BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

NAVIGATION OF MOBILE ROBOTS IN UNKNOWN ENVIRONMENTS USING RANGE MEASUREMENTS

NAVIGACE MOBILNÍCH ROBOTŮ V NEZNÁMÉM PROSTŘEDÍ S VYUŽITÍM MĚŘENÍ VZDÁLENOSTÍ

DOCTORAL THESIS
DOKTORSKÁ PRÁCE

AUTHOR
AUTOR PRÁCE

Ing. ONDŘEJ JEŽ

SUPERVISOR
VEDOUCÍ PRÁCE

doc. Ing. LUDĚK ŽALUD, Ph.D.

BRNO 2008

First of all, I wish to thank Luděk Žalud, my supervisor, for his support during my research and also for his valuable suggestions.

Also, I would like to thank all members of the L.T.R. team, who have helped me in various ways, especially František Šolc who introduced me to the theory of kinematics and dynamics of robots and also Lukáš Kopečný and Ondřej Lebeda who both helped me with the hardware part of the research. Special thanks to Tomáš Neužil for carrying out the experiments with me.

My thanks are also due to Jan Vodička who has built the prototype of the scanning platform.

LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami:

1. Pan/paní

Jméno a příjmení:	Ing. Ondřej Jež
Bytem:	Jiráskova 49, 60200, Brno - Veveří
Narozen/a (datum a místo):	20.10.1979, Brno

(dále jen "autor")

a

2. Vysoké učení technické v Brně

Fakulta elektrotechniky a komunikačních technologií
se sídlem Údolní 244/53, 60200 Brno 2
jejímž jménem jedná na základě písemného pověření děkanem fakulty:
prof. RNDr. Vladimír Aubrecht, CSc.

(dále jen "nabyvatel")

Článek 1
Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):

   ☑ disertační práce
   □ diplomová práce
   □ bakalářská práce

   jiná práce, jejíž druh je specifikován jako ....

(dále jen VŠKP nebo dílo)

Název VŠKP:	Navigation of Mobile Robots in Unknown Environments Using Range Measurements
Vedoucí/školitel VŠKP:	doc. Ing. Luděk Žalud, Ph.D.
Ústav:	Ústav automatizace a měřicí techniky
Datum obhajoby VŠKP:	....

VŠKP odevzdal autor nabyvateli v:

☑ tištěné formě	- počet exemplářů 1
☑ elektronické formě	- počet exemplářů 1

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.

4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2
Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizovaní výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti

   ☑ ihned po uzavření této smlouvy
   ☐ 1 rok po uzavření této smlouvy
   ☐ 3 roky po uzavření této smlouvy
   ☐ 5 let po uzavření této smlouvy
   ☐ 10 let po uzavření této smlouvy

   (z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3
Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísni a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: ....

....	....
Nabyvatel	Autor

Abstract

The ability of a robot to navigate itself in the environment is a crucial step towards its autonomy. Navigation as a subtask of the development of autonomous robots is the subject of this thesis, focusing on the development of a method for simultaneous localization an mapping (SLAM) of mobile robots in six degrees of freedom (DOF).

As a part of this research, a platform for 3D range data acquisition based on a continuously inclined laser rangefinder was developed. This platform is presented, evaluating the measurements and also presenting the robotic equipment on which the platform can be fitted.

The localization and mapping task is equal to the registration of multiple 3D images into a common frame of reference. For this purpose, a method based on the Iterative Closest Point (ICP) algorithm was developed. First, the originally implemented SLAM method is presented, focusing on the time-wise performance and the registration quality issues introduced by the implemented algorithms. In order to accelerate and improve the quality of the time-demanding 6DOF image registration, an extended method was developed. The major extension is the introduction of a factorized registration, extracting 2D representations of vertical objects called leveled maps from the 3D point sets, ensuring these representations are 3DOF invariant. The extracted representations are registered in 3DOF using ICP algorithm, allowing pre-alignment of the 3D data for the subsequent robust 6DOF ICP based registration. The extended method is presented, showing all important modifications to the original method.

The developed registration method was evaluated using real 3D data acquired in different indoor environments, examining the benefits of the factorization and other extensions as well as the performance of the original ICP based method. The factorization gives promising results compared to a single phase 6DOF registration in vertically structured environments. Also, the disadvantages of the method are discussed, proposing possible solutions. Finally, the future prospects of the research are presented.

Contents

List of Figures 5

List of Tables 8

List of Abbreviations 9

1 Introduction 10

2 Navigation of Mobile Robots 12

2.1 Navigation's Key Elements: Localization, Mapping and Path Planning 13
2.2 Navigation Methods Related to Sensing Approaches 14
2.2.1 Dead-reckoning 14
2.2.2 Dead-reckoning - Odometry Implementation 15
2.2.3 Dead-reckoning - Inertial Navigation 17
2.2.4 Guide-path Following 17
2.2.5 Tactile and Proximity Sensing 18
2.2.6 Range Measuring 18
2.3 Systems of coordinates and elementary rotation angles 18
2.4 SLAM's Theoretical Perspectives 21
2.4.1 Global frame of reference 21
2.4.2 Preprocessing of the range data 22
2.4.3 Semantic analysis option 23
2.4.4 Data registration 24
2.4.5 Post-processing issues 27
2.5 Path planning 28
2.6 Current State of Range Sensing Methods and Technology 28

3 Platform development 31

3.1 State of technology in 3D range measurement systems 31

3.2 3D range measuring system development ... 34

3.2.1 SICK LMS 200 sensor ... 35

3.2.2 Validation of measurement's reliability ... 39

3.2.3 Inclining mechanism ... 44

3.3 Robot platforms development and integration of sensors ... 49

3.3.1 Focus in the Laboratory of Telepresence and Robotics ... 49

3.3.2 Hermes: omnidirectional robotic platform ... 50

3.3.3 U.T.A.R: Universal Telepresence and Autonomous Robot ... 52

3.3.4 Orpheus: the rescue and reconnaissance remote-operated robot ... 53

4 Original Implementation of the 6DOF SLAM Algorithm ... 56

4.1 Structure of the Initial SLAM Method ... 58

4.2 Iterative Closest Point Algorithm ... 60

4.3 Implementing the Closest Point Search ... 63

4.4 Decomposing the Matrix H ... 67

5 Extending the Original Implementation: Two Stage Leveled Map Accelerated 6DOF SLAM ... 69

5.1 Identifying the Bottlenecks of the Original Method ... 69

5.2 Quality of the Registration and Mapping ... 70

5.3 Structure of the Extended Method ... 73

5.4 Preprocessing the Range Data ... 74

5.5 Data Reduction – Saving the Effort ... 77

5.6 3DOF Invariance: Leveled Structure Extraction ... 77

5.7 Factorizing the Task: Leveled Data Sets Registration in 3DOF ... 81

5.8 Registration of the Pre-aligned 3D Data Set in 6DOF ... 82

5.9 Modifying the Implemented ICP Algorithm: Unique Matching of Model Points ... 82

6 Experiments ... 86

6.1 3D Range Scanning Platform Evaluation ... 87

6.2 Experiment #1: Close to Artificial Conditions: Ideal Match ... 90

6.3 Experiment #2: Vertically Structured Environment, Simple Floor Projection ... 98

6.4 Experiment #3: Vertically Structured Environment with Scattered Objects 107

6.5 Experiment #4: Vertically Structured Environment, Scattered Objects, Multi-Floor Vertical Structuring 112

7 Conclusions and Future Prospects 116

References 121

List of Author's Publications 125

List of Figures

Figure 1: Orpheus robot’s platform developed in our department implementing differential steering (Source: Zalud et al. [37]) 16

Figure 2: Representations of points in three-dimensional space: Cartesian system of coordinates (left), spherical system of coordinates (middle), cylindrical system of coordinates (right); Source: Wikipedia (en.wikipedia.org) 20

Figure 3: Demonstration of the flight dynamics angles on a wheeled robot: pitch (green), roll (blue), and yaw (red)............ 21

Figure 4: The principle of the time-of-flight operation, scanning horizontal angle range; Source: SICK Technical Description [28]............ 36

Figure 5: Dependence of standard deviation of the range measuement on the distance from the obstacle for objects of different reflectivity............ 41

Figure 6: Dependence of standard deviation of the range measurement on the distance from the obstacle for objects of different colors............ 42

Figure 7: Histogram of range measurement, measured distance approx. 9m, target object - white paper 43

Figure 8: The options to incline the scanner: pitching inclination (left); rolling inclination (right)............ 45

Figure 9: Hermes robot equipped with the SICK laser rangefinder and a servo-driven inclining mechanism; a) detail of the inclining mechanism; b) the robot 51

Figure 10: The U.T.A.R. robot with continuous inclining device............ 52

Figure 11: Screenshots from the ARGOS system, a) with the point clouds directly fused with the camera image (other virtual displays are disabled) and b) with 3D scans included in the virtual HUD displays............ 54

Figure 12: A block diagram of the Simultaneous Localization and Mapping process in six degrees of freedom............ 59

Figure 13: A kd-tree built in a three-dimensional space (splitting planes in the tree hierarchy: first level - root node – blue; second level nodes – red; third level nodes - leaves – green) 66

Figure 14: Proposed block diagram of the navigation system............ 74

Figure 15: Estimating the absolute inclination angle for data points: linear regression applied to the detection of inclination construction (left: whole characteristic; right: detail) 76

Figure 16: Vertical object detection: slope, height and gap criteria 80

Figure 17: Leveled map building: the algorithm structure 81

Figure 18: Different point density introduced error in finding the corresponding points 84

Figure 19: The cumulative pitch angle error as a function of the absolute pitch inclination; angular inclining velocity at 25.6 °/s 88

Figure 20: Demonstrating the details in the 3D image: a 3D scan of a woman (left) and a man (right); discontinuity area due to dynamics of the environment (movement of the person) is highlighted by a red circle... 89

Figure 21: Registration of identical but rotated and translated point clouds from office environment: original sets (left); registered solution (right).................................. 92

Figure 22: Leveled maps: extracted maps (left) and registered solution (right) 93

Figure 23: Duration of the Leveled maps registration as a function of the applied transformation's rotation angle.. 94

Figure 24: Number of ICP iterations in the Leveled maps registration as a function of the applied transformation's rotation angle 95

Figure 25: Duration of the 6DOF ICP registration as a function of the applied transformation's rotation angle.. 97

Figure 26: Number of ICP iterations in the 6DOF ICP registration as a function of the applied transformation's rotation angle 97

Figure 27: Unregistered model and data sets in the experiment #2.................................... 100

Figure 28: Registered solution: uniqueness of model points use in closest point queries not enforced (left) and enforced (right).. 101

Figure 29: Unregistered (left) and registered (right, enforcing uniqueness of model points use) leveled model and data sets in the experiment #2 101

Figure 30: Registering a problematic 3D pair: other objects are in the view of the data set; unregistered sets (left) and registered (right) 102

Figure 31: Data and model leveled maps to be matched (problematic solution due to variable point density and appearance of new objects in data set)......................... 103

Figure 32: Registered leveled maps in 3DOF: using a single pass method (left) and using two passes and updated centroids (right)... 103

Figure 33: Registration of data sets acquired in office environment: scale of rotation approximately 20° (left: unregistered; right: registered) 109

Figure 34: Leveled model and data sets from the office environment (scale of rotation 20°; left: unregistered; right: registered)... 109

Figure 35: Registration from office environment: scale of rotation approximately 45° (left: unregistered; right: registered) 110

Figure 36: Leveled model and data sets from the office environment (scale of rotation 45°; left: unregistered; right: registered)...................... 110

Figure 37: Unregistered scans taken in a laboratory environment with large amount of scattered objects...................... 111

Figure 38: Registered scans taken in a laboratory environment with large amount of scattered objects...................... 112

Figure 39: Leveled model and data sets from the laboratory environment (scale of rotation 15°; top: unregistered; bottom: registered) 112

Figure 40: Leveled maps of multi-storey data and model sets: unregistered (left) and registered (right)...................... 113

Figure 41: Multi-storey unregistered data and model sets: transformation approx. 15° rotation around z-axis direction 114

Figure 42: Multi-storey registered data and model sets; the object contained only in data is marked with a red circle...................... 114

Figure 43: Second experiment in the multi-storey environment: unregistered model and data (left) and registered solution (right) 115

List of Tables

Table 1: Measurement ranges and the resolution on different regimes 37

Table 2: Standard deviations of the measurements for each color and shade of gray depending on the distance from the scanner 41

Table 3: Universal Telepresence and Autonomous robot’s parameters 53

Table 4: Point cloud sets selected for the experiment #2........ 106

Table 5: Impact on the 6DOF SLAM when accelerating the method using the leveled map factorization 107

Table 6: Results of the pairwise registration in experiment #2........ 107

List of Abbreviations

DOF	Degree of Freedom
SLAM	Simultaneous Localization and Mapping
ICP	Iterative Closest Point Algorithm
3D	Three dimensional
SVD	Singular Value Decomposition

1 Introduction

The history of navigation

The main purpose of this thesis is to develop a strategy towards navigation of mobile robots in unknown environments. The concept of navigation is as old as the human civilization; it is connected to human migration and the development of the means of transportation. In order to navigate themselves through miles of open ocean, the Polynesian sailors had to memorize extensive facts such as the motion of stars, weather influences, directions of swells etc. European explorers were initially using dead reckoning and slowly developed celestial navigation, using passive landmarks (planets and stars) to find ship's current position. Today's navigation methods used in transportation are mainly dependent on artificial landmarks (such as beacons) or even active systems of moving landmarks (satellites). The development of precise mechanical, optical and MEMS technology gyros and accelerometers has also allowed high dead reckoning precision in systems where independence of the navigation on the outside world is critical.

Motivation of the task

The motivation for the development of autonomous navigation is that autonomous navigation is an essential step towards the autonomy of a robot. Only after the robot can localize itself in the environment and sense its surroundings in an organized way, it could perform various tasks. Such tasks could range from utility of households, transportation of people and goods, exploration of unknown environments, automatic civil construction etc.

Navigation of robots and the autonomy

The development of navigation methods for mobile robots started in a similar way as of the human navigation: at first, robots used dead reckoning for pose estimation in known environment; further development of the robotic navigation went hand in hand with the developments in sensing technologies. Today, mobile robots are equipped with a variety of sensing devices such as cameras, range sensors (ultrasound, optical, microwave), tactile sensors, gyros etc. which allow the solution of the task of localization and exploration. The state of technology makes it possible for us to advance further with the theory and strategy to navigate the robot in the environment. Current sensing devices are able to explore the surroundings of robots in such a precision and detail that the possibilities of using this information to navigate the robot are so far unused. In the current scientific discourse in the field of mobile robot's autonomy and navigation, the use of three dimensional environment perception is slowly breaking through. This is the area on which the thesis focuses, trying to bring a new strategy and make use of new technological potential.

Research focus

The research documented in this thesis intended to cover some of the major challenges that are yet to be approached or further investigated in the navigation methods for mobile robots. The focus is one of the most challenging ways of navigating mobile robots in unknown environments: using spatial 3D range measurements. The existing methods for localization and map building are very computing-power demanding especially due to the high data flow and difficult optimization tasks. Also the quality issues are often to be resolved when working with real data. The work summarized in this thesis was aiming to innovate the state of the art in both aspects.

2 Navigation of Mobile Robots

Navigation typology, technology determination

Differentiation of navigation methods is quite a complex issue and it has to be perceived from more points of view. One of the common ways to split the methods is to concentrate on the technology through which is the environment being explored: the measuring method from which our pose is estimated. This typology is very common especially due to the fact that the environment-exploring technologies are often very different in the impact they deliver for the navigation task [11][8]. Some measuring technologies allow only estimating the relative position change of the vehicle, not giving any information about the environment. Other sensing technologies give position updates requiring active landmarks or beacons, allowing operation only in known environments and provide the same information as previous methods. Other sensing technologies provide proximity information about the environment and thus allow not only estimation of the robot's pose, but also building of complex maps of the environment and autonomous path planning of the vehicle.

Chapter structure

This chapter will first focus on the key concepts of navigation. Localization, mapping and path planning will be defined as these represent elementary (though sometimes inseparable) procedures in the navigation process. This will be followed by a typology of the different navigation approaches with regards to the sensing technologies. In the meantime, the problem of self-containment will also be investigated, trying to point out whether the systems are self-contained or if they require external devices to operate in known or unknown environment.

Then the SLAM concept - Simultaneous Localization and Mapping will be explained, followed by the analysis of the different parts of the navigation process in more detail providing the basic theoretical basis for further analysis: defining the global frame of reference, looking and data preprocessing, registration, map enhancement/building, semantic analysis of the data and finally the path planning.

2.1 Navigation’s Key Elements: Localization, Mapping and Path Planning

Navigation subtasks

The task of navigation in unknown environments implies three basic subtasks: *localization*, *mapping* and *path planning* [11]. These three subtasks correspond to the fact that we have to obtain information about the environment, find our position related to some frame of reference in the environment, and decide where we should and where we are able to move and how to do it. This is a very complicated process consisting of data acquisition and preprocessing and various optimization tasks such as data registration, occlusion detection, error diffusion and optimal path planning. In our case the environment is apart from global characteristics (such as gravity and other physical laws, possible presence of objects of known characteristics etc.) unknown. When designing a navigation system which operates in unknown environments, the first two subtasks – localization and mapping – can be treated as simultaneous processes. Since we have no prior map in which we are determining our position, in every moment we get some information about the environment, we update our maps and also our current position on this updated map. Thus the localization and map building form one procedure called simultaneous localization and mapping. In case of three dimensional environments, this task will be solved in six degrees of freedom (three translations and three rotations). Although the position of a mobile robot is expressed in six degrees of freedom, its possible trajectory is often limited by the terrain and ground constrains, thus allowing us only certain movements and achievable locations. This is what makes the task of path planning of wheeled vehicles more difficult than e.g. path of planning of an aircraft. On the other hand the proximity of an observable environment allows the selflocalization with respect to the environment structures [8].

SLAM and image registration

When proceeding with SLAM in unknown environments, we have to measure new range information about the environment once in a while during robot’s movement in order to keep track of the current position and to expand the current map. The transformation of this new set of information (image, range data) into the original coordinate system (in which the built map is referenced) is called image registration and is serving for both purposes of SLAM (the new data are compared to the reference map and consequently integrate in it).

2.2 Navigation Methods Related to Sensing Approaches

Chapter structure

In this chapter the traditional navigation methods typology differentiating the methods according to the measuring principle will be presented. Although it may seem as if the sensing and navigation procedures should be treated separately, the type of data we measure to certain extent determines its potential use and also the method's constraints. A typical constraint of the method is that it is only limited to localization and does not allow mapping since some measurements are only taken to specify the relative positioning to the original pose. First, we will look at the most basic (though often not simple) method – dead reckoning, followed by an introduction to a two most common implementations of the dead reckoning method – odometry based navigation and inertial navigation. This will be followed by a brief note on a relatively outdated method – guide-path following. Then tactile and proximity sensing will be presented, which are connected to the collision avoidance problem. Finally range measuring ultrasound, optical and radar methods will be discussed, followed by an introduction into Global Positioning System.

2.2.1 Dead-reckoning

Main principle – the process

Dead reckoning is a navigation method which is in fact a process: we are updating a robot's current location using the vehicle's original known location and the measurements of its speed, elapsed time and course. If using this method in planar navigation, the speed is the horizontal component of the velocity vector at the given moment and course is the direction in which the vehicle is traveling. This implies that this method is usable purely for localization; we cannot build maps using this method nor plan trajectory (apart from return trajectory to the original location). Dead reckoning is thus a navigation method which does not use observations of the outside world. This is most likely where the name of the method comes from: "dead" has often been though to be an abbreviation of "deduced reckoning" but the meaning of the word may also be in the inability to observe the environment [8].

Air and marine navigation

Dead reckoning combined with stars observation was historically the first navigation method used in aircraft and marine navigation. In marine navigation, the speed of swells and wind has not been traditionally taken into account when determining the dead reckoning plot, resulting in error integration. In air navigation, the navigator traditionally takes the wind speed into account, correcting the estimated pose using so called wind triangle [11].

2.2.2 Dead-reckoning - Odometry Implementation

Odometry in robotics

The most common way of implementing the dead reckoning method in mobile robotics is the use of the odometry of the vehicle. Odometry is a special type of dead reckoning navigation method which uses the rotation angles of wheels to estimate the total displacement of a vehicle over a period of time. Odometry is therefore only applicable to a vehicle equipped with wheels or tracks. The rotation of wheels over time is often measured using optical encoders which are coupled with the motors or the vehicle's wheels axis. Alternatively other sensors could be implemented such as induction and capacitive sensors.

Determining the robot's pose

The transformation of the total wheel rotation to the overall displacement and rotation of the robot is mainly dependent on the geometrical properties of the robot's platform. Basic types of platforms where one can use odometry are differential (and its subtype skid-steered) platform, Ackerman platform and omni directional platform. The position of the robot is generally determined by using the displacement and the heading of the robot; in case of forward motion on can use the equations (1) in case of movement of a robot on a plain [11]. The displacement of the robot is calculated based on the model of the robot's platform. In the following paragraphs, the differential steering platform will be further examined since it is the main platform implemented in the robots developed in our department.

$$\begin{aligned} x_{n+1} &= x_n + D.\sin(\theta) \\ y_{n+1} &= y_n + D.\cos(\theta) \end{aligned} \tag{1}$$

$$D = \frac{D_L + D_R}{2} \tag{2}$$

$$\theta = \frac{D_L - D_R}{d} \tag{3}$$

Differential platform and skid steering subtype

The main principle of a differential platform is that each side of the platform has one or more independent drives. It can have one or more wheels on each side and these are often mechanically or electrically linked, or it could have tracks on both sides. The wheels or the tracks cannot change its' orientation; it is always aligned with the robot's platform. If there is more than one wheel or there are tracks on each side, the platform is considered to be skid-steered. Figure 1 shows the model of wheeled platform together with the propelling units. This model of a platform was actually implemented in the Orpheus Robotic System developed at our workplace [37]. The total displacement of the robot is given in equation (2), where the D_L and D_R are the displacements of the left and right side [11].

