaunay triangulation," *Int. J. Numer. Methods Eng.*, vol. 65, no. 5, pp. 734–751, 2006.

[16] C. L. Bajaj, E. J. Coyle, and K.-N. Lin, "Arbitrary Topology Shape Reconstruction from Planar Cross Sections," *Graph. Model. Image Process.*, vol. 58, no. 6, pp. 524–543, 1996.

[17] S.-W. Cheng and T. K. Dey, "Improved constructions of Delaunay based contour surfaces," in *Proceedings of the fifth ACM symposium on Solid modeling and applications - SMA '99*, 1999, pp. 322–323.

[18] V. Chatzis and I. Pitas, "Interpolation of 3-D binary images based on morphological skeletonization.," *IEEE Trans. Med. Imaging*, vol. 19, no. 7, pp. 699–710, 2000.

[19] J. J. Leonard and H. F. Durrant-Whyte, "Simultaneous map building and localization for an autonomous mobile robot," in *Proceedings IROS '91:IEEE/RSJ International Workshop on Intelligent Robots and Systems '91*, 1991, pp. 1442–1447.

[20] M. Montemerlo and S. Thrun, "Simultaneous localization and mapping with unknown data association using FastSLAM," in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, 2003, vol. 2, pp. 1985–1991.

[21] D. Hahnel, W. Burgard, D. Fox, and S. Thrun, "An efficient fastslam algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in *Proceedings 2003 IEEE/RSJ International*

Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453), 2003, vol. 1, pp. 206–211.

[22] L. Guo, H. Wang, Q. Meng, and Y. Qiu, "FastSLAM and its data association approaches for mobile robots," in *Proceedings of the IEEE International Conference on Automation and Logistics, ICAL 2007*, 2007, pp. 1952–1957.

[23] A. Ozgur and A. Saranli, "A novel mobile robot navigation method based on combined feature based scan matching and FastSLAM algorithm," Middle East Technical University, 2010.

[24] A. Diosi and L. Kleeman, "Laser scan matching in polar coordinates with application to SLAM," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 3317–3322.

[25] S. Riisgaard, "SLAM for Dummies: A tutorial approach to Simultaneous Localization and Mapping," *citeulike.org*, 2005.

[26] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. Robot. Autom.*, vol. 17, no. 3, pp. 229–241, Jun. 2001.

[27] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot, "Consistency of the EKF-SLAM Algorithm," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 3562–3568.

[28] G. Grisettiyz, C. Stachniss, and W. Burgard, "Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 2432–2437.

[29] G. Grisetti, C. Stachniss, and W. Burgard, "Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters," *IEEE Trans. Robot.*, vol. 23, no. 1, pp. 34–46, Feb. 2007.

[30] M. Montemerlo, "FastSLAM : A Factored Solution to the Simultaneous Localization and Mapping Problem With Unknown Data Association," *Techniques*, no. June, p. 123, 2003.

[31] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, 2000, vol. 1, pp. 476–481.

[32] A. Birk and S. Carpin, "Merging Occupancy Grid Maps From Multiple Robots," *Proc. IEEE*, vol. 94, no. 7, pp. 1384–1397, Jul. 2006.

[33] N. Adluru, L. J. Latecki, M. Sobel, and R. Lakaemper, “Merging maps of multiple robots,” in *2008 19th International Conference on Pattern Recognition*, 2008, pp. 1–4.

[34] K. Wang, S. Jia, Y. Li, X. Li, and B. Guo, “Research on map merging for multi-robotic system based on RTM,” in *2012 IEEE International Conference on Information and Automation*, 2012, pp. 156–161.

[35] S. Pontier, B. Shariat, and D. Vandorpe, “Implicit surface reconstruction from 2D CT scan sections,” in *Proceedings. Computer Graphics International (Cat. No.98EX149)*, 1998, pp. 583–586.

[36] L.-P. Nolte, “Surface Reconstruction of Bone from X-ray Images and Point Distribution Model Incorporating a Novel Method for 2D-3D Correspondence,” in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR’06)*, 2006, vol. 2, pp. 2237–2244.

[37] K. T. Rajamani, M. Styner, R. U. Thoranaghatte, L.-P. Nolte, and M. A. G. Ballester, “Accurate and Robust Reconstruction of a Surface Model of the Proximal Femur From Sparse-Point Data and a Dense-Point Distribution Model for Surgical Navigation,” *IEEE Trans. Biomed. Eng.*, vol. 54, no. 12, pp. 2109–2122, Dec. 2007.

[38] K. K. Sareen, G. K. Knopf, and R. Canas, “Surface reconstruction from sliced point cloud data for designing facial prosthesis,” in *2009 IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH)*, 2009, pp. 6–11.

[39] G. Zheng, “3D volumetric intensity reconsturction from 2D x-ray images using partial least squares regression,” in *2013 IEEE 10th International Symposium on Biomedical Imaging*, 2013, pp. 1268–1271.

[40] V. Nguyen, A. Martinelli, N. Tomatis, and R. Siegwart, “A comparison of line extraction algorithms using 2D laser rangefinder for indoor mobile robotics,” in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 1929–1934.

[41] S. M. LaValle, “Rapidly-Exploring Random Trees: A New Tool for Path Planning,” *In*, vol. 129, pp. 98–11, 1998.

[42] K. Mikolajczyk and C. Schmid, “Performance evaluation of local descriptors.,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, 2005.

[43] F. L. F. Lu and E. E. Milios, “Robot pose estimation in unknown environments by matching 2D range scans,” *Comput. Vis. Pattern Recognition, 1994. Proc. CVPR ’94., 1994 IEEE Comput. Soc. Conf.*, 1994.

[44] T. K. Dey and J. A. Levine, “Delaunay meshing of isosurfaces,” *Vis. Comput.*, vol. 24, no. 6, pp. 411–422, 2008.