Heading of the robot; errors

Heading sensors are often being implemented in the vehicles, allowing a more precise determination of the current heading position. If this is the case, the heading is determined from these measurements. In some cases, heading information is not available and it has to be computed from the displacement of either tracks or wheels. This can be done using equation (3) [11]. The displacements D_L and D_R can be easily determined e.g. from the measured number of detected edges on the output of optical encoders attached to the wheels' axis. The denominator in the equation d is the displacement between the effective points of contact of wheels or tracks, while this constant is in fact a significant source of errors: in a simple differential platform (a wheel on each side of the robot), the effective point of contact is not always in the center of the tire and is actually changing with the wear of the tire's thread and with changing friction conditions on the surface. Sometimes the errors introduced by the variable friction of the surface are symmetrical, bumps and uneven areas are though introducing errors in one particular displacement D_L or D_R. For the skid-steered differential platform, the skid is a natural way of changing the direction of the vehicle and therefore the determination of the heading is much more imprecise since the effective contact point is changing in two degrees of freedom (sideways and forward – backwards). This uncertainty implicates that this type of navigation method is often coupled with additional methods such as inertial navigation or at least a heading gyro – known as gyrodometry, GPS or a range measurement aided SLAM method.

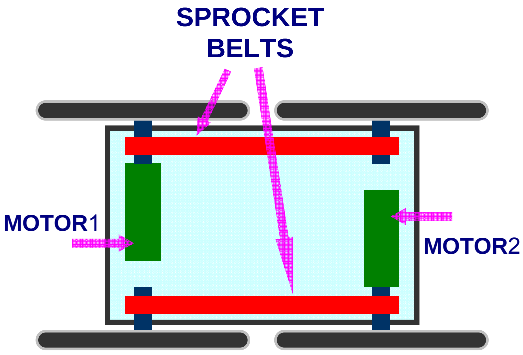


Figure 1: Orpheus robot's platform developed in our department implementing differential steering (Source: Zalud et al. [37])

2.2.3 Dead-reckoning - Inertial Navigation

Main principle

Inertial navigation could be considered as a subtype of dead reckoning since the method does not make use of any external measurements (such as range etc.). It therefore augments the simple odometry based dead reckoning algorithms. The technology was first used in an aircraft and missile development because both have to operate completely independently on measurements of the outer environment. The method was also further applied in submarines [11]. The main principle lies in the continuous measurements of accelerations in three perpendicular directions and three rotation angles around, which are further used to calculate the pose change in six degrees of freedom. Usually a gyroscopically stabilized platform is being used to keep the accelerometers in a constant position in terms of the remaining three degrees of freedom – rotations.

For both inertial navigation and dead-reckoning methods these do not provide measurements of the external environment but only localize the robot in the global frame of reference. Map building using dead reckoning and inertial navigation is not possible.

2.2.4 Guide-path Following

Compexity

This method for navigating of mobile platforms is rather simple compared to the other methods described in this section. The use is limited to static environments where the movement of the robot is limited by the preset guidelines and therefore the environment has to be adapted for the robot's operation.

Implementations

Different sensors (e.g. optical, induction etc.) are being implemented in the robot's platform in order to observe the guide-line which is to be followed. This line could be a wire, optical stripe, magnetic tape or in some cases heat is being used as a signal to be followed [11]. Common applications are e.g. automated warehouse operations, nuclear waste disposal sites etc. The guide-path following method is mainly intended for localization purposes, though in theory the robot could follow an unknown guide-path explore unknown though artificial environments. One can imagine this situation in an unknown mine when following unmapped tracks.

2.2.5 Tactile and Proximity Sensing

Collision avoidance

Instead of implementing a complex navigation algorithm, tactile and proximity measurements serve mainly for a last-resort collision avoidance strategy [11]. Close surroundings of the robot are sensed using a large variety of sensors: contact closures, photoelectric, magnetic, piezoelectric, capacitive, ultrasonic and other. Some of these are detecting the collisions by detecting a direct physical contact with the environment, other are using proximity non-contact measurements to determine an upcoming collision.

Lower level of implementation

This method is being used in both unknown and known environments while the dynamics of the environment determines the unexpected collision risk in known environments. It also influences the decision about implementing the tactile and proximity sensors to a certain extent. The collision avoidance algorithms which use data measured from the tactile and proximity sensors are usually decision based algorithms, while modeling of the surrounding environment is often not implemented and would result in rather incomplete maps. In fact this subtype of collision avoidance algorithms is implemented at a lower level of the navigation algorithms, supporting the localization and mapping methods as a security element. This group of navigation resources will therefore not be considered or explored in this research.

2.2.6 Range Measuring

Main principles

Measuring of ranges of obstacles to the robot's location is the only method which allows localization of the robot as well as map building of the surrounding environment. Distance or range measurement is the only way to build maps of the environment. Most range measuring systems are based on one of four measuring principles: time of flight sensors, phase shift sensors, sensors based on frequency modulated radars or triangulation based optical sensors (active and passive) [11] [8]. As the navigation methods developed in this research use range measurements, a separate chapter further in this thesis (2.6) is focused on the state of the art in range sensing methods and their use in mobile robotics.

2.3 Systems of coordinates and elementary rotation angles

Coordinate systems and transforma-

In the three-dimensional space, points can be represented in the same space by different coordinate systems. A coordinate transformation is a conversion from one system to another. The coordinate systems which are commonly used for description of points in a three-dimensional space are of two main

tions

subtypes: Cartesian coordinate system and polar coordinate system.

Cartesian system

In Cartesian coordinate system, a point is represented by three physical dimensions of space: length, width and height. The three Cartesian axes are perpendicular to each other and a point in the Cartesian space is represented by a vector (*x,y,z*). [12]

Polar systems

Polar coordinate systems have three representations: circular coordinate system, cylindrical coordinate system and spherical coordinate system. Circular coordinate system (often called polar coordinate system) is only two dimensional. In cylindrical and spherical systems, points are described in the three-dimensional coordinate space. Spherical system's point representation (*ρ,φ,θ*) is given by the radial distance of the point from the fixed origin *ρ*, zenith angle *φ* from the positive z-axis (in the Cartesian coordinate system) and azimuth angle *θ* from the positive x-axis. Cylindrical coordinate system in fact extends the circular coordinate system: a point's representation (*r,θ,z)* is given by the radius *r* which is the distance of point's orthogonal projection to the *xy* plane from the fixed origin, azimuth angle *θ* from the positive x axis and the height *z*. [12]

Coordinate transformations

The transformations between the systems are given in equations (4) to (10). The conversion for all above mentioned three-dimensional point's representations is given. In Figure 2, the systems for representations of points in three-dimensional space are illustrated.

$$x = \rho.\sin(\phi).\cos(\theta) = r.\cos(\theta) \tag{4}$$

$$y = \rho.\sin(\phi).\sin(\theta) = r.\sin(\theta) \tag{5}$$

$$z = \rho.\cos(\phi) \tag{6}$$

$$\rho = \sqrt{x^2 + y^2 + z^2} \tag{7}$$

$$\phi = \arctan\left(\frac{\sqrt{x^2 + y^2}}{z}\right) \tag{8}$$

$$\theta = \arctan\left(\frac{y}{x}\right) \tag{9}$$

$$r = \sqrt{x^2 + y^2} \tag{10}$$

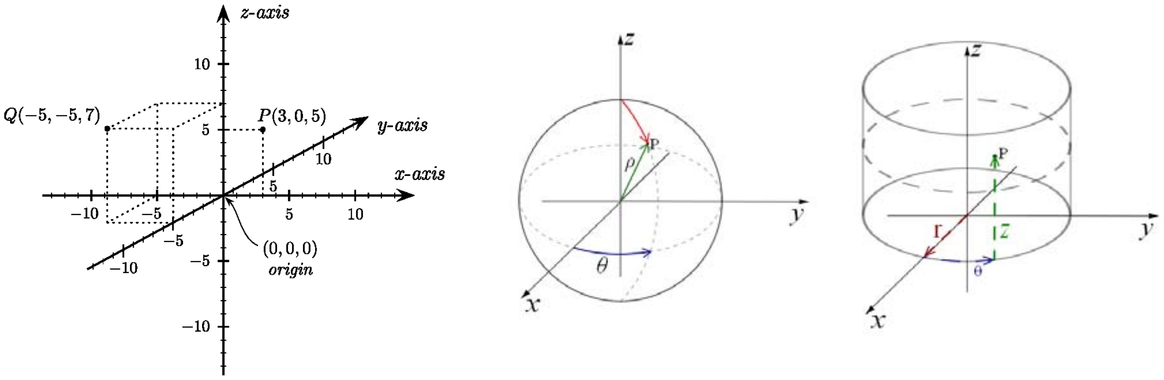


Figure 2: Representations of points in three-dimensional space: Cartesian system of coordinates (left), spherical system of coordinates (middle), cylindrical system of coordinates (right); Source: Wikipedia (en.wikipedia.org)

Rotation angles: flight dynamics

In this thesis transformations will be applied on data in systems of coordinates. Such transformations will be consisting of translations and rotations. A translation transformation is easy to describe as it is always given in the metric coordinate(s) of the given system (e.g. in the Cartesian coordinate system, the translation will be given by (x,y,z) vector). Rotation though is more difficult to be described though any rotation can be decomposed to subsequent three rotations around the Cartesian system of coordinates. The most common terms to describe rotations around axes of Cartesian system of coordinates originate in the flight dynamics. Even though the mobile robots presented in this thesis are wheeled, these terms to describe rotations will also be used in this thesis. These three rotations in three dimensions around vehicle's coordinate system origin are given by three angles: pitch, roll and yaw. In respective order, these are giving rotations around x, y and z axes in the Cartesian coordinate system [12]. These angles are demonstrated in Figure 3 on a wheeled robot; the y axis is aligned with the horizon and is in the direction of the robot's movement, x axis is aligned with the surface plane and is perpendicular to the direction of movement and z axis is "upwards", perpendicular to the surface plain. The description of these rotations was included in this section as these notations will be used frequently throughout this thesis.

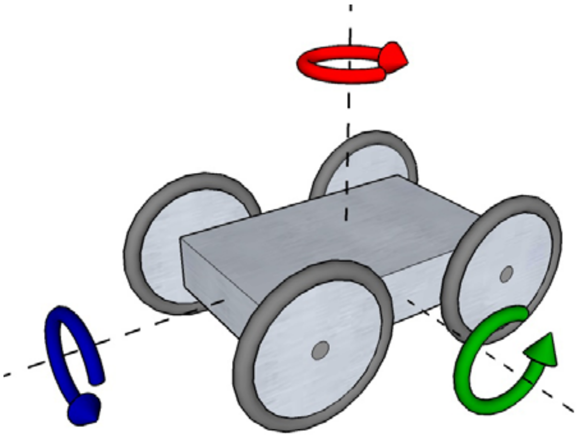


Figure 3: Demonstration of the flight dynamics angles on a wheeled robot: pitch (green), roll (blue), and yaw (red)

2.4 SLAM's Theoretical Perspectives

In this chapter, the theory and structure of simultaneous localization and mapping will be presented, focusing on the different aspects and processes common to most SLAM methods.

2.4.1 Global frame of reference

Global frame of reference selection

First step towards localization and map building is data acquisition. At the beginning of the exploration, we have to select a global frame of reference. Here we have two options: we can assume that the initial position of the robot is the reference frame, or we can correct it according to a specific requirement. The orientation of the reference frame and the position of the origin define the global frame of reference in 6DOF. The desired global frame of reference orientation is often the alignment of the reference frame with the earth's surface. This can be done automatically by installing absolute inclinometers into the robot and measuring the gravity vector in this way (assuming we're on a plain, this alignment with the horizontal plain is limited due to the curvature of the Earth's surface). Also, the origin of the reference frame can be specified by installing a GPS navigation unit and measuring the absolute latitude and longitude.

Once we have the reference frame, the position of the robot can be given within this reference frame as a six dimensional vector *P*, given in equation (11) (*x, y,* and *z* are the displacements from the origin, θ_x, θ_y, θ_z are the angles of deflection from x,y,z axis), or as a matrix of homogenous transformation $P_{MHT,}$ given that this matrix transforms the robot's orientation in its own frame of reference to the original frame of reference, consisting of a rotation matrix *R* and a translation vector *t* as in equation (12) [12].

$$P = (x, y, z, \theta_x, \theta_y, \theta_z) \tag{11}$$

$$P_{MHT} = \left(\begin{array}{ccc|c} & R & & t \\ \hline 0 & 0 & 0 & 1 \end{array} \right) \tag{12}$$

2.4.2 Preprocessing of the range data

Range data acquisition, point clouds

Now when the frame of reference is set, one can focus on the observation of the real world. In our case we will be observing the world through range data. With the first scan, we are presuming that we are at a known location, and thus the first preprocessed data are directly input into the map of the environment. Range data are in the form of point clouds, where each point is representing the measured range point. These points are often expressed in a Cartesian or a cylindrical coordinate system, as in equation (13).

$$p_{i,j} = (\phi_{i,j}, r_{i,j}, z_{i,j})^T = (x_{i,j}, y_{i,j}, z_{i,j})^T \tag{13}$$

Filtering the noise

The acquired range data have to be preprocessed in order to create a representative map of the environment. The main focus of the preprocessing phase is the removal of noise present in the data. For example, in the navigation system proposed by the team from Fraunhofer AIS, the Gaussian noise is being eliminated by preprocessing the single 2D range scans using a median filter with the size of adjacencies n=7. This filter substitutes the value of every point by the value of a median computed from the seven closest points of the surroundings, taking only the scanned line into account (not previous or next lines) [22].

Noise versus detail

The selection of appropriate filters could lower the noise present in the data but a side effect of the filtration could be effacing of details in the images. The loss of such details in the image could be critical to the following phases such as edge or object detection and data registration. The use of filters should be carefully considered especially with regards to the character of the measured data: if the noise present in the data is very low compared to the details present in the image which are necessary for the successful navigation and mapping process, the filtering should not affect the quality of the navigation method. On the other hand, if the scale of the noise present in the data influences to a large extent the perception of the details that are to be detected in the image, noise filtering could lead to loss of information and could contribute to the effacing of the image.

2.4.3 Semantic analysis option

Semantic analysis

One of the options in the processing of the range data is to extract semantic information from the data and in this way to create a map with cloud point groups that correspond to the extracted semantic information. The semantic information should be well suited to the environment. For example in indoor applications the categories should be selected in a generalized way so that they appear in most human constructed structures (such as walls, pillars, floors, ceilings etc.). On the contrary in natural environments, semantic information that could be extracted from the obtained 3D images is dependent on the characteristics of the particular environment (e.g. trees, overhang or water surface). This semantic analysis could help in the next stages of map building, when observing the same environment from different points of view.

Segmentation and feature detection

In order to detect objects in the image and to extract any information about these objects, the data have to be segmented. Segmentation is a process of partitioning of data into more regions using segmentation criteria. These criteria are often set to detect discontinuities of certain kind, e.g. detecting contrast areas representing boundaries in the image or distance discontinuities representing the borders of objects in the environment.

Surface typology analysis

The team from Fraunhofer AIS institute proposed a semantic analysis of 3D images in the article *Semantic Scene Analysis of Scanned 3D Indoor Environments* [21]. This semantic analysis was originally used only to analyze 3D maps, distinguishing between various semantic categories. The approach of segmenting the image using gradient information and extracting semantic information from this (detecting segmented areas) was further developed to be integrated in the process of acquiring three dimensional models [21][22]. The gradient function between neighboring points is being used to differentiate between three categories which are floor, object and ceiling. Such labels are assigned to all points meeting gradient criteria, given in the cylindrical coordinate system, while gradient *tanα* is obtained as in equation (14). The criteria in fact represent a decision tree where the decision is based on falling into these intervals: $\left(-0.5\pi,\tau\right)$ for floor, $\left\langle\tau,\pi-\tau\right\rangle$ for object and $\left(\pi-\tau,3/2\pi\right)$for ceiling points. The purpose of this semantic classification is not a 2D projection as in the case of previous paragraph, but the semantic information is used when building k-d trees for ICP matching (which serves to image registration and will be discussed in following chapters).

$$\tan \alpha_{i,j} = \frac{z_{i,j} - z_{i,j-1}}{r_{i,j} - r_{i,j-1}} \text{ with } -\frac{1}{2}\pi \le \alpha_{i,j} < \frac{3}{2}\pi \qquad (14)$$

Vertical landmarks segmentation

An interesting segmentation approach has been presented by Wulf et al. from the University of Hannover, who apply an edge detection segmentation using gradient [34]. The principle of the algorithm is that vertical lines and subsequently horizontal lines are analyzed and the gradient of the measured distance is computed between each two points, while the edge is defined as a point where the gradient of the distance is not continuous (of course certain threshold values are being defined). This edge detection algorithm is in fact serving mainly for the segmentation, the subsequent operation is dilatation (to reduce noise and remove border edges) and finally, surface segments are labeled using flood fill algorithm. This approach leads to landmark extraction extension, proposed by Wulf, Brenneke et al. [34], which was originally motivated by an attempt to create a leveled range scan in which landmarks would be displayed, leaving out the traversable areas. This method was developed to aid teleoperation, displaying a traversibility landmark map for the user.

Semantic analysis aiding localization

Described semantic analysis methods are primarily designed as supplementary map building processes, which of course could be highly valuable when mapping unknown environments. Both mentioned methods output information for map building. It could be highly valuable if these were to aid the localization process, as to certain but limited extent implements the method proposed by Fraunhofer AIS (by using the semantic information in the kd-tree building and searching).

2.4.4 Data registration

Scan/image registration

In order to explore the environment and to create a map which is representing the environment without occlusions, there have to be multiple scans taken from different perspectives. After each image is taken and the semantic analysis is performed, the data set has to be processed to expand the current map of the environment. This step is called image (scan) registration and the principle is that the current image has to be transformed into the original frame of reference. This step is based on finding the optimal rotation and translation corresponding to the displacement *t* and rotation *R* of the robot as in equation (12), for which the matching of the new scan into the global frame of reference with previous scans is most consistent. This means that the point clouds in the new scan referring to environment areas which appear also in previous scans are as close as possible to the corresponding point clouds in the map. As already stated, this rotation and translation in fact corresponds to the change in robot's pose (equation (12)). Thus the robot is localized in the global map of the environment after each image registration. The operation of scan registration (matching) is starting with initial values of the rotation and translation taken from the odometry of the vehicle, while the matching (registering) algorithm is from the localization point of view correcting the errors which have integrated from the vehicle's odometers measurements. Here we can actually observe the meaning of the simultaneous localization and mapping since image registration serves to both purposes. The registration procedure is in general a minimization algorithm, which minimizes the error caused by incorrect placement of new scans into the global map based on global frame of reference. The error function *E* can be in general expressed as in equation (15), where *R* and *t* are the rotation and translation variables which should optimally converge to the variables from equation (12), *M* is a model set of previously scanned points and *D* is a data set of new points from the current scan.

$$\begin{aligned} &E = f(R,t,M,D) \\ &M = \{m_1,...,m_i\}, D = \{d_1,...,d_j\} \\ &i \in N, j \in N \end{aligned} \tag{15}$$

Methods for data registration, ICP

The data registration can be done in an automated way using correspondence based methods. The mainstream in the 3D data matching is the use if Iterative Closest Point algorithm proposed in 1992 by P. Besl and N.D. McKay [6][5][3]. The navigation method proposed in this paper is using the Iterative Closest Point algorithm. In order to clearly present the proposed method and the modification to the existing implementations in a consistent way, this method will be presented in a greater detail later in this thesis in the part of the thesis focusing on the implemented navigation method, chapter 4.2. In general, the ICP method is minimizing the least-square point correspondence sum proposed by Arun et al. in [3] through finding corresponding points in datasets and estimating the optimum rotation to match them. It decouples the calculation of translation from the estimation of the rotation matrix. After determining all closest points for data and model sets, these are used to build a matrix which is decomposed using e.g. singular value decomposition (SVD) or quaternion based methods. The rotation matrix is determined from the results and further used for translation calculation using centroid points of the datasets. This is done iteratively until the best (optimum) match is reached when the rotation matrix is close to ones matrix. This match corresponds to the minimum of the least square point correspondence sum as it will be explained later in the thesis.

Evolutionary search based method

There are also other methods for minimizing the function expressed in equation (15). One interesting method is parallel evolutionary registration method proposed by Robertson and Fisher [25]. This method is using an evolutionary search while the chromosomes used in this search consist of six parameters: three angles representing the rotation of the system and three compounds of the translation. First the chromosomes are initialized using a random function. This is followed by the evaluation of the fit function, results of which are used in the following phase: adapted crossover and mutation. In the algorithm, there are in fact parallel processes for the evaluation of the fit function accelerating the computation for the whole population, since this could be seen as a bottleneck of the algorithm. The criteria – fit function is in fact very similar to the one used in the ICP algorithm and comes from Arun’s publication on Least-squares fitting of two 3-d point sets [3]. Hence the nearest neighbors have to be determined in order to calculate the fit function: in this matter the evolutionary search algorithm is limited by the time demanding nearest neighbor search which also appears to be the bottleneck of the ICP based methods.

Extending the global map

The data set which is registered with the model (the existing map) is then used to expand the map. In case of optical time of flight measurements, the measured data are often considered precise enough to directly expand the global map [35][23]. On the other hand, when the range data are input from e.g. from ultrasound sensors, the uncertainty of the measurements require the introduction of probabilistic approach, using sensor models to update the probabilistic representation of the environment. A widely used approach to this map-building is the Robot evidence grids approach proposed by Moravec and Martin and presented in [15]. The grids are separating the space to identical parts; each part of the space is then represented by the rate of occupancy or unoccupancy. There are different approaches of how to estimate this rate, the description of these is beyond the scope of this thesis. Since the range measurements used in this research are highly reliable (as will be shown later), the probabilistic approach would result in an inefficient rise of computing-power requirements. Also in case of 3D mapping, the memory requirements for the use of evidence grids would be very high. Therefore the map building in this research will be considered as simply expanding the model by the registered data points.

2.4.5 Post-processing issues

Enhancing global map, data fusion

After all images have been registered, global map has been expanded and the robot has been re-localized, we can further enhance the global map. There are various ways of doing this, the options mainly depend on the information from sensors which we have available. Some optical scanners allow us to get reflexivity data together with the distance measurements, which we can either utilize in the step of segmentation or now for the map enhancement, adding this information to the map. Other applications have multiple other sensors such as cameras, the signal of which can be used to create textures which can be applied to the 3D model after planar or other interpolations of the point clouds. This process is in general called data fusion. An example of data fusion use in the map building can be seen in work by Baltzakis et al., focusing on laser and visual data fusion for map building and collision avoidance [4].

Error diffusion

Closing the loop when mapping in unknown environments means that the robot finally returns to the area which was previously mapped e.g. acquiring the first 3D scan. During the mapping process, the added error in each image registration on the robot's path results in an accumulated error. This accumulated error causes the path-wise distant acquired images (the distance traveled by the robot between acquiring the scans is long) not to overlap as they should. The accumulated error can be ideally diffused over the path so that the map when closing the loop is consistent.

Relaxation methods

There are different strategies of how to diffuse the error after closing the loop when mapping in unknown environments. An interesting strategy to limit the global error is presented by K. Pulli [24]. In the proposed registration process, the newly acquired scan is registered to all neighboring scans in order to minimize the error and to eliminate inconsistencies in the scene. An interesting extension of the Pulli's method has been proposed by Nuchter et al., whose relaxation method was called simultaneous matching [21]. This process is in fact using a queue for image registrations. The current scan first in the queue is matched to the neighboring ones and if it is transformed, all neighboring scans are added to this queue. This ensures the diffusion of the error between all scans in the map.

2.5 Path planning

Path planning

An inseparable part of the mobile robot navigation is path planning, which is necessary for the autonomous movement of the robot in the environment and its exploration. The uses of three dimensional world representations have one important advantage over the use of only two dimensional world representations: any disparities of the terrain, obstacles and overhangs can be observed in the three dimensional model. Thus its application in path planning and collision avoidance is obvious. There are many options how to deal with such task; the main and most direct approach seems to be data reduction into 2D traversibility maps and thereafter path planning using existing methods. This thesis will though be focused on three dimensional map building and localization, leaving the task of traversibility and path planning open.

2.6 Current State of Range Sensing Methods and Technology

Range measuring for 3D perception

Although range measurements are very common in the field of automation and there is great number of available sensors, affordable range measuring systems which are precise, light enough and suitable for the use in three dimensional measuring in mobile robots are being produced in the last couple of decades. Thus the discipline of automated 3D navigation is fairly new. [1]

Passive and active triangulation

The triangulation range determination method is based on observation of certain point by two apart-situated sensors. The computation of the distance is performed by taking into account the angles under which the sensors perceive the target. This is commonly done in a method called passive

[1] There were more constrains in the development of 3D navigation besides measuring problems, as a second major constraint can be considered the lack of affordable and powerful onboard computing and memory devices

triangulation using two relatively cheap CCD cameras, where the whole scene is captured at a high refresh rate and consequently corresponding points are detected in both pictures, determining the range by the difference of the positions of points in the images. This passive triangulation method is therefore limited by the computational effort required to find matching points in the images, and the precision is limited by the resolution of the camera and the validity (and precision) of the optical system's model. Another issue is also the synchronization of the video acquisition of the two CCD sensors. Sensors such as CCD cameras are on the other hand widely available and are extensively used in mobile robots, though due to the computational and precision issue they have not been commonly used for the purpose of 3D range data acquisition. An active triangulation method is using a directed source of energy (e.g. a laser beam) and a fixed detector of the reflection, determining the distance by knowing the angle in which the beam spread and the position of the reflected beam on the detector's surface. The detectors are called position-sensitive devices and in active triangulation, the computation of correspondences is in fact substituted by hardware means – creating "detectable" energy sources in the environment by directing energy sources towards the environment [11][8].

Time of Flight sensors

Time of flight (TOF) ranging systems determine the range by measuring the time required for a pulse of emitted energy to travel to and from an object which is expected to reflect the emitted energy. Typically optical, radio frequency and ultrasound energy sources are applied. Due to the precision and availability of laser based TOF sensors (laser radars, or lidars), these now dominate the use in mobile 3D perception. The first laser radars were unidirectional, but today's common laser rangefinders have planar coverage with rotating mechanism, allowing up to 360 degrees scanning angle. Two most common products are SICK LMS and HOKUYO URG laser rangefinders. SICK scanner allows ranges up to 80 meters with outstanding precision in centimeters; disadvantage is the weight of the sensor (approx. 5 kg) and also its size (approx. 160x160x200 mm) [28]. The HOKUYO's URG scanner is very lightweight (approx. 160 g), small (approx. 50x50x70) and also precise (1 percent of measurement), but the range is limited to four meters [13]. Since these products have a planar field of view, in order to obtain 3D data it is necessary to incline the sensor with an actuator, commonly a servomotor.

Genuine 3D rangefinders

Even more advanced are expensive 3D rangefinders, the rotational mechanism of which is applied to both horizontal and vertical axis, thus it is possible to directly obtain 3D data. These are e.g. RIEGL and CYRAX products, which are commonly used for creating models of urban structures and archeological sites. Because of their price, these are not very common in navigation applications.

3 Platform development

Chapter's focus

In this chapter, the development of a platform capable of three dimensional data measurement and processing will be presented. It was necessary to perform this development prior to implementing and validating the navigation methods proposed further in this thesis.

Chapter's structure, state of technology at LTR

Since this is the first project in our workplace (Laboratory of Telepresence and Robotics – LTR) focusing on a three dimensional navigation system, an inseparable part of this research was the development of an efficient 3D range measuring and processing system. This system – as one can observe further – can be crucial in developing acceleration steps in existing 3D navigation methods. Therefore at first attention will be given to the state of technology which is being used in the robotic laboratories focusing on 3D data acquisition and navigation in unknown environments, estimating the advantages and disadvantages of the implemented technologies. Based on this research, an innovative 3D measuring system was developed and is presented here. The implemented sensor is shown also investigating its measurement characteristics. The hardware developed for sensor inclining-mechanism is presented, showing its capabilities and describing the electrical and mechanical parts of the system. Finally, robotic platforms which are available at the workplace are presented, followed by the insight into the innovation of robot U.T.A.R. which was part of this research.

3.1 State of technology in 3D range measurement systems

Stereo vision

The current mainstream in 3D mapping is based on the use of 2D laser rangefinders which are installed in an actuating mechanism allowing the inclination of the sensor and thereafter 3D data acquisition. We will though start this subchapter by discussing a minority approach to 3D mapping based on the use of stereo vision by CCD cameras. This approach is applied e.g. in research by Se et al. [26] and Biber et al. [7]. Extracting three-dimensional information from stereo vision means in fact estimating the depth from two images. This method is in fact called passive triangulation method. First, an object has to be identified in the images, the correspondence has to be calculated and then the procedure uses the information of the camera’s setup geometry and the displacement of the object within images to compute the depth. The main constraint and also the main reason why this method is quite rare in the robotics research is that the stereo base line is limited by the robot’s dimensions and often cannot exceed more than half of a meter. This together with low resolution of currently available cameras leads to relatively low resolution and high range measurement error, often being about ten percent in at distances approx. ten meters (the error gradually grows with growing distance). Also both processing and finding of corresponding points or objects in the images are difficult to run in real time, not even mentioning that higher errors lead to more difficulties in scan registration when building the map. This does not mean that cameras in general are not useful in 3D mapping; these are commonly used when enhancing the model of the environment e.g. for generation of textures. An example could be seen in work by Allen et al. in [2] .

The use of 2D laser rangefinders

As stated in previous paragraph, the most common method of measuring range data for three dimensional modeling of the environment is the use of 2D laser rangefinders. 2D planar laser rangefinder’s operation principle was described in the previous section. Its main advantage is in a high precision of measurements over long distances, while the minimal measured distance is with current technologies quite low (down to two centimeters). The mechanism used to incline the sensor is different for various research applications and in the following paragraphs different research teams’ solutions will be presented.

Rotating (inclining) 2D rangefinders

The most comprehensive analysis of the application of 2D laser rangefinder for fast 3D scanning by rotating it around different axes was performed by Oliver Wulf from the University of Hannover [35]. The team from the ISE/RTS department has tested four methods of inclining the SICK laser rangefinder with scanning angle of 180°. The methods of inclining the sensor are pitching, rolling and two different yawing methods. During the pitching scan, the scanning plane is horizontal and is pitching up and down. Rolling method is used when the scanner rotates around the middle measuring point (90° - forward, the center of the scanner), thus there is only one focus point. The yawing scans have vertical scanning plane and are rotating around the z-axis; in the first method the scanner plane is oriented sideways and in the second upwards (therefore not scanning the area under the scanner). The analysis focuses mainly on the measurement of point distribution density of various inclination principles, while the density of measured points is always higher in

Measurement density analysis

areas where there are beams whose direction is closest to the axis of rotation. Therefore, if the scanner is being inclined in the rolling sense, the highest density of points is in the direction where the middle measurement beam is pointed (that is ahead). During pitching scanning, the highest density is in the direction of the 0° and 180° beam which are on the sides of the scanner. Yawing scans have the highest density of measurements in the upper and bottom beam direction (since the sensor is vertically mounted).

ISE/RTS prototype

The ISE/RTS research finally led to a creation of a prototype of the scanner inclining device. The selection of the inclining method was motivated by the highest measurement density in the ground surrounding the robot, thus the team selected yawing scanning (with central beam facing the horizon). In this prototype, slip rings were also used for the unlimited continuous rotating operation. The success of this team in developing the inclining hardware was confirmed in the cooperation with Fraunhofer AIS and the University of Osnabrück in Robocup team Deutschland1, while the inclining device selected for the competition was the ISE/RTS prototype [22]. The actuator in this prototype is a servo-motor while real time control and measurements are done using a Linux based real-time OS. Fraunhofer institute's group AIS has so far been using a horizontally mounted pitching scanning device, also using SICK rangefinder.

Fraunhofer AIS

Non-rotating 2D rangefinders

Thrun et al. and Zhao et al. are using multiple static mounted 2D laser rangefinders, which are oriented in different axis and the planar data are merged into 3D model based on the current pose of the robot [39][33]. Thus creating a 3D model requires a movement of the robot. Also the errors which arise from the odometry measurements are projected into the 3D model. One scanner is oriented horizontally and one vertically. Zhao et al. use two additional scanners shifted by 45° from the previous two in order to assure a better model with less occlusions. The pose of the robot is determined by using a 2D SLAM on the data from the horizontal scanner.

3D scanners

RIEGL and CYRAX 3D scanners mentioned in the previous section are not very common in robotics due to the high price of the sensor. There are in fact only two applications, one in the RESOLV project and one in the AVENUE project [27][1]. Both of these have one in common feature: the three dimensional data are not used for navigation because it is not the primary purpose of the project (which is the visualization and the creation of the 3D model).

3.2 3D range measuring system development

Discussion on previously described imple-mentations

The current research implementations of three dimensional range measurements utilize mostly precise 2D SICK LMS proximity laser scanners which are much more affordable than “all in one” 3D devices. The design of the actuators is often limited to the “stop and go” fashion of movement not only in terms of the movement of the robot, but also in terms of the movement of the scanner itself since servomotors are used as actuators. If the regulated variable is the inclination angle of the scanner, it implies that in order to reach certain inclination angle precisely during 3D scanning, long pauses between each scanned line have to be taken. The resulting movement of the sensor is abrupt, leading to sensor attrition and high demand on mechanical robustness of the optical parts. Accelerating this type of scanning procedure leads to non-uniform inclination angle distances between lines and therefore to additional errors in the range estimation.

Selecting the sensor

In this research, the fast and robust SICK LMS sensor was selected for the design of the 3D scanning system due to its capabilities and reliability. There were also other sensors available at the workplace, such as Hokuyo URG and high speed Messa SR-3000 [16][13]. The main objective was to develop a scanning system capable of taking 3D scans with the highest possible range and precision. The mobility of the system was also an issue though not the priority. Both Hokuyo URG and Messa sensors are more suitable from the mobility perspective. But the maximum range for the Hokuyo URG is approximately five meters, the scanning period is 100 ms per scan and these parameters are not sufficient. The Mesa SR-3000 sensor’s range is approximately seven meters and although the sensor acquires range images in video refresh rates since it is a CCD based sensor, this was also not sufficient especially if it was to be used in general indoor and outdoor applications at any point of research. Therefore the heavy though capable SICK LMS 200 sensor was selected [13] [16] [28].

Proposed method of inclining

After selecting the sensor, the objective was to develop an inclining hardware which eliminates the above named disadvantages of the use of servomotors. Instead of servomotors, a DC motor was used with nominal torque sufficient enough to be able to operate in a quick stop and go regime. The main operation mode though is a precise angular velocity regulation, meaning that the inclination is changing at a constant rate. In the following subchapters, the sensor itself and its reliability (measurement repeatability) will be discussed, following by the description of the inclining mechanism development.

3.2.1 SICK LMS 200 sensor

SICK LMS operating principle

The SICK LMS 200 sensor's intended use is mainly in the industry as a stand-alone remote measuring system. The SICK LMS scanners are operating as time of flight sensors: measuring the time it takes a laser light pulse to travel to an obstacle, where it is reflected, and returns back to the sensor, where it gets registered by the sensor's receiver. The pulse laser beam is reflected by an internal rotating mirror, so that a cross-cut of the space on which the range measurements are taken in equidistant angles is perpendicular to the axis of mirror's rotation. The light pulse is emitted every 1°of the angular range 180°, while the offset of the first angle value can change in every scan from 0 to 0.75° by +0.25°step, depending on the operational mode. These principles are illustrated in Figure 4. The sensor SICK LMS 200 was chosen mainly for the following paper characteristics: it is capable of data acquisition at speeds of up to 75 Hz (scans per second ~ rotations of the inner mirror per second), its angular range is 180 degrees and resolution at this speed is 1 degree. In slower modes of operation, its maximum resolution is 0.25 degree, quadrupling the time required for the measurement. The maximum distance range of the scanner is 80 meters and the systematic error at the distance of 8 meters is ± 15 mm [28].[2] The operation modes of the scanner will be described in more detail later in this chapter.

Communication subsystem

The sensor communicates with a PC or a standalone processing unit using a serial link; both standards RS-232 and RS-422 are implemented. The communication speed can be set according to the requirements of the application, though the highest supported RS-232 baud rate is 38,400bps and thus high speed measuring modes can only be set when communicating through RS-422 serial link. Therefore in our application we had to use the

[2] Given 60 % reflectivity and good visibility conditions (unpolluted air, no fog); the range is related to visibility conditions and reflectivity of the surface.

RS-422 connection, targeting to create a 500 kb/s link. This was the fastest supported baud rate by the SICK sensor. Several RS-422 PC adapters were tested for this purpose (both PCI and USB connected), though only one adapter – implementing FTDI FIFO-UART converting circuit - turned out to be capable of running at this non-standard baud rate (given a very high onboard oscillator frequency which could be divided by an integer to achieve 500kbaud). The only disadvantage is that it is converting the serial link to a USB connection, therefore causing delays in the communication and this implies difficulties in the synchronization of the currently read range data with the measurements of the inclination from the driving unit. This will be discussed in the following chapters, where we will focus first on the inclining mechanism and then on the synchronization of the data with the inclination measurements. The implemented module for the RS-422 to USB conversion was previously developed in our laboratory for universal applications [14]. This device was the only available which was successfully tested at high speeds with the scanner.

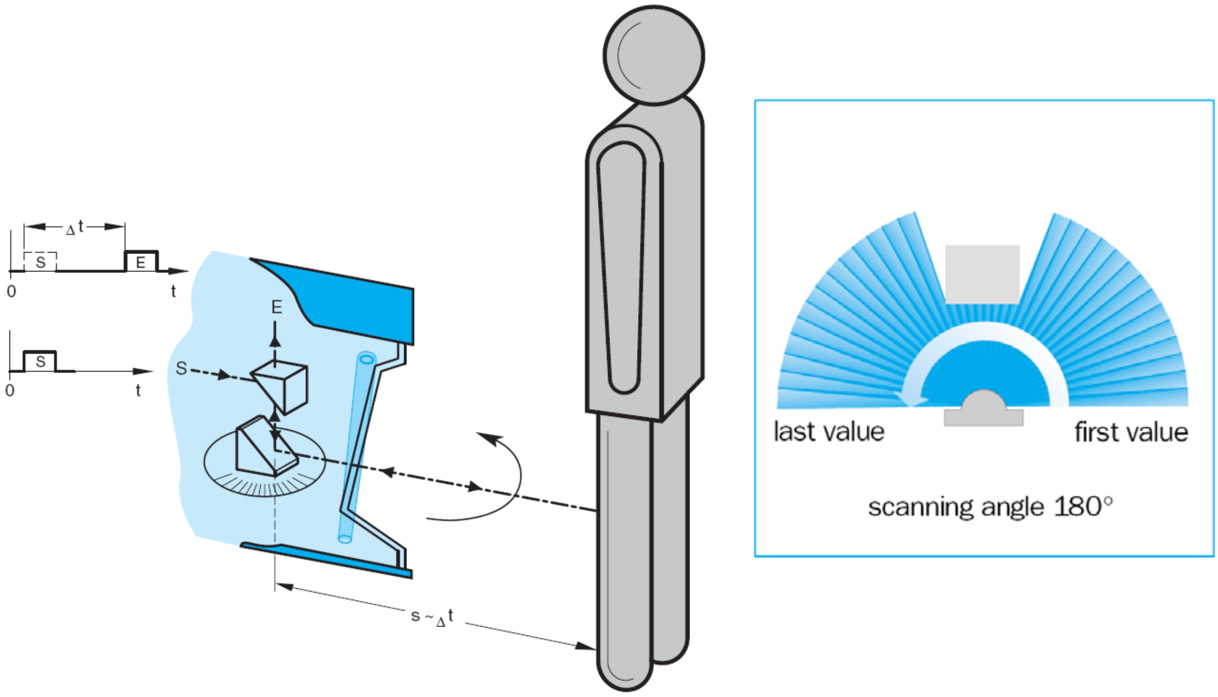


Figure 4: The principle of the time-of-flight operation, scanning horizontal angle range; Source: SICK Technical Description [28]

Measurement regimes and resolution

The scanner can operate in three different regimes: measurement of distance values, measurement of energy values received (can be interpreted as the "reflectivity" of the material) and a combination of both. The coding of the measuring values can be set and this setting alters the maximum measurement range representation and also the resolution of the returned data. The resolution and range representation values are shown in Table 1. In indoor applications, the 32 m range setting was applied, since it offers the best resolution (1 mm) and the maximum range value is still acceptable. In outdoor applications we would have to consider the 80 m setting [29].

Table 1 : Measurement ranges and the resolution on different regimes

Measurement Range	Data bits used	Max. representation	Max. measurement range representation	Resolution [mm]
8 m	13	1FF7h = 8183dec	8.183 m	1 mm
16 m	14	3FF7h = 16375dec	16.385 m	1 mm
32 m	15	7FF7h = 32759dec	32.759 m	1 mm
80 m	13	1FF7h = 8183dec	81.83 m	10 mm

Basic communication principles

The telegrams for the communication with the scanner always contain the address of the scanner, command or type of response, appropriate data and a checksum. Response telegrams from the scanner are no longer than 812 bytes and the data is transmitted with accordance to the Intel standard. The scanner also uses "Acknowledged" or "Not-acknowledged" proprietary handshake responses. There are three main modes in which the scanner can be switched: installation mode, monitoring mode and calibrating mode [29].

Installation mode

In the installation mode, the servicing unit communicating with the scanner can perform a very complex setting of the scanner. When switching to the installation mode, a correct password has to be provided to the scanner and then the configuration telegram can be sent. These important settings were done in the configuration for the 3D range measurement purposes: transmission of real-time indices was switched on: these are the number of current telegram and the number of mirror rotation. These numbers are checked with every received telegram during scanning procedures in order to detect missing measurements. Also, multiple evaluations of measured data was switched off in order to accelerate the measuring process, sensor sensitivity was set to a default value and the data resolution was set to 15 bits. This setting of data resolution allowed maximum resolution (1 mm) and necessary range (32 m) but unfortunately also disabled the transmission of reflectivity data (only 1 bit is left in each 16-bit measurement value which is not enough to encode the reflectivity measurement, though the priority in this research are 3D ranges). In the installation mode one can also specify a security area which is further used by the sensor in a special regime in monitoring mode serving to detect intrusions in this area. Since this feature is unused and might be disturbing in this research, this area was limited to the closest possible surroundings of the sensor [29].

Monitoring mode

Monitoring mode is the mode in which the sensor operates most of the time, allowing reading of range measurements. There are more ways for the sensor to operate in the monitoring mode: the basic regime is that the measured values are transmitted only if these are requested by a data request telegram. Other setting is that the range values are transmitted either if these are requested or if an object enters a security area specified in the installation mode. For our application the most important regime of the monitoring mode is the continuous output of all measured ranges which ensures that the data is output after every mirror rotation together with the real-time indices (if these are specified in the installation mode). Although the internal manual from the SICK manufacturer specifies two ways of value transmission, only one of these options was supported by our LMS-200 scanner: the interlaced continuous regime. In this setting the data is output after each rotation in the following fashion: first, 181 data values ranging from 0° to 180° with a 1° step are output; then 180 range values ranging from 0.5° to 179.5° are output, then again 181 values as in the previous step are output and the transmission continues changing the number of values and the offset. The number of scanned angles in 180° view is 361, giving the horizontal angular resolution of half degree, although the regime is in fact interlaced. The manufacturer specifies that the sensor in this mode should have 4 different offsets (0, 0.25°, 0.50°, 0.75°) and the interlaced cycle should be 4 rotations of the mirror, though the experience in this research is different and also other teams have confirmed same results with this type of scanner. Sensor can also output a specified number of mean values on request (the number of values to be transmitted is set in the data request telegram) and also a specified measurement sub-range can be transmitted continuously [29].

Calibration mode

The calibration mode is an internal SICK manufacturer's mode which is used during manufacturing to calibrate the sensor. Assuming it was correctly calibrated during the manufacturing process, the sensor was not calibrated in our laboratory.

3.2.2 Validation of measurement's reliability

Reliability validation

The measurements taken by the LMS laser rangefinder are according to the datasheets very precise, though for the comprehensive analysis it was necessary to validate these characteristics. Since the sensor's resolution is in millimeters and we did not have equipment in our laboratory to validate the absolute precision of the scanner, we decided to focus more on the reliability of the range measurements. Therefore the objective was to estimate standard deviation for different distance measurements, altering the targeted surface's characteristics – influencing reflectivity and focusing also on color characteristics. The reliability of the measurements is important when using the sensor in robot navigation, so that the objects are measured in a way consistent over time.

Measuring

The experiments which were carried out in order to investigate the reliability of the scanner's range measuring focused on three properties: distance to the obstacle, reflectivity of the surface and color properties of the surface. In the following paragraphs the way of influencing these properties will be discussed, followed by the summary of the results from the experiments.

Distance from the obstacle

The distance to the measured obstacle was the easiest characteristic of the measuring circumstances to be influenced. An artificial obstacle was created and the distance from the scanner was measured using measuring tape – as previously mentioned, the main focus was not on the absolute precision of the scanner but rather on the reliability of continuous measurements.

Reflectivity

Reflectivity is a fraction of the spectral intensity of reflected radiation over the spectral intensity of the incident radiation. It is dependent on both the wavelength of the emitted radiation and also on the angle of incidence (measured from the normal of the surface). If the surface of the material is specular, the reflectivity will be very low at zero angle of incidence (understand that the majority of the light will be transmitted through the surface) and close to one at angle of incidence close to the right angle (majority of light will be reflected). In fact, reflectivity for specular surfaces is approximately one for angle of incidents higher than the critical angle of incidence (which is determined from the refractive indexes). An example of such surface could be polished metals and minerals. On the other hand when the surface of the material is diffuse, the reflectivity will not significantly change with the change of angle of incidence. This is due to the fact that diffuse surfaces have granular surface and the radiation is reflected at a number of different angles at the same time. An example of such surface would be matte and glossy paints. An extremely granular surface will cause a spread of the reflected light over a hemisphere surrounding the surface resulting in independence of reflectivity on the angle of incidence [12].

Changing the reflectivity

In order to validate the influence of the reflectivity of an obstacle on the measurement reliability, it was necessary to manipulate the reflectivity of certain material. It is important to note here that our laboratory was not holding any special equipment to measure the spectral intensities. Therefore our results are only approximate and provide only limited information about the reliability of our measurements. If a specular surface was to be used, it would be possible to change the reflectivity of the surface by changing the angle of incidence. Though we would not only have to determine the dependence of the reflectivity on the angle of incidence, but since the reflected energy is emitted to the direction of negatively taken angle of incidence, we would have to place a mirror in this direction so that the sensor could pick up the reflected signal, having to determine the reflectivity of the mirror as well. If a diffuse material was used, the change of reflectivity would need to be changed by the change of material properties, which in the case of this research seemed to be much easier. It was easier to proceed this way by using a matte paper and printing different shades of gray, controlling the percentage of coverage by black ink. To limit the number of measurement to be taken at different distances, the number of shades was set to five, where five corresponds to white paper and thus the highest reflectivity and one corresponds to 100 % coverage by black ink. Each step up removes 25 % of the original paper's (num. one) black ink coverage.

Changing the color

In order to estimate whether the reliability of the measurements changes over the distance differently for objects of different colors (which means objects whose reflectivity is sensitive to the incidental radiation wavelength), a set of colored papers was created. Five colors from the visible spectrum were chosen: violet, blue, green, yellow and red (these colors are in the order of their wavelength). This part of the experiment unfortunately lacks important validation of the sample objects – a device to measure spectral intensities was not available, therefore reflectivity values for different colors could not be measured, and thus the results are only illustrative and cannot be used for modeling of the sensor taking color sensitivity into account.

Experiment

For the experiment, twelve distance values between 1 m and 15 m were selected and approximately 2,000 range values were measured for each artificial object: five black & white papers and five colored papers. In Table 2 the standard deviations of range measurements are presented for both colored objects and also the grayscale objects. Figure 5 represents the dependency of the standard deviation of the measurements on the grayscale objects with positive (though not linear) steps in reflectivity. Figure 6 shows the same results for the colored objects.

Table 2 : Standard deviations of the measurements for each color and shade of gray depending on the distance from the scanner

Distance [m]	Standard Deviation [mm] of the Measurements for each Color / Shade of gray									
	Violet	Blue	Green	Yellow	Red	100% black	75% grey	50% grey	25% grey	white
1	3,48	2,37	4,04	4,90	2,34	3,48	4,31	1,43	1,96	4,19
2	1,59	1,60	1,63	1,85	2,62	0,95	1,61	2,36	3,91	2,39
3	1,16	1,25	1,71	1,52	1,27	1,29	0,98	1,12	0,87	0,77
4	2,03	3,41	3,83	2,49	2,43	5,14	5,16	3,56	3,53	1,77
5	4,39	2,02	1,98	4,09	4,54	4,61	1,80	4,85	3,10	2,32
7	8,22	8,12	8,15	8,96	7,35	2,88	9,23	3,81	8,08	9,15
9	5,96	5,61	3,79	4,12	5,52	3,83	5,15	6,11	6,12	5,18
11	3,47	3,56	3,24	2,88	2,87	4,95	4,41	3,83	3,87	3,22
13	4,75	5,08	4,20	4,22	4,35	5,03	5,10	5,48	5,58	5,53
15	5,04	5,12	4,54	4,77	4,25	5,07	5,34	5,40	5,01	5,01

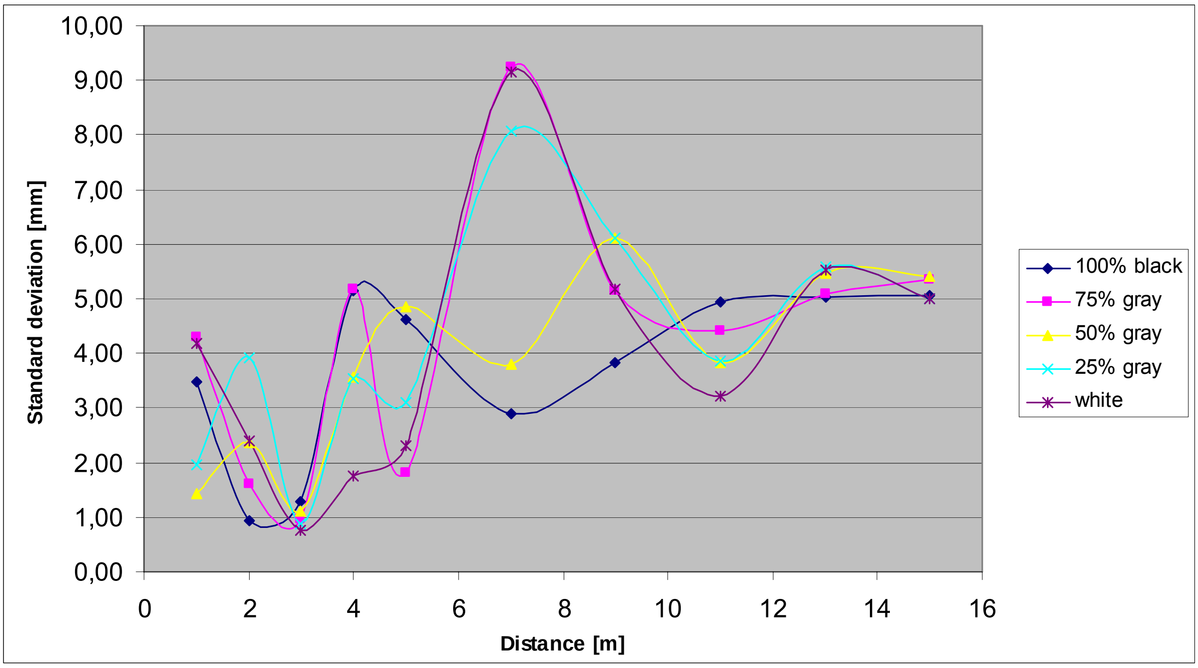


Figure 5: Dependence of standard deviation of the range measuement on the distance from the obstacle for objects of different reflectivity

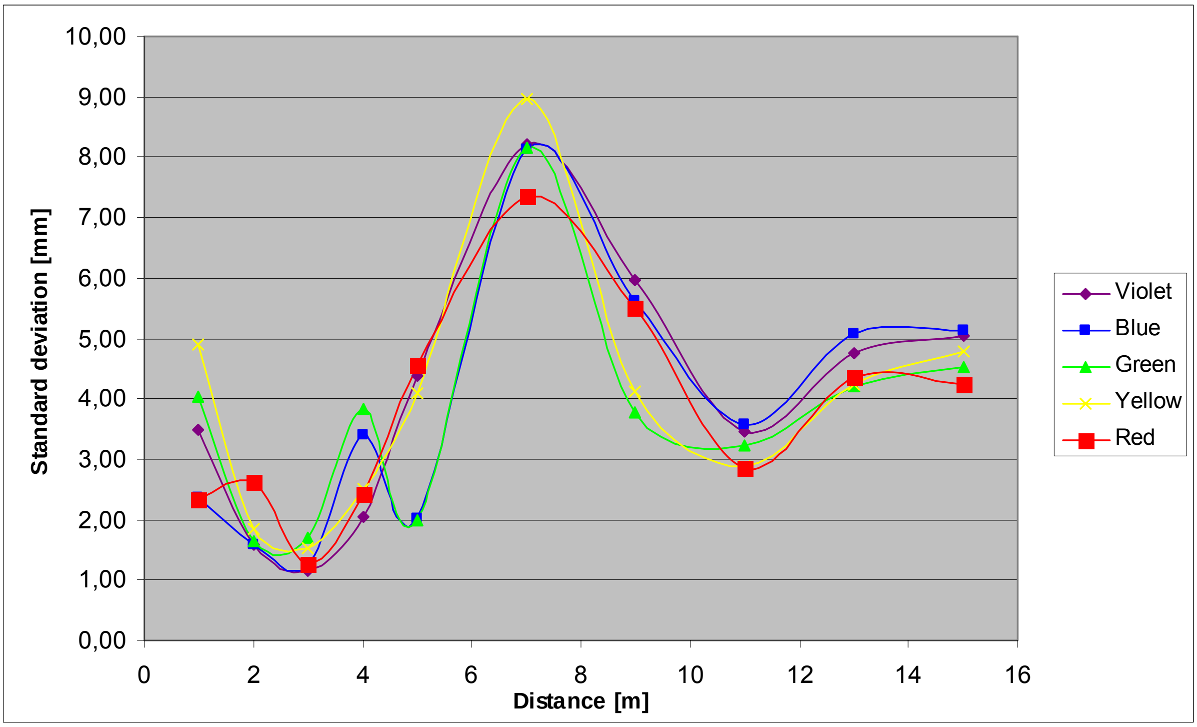


Figure 6: Dependence of standard deviation of the range measurement on the distance from the obstacle for objects of different colors

Discussion on the results

It can be observed that the standard deviation does significantly change over the measured distance, though the dependence on the reflectivity or the color of the object could not be clearly interpreted from the data which was measured. The reflectivity influence on the standard deviation is not quite clear from Figure 5 or Figure 6, since there is not a single consistency with the changing reflectivity. One result that is observable both in the grayscale and colored objects detection is that the standard deviation rapidly rises at the distances of approximately 7 m. Also, and this is visible especially in the colored object detection graph, the standard deviation seems to rise with the growing distance, although there are few local minimums and maximums. The maximum measured standard deviation was around 9 mm, which means that 68 % of all measurements of a specific object will lie in the interval <mean – 9 mm; mean + 9 mm> and approx. 95 % of the data will be in the interval <mean – 18 mm; mean + 18 mm>, where mean is the mean value of the maximum possible measurements one would take.

The average standard deviation though is around 5 mm (even for distance 15 m), which implies the 95 % confidence interval +-1 cm from the mean value (although the absolute precision is unknown, the sensor would place 95 % of its measurements within this interval, which is an assurance of the measurement reliability).

Histogram of the measured data

In Figure 7 a histogram of one of the measurements is displayed. This measurement had 2,000 values, the target object was a white paper and the distance was approximately 9 m. As it can clearly be seen in the image, the sensor has a bimodal distribution since there are two almost equal local maximums in the histogram – approximately one cm far from each other. This tendency was also observed in histograms of other measurements, while the second maximum is usually less significant than the first one.

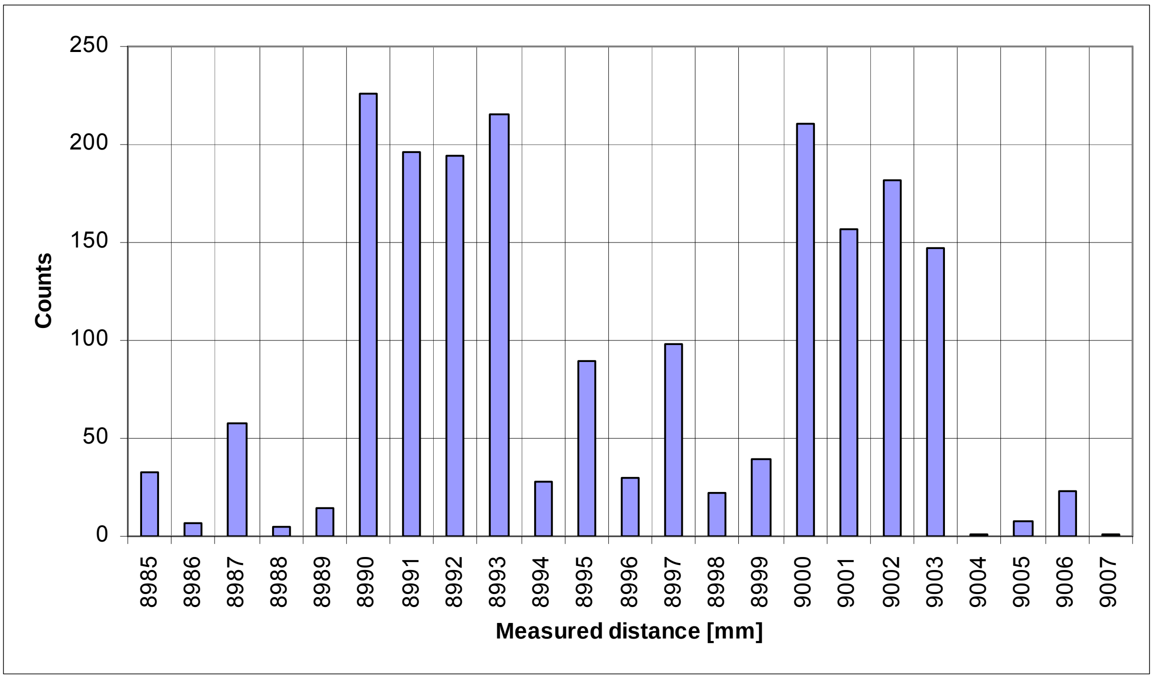


Figure 7: Histogram of range measurement, measured distance approx. 9m, target object - white paper

3.2.3 Inclining mechanism

Continuous inclining method

A DC motor was used as an inclination actuating device. The main operation regime is a precise angular velocity regulation, meaning that the inclination is changing at a constant rate. Presuming that the SICK LMS sensors have a very stable and precisely determined duration of one rotation of the mirror inside the sensor (13.30ms), if we obtain an information about one single inclination angle position of the scanner ϑ_0 at a given time t=0 when distance at rolling angle ψ_0 was measured, we can compute the remaining pitch angles ϑ of each scanned point using the following equation:

$$\vartheta = n.\omega_\vartheta.t_{rot\psi}.\frac{\psi - \psi_0}{360} + \vartheta_0 \qquad (16)$$

In the equation (16), ψ is the roll angle, ω_ϑ is the inclining angular velocity; $t_{rot\psi}$ is the time it takes the mirror inside the scanner to rotate around the axis, *n* is the number of completed turnarounds of the mirror and $(\psi - \psi_0)/360$ is the remaining fraction of the mirror's rotation at the given point of measurement.

Hardware limitations

In order to ensure such operation of the hardware, it was necessary to ensure that the DC motor has a precise incremental encoder installed and that the motor itself will be powerful enough to overcome the disturbance of gravity and other influences (friction etc.). The resulting image will not be in the form presented e.g. by Wulf et al. and Nuchter et al., since the measured points will not be positioned in a perpendicular grid but will be positioned as if the scanned lines were tilted [22][35]. Therefore, the data processing is more complicated, requiring computation of one additional parameter: the inclination of each scanned point.

Inclination methods

The orientation of the scanner was also an issue. The analysis performed by Wulf et al. already mentioned in chapter 3.1 concluded that the orientation of the scanner is very influential on where the highest density of scanned points would be. In each application, this high point-density area should be directed to the area of interest, which in the case of this research would be in front of the robot. This would imply the rolling inclination of the scanner: positioning it so that it is facing towards the front area of the robot with the planar scanning area aligned with the horizon when at zero inclination and inclining the scanner around an axis going forward from the robot (the axis of inclination positioned in the 90° scan point of the scanner). If slip rings were used for powering and communicating with the scanner and also extremely precise odometry and inertial measurements were taken, this rolling inclination method could operate in a continuous rotating regime and thus uninterrupted 3D data acquisition would be possible even when the robot would be moving.

Criteria: field of view

Nevertheless there were other criteria influencing the selection of the inclination method. A very important factor was the possible field of view of the sensor while inclining it. This could change – if inclining as described above in the rolling sense around the 90° scan axis, the field of view would be only half of a theoretical sphere. If inclining around the axis going through 0° and 180° scanning angle of the SICK sensor, the field of view is full - spherical. This is though only in theory since the robot itself and the construction of the mechanism always create a field of view obstacle.

Criteria: 2D scanning when moving

Another aspect was the possibility of real-time SLAM in 3DOF (x, y axes translation, rotation around z-axis) during the movement of the robot which requires horizontal alignment of the scanning plane with the ground. In case of rolling inclining when axis of inclination goes through the 0° and 180° scan angles of SICK sensor and if the scanning plane when still is aligned with the horizon, the scanner would not have the whole area in front of the robot in its field of view. It would only be able to see half of it for performing the real-time 3DOF SLAM. If pitching the scanner with the plane at zero position aligned with the horizon, the scans would be ideal for 2D scanning during movement. If yawing the scanner, the scanning plane is always perpendicular to the horizon plane and therefore the scans would not be suitable for 3DOF real-time SLAM during robot’s motion.

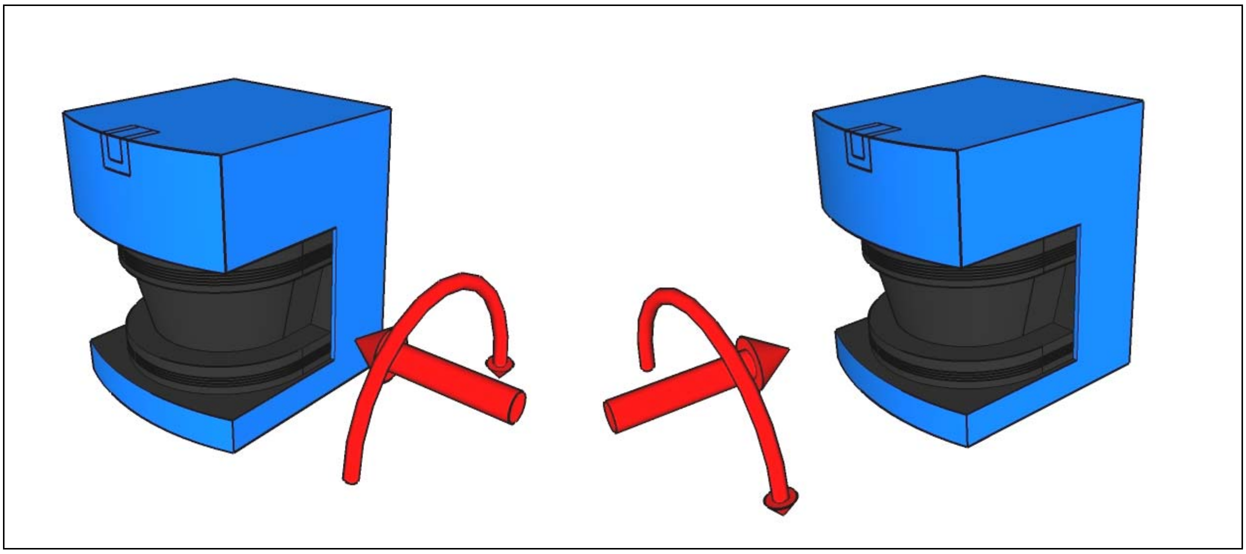


Figure 8: The options to incline the scanner: pitching inclination (left); rolling inclination (right)

Criteria: Extraction of leveled maps

Last aspect of the selection was the possibility of accelerating extraction of certain type of objects from the 3D data in order to create a leveled map. The objects which will later be extracted from the 3D image are vertical structures. The motives and principles of the extraction will be explained later in this thesis - in the description of the navigation method since these objects are being used for the 3DOF SLAM. If the vertical structures extraction is to be accelerated, the data should be organized in such way that they contain vertical columns – clusters of points with identical yaw angle. Since this localization method is to be operating in unknown environments, we cannot ensure that the robot always scans the environment when in horizontally aligned environment. Therefore the only way to satisfy this condition would be the installation of a standalone 3D scanning system positioning mechanism which would keep it horizontally aligned. In addition to this, the scanner would have to use yawing inclination method and the inclination would have to be performed in the stop and go fashion (so that the scanned lines would form vertical columns). Such configuration is very complicated and technologically challenging.

Selected inclination method

Taking all these factors into account, the “pitching” inclining method was selected: the scanner’s scanning plane in zero position is even with the horizon and the sensor is being inclined / tilted up and down, theoretically allowing whole spherical view of the surroundings. The scans in zero position can be used in real-time 3DOF SLAM when robot is in motion. Such navigation methods are being developed at the workplace and these are not part of this thesis, although the principles of 3DOF SLAM are implemented for the use on leveled maps extracted from 3D data (this SLAM method could be easily used on the real-time horizontal scans). The selected configuration therefore allows the use of the robot equipped with this scanning system for other experiments. The resulting 3D data are though not suitable directly for the vertical structure extraction and vertical column extraction method has to be implemented in the navigation system.

HW: actuator and sensor

As already mentioned, a DC motor is used to power the inclining mechanism. The motor has an integrated snail transmission and also an incremental encoder attached to its shaft. The motor brand is Valeo, having a nominal torque 2.0 Nm, starting torque 17 Nm and maximum rotation speed 50 rotations per minute. Given the transmission ratio 99/1, the available torque is strong enough to quickly manipulate the heavy SICK LMS sensor, and since the maximum rotation rate is approximately 180° per second, it is sufficient to perform high speed 3D scans. The implemented incremental rotation sensor LARM IRC 340, attached on the fast motor shaft, is a two channel incremental optical encoder with 100 pulses per rotation, giving a theoretical resolution of 39,600 steps per rotation.

Motor driver - electronics

The electronics for driving of the motor and controlling the inclination of the laser scanner had to be built. A universal dual driver unit was assembled for this purpose, using a design already implemented in our department for different purposes (e.g. driving of the main motors in the Orpheus robot) and developed by L. Kopecny [37]. The driver is based on a Vishay configurable H-bridge driver circuit, which is controlling the opening and closing of FET transistors according to the PWM signal output by an ATMEL microcontroller. Also, the “direction” and “enable” signals are output from the ATMEL, setting the appropriate direction of motor current / rotation and enabling the opening of the transistors.

Control SW for the inclining device

The Atmel microcontroller’s SW was developed to specifically fit this application. The communication is implemented using an RS-232 serial link and the packets are interpreted in the following way: first two bytes contain the correct identification number of the driver, second byte is interpreted as a starting position of the scanning, third byte is the stopping position and the fourth byte is the desired speed. This is followed by a simple checksum. The start/stop positions are in hundreds of encoder impulses, the desired speed is in pulses per control loop duration, which is set to 4ms. The two channels output from the incremental encoder are input in the microcontroller, interrupts are set for appropriate edges detection on these ports and an accumulator is updated when this interrupt is called. Due to the required precision of the scanner movement and the speed of the microcontroller, the resolution was left on the optimal 19,800 pulses per one rotation of the inclining mechanism (detecting only one type of edge on each channel, thus doubling the basic resolution 9,900 pulses per rotation). An interrupt from the controller’s timer is set to call the controlling loop each four milliseconds for determining the PWM and “direction” signals which are to be output to the bridge controller. A PS regulator was implemented in the controlling algorithm, where the regulated variable is the rotation speed of the motor. Depending on the current stage, the desired value is either set to 0 (when the scanner is to be still), to the transition speed $-\omega_{trans}$ when moving the scanner to its start position or returning it to zero position after scanning and finally to the desired speed $\omega_{desired}$ when in the scanning procedure. The transition speed is set in such way so that the scanner could move quickly to the starting position. The desired speed is received in the packet from a PC or other device. The scanning procedure is started immediately after receiving a correct packet in the following fashion: first, the scanner is moved to the starting position (in the positive – up direction). When reaching the starting position, the driver transmits this information to the processing station (to give the processing station enough time to enable the reading of the data from the laser scanner) and still drives the scanner a little further (by a specified offset) in the same direction. This is to ensure that that the system has a trajectory long enough to reach the desired rotation speed on the way down. When it reaches the starting point position minus the mentioned offset, the controller starts to incline the scanner in the negative – down direction, controlling the speed to the desired speed value received previously in the packet. When passing the starting point again, it transmits this information again for synchronization purposes (the data currently read from the scanner will be marked as starting in the processing station). The driver then continues to incline the scanner at the desired speed until it reaches the stopping position, transmitting this info again through the serial link, stopping the scanner movement and then returning the scanner back to zero position at the transition speed. Then the system waits until another controlling packet is received.

Synchronization of range data and inclination

An issue which was quite difficult to be solved was the synchronization of the range data which is read from the scanner with the inclination information from the driver. Since the communication with the scanner is done using a USB – RS422 converter, significant delays are introduced in the communication between the scanner and the processing machine. When the information about the current scanner's pose at the start inclination angle is received, it is recorded to the appropriate range line metadata. Since the delays could be easily in tens of milliseconds, these could cause an additive error in inclination angle of each data point, while this error would depend on the inclination speed and the delay which was introduced. This issue was finally resolved using an algorithm for detecting the mechanical construction in the 3D data inclination device, since this construction is always present in the data as an obstacle. As the objective inclination of this construction to the robot's pose is known, it provides necessary information to align the data with the robot without having to synchronize the range data reading with the inclining measurement. The algorithm will be discussed in chapter 5.4.

Operating parameters

Since the axis of rotation when inclining the sensor is not placed precisely in the scanner's centre of gravity, the gravity introduces disruption in the

of the device

regulating algorithm. PS algorithm can regulate steady state deviation though when the disturbance changes in time, error is introduced. This error was measured and when realistic inclining speed was set (approx. 45° per second), the maximum measured error from the desired inclination speed was about 2.5 %. These results are satisfactory. Also, the maximum inclination speed of the device is about 170° per second, meaning that a scan of the front view (approximately 160°) can be finished within one second. At this speed the density of points is quite low, since the number of scanned lines would be around 70. The optimal scanning inclination speed is about 35° per second. Lowering the speed is also possible down to approx. 15° per second, though one has to take into account that the lower the inclination speed, the larger the produced 3D data sets. Lower inclining speeds only improve vertical resolution of the scanned data since the horizontal resolution is given by the discrete number of scanned angles in the plane of the planar laser scanner. Figure 10 shows the inclining mechanism together with the robot on which the platform was fitted – U.T.A.R.

3.3 Robot platforms development and integration of sensors

Available platforms

There are currently three main robots for autonomous navigation implementation available in the Laboratory of Telepresence and Robotics: Hermes, Orpheus and U.T.A.R.. In this chapter, the current sensory and other equipment of the robots and the possibilities of their use for the implementation of 3D autonomous navigation methods will be discussed. There are also other platforms being developed, such as low-cost mobile robot platform. Also walking robots are currently being studied. Either the state of the research or its low-cost nature does not allow the 3D sensor deployment, therefore only the above name three platforms will be discussed in this thesis.

3.3.1 Focus in the Laboratory of Telepresence and Robotics

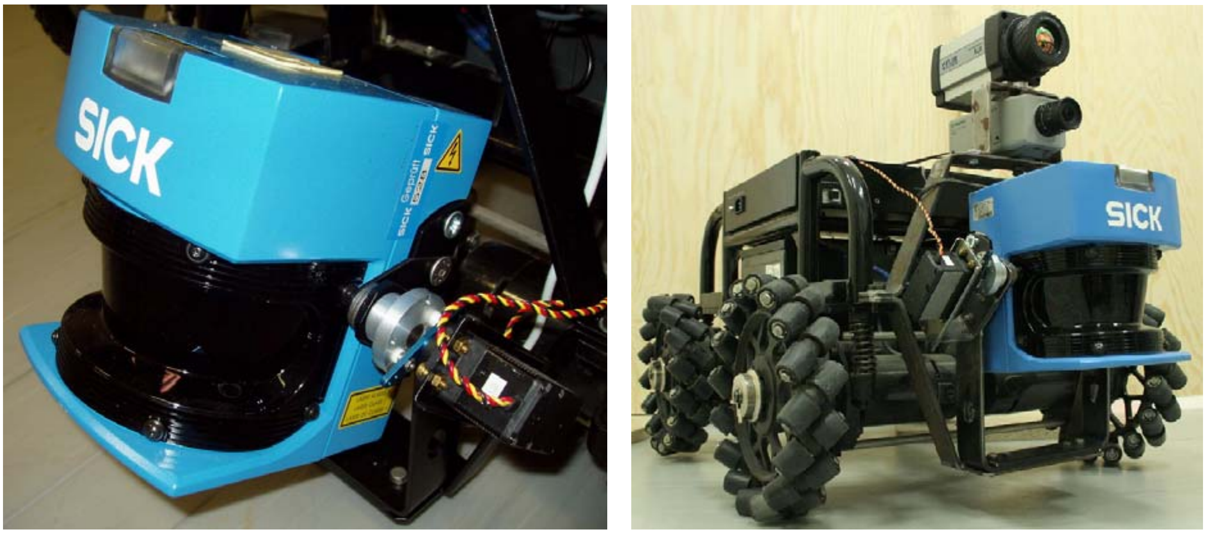
Teleoperation and fusion of sensors

In the author’s workplace - the Laboratory of Telepresence and Robotics in Department of Control and Instrumentation, there is a long tradition in the development of mobile robots focused both on both teleoperation and the development of methods for the robot’s autonomy. The teleoperation of a robot is a feature which is very practical: the research in teleoperation mainly focuses on methods to make the operation of a remote vehicle as easy as if the person himself/herself was present at this remote place. This implies different methods of displaying the information from the sensors such as cameras, range scanners, thermo cameras etc. to the operator. Data fusion of various data (namely array data like CCD, thermo vision and range) is very important and also a simple way of controlling the mobile robot has to be implemented. Localization and mapping methods are primarily intended to provide information necessary for the autonomy of the robot, though as a side effect, this information can be used in the teleoperation regime. For example a person operating a robot in an unknown environment supported by a view of this environment from a bird's perspective (provided by a map which was created using a SLAM method) has much better chances to avoid obstacles and to find his/her way out. The development of localization methods in the LTR, so far only in 3DOF, was always used as a support for the teleoperation systems developed at the workplace. Therefore although the system for the autonomy equipped with a 6DOF SLAM system (missing path planning and mission objectives planning algorithms) is incomplete, it can be already useful for the support of the teleoperation. Both areas of focus in the LTR are therefore coherent and the methods for navigation can support teleoperation of robots and vice versa, assuming the operator can correct errors in the navigation systems. In longer term the aim of our department is the progressive implementation of autonomous methods, taking the burden of operation from the human operator. This would only leave the mission planning work on the operator and also help to ensure safe operation of missions e.g. by letting the robot to return to its original position in case of signal loss.

3.3.2 Hermes: omnidirectional robotic platform

Hermes's platform

Hermes is a robot based on an omni-directional driving mechanism using Mecanum-type wheels. The Mecanum wheels are designed in such way that the vehicle equipped with them can move in a conventional way and also, if the opposite wheels are rotating in reverse directions, it can move sideways. The wheels have a series of rollers attached to its' circumference. These rollers are oriented so that their axis of rotation is at 45° to the plain of the wheel.



a) b)

Figure 9: Hermes robot equipped with the SICK laser rangefinder and a servo-driven inclining mechanism; a) detail of the inclining mechanism; b) the robot

Robot's use

The main use of this robot is currently in experiments. Since it allows movement in 3DOF, the 3DOF SLAM methods developed in the LTR were evaluated using this robot. Hermes is equipped with three main sensors: a CCD camera, thermo vision camera and the SICK LMS 2D laser rangefinder. The robot is displayed in Figure 9 where a first 3D inclining mechanism developed in the LTR can be seen in detail. The actuator here is a digital servo which has a desired angle value as input. The precision of the digital servo is relatively high but the principle of its operation is limited to the stop and go regime. Therefore data acquisition is very time demanding (the approximate duration of taking one 3D scan of 180 lines is about one minute). In the most recent series of experiments using this robot, data fusion method using all three sources of information about the environment was successfully evaluated and further optimized [37][38]. To sum up, Hermes's inclining mechanism is the limiting factor for the implementation of the accelerated 6DOF SLAM method since the data acquisition is very time demanding.

3.3.3 U.T.A.R: Universal Telepresence and Autonomous Robot

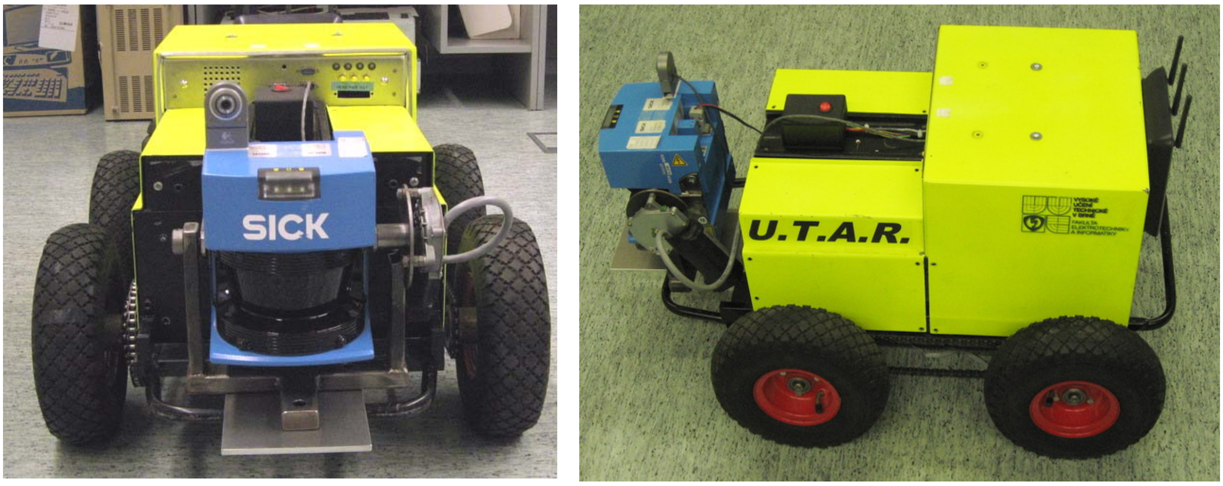


Figure 10: The U.T.A.R. robot with continuous inclining device

U.T.A.R.

The U.T.A.R. robot is an older experimental robot in the LTR and this robot recently went through a major hardware innovation. Its major role in the future is experimental application of new autonomous navigation methods. While the robot is robust enough to carry heavy sensory equipment including multiple laser rangefinders, powerful computer and heavy batteries. Also, compared to HERMES, this robot is suitable for operations in light terrain.

Innovation of U.T.A.R.

As a part of this research, the robot was recently innovated including the development of a new laser rangefinder pitching device, using a dc motor and a driver in order to operate in an angular velocity regulating regime (as specified in the previous section). In the second part of the innovation, an industrial PC based on mobile Pentium processor was integrated into the chassis. This processing and control station was selected so that it is powerful enough to process three dimensional data and to perform 3D image registration. It is also capable of setting, controlling and reading from all sensors and controllers in real time. The station is equipped with multiple serial and USB links. A wireless access point equipped with MIMO technology (using Airgo Networks True MIMO chipset) was also implemented in the robot, taking into account the reflecting effect of the walls in the indoor applications and extending the transmission range. Figure 10 displays the U.T.A.R. robot with the new inclining device attached. The robot is also equipped with a portable inclinometer, namely Analog Devices ADXL 202 MEMS dual axis accelerometer, and a RS422 communication module for a

500 kbaud communication with the SICK sensor.

Technical parameters

The technical parameters of U.T.A.R. are summarized in Table 3. The robot was accustomed for the 6DOF SLAM data acquisition and navigation.

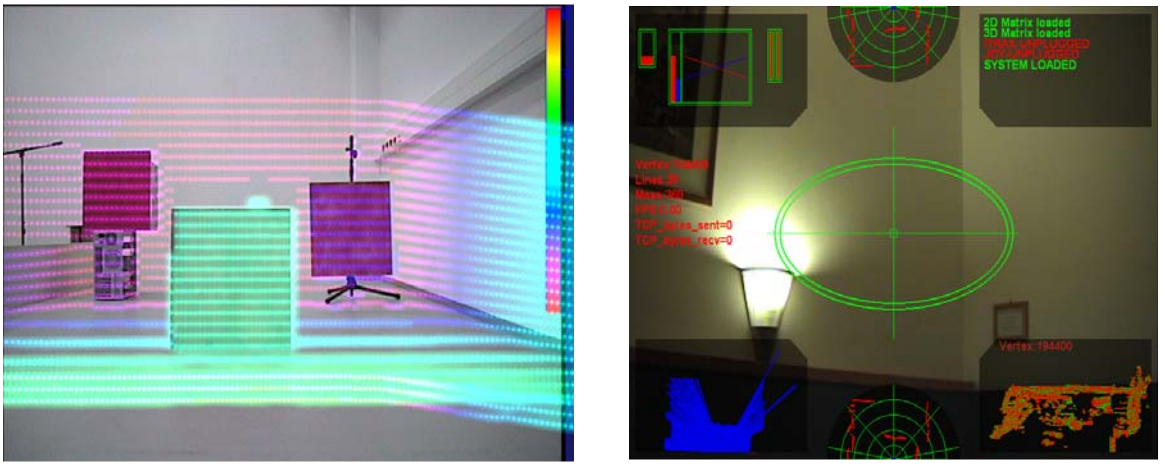
Table 3 : Universal Telepresence and Autonomous robot's parameters

Characteristic	Parameters
Platform	Skid-steer
Areas of use	interior, exterior (light terrain)
Dimensions	500x600x500 (mm)
Highest traversible obstacle	100mm
Maximum rise angle	25°
Maximum velocity	2km/h
Whells radius	260mm
Total weight	50kg
Communication subsystems	WiFi, RS232&422, USB, FireWire
Processing Station	Pentium M based embedded PC
Driver	Controllable H-bridge based digital dual driver
Motor	Two Portescap 200W motors with harmonic transmission

3.3.4 Orpheus: the rescue and reconnaissance remote-operated robot

Orpheus

Orpheus is the rescue robotic platform in the LTR. The objective of this robot's operation is primarily surveillance of remote environments. Therefore the main objective is telepresence and remote operation support. The navigation systems are implemented into this type of robots only if these prove to support the surveillance and rescue operations in unknown environments. The robot's hardware currently includes a Hokuyo PBS planar triangulation range scanner, which will soon be replaced by a new and more precise Hokuyo URG version. This sensor is inclined using a small servomotor; the advantage of both the sensor and the actuator is that these are very light and small enough not to increase the robot's weight and interrupt the surveillance operations. Also inclinometers and cameras and implemented. The onboard computer is also a Mobile Pentium based industrial board, capable of performing demanding computation tasks [37].



2D Matrix loaded
3D Matrix loaded
JOY:UNPLUGGED
SYSTEM LOADED
TCP_bytes_sent=0
TCP_bytes_recv=0
Vertex:194400

a) b)

Figure 11: Screenshots from the ARGOS system, a) with the point clouds directly fused with the camera image (other virtual displays are disabled) and b) with 3D scans included in the virtual HUD displays

Controlling SW platform

Apart from the sensory equipment fitted in the Orpheus-type robots, these are also equipped with a special software platform, under which all teleoperation and surveillance tasks are to be programmed and executed. The operation system which is currently being used in Orpheus and Hermes robots and in their control units is called ARGOS (Advanced Robotic Graphical Operating System). The innovation of the U.T.A.R. robot also allows the application of this system, thus all three robots may be controlled by a uniform operating and controlling system.

ARGOS 3D view imlementation

The currently implemented 3D-data vision in Hermes robot is utilized to support the teleoperation of the robot, giving the operator a clearer information about the surrounding obstacles and showing a three dimensional scan in the virtual HUD viewing regime. This situation where the three dimensional visualization support is activated in a separate HUD is shown in Figure 11 b). Also a screenshot from the ARGOS system where the 3D scanned point clouds are directly fused into the camera image is shown in Figure 11 a). It is especially in very complicated environments where there are many objects scattered in the robot's view where the 3D maps can enhance the safety of teleoperation and decrease the reaction time of the operator [38].

Possible applications of the 3D navigation subsystem in teleoperation

When the 3D navigation and mapping method is successfully tested with the operation system ARGOS and interference with the quick telepresence response time is eliminated (e.g. setting the localization process priorities to the minimum), the system can also support the teleoperation by not only showing the currently acquired 3D data, but also by building overall 3D maps of the environment and implementing collision warning systems during telepresence operation. Afterwards, the 6DOF SLAM and mapping could be further used for the development of autonomous mapping and navigating operation regimes, which will also require path planning task to be solved. The task of this thesis is though only to develop the 3D self localization system in such way that this could be successfully implemented in either the robot's and the operator's processing stations.

Path planning using 6DOF SLAM method

The path planning aiding system for the remotely operated robots could be a very effective method for making the teleoperation decisions easier and thus accelerating the reconnaissance operations of the robot. Such path planning algorithms could use the 3D scans to build traversibility maps and to find a low cost path to a place of interest.

4 Original Implementation of the 6DOF SLAM Algorithm

6DOF SLAM constraints

The objective of this research was to develop a method for Simultaneous Localization and Mapping of unknown environments using 3D range measurements. It is a crucial step towards the autonomy of the robot. The task is to create an organized and consistent model of the environment and to place the robot based on the most recent measurements into this system. Such task is very complex and there are many constrains in the overall process. First problem was encountered when designing the system for the 3D data acquisition; it is a time demanding process when such amount of measurements is to be taken. Also, the robot also has to stop its movement before starting the scanning as we are using a planar scanner to acquire 3D images. The next step – performing the SLAM algorithm, is even more complex: multiple 3D images are to be registered in order to form an overall model of the environment. Such registration naturally requires comparisons between the 3D scans which are represented by thousands of points. Andreas Nuchter et al. from the Fraunhofer AIS use the abbreviation 6D SLAM for this process, but the obvious connotation of 6D is six dimensions instead of correct meaning “six degrees of freedom”. Therefore in this thesis the abbreviation 6DOF SLAM is being used.

Objectives of the method

Although the existing methods are robust and were already accelerated by other research teams focusing on different parts of the registration algorithms, the task of image registration was perceived in this research as yet not completely resolved. The reason was especially the fact that the most recent CCD range sensors are able to acquire the 3D images with such speed that even when using the most powerful and task dedicated computing hardware, the localization process could not be solved with a reasonable computing effort. The localization task is solved in 6DOF and the overall registration process is very time-demanding. Therefore the main objective of this thesis was to develop such approach to the registration of multiple 3D scenes that the time-wise performance of the existing methods could be improved.

Robustness of the method

Before the method for the localization and mapping will be introduced, it has to be noted that often the improvements in existing methods are environment specific and that the results and improvements are thus limited to certain circumstances. If the task of SLAM was perceived as a minimization of certain error (or energy) function, the acceleration of such minimization often does not do without the drawback on the side of the method's robustness. In image processing, this could be referred to as data reduction and semantic analysis based optimization, which under extreme circumstances could lead to the unprecedented results where more robust methods based on unreduced data use could perform better. The second objective of this research was therefore to create such acceleration method which would not disturb the overall robustness of the existing methods and the results of which would be comparable to the results of currently existing SLAM methods.

Determination by the technologies

Every theoretical method's evaluation and applicability is limited by the current state of technology. The data which are acquired and the computing power which is available to certain extent determines the focus and also success of each navigation method and in fact it is the technological drive that stimulates the research to develop new and more efficient methods to process and utilize the data acquired by the newest sensing devices. This corresponds with the criteria related to applied research and in this way the research in the LTR is no different. The method developed in this research is also motivated and in certain way also limited by the character of the data which are acquired from the 3D ranging system. For example if ultrasound range measurements were substituted for the laser rangefinder measurements, the registration of such images would require a probabilistic approach since multiple reflections would occur more often and also the error of the measurements would be much higher compared to the system used in this research. This does not imply that the method described further is not suitable for any other ranging system than laser range measuring. The method would though have to be modified and uncertainties would have to be introduced in the system in order to preserve the robustness. There are also other aspects than the determinism of the measurements which relate to each technology applied in robot's navigation. Fortunately most researchers today are using laser scanners to measure distances and therefore the community in which the research contributes is quite large. This is especially due to the already mentioned deterministic character of the currently available laser rangefinders while the main disadvantage of the system – the time required to obtain a 3D scan – is often assumed to be a limitation of current state of technology which is to be resolved in further developments.

Chapter structure

This research was in general divided into two main phases: first, a method for the 6DOF SLAM had to be developed based on the recommendations which appear in the literature published from worldwide research. In the following phase, this method was to be further developed based on the found constraints of method's performance. In this chapter, the original method developed in this research will be introduced and it will also be related to the existing research and applications. This introduction is a natural way of presenting this research as the extensions of the initially developed method are based on the foundations of the originally implemented method. This chapter will start with a description of the original version of registration (SLAM) method which was developed in the first part of this research. The different processes which form the original method will be introduced.

4.1 Structure of the Initial SLAM Method

Chapter outlook

In the 6DOF SLAM algorithm, one can identify important separate processes which lead to a successful performance of the registration of multiple 3D range data sets. Although the SLAM in 6DOF is a very complex task, the implementations often follow a similar pattern and the further described original method implemented initially in this research is no different: it in fact represents a classical pattern often seen in general image processing and optimization implementations. The processes in the method will be briefly introduced in this chapter, focusing on the main principles. The core of the algorithm – Iterative Closest Point algorithm – will be discussed in detail in the following chapter. All parts of the original method will though be shown in more detail in the chapter focusing on the further modified navigation method.

Basic 6DOF SLAM process

In Figure 12, the original approach to the simultaneous localization and mapping in 6DOF is illustrated in a block diagram. The overall SLAM process is traditionally divided into the following steps: first, data acquisition takes place, obtaining the range measurements as raw data and communicating these to the processing unit. The data then have to be preprocessed and for the purpose of reducing the number of overall operations in further steps these are also reduced. After the data reduction, the altered data set is usually passed into an initial estimation algorithm. This process uses other external measurements to pre-align the data with the existing 3D model. After this, the main process in the 6DOF registration is performed, using a special algorithm for minimizing the difference between the data set and the environment model. After this is done, the model is updated with the aligned data set and therefore the map or environment model is extended.

Possible extensions to the general scheme

Such framework for the 6DOF registration is usually sufficient for the localization and mapping, but can be extended in a number of ways. Two ways to extend the SLAM algorithm are mentioned in the introduction to the simultaneous localization and mapping in chapter 2.4. First interesting area which is currently being underestimated in different researches is the semantic analysis of the 3D models: this semantic analysis could not only provide information about the environment and extend the existing maps but it could also aid the registration process itself. Another area which could be interesting is data fusion with information from different sensors such as color measurements. Such fusion could also contribute to the registration and enhance the global map, although in general it would mean adding other dimension(s) to the existing data.

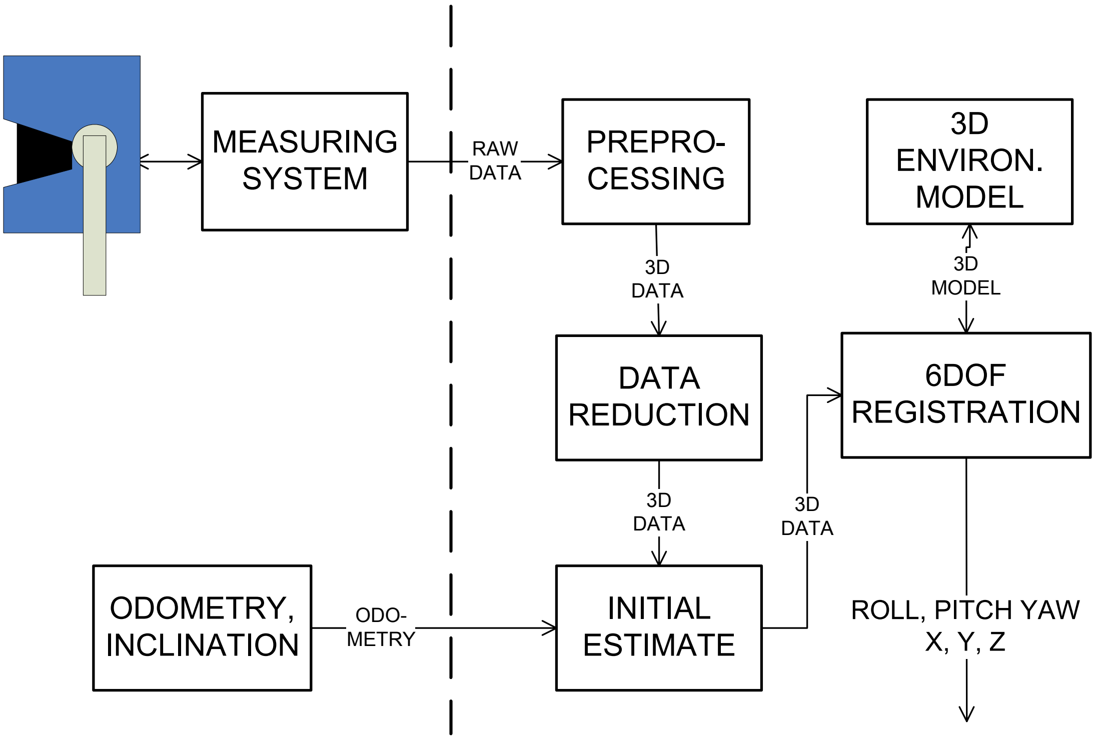


Figure 12: A block diagram of the Simultaneous Localization and Mapping process in six degrees of freedom

6DOF registration

The particular steps of the above shown SLAM process will be further discussed in the section appointed to the further extended and developed navigation method. Nevertheless, there is one area which needs more attention prior to the extended navigation method description: the 6DOF registration process. This process is in fact the core of the 6DOF SLAM algorithm and in the worldwide researches there is one major method being used for the registration of 3D range images: the Iterative Closest Point algorithm, a method proposed by Besl and McKay in 1992 [6]. This algorithm was also implemented in the first phase of this research and it is presented in the following chapter.

4.2 Iterative Closest Point Algorithm

Computing constraint

As already mentioned, the image registration is from the computational point of view the most complicated task of all processes in three dimensional mapping and model building. In order to align multiple datasets, the algorithm naturally requires processing of all data points in an organized way; since the sets in case of 3D images contain thousands of points, this process could be very time demanding.

Iterative Closest Point Algorithm

The Iterative Closest Point (ICP) algorithm is an iterative aligning algorithm, as already mentioned it was first proposed in 1992 by P. Besl and N.D. McKay [6]. The application of this algorithm in 6DOF SLAM by Fraunhofer AIS institute is very comprehensive and it inspired the development of the original SLAM method to a great extent [20]. The algorithm serves to merge new scans into one – reference – coordinate system. Let's assume that we have a data set *D* as it is defined in the equation (15) (chapter 2.4), which is partly overlapping with the existing model *M* also as defined in equation (15).

As stated in Chapter 2.4 the criteria function which is to be minimized is a function $E = f(R,t,M,D)$. In the case if ICP, this function is expressed in equation (17), where $w_{i,j}$ is 1 if the point d_j in *D* describes the same point as m_i in *M* (the points are corresponding), otherwise it is set to 0.

$$E(R,t) = \sum_{i=1}^{m} \sum_{j=1}^{n} w_{i,j} \left\| m_i - (Rd_j + t) \right\|^2 \qquad (17)$$

Singular Value Decom-position

In each iterative step, the algorithm selects closest points as corresponding and calculates the transformation (*R,t*) according to the minimization of *E*(*R,t*).

There are four known methods for the minimization of this function. Easy to implement are quaternion based method and singular value decomposition method (SVD). Both are applied by Fraunhofer AIS, in later work SVD is preferred due to its robustness and an easy implementation [20]. The algorithm supposes decoupling of rotation *R* from translation *t* which is done by using centroid points given in equation (18).

Decoupling of *R* from *t*

$$c_m = \frac{1}{N}\sum_{i=1}^{N} m_i \qquad c_d = \frac{1}{N}\sum_{j=1}^{N} d_j$$
$$M' = \{m'_i = m_i - c_m\}, D' = \{d'_j = d_j - c_d\} \tag{18}$$
$$i \in N, j \in N$$

Minimization of *R* and *t*

In order to minimize the rotation matrix *R,* a matrix *H* has to be created. This matrix is defined as in equations (19) and (20). Its elements $S_{xx},...,S_{zz}$ have to be calculated by multiplying the corresponding model and data points (these are the closest points). The H is calculated as in equation (20).

$$S_{xx} = \sum_{i=1}^{N} m^{'}_{ix} d^{'}_{ix}, S_{xy} = \sum_{i=1}^{N} m^{'}_{ix} d^{'}_{iy} \tag{19}$$

$$H = \sum_{i=1}^{N} m'_i \cdot d'_i = \begin{pmatrix} S_{xx} & S_{xy} & S_{xz} \\ S_{yx} & S_{yy} & S_{yz} \\ S_{zx} & S_{zy} & S_{zz} \end{pmatrix} \tag{20}$$

Correlation matrix *H*

After obtaining the matrix H, it has to be decomposed e.g. using singular value decomposition method:

$$H = U \Lambda V^T \tag{21}$$

The final *R* is calculated in equation (22), where the matrices *U* and *V* are obtained by a singular value decomposition of matrix *H.* Final translation *t* is derived from equation (21).

$$R = VU^T \tag{22}$$

$$t = c_m - Rc_d \tag{23}$$

Computing the

The most time and computation demanding task of the ICP method is the determination of the closest points necessary for calculating the elements of

correspondences

Tree based approach

matrix *H*. Using brute force, this is a task of searching for the closest points and its complexity is O(*n*) single steps given the number of points *n*; the total processing time is in the order of tens of hours when using current computers. Though there are methods to speed up the search of corresponding points, commonly used in 3D graphics programming. Majority of them is based on the structuring of point clouds using planes, thus creating a tree structure. Nuchter et al. are using octrees, box decomposition trees and kd-trees for accelerating the search [22]. The kd-trees are a generalization of binary trees, where every nod represents a partition of a point set to the two successor nodes. Nuchter et al. have optimized kd-trees, so that the buckets are being chosen in such a way that the separating planes create two subgroups of equal amounts of points and the planes are also perpendicular. The algorithm of searching for the closest point is iterative, the point is compared to the separating plane and the decision about the direction of searching is determined from this. The implemented kd-tree acceleration reduces the search space and this can be expressed in the big O notation as $O(n^c)$ where $0 < c < 1$.

Iterative character

After estimating the optimum rotation of the data set, the centered data points are rotated using this matrix and the iterative closest point algorithm is run again. The optimum state is reached when the rotation matrix output from the SVD or quaternion based evaluation is close to ones matrix, therefore the data set is not to be transformed any further. Therefore the iterative character of the ICP is in the iterative running of the process to find the rotation matrix which is used for rotating of the centered data. The method finishes when the output rotation is not transforming the data any further and this state is expected to be the optimum.

Closing the loop

Finally, when all scans are merged into a single coordinate system, the errors of the single scan matches will cause a problem when the last scan overlaps with the first scan and the integrated errors will prevent the last scan from correct placement. This problem can be solved through uniform error redistribution, which eliminates major inconsistencies in the scanned scene. A solution to this was proposed by Nuchter et al. and Pulli et al. [24][23]. There also exist other solutions based on virtual mechanical interconnections between images. A simple approach is iterative reregistering of the whole scene in an organized way, so that the errors are evenly distributed.

4.3 Implementing the Closest Point Search

Search in the ICP process

Determining the closest points in order to build the matrix H correctly is critical for the computing-time performance of the method. The state of the art in implementing the ICP as a core of the SLAM method favors the tree based methods to reduce the search space for each data point in the model point space.

Nearest neighbor specification

The suggested kd-tree acceleration method incorporates a tree structure which is built for one of the matching point sets. A k-dimensional tree is a space-partitioning data structure which serves for organizing and searching in a k-dimensional space. In order to specify a kd-tree in terms of the nearest neighbor search, let us first define the nearest neighbor search as A. Moore [18]. Say we have two multi-dimensional spaces *Range* and *Domain* defined as in equation (24):

$$Domain = \Re^{k_d} \qquad Range = \Re^{k_r} \tag{24}$$

Let the exemplar be a member of Range x Domain, the exemplar-set E is a finite set of exemplars and a target domain vector is d. The nearest neighbor of d is then any exemplar $(\mathrm{d},\mathrm{d}') \in E$ so that it satisfies NN(E,d,d‘) where NN stands for nearer non-existing. The *NN* is defined in equation (25). Please note that the “is equal or less” condition expresses the possible ambiguity of the nearest neighbor search since there could be multiple points with the same distance from the target domain vector *d*.

$$NN(E,d,d') \Leftrightarrow \forall (d'',r'') \in E : |d-d'| \leq |d-d''| \tag{25}$$

Kd-trees

Kd-trees are a special form of binary space partitioning search trees which also serve for recursive subdividing of space into convex sets by hyperplanes. The difference between the binary space partitioning trees and kd-trees is that kd-trees only use splitting planes which are perpendicular to one of the coordinate system’s axes in contrast with the arbitrarily set splitting planes for binary space partitioning trees. The kd-trees are therefore simpler to construct [18].

Partitioning principle

Every node in a kd-tree is therefore partitioning the space into two successor nodes. The splitting plane is led through one of the data points. The data set is therefore represented by the root of the kd-tree and since the two partitions are disjunctive, the two successor nodes are representing two disjunctive partitions of the point cloud. The last nodes in the tree (that is the highest nodes at the highest branches) are called leaves and these can contain one point as the part of the partitioning plane plus possibly multiple points which are contained in the created further not split cell. The number of points contained in the leaves is called the bucket size; the leaves which form a disjunctive partition of the space are thus also called buckets [20]. The representation of the set *E* in a kd-tree is the set of nodes. Each node is therefore representing one exemplar, the index of the node in the kd-tree which is represented by the coordinates of the point contained in the splitting plane is actually the domain vector *d*. This index also defines the splitting plane. The tree could be defined by definition of a mapping *exemplar_rep* which maps the kd-tree into the exemplar set as the following equations [18]:

$$
\begin{aligned}
&exemplar\_rep : kd\_tree \to E \\
&exemplar\_rep(empty) = \varnothing \\
&exemplar\_rep(d, r, -, empty, empty) = \{(d, r)\} \\
&exemplar\_rep((d, r, split, kd\_tree\_left, kd\_tree\_right) = \\
&= exemplar\_rep(kd\_tree\_left) \cup \{(d, r)\} \cup exemplar\_rep(kd\_tree\_left)
\end{aligned}
\qquad (26)
$$

Building the tree

The tree is built by partitioning the space contained in the area belonging to the currently processing node, always splitting the remaining space by creating a splitting plane perpendicular to the plane which demarcated the space creating the parent node. The position of this splitting plane is set according to preset criteria, which will be discussed further in the text; the splitting plane should though pass through one of the data points. The planes are also aligned with two of the three orthogonal dimensions in the space giving three possible plane orientations. Which one is determined by the specified order in which the planes are cycled: as one moves through the tree which is being built, the splitting plane is cycling through available splitting planes of the space. In 3D space these would be the plains aligned with the *x* and *y* axes, *y* and *z* axes and finally *x* and *z* axes.

Positioning the splitting planes

One can set a rule which determines where the splitting plane is positioned on the splitting plane of the node one step above in the tree hierarchy. A common way to place the splitting plane is to ensure that the plane divides the point cloud contained in the space of the parent node into two subsets containing approximately equal number of points. Therefore the splitting plane is placed in the median (in the splitting dimension) of the contained points. D. Mount also proposed other options in the splitting rules and selecting the splitting plane: a *standard kd-tree splitting rule* is using the median point split placement criteria, although the splitting plane alignment is selected according to the highest spread of points in the dimensions [19]. Another possible rule which shows very promising results is the sliding midpoint splitting rule: according to this rule, the cell is cut along the longest side and the cut is done in the midpoint of the cell as long as each part of the split cell is containing at least one point from the data set. If one of the created cells would be empty, such cut is called trivial. In this case, the splitting point is moved along the given dimension so that the created cells contain at least one point. A tree is called a balanced tree, if the distance of all leaves from the root point is approximately equal. Also if the cycling of the splitting plane orientation is enforced, the kd-tree is in the canonical form [10]. In general the median splitting rule generates a balanced tree, while the sliding midpoint splitting rule is generating an unbalanced tree which is very similar to the box-decomposition tree. The space is though divided to cells much more uniformly than in canonical kd-tree and trivial cuts are avoided using the sliding principle [19].

Searching the tree

The search performed in the kd-tree is performed recursively in the following fashion: the search starts at the root of the search tree. It has to be determined to which way the search is to continue by comparing the queried point's coordinate to the splitting position of the given splitting plane axis. This determines which way does the search continue and finally a leave (a bucket) of the tree is reached. The possible problem here is that the queried point can be further from the points contained in the bucket than from the boundary of the partition of the space contained in the bucket. In such a case, backtracking has to be performed and neighboring buckets have to be checked. The number of neighboring buckets to be checked can be limited by searching only those neighboring buckets whose distance to the queried point is smaller than 1/(1+ε) where ε is a positive real value. E.g. when setting ε to 1, only neighboring buckets which are closer than half of the distance to the closest point in the current bucket are checked. This limits the number of buckets to be backtracked and therefore significantly increases the speed of the search. The cost is a limited probability that the found closest point is actually the closest one [19].

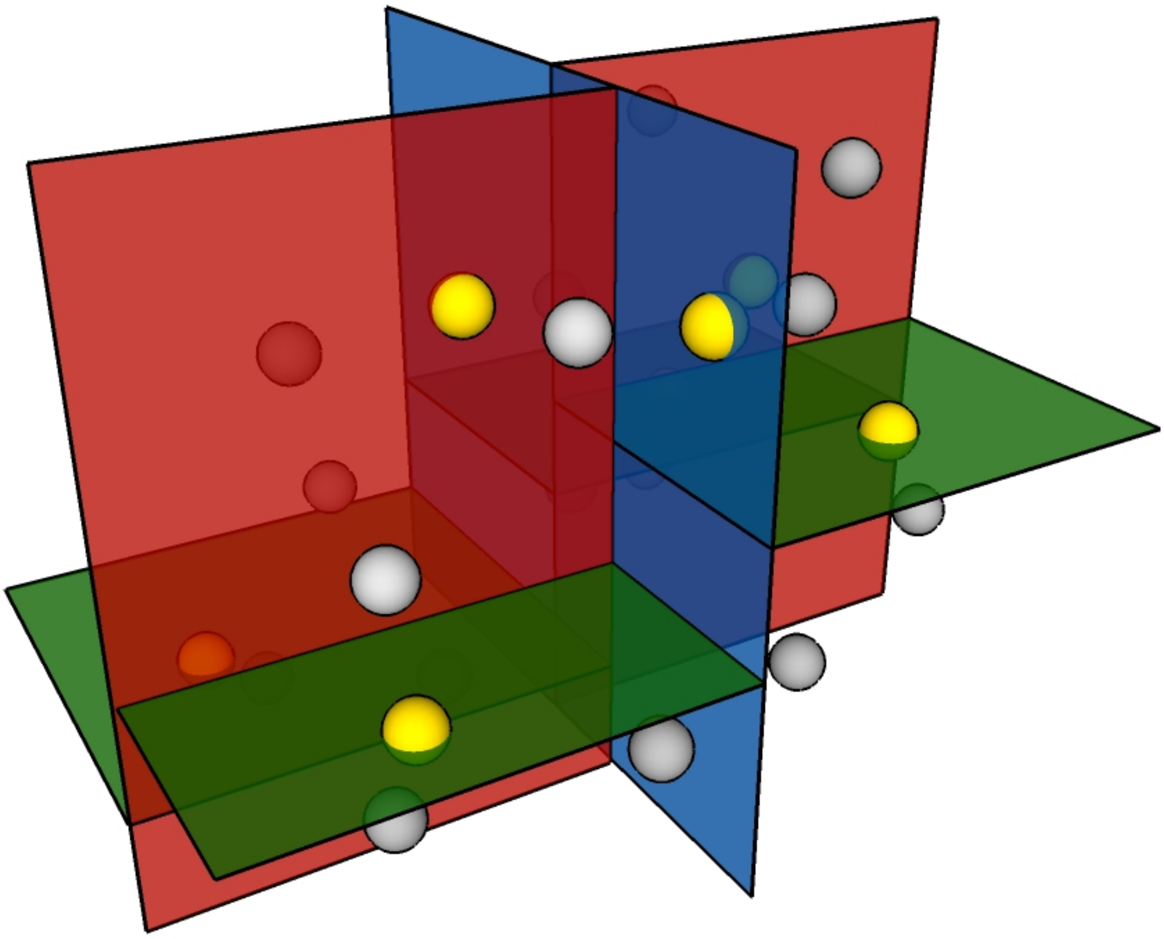


Figure 13: A kd-tree built in a three-dimensional space (splitting planes in the tree hierarchy: first level - root node – blue; second level nodes – red; third level nodes - leaves – green)

Illustration of a kd-tree

A kd-tree built on 3D data with bucket size of three points is illustrated in Figure 13. The yellow data points actually form nodes of the tree (these determine the splitting planes) and the grey points are bucket points – these belong to the lowest level leaves - buckets. In this figure, one can see the dividing planes in the following hierarchy: blue (root), red, green. Each cell which is not further split contains at most two points from the point cloud plus the one which formed the dividing plane. As can be seen in the figure, the presented tree is a balanced type of tree.

Efficiency of search

The efficiency of the tree depends upon the number of nodes which are to be inspected when performing a nearest neighbor query. This number of nodes is not constant; the search could be terminated if the tree is unbalanced and the point lies in the bucket of a node which is closer to the root. The search performance also depends on whether backtracking has to be performed which is always dependent on how close is the query point placed with respect to the bucket cell's splitting planes [20].

Implementation

The nearest neighbor search implemented in this research to large extent uses the Approximate Nearest Neighbor library written by D. Mount and S. Arya [19]. Kd-trees are built for the nearest neighbor search acceleration and various splitting and building rules were tested. Although kd-trees are used, the main contribution of this thesis was not focused mainly on the acceleration of the ICP nearest neighbor search but on the overall performance of any existing ICP based method, whether or not these use kd-trees to reduce search space. The selection of the kd-tree and the appropriate rules will have impact on the improvement of the overall performance in absolute number. Though the relative ICP based SLAM performance improvement which is the main objective of this thesis can be independent on the selection of the nearest neighbor search performance. Regardless of this note, accelerating the search is clearly addressing the computing time reduction issue.

4.4 Decomposing the Matrix H

Pre-processing of the data

As stated in the ICP description, one way to obtain the optimum rotation is to decompose the matrix *H* using singular value decomposition. The results of such decomposition of a m-by-n matrix *H* are three matrices given in equation (21): a unitary m-by-matrix *U* containing a set of orthonormal output basis vector directions, m-by-n matrix *Λ* which is the matrix of singular values, containing only nonnegative elements in the diagonal which are the singular values of *H* (other elements are zeros), and a n-by-n matrix V^T containing a set of orthonormal basis vector directions of M called inputs. The singular value decomposition is a factorization method which can be applied to both real and complex matrix and has many applications in statistical analysis and signal processing [31].

Note on conjugate transposition

In case a complex matrix is being decomposed the V^T matrix should rather be noted as *V**, since the transposition is in fact a conjugate transposition, meaning that the complex element of the V matrix is obtained by taking a complex conjugate of the elements in the simply transposed matrix. This should not be mistaken with an adjugate matrix which is noted as adj(V). The fact that the V matrix is in the form of conjugate transposition is however often neglected in SLAM applications since the *H* matrix we are decomposing is not complex; therefore common notation in robotics literature is V^T.

Non-uniqueness of the SVD

It should be noted that the singular value decomposition is not unique; therefore multiple valid singular value decompositions may be computed.

SVD implementation

In this research, the implementation of the SVD numerical algorithm is a subroutine of the LAPACK Linear Algebra Package, originally developed for Fortran programming language but translated to C++ as a part of ALGLIB library. The matrix that is being processed in the localization task is always square (2x2 in case of 2D matching, 3x3 in case of 3D matching). LAPACK as well as many other technical tools (e.g. also in Matlab) is using QR algorithm for computing the singular value decomposition [17]. This algorithm was first implemented and described by J.H. Wilkinson and is based on repeated use of QR factorization. The factorization mainly serves for solving of over-determined systems in the least squares meaning. The C++ implementation is fast and if we take into account that it is being evaluated only once in every iteration of the ICP algorithm, the determination of the optimum rotation matrix does not introduce any significant limits to the method.

5 Extending the Original Implementation: Two Stage Leveled Map Accelerated 6DOF SLAM

Chapter structure

After introducing the original method, this chapter will mainly focus on the Author's further extensions of it. First, an insight into the bottlenecks of the original method and the analysis of the quality based concerns will be presented. This will be followed by the description of the proposed modified mapping and localization method, relating it to the initially described original algorithm and discussing the implications of the changes made.

5.1 Identifying the Bottlenecks of the Original Method

Areas requiring attention

Before proceeding to the structure of the innovated navigation method, the areas in the original navigation method which require further attention should be discussed. The first problem which was already dealt with in previous chapters is the data acquisition: a platform for fast 3D range data scanning was developed and successfully tested with the robotic system.

Time wise optimization

After data are acquired the preprocessing and data reduction is performed. Such preprocessing and data reduction follows usual imaging patterns - e.g. filtering the noise in the data, removing unwanted objects etc. Certain computing effort is required though this cannot be expected to make a bigger share of the overall computing time. This is followed by a non-compulsory initial estimate based on other sensor's data, which is the application of a rotation and translation based e.g. on the odometry of the vehicle. Here again the computing effort is not expected to be critical. The final part is the Iterative Closest Point registration which in fact is a bottleneck of the method.

Operation regimes of current implemen-

This can be also noticed in other researches: the navigation has to be supported either by different methods (Thrun et. al.) or the robot is teleoperated or operated by a human, while the actual 3D map building and 6DOF SLAM is computed offline when processing the collected data (Wulf et al.) [33][35]. There could also be other reasons for this mode of operation

tations

(such as mission and path planning algorithms) but the required computational effort certainly contributes to the selection of such operation regime. There exist teams worldwide which use the 3D acquired data for finding the best path, though the localization in the map is done using different methods and the 3D maps are not merged until offline processing [33][35].

Further optimization of the ICP

In the registration of two range images, data points are iteratively used to find a corresponding point in the model set, which is obviously the most challenging part of the ICP algorithm. The search can be accelerated by limiting the search space using the tree structures or by projections of data points to model surfaces. Accelerations of other parts of the ICP algorithm are less likely since the estimation of the optimum rotation matrix *R* using either a quaternion based method or the singular value decomposition method requires computing time of approximately one to two lower orders. From the computer graphics point of view – not taking the physical characteristics of the environment into account - further optimization of the ICP method seems difficult.

Number of iterations reduction

The time required for the optimization is very close to be directly proportional to the number of iterations of the ICP algorithm. Another way to approach the improvement of the method would be to reduce the overall number of iterations. This could be done in two ways: first way is to ensure that the images are very close to the optimum match; the iterative closest point algorithm has to perform less iterations and this could save the processing time. Such condition could be ensured only when pre-aligning the matched data using a different method. Another solution would be a complex and robust data reduction. Data reduction would though introduce system error since the registration would be performed on the simplified data set, neglecting details in the scans. The first mentioned way to improve the registration is in fact much more attractive especially if it wouldn’t require extensive additional inputs from the environment.

5.2 Quality of the Registration and Mapping

Quality of the results and errors

Prior to advancing to the extension of the original method which was implemented as part of this research, attention also has to be given to the quality performance of the algorithm and the identification of potential risks in this area. The quality of the SLAM method is reflected by the error of the localization. The introduced error is in case of the map building very important since the errors are additive. When building a 3D map using multiple environment range scans, the error could build up so that the overall scene is inconsistent.

Sources of errors

The SLAM ICP based registration algorithm has multiple sources of errors: first source is the measurement and the characteristics of the sensing platform which were already discussed in chapter 3.2.2 when discussing the measurement's reliability. As stated in the chapter, the SICK LMS sensors are very reliable and the characteristic of the measurements is close to Gaussian distribution. Therefore most of the errors in the localization will be contributed from the registration method itself.

Preprocessing and data reduction

The preprocessing part – especially when filtering the acquired data - also contributes to the overall error although it can in fact smooth the data and help the registration. If data reduction is used, the influence on the overall quality of the matching algorithm strongly depends on the fashion of reduction. Nevertheless data reduction clearly contributes to the reduction of information and in this way it could also contribute to the decrease of the quality performance of the method.

Quality of the ICP match

The core of the method – the registration algorithm ICP – has a significant influence on the quality of the localization. The efficiency of the algorithm's determination of the rotation matrix and translation vector is critical for the precise localization. The optimum solution is guaranteed by the Iterative Closest Point method when working in ideal conditions: when each point in the data set has a corresponding point in the model set and these points are identical. Any difference between the acquired data set and the model set introduces error: ideally this error is compensated if the coverage of both sets is identical, only points are distributed differently but with identical density over the measured surfaces. This condition is though often not fulfilled since there are occlusions in the images and also the overlaps of the datasets are only partial, therefore data set contains data of other real objects measurements than the model set and vice versa.

"Catch 22" paradox

If there are overlaps and occlusions in two data sets, we can determine these only after we know the transformation that matches these sets. The problem is that in order to match these sets, the corresponding points have to be determined and ideally (to ensure close to zero error) when the optimum is found, these closest points should be measurements of identical points in real world. This is only true when occlusions are removed and only overlapping part of sets is used. This situation is in fact a so-called Catch-22 paradox:

precisely determining the overlap and removing occlusions requires the knowledge about the transformation between sets but the transformation can only be determined precisely after the overlap is found and the occlusions are removed. Nevertheless, the situation is not so extreme as even unaligned 3D images converge to the optimum solution using ICP. But if one was to strictly enforce zero error solution, it would require the use of some other method for the determination of the overlaps and occlusions. This means that ideally the transformation of the two sets should be initially roughly estimated by other means than the ICP algorithm.

Use of centroids

As already stated, the ICP is though a very robust method and the algorithm is capable of matching two datasets even when the initial conditions (e.g. the scale of the rotation *R* required to match the data) are far from ideal. In the determination of the rotation matrix, the method is robust. The weakness could be seen in the initial decoupling of the translation from the rotation transformation. Since the data and the model are first centered and then the centered data are matched rotation-wise, this supposes that the rotated sets are overlapping. The precondition for this state is that the model and the data contain the same points which are translated and also rotated. If points of certain object are introduced only in one of the sets (the model or the data), the two centroids will be offset from the *ideal centroids* which can be expressed as centroids of all correctly corresponding - matched points. This offset will not only introduce error in the determination of the computed optimal translation *t* determined from equation (21), but it will also affect the rotation calculation. The reason for this is that the dataset will be offset and even when rotating it, the ideally corresponding points will not be the closest points detected in the iterative algorithm.

Implementing other methods

This quality issue can not be resolved by other means than support of other method such as dead reckoning. There are approaches to SLAM which do not use closest points in the matching algorithm and therefore to certain extent avoid the above described problem [38]. In this research such approaches were not implemented as the focus was on the acceleration of a general ICP algorithm. A SLAM method which could be less sensitive to the mentioned problem could use landmarks for the correct matching. If a limited number of important landmarks is detected, these could be used for the estimation of occlusions and overlaps rather than equalizing the number of points used in the ICP. Also due to the uneven distribution of density of scanned points over the covered space, equalization of number of points in two datasets in order to avoid occlusions and increase overlap would be very difficult because the higher density point areas in different scans could be focused on different areas in the real environment.

5.3 Structure of the Extended Method

Method's block diagram

A block diagram of the proposed extended navigation method is shown in Figure 14. In this diagram one can differentiate three main parts of the navigation system. The first part is naturally the sensor subsystem, using the above described scanner and inclination module. Then in the second part (not highlighted in the diagram), the 3D data are preprocessed and reduced. So far the pattern follows the original method described in the previous chapters. The next step of the navigation method is the SLAM core highlighted in the figure. This is the part where the major innovation takes place. Its details will be explained in the following chapters. The output from the SLAM core is the localization of the robot in six degrees of freedom (roll, pitch, yaw angles and x, y, z translations) and also the created model of the environment could be perceived as an output. In the following paragraphs, the blocks of the diagram will be briefly described, giving more attention to the particularities in the following subchapters.

Initial alignment and measuring

The measurement of the data serves as the input of the point clouds into the navigation and mapping system. The preprocessing mainly serves the purpose of computing the inclination angles and spherical and Cartesian coordinates of each scanned point. Also, the aligning of the measured data with the robot's construction is done since there are problems in synchronizing the range data readings with the absolute inclination values.

Data reduction block

Then data reduction takes place, removing the robot and the inclining mechanism from the 3D image since these would introduce additional error in the registration (the robot and this mechanism is always in the same position in the data).

Two sub-models

In this method the model of the system composes of two sub-models: the leveled map model and the 3D environment model. This corresponds to the fact that the SLAM core composes of two phases: in the first phase, leveled maps are extracted (factorized) from the 3D data and these are registered in a 3DOF ICP based matching algorithm. Then the 3D data are transformed according to these results in 3DOF and passed to the robust 6DOF ICP based matching algorithm.

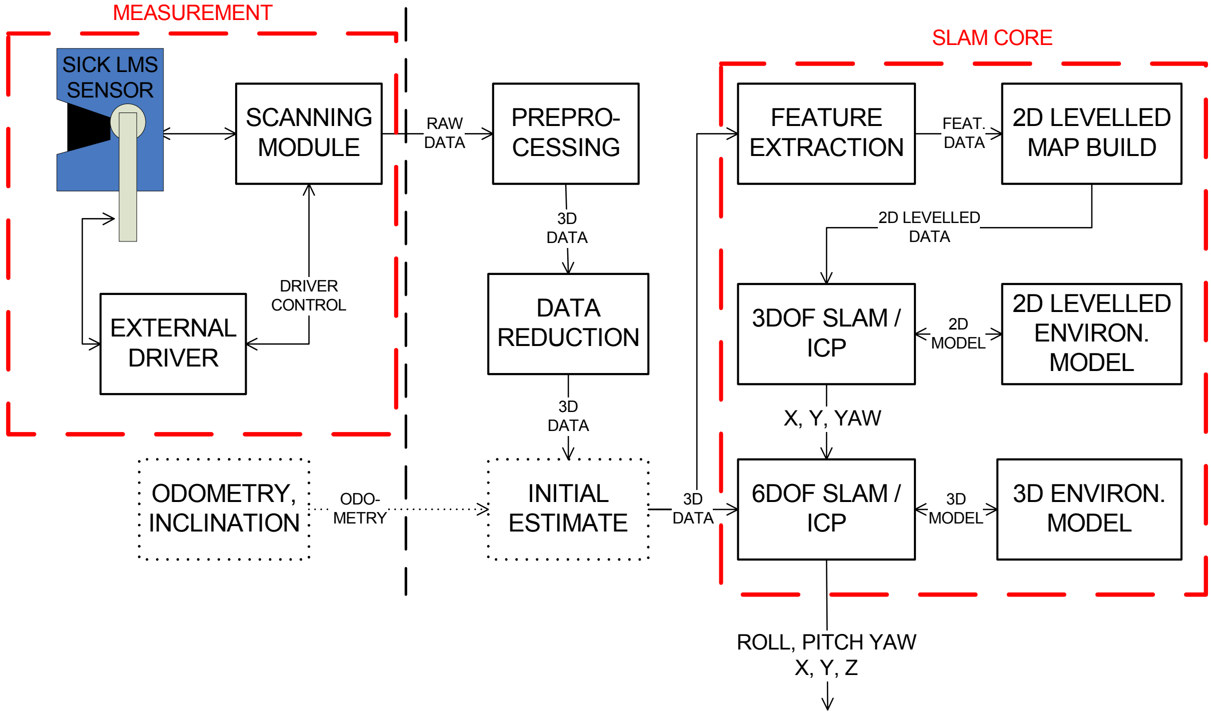


Figure 14: Proposed block diagram of the navigation system

Following chapters

In the following chapters, the different parts of the navigation method will be discussed in detail, especially focusing on the factorization of the overall 6DOF SLAM into two separate tasks: 3DOF SLAM using extracted vertical objects and 6DOF robust registration.

5.4 Preprocessing the Range Data

Pre-processing of the data

The main function of the preprocessing is the preparation of the data for the following matching phases: first, the inclination angles have to be computed for each scanned point. This inclination (zenith) angle is computed using the known inclination speed and the duration of the mirror rotation as expressed in the equation (16).

Filtering the data

Although the optical time of flight based measurements are very precise with low standard deviation, noise is naturally present in the data. Before performing higher level processing, this noise should be minimized in the data so that it does not affect further steps of the navigation method. The proposed filter is an adaptive median filter. Using such filter, the median is computed from *k* points of the surrounding area $A_{k,w}$, where *w* is the side length of the square area *A*, and the *k* points are the ones that have the closest range values to the evaluated point. The advantages of an adaptive median filter are that it eliminates the additive noise and effaces the image without affecting the shapes and borders in the image as long as the side length is not chosen too large. The surrounding area could be viewed as a window consisting of odd number of points. The computation of the median requires sorting of such window by the numerical value of each point and selecting the median between these values. In case of the range data acquired from the laser scanner, the numerical value is represented by the range measured by the scanner and the neighbors are points with the closest zenith – inclination – and azimuth angle from the robot’s perspective.

Detecting the absolute inclination

Due to the difficulties in synchronizing the data readings with the system for the inclination driving, the data have to be aligned with the robot’s construction, detecting the mechanical parts of the inclining device in the 3D image. The algorithm for the detection of inclining device’s construction is based on the experimentally measured characteristics. The variable which is to be measured is the number of a line in which the examined beam of the scanner (meaning this beam is of a constant yaw angle related to the scanner’s coordinate system, e.g. 0°) encounters the obstacle represented by the inclining device (the bar which can be seen on the sides of the construction) while the scanner is being inclined at a constant pitch angular speed. If the dependency of this variable on the yaw angle of the measurement beam is measured, for the first twelve angle values this dependency can be approximated using linear regression model. This characteristic (number of line where inclining construction detected is the dependent variable, beam’s yaw angle with regards to the scanner’s coordinate system is the independent variable) is naturally only valid for the first few data points for which this statement has been verified. The regression model is then used for approximating the time when the scanner’s zero yaw angle beam first encountered the inclining device’s construction – by determining the value of this linear characteristic at a zero yaw angle value. Since the line numbers are directly proportional to the inclination angles, the resulting detected number of the line corresponds to the construction detection angle and is further used for determining the offset of each data point’s pitch angle from this known scanner’s inclination. Using the linear characteristic rather than simply the detection line number for a particular beam (zero yaw angle) minimizes the error which could be caused by the frequency of the measurements at the given beam (75 Hz) – the construction of the inclining device could be detected up to 13.32 ms later than the beam actually crosses the inclination construction's bar. Such introduced error of absolute inclination determination would be a function of the inclining speed and the use of linear approximation of more measurements minimizes this error. The importance of this preprocessing issue is obvious: the error which would be introduced at this point (when acquiring and preprocessing the images) would not be eliminated in the registration algorithm. The map would be extended in a correct way but the robot would be localized with an error in pitch angle determination.

Spherical density effect

In Figure 15, the linear regression applied to the approximation of first nine values of the detection line of the first nine scanner beams (roll angle range 0° to 4°) is presented. This regression is used to determine a more precise zero angle beam detection line, which is further used as a measured absolute inclination angle data point. The regression was implemented in the navigation application using least squares minimizing method.

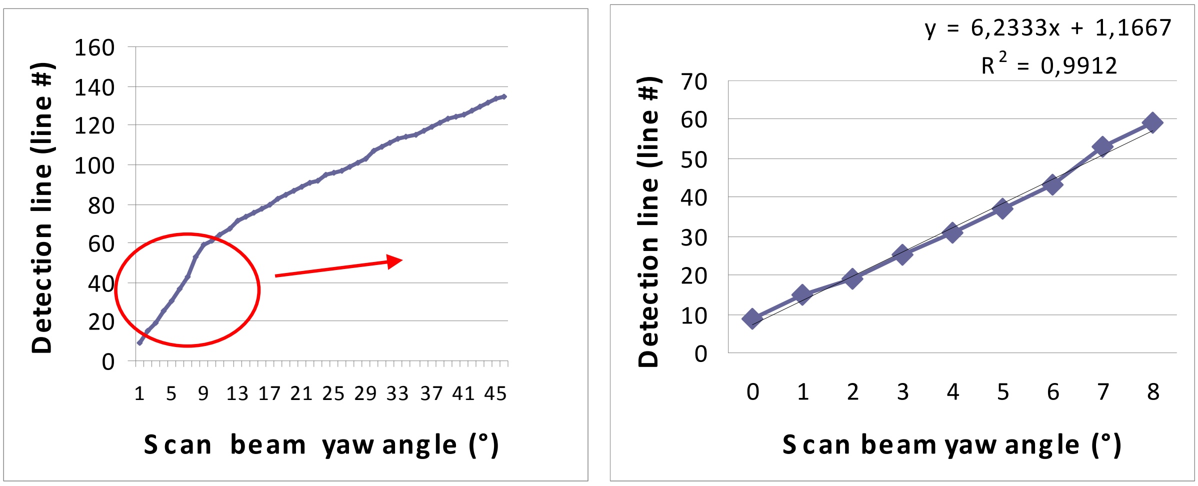


Figure 15: Estimating the absolute inclination angle for data points: linear regression applied to the detection of inclination construction (left: whole characteristic; right: detail)

5.5 Data Reduction – Saving the Effort

Spherical density effect

The principle of the uneven distribution of points is usually resolved in the data reduction phase. The measured point density is larger in areas closer to the scanner because the laser beams propagate in a spherical way. Therefore multiple points that are very close to each other could be substituted by a single point. The loss of information introduced by such fashion of processing could be balanced by the reduction of the computing time required for the scan registration – the lesser the points there are to be processed, the shorter the time to perform the registration.

Introduction of errors

The problem in point density correction is that we would have to choose targeted density in the measured space. Such targeted density could be expressed as the distance in which the scanner places points in the targeted density given the preset scanning parameters. The areas closer to the scanning device would have higher point density and the areas further would have lower point density. Higher density areas could be processed reducing the number of points, though lower density areas could only be corrected using interpolation and possibly introducing errors into the image.

Map and registration quality concerns

For the purpose of mapping, the reduction of points could mean loss of detail in the built map. Also from the registration point of view, quality of the registration is very important and therefore in this research the reduction of points according to the spherical density effect was not implemented. Acceleration of the classical ICP method will rely more on the introduced two-stage factorized registration rather than reduction of valid data points.

Necessary data reduction

On the other hand, in the data acquired from the scanner, we cannot avoid the appearance of the robot and the scanning platform. These have to be removed so that additional registration error is not introduced since these objects' position is invariant to the robot's displacement and rotation. Removal of the robot itself and the scanning platform can be easily done since their position relative to the scanner is known.

5.6 3DOF Invariance: Leveled Structure Extraction

Targeted state

The main objective of the research was to implement and accelerate the 6DOF ICP SLAM method. The focus of the improvement was not limited to the ICP algorithm itself, the objective was to accelerate the overall performance including the search of the closest points. Regarding the ICP method, the algorithm was modified so that it works with realistically overlapping data sets which will be further explained in this chapter. The main innovation is though based on the inevitable problem that in 6DOF ICP based SLAM, the number of points for which the closest point search has to be performed in many iterations is determined by the number of points in the dataset. This is especially worrying in case of three-dimensional images due to the size of the 3D data sets. Therefore the focus was on how to perform the registration initially in fewer dimensions and thus fewer degrees of freedom, decoupling the registration in certain degrees of freedom from the overall robust but slow 6DOF SLAM. Ideally, this 3DOF registration would be invariant to the remaining 3DOF pose of the robot.

Reducing the dimensions: creating leveled map

In this solution, one aspect of the physical environment is being used to decouple the 3DOF registration from the 6DOF SLAM - the gravity. It is quite easy to measure the robot's pose in two degrees of freedom: the pitch and yaw inclination. These are easy to measure when the robot is still, which is the case when acquiring 3D images. Since we can measure these angles, the 3D data can be pre-aligned so that the two dimensional data which are to be extracted for the 3DOF SLAM can be independent on the pitch and roll inclination of the robot. This 2D extracted data set used in this thesis was called *leveled map* since it is aligned with the horizontal plain.

Invariance to the remaining 3DOF

In other words the leveled map created from the 3D data is invariant on the pitch and yaw inclination of the robot since we can measure these angles and align the 3D data in these two degrees of freedom before the extraction. Then we have to ensure that the 2D leveled map is also invariant on the remaining degree of freedom – the z ("upward") translation. This implies the type of objects we will extract into the leveled map - vertical objects. It is assumed that the environment has such characteristics that in most times the robot will be able to see vertical objects in all heights (that is z translations). Thus the extracted leveled map will be independent on the robot's pose in terms of z translation. This actually holds true for most single-floor indoor environments and also for some outdoor environments.

The decoupled three degrees of freedom are therefore the remaining DOFs: horizontal translations x and y and the yaw angle ψ. In the first phase of the registration, these three degrees of freedom will be registered.

The leveled map extraction is done in the following algorithmic fashion:

1st phase:

First it is necessary to create vertical "columns" in the data. This could be done either by adapting the scanning platform (aligning the scanning planes

creating columns

with the vertical axis – not in case of this research) or by sorting the data by the yaw angle and extracting equidistant columns. The adaptation of the scanning algorithm would mean that the order in which the data are scanned is in fact the order of the data yaw angle; clearly this is only true for the "yawing" type of inclination mechanism which was not used in this research. Therefore the sorting of the data by the yaw angle value had to be implemented.

Comb Sort algorithm

For this purpose, a general Comb Sort algorithm was implemented. This algorithm was first presented by S. Lacey and R. Box and is especially efficient when there are values of very low number at the end of sorted list. The algorithm is a modification of a very common bubblesort algorithm which unfortunately loses efficiency in the above mentioned situation [9].

2nd phase: sorting columns

The created columns are then to be sorted by the vertical height of points. This ensures that vertically neighboring points in the physical environment are also neighbors in the column point list. The vertical height is directly taken from the "z" coordinate in the Cartesian coordinate system. The sorting method used in this phase is also Comb Sort.

3rd phase: finding vertical clusters

In the next phase, a search for clusters of points with similar horizontal distance (within a specified tolerance) in the columns is performed. The horizontal distance is determined simply using the point's coordinates in the Cartesian system. In this search, it has to be ensured that for each vertically neighboring pair of points contained in the cluster, the slope that these points create is higher than a specified minimum slope s_{min}. This ensures the vertical character of the cluster. The evaluations of the distance for i-th point and of the offset criteria are expressed in equation (27). Also, the cluster has to be higher than a specified minimum height h_{min} and it should not be interrupted by more than a specified maximum height gap h_{gap}. These conditions are illustrated in Figure 16.

$$d_i = \sqrt{x_i^{\,2} + y_i^{\,2}}; \left|\frac{z_i - z_{i-1}}{d_i - d_{i-1}}\right| > s_{\min} \qquad (27)$$

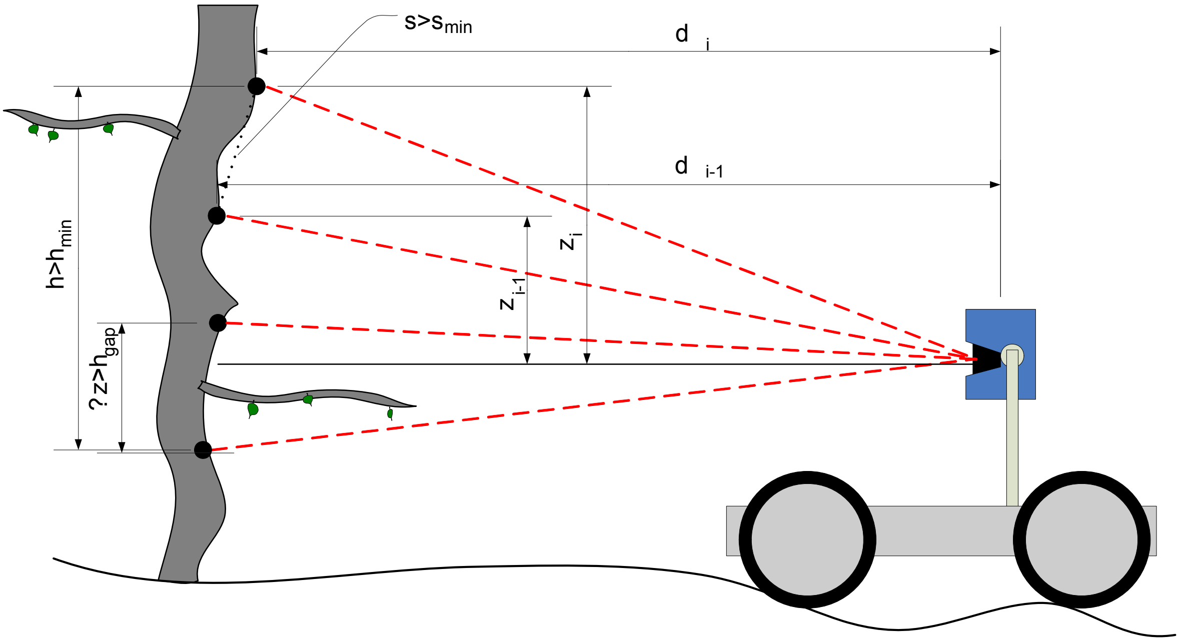


Figure 16: Vertical object detection: slope, height and gap criteria

4th phase: recording clusters into map

The extracted clusters which satisfy all above mentioned conditions are considered to be vertical landmarks. These are therefore recorded into a 2D point map, where each point represents a cluster. The coordinates of the vertical landmarks are computed as mean values of the cluster's points' coordinates. This ensures that in case the vertical landmark is slightly tilted (strong tilt would not satisfy the criteria for a vertical landmark) its horizontal coordinates are in the mean point of the points that were detected on this vertical obstacle.

Block diagram

The algorithm of the vertical structure extraction is presented in Figure 17. It can be seen that the algorithm terminates when there are no columns to be read. The algorithm itself can partly be parallelized: the first part – creating the columns from the data – cannot be divided into separate parts. The second part – processing the columns and extracting the vertical structures, can easily be run in multiple threads or processes. Therefore, if this part proved to be computing-time demanding, it could be extended to a parallel version.

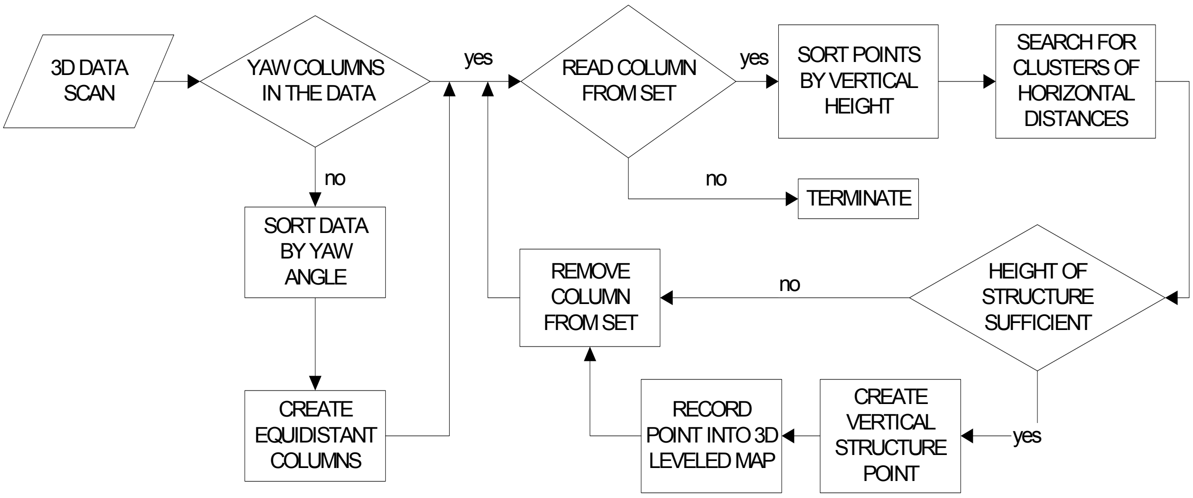

3D DATA SCAN
YAW COLUMNS IN THE DATA
yes
READ COLUMN FROM SET
yes
SORT POINTS BY VERTICAL HEIGHT
SEARCH FOR CLUSTERS OF HORIZONTAL DISTANCES
no
SORT DATA BY YAW ANGLE
no
TERMINATE
REMOVE COLUMN FROM SET
no
HEIGHT OF STRUCTURE SUFFICIENT
CREATE EQUIDISTANT COLUMNS
RECORD POINT INTO 3D LEVELED MAP
CREATE VERTICAL STRUCTURE POINT
yes

Figure 17: Leveled map building: the algorithm structure

5.7 Factorizing the Task: Leveled Data Sets Registration in 3DOF

Factorization principle

Factorization as a term relates to multiple areas of mathematics. Simple factorization is in fact a decomposition of an object into a product of other objects or factors. Similarly, in statistical analysis, factor analysis is a method which is widely used for determining or reducing the number of dimensions - variables - which are influencing the measured characteristic. The method proposed in this thesis – decomposition of the 6DOF registration problem into two separate tasks, reducing the number of dimensions in the first step, follows both patterns in which the term factorization has been constituted. The factorization itself in this research is based on detecting such characteristics in the environment that the task of the localization can be decomposed into two separate phases. If such decomposition is expected to bring positive performance change, the dimension reduction must significantly reduce the effort necessary for the completion of the registration task.

Registering the leveled maps

When the leveled 2D map is constructed, it is passed to the 3DOF ICP algorithm. First, the leveled maps have to be centered using the computed centroids. In order to accelerate the search of the closest points in the ICP algorithm, a kd-tree is built for the leveled model set and then closest points to the leveled data points are queried. After the closest points are determined, the matrix H is calculated and finally the rotation R is obtained using the SVD algorithm. This is iteratively repeated until the rotation matrix is close to the ones matrix (the criteria set for this state is when each non-diagonal element is smaller than a preset limit ε). After this is fulfilled, the final rotation (which is the product of the particular rotation matrixes from all iterations) is determined and also the translation is computed. All steps are identical to the Iterative Closest Point algorithm described in the original method's description in chapter 4.2. The result from this ICP registration is the rotation and translation in 3DOF (yaw angle – rotation around z axis, x and y translation). The reduction of the dimension is expected to significantly reduce the number of points for which the closest points are to be found when compared to registering of 3D data sets. This should naturally reduce the time required for the calculation of the H matrix which is to be decomposed. This is the first hypothesis which will be empirically evaluated in the experimental section and should contribute to the success of this method in the acceleration of the overall registration process.

5.8 Registration of the Pre-aligned 3D Data Set in 6DOF

Pre-alignment

After the matching of the leveled map, the obtained estimate of the yaw rotation is applied to the 3D data set. Since the ICP works with centered data sets, the application of the translation is not necessary at this point. The input 3D data should also be aligned in the two rotation angles: the roll angle and the pitch angle. These were also the source of information for the leveled map building and are necessary for this method to accelerate the classical ICP. The two angles correspond to the inclination of the robot and could be easily measured in any environment where gravity is present. The pre-alignment is applied on the data set by applying the rotation on each of the data points. This transformation is expected to significantly reduce the number of iterations in which the following registration method is to determine the optimum rotation and translation to correctly match the data set to the model.

6DOF ICP

The ICP algorithm as described earlier is applied on the 3D data, iteratively calculating the optimal rotation R, finally resulting in the full 6DOF match of the data. Then the data can be used to expand the overall model of the environment.

5.9 Modifying the Implemented ICP Algorithm: Unique Matching of Model Points

Correspondence problem

The originally implemented ICP algorithm had to be modified according to the problems identified in the section related to the quality issues. The problem which had to be dealt with is in the fact that the Iterative Closest Point algorithm requires that there are corresponding points in the model to the points in the data and also that the number of points corresponding to the same physical objects in the environment is identical for both point sets which are being registered. As it was already mentioned in the section related to data reduction, the density correction algorithm was not implemented in this research due to loss of information reasons and also due to the inability to generate additional range information for areas with lower than average density. Therefore the point density is changing with respect to the distance of objects from the scanner; due to the spherical character of the beam spread this density reduction is very significant. One cannot assume that the scans will be acquired from reasonably close locations, since this would undermine the whole task of localization in unknown environments as the robot's movement in this environment would be limited by the poses from which the scans are to be acquired.

Illustrating the problem

The problem is illustrated in Figure 18, where the density of the data points compared to the density of model points is approximately 4 times higher. This situation is realistic when the ratio of the distance of detected objects to the scanning platform for model and data sets is approximately 1:4. If the nearest neighbors are searched for the data points, the kd-tree is built for the model point cloud. In case the uniqueness of the model points' use is not enforced, the unbalanced number of points will result in the correspondence being estimated as the aggregate of blue and red correspondences in the figure, meaning that each model point will be found approximately four times as the closest point for a particular data point. In such case, the error estimated in the *H* matrix that drives the ICP to the optimum solution will be four times higher than if the point density for model and data points was equal. This does not necessarily mean that the transformation which will be computed from the *H* matrix will be incorrect as the transformation minimizes the overall distances in the closest point queries and therefore this increased error sensitivity area could help to find the optimum solution.

Source of errors

Nevertheless this situation will be problematic in the areas when in the two point sets we are looking for closest points of points which are close to the border of the overlap of the sets. In this case, it could happen that for the data points which do not have real correspondences in the model (meaning that the objects represented by the data are not present in the model set), correspondences of closest points will still be found but will introduce significant errors since the real correspondences are not possible. Such situation could be avoided if the overlaps would present 100 % of the sets, meaning that the overlap would be known and the transformation of the sets could hence be determined. In this case the registration would no longer be necessary which is not a realistic situation.

Implemented solution

A possible solution to this problem is the establishing of a mechanism which enforces the uniqueness of the use of model points in the closest point queries. Such mechanism was implemented in the navigation method and it has become part of the closest point query. It works according to the following principle: when a closest point is found, it is first checked whether it has already been allocated for another - previous query. If this is true, the distance between the queried points and the closest point are compared and the queried point which is closer to the found closest point is registered in the records of found closest points. The matrix *H* is therefore updated at the end of the overall process of finding closest points for the data. Otherwise the matrix would already reflect the previously added points which would sometimes (in case a closer data point was queried) no longer be valid and it would have to be corrected to drawback the previous update.

Computing-time change

This modification is not without an impact on the time-wise performance of the ICP algorithm. This change of performance and the impact on the quality will be assessed in the experiments section. The hypothesis though is that the quality of the match compared to the standard ICP algorithm where the uniqueness of the use of model points is not enforced will be better for the modified closest point search.

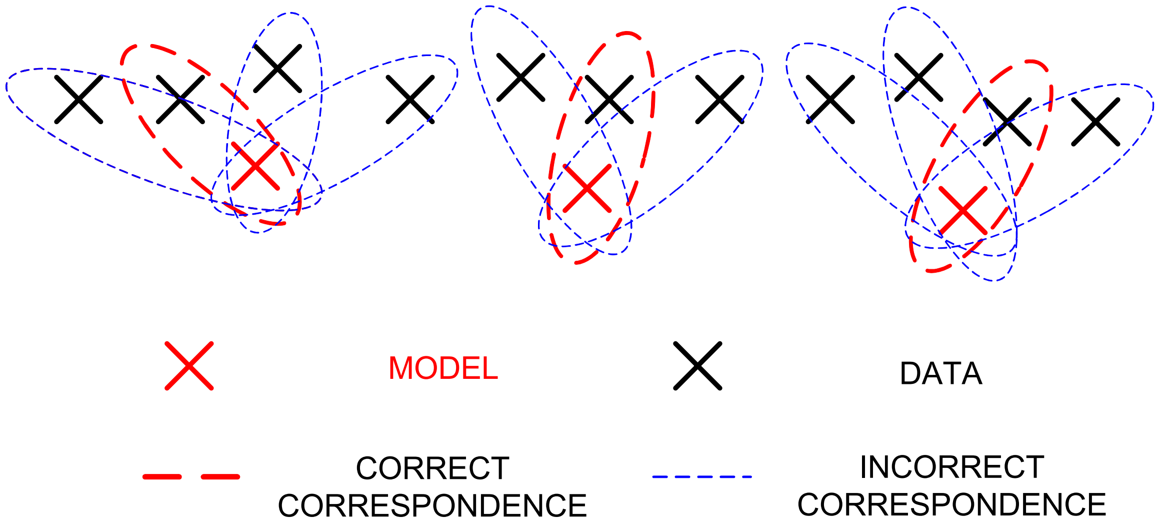


Figure 18: Different point density introduced error in finding the corresponding points

Correspon-ding points distance limit

Another modification which was implemented after testing the original version of the ICP algorithm and which is also related to the closest point queries is the introduction of a maximum acceptable distance between the corresponding points. In real environments, the reflectivity of materials is often dependent on the angle of incidence and therefore outlier points could appear in the image in some locations where in other scans representative points were measured. In such case, these would normally be taken as representative correspondences. To avoid such cases a distance upper limit for corresponding points was introduced. It has to be noted that this upper limit influences the performance of the method in terms of the maximum transformation allowing the convergence to the optimal solution. If this distance upper limit was set too low, it could happen that the correct point correspondences would not be detected and therefore the computed optimal transformation would not necessarily converge to the correct solution of the matching task.

Impact on the robustness

The implemented modifications of the ICP corresponding point's determination are in fact customizing the general ICP algorithm for the scanning platform and robotics applications circumstances (partial data overlaps, variable point density). These modifications could influence the robustness of the method but they should contribute to the quality enhancement of the method's performance. Such impact on the robustness will not necessarily be detected in the experiments following in the next chapter since the environment conditions could not necessarily provide the right initial state for this robustness reduction to prove. This possible drawback of the otherwise necessary modifications though should be kept in mind when further extending the method.

6 Experiments

Focus of the chapter

In this chapter, the experiments which were carried out during this research related to the 6DOF SLAM will be presented. The evaluation of the navigation method implemented in this research is based on the presumption that the proposed algorithm is mainly useful when operating in structured, artificial environments. Therefore special attention will be given to different aspects of the indoor human-constructed environment which are thought to be related to the performance of the 6DOF SLAM.

Aspects of experimental evaluation

The performance of the method can be evaluated from different perspectives: the most important two categories in which the evaluation of a navigation algorithm can be performed are the *computing effort performance* and the *quality performance* of the developed method. Both of these aspects were already addressed in the analysis necessary for addressing the development issues. The objective of this experimental section is therefore to confirm the issues which were raised in the analytical section and to validate the means which were developed to address the identified problems or to enhance the performance of the existing methods in the mentioned aspects. Validating the hypotheses directly or indirectly stated in the previous chapters is not the only subject of experiments: as a complement to the confirmatory function, these are also to further broaden the previous analysis through identifying new problems. This exploratory function of experiments is very important for the drawing of not only the conclusions of the research but mainly for future prospects and tasks to be completed in the following research.

Contents of the Evaluation

The evaluation will be commenced by evaluating the developed 3D range scanning platform as this is the key element which influences all other segments of the developed method. The experiments will be structured in the following way: first, an analysis under close-to-artificial conditions will be performed, evaluating the performance of the method under predefined close-to-optimum conditions. This analysis will though be performed on data from real environment. Following three evaluations will represent three different environment model conditions which correspond to the changing difficulty of the image registration task. The three evaluation environments were selected with regards to the expected weaknesses of the developed method. The changing attributes of the environments are especially the structuring factor and the presence of vertically consistent structures in the environment. Another aspect is the presence of randomly scattered objects. The last and most challenging aspect is the horizontally structured environment meaning the presence of multiple floor zones.

6.1 3D Range Scanning Platform Evaluation

The quality of the range measurements of the used SICK LMS-200 scanner was already presented in the platform development section of this thesis. This section will focus on the performance of the 3D data acquisition platform which is the key element of the whole navigation system; as stated previously, the character of the measured data to a great extent determines the overall success and performance of the implemented navigation methods.

Regulation and errors

The platform which was developed uses the DC-powered inclination to acquire fast and accurate 3D range data. The PS regulator is used to ensure that the inclining speed in the angular-velocity regulating regime is as close to the desired value as possible. The implemented processing unit allowed the measurement of desired value deviation. Since the regulated variable is the angular velocity which first derivation of the inclination angle, the local deviations in the angular velocity regulation result in the additive error in the current pitch position of the scanner at a given time compared to the inclination angle in case of precise inclination at the exact angular velocity. Since the deviation is being measured, it can be used to determine the current angular velocity and to update the inclination angle minimizing this additive error. In Figure 19, the deviations from the absolute desired pitch angle value which are a result of the angular velocity deviation are displayed. As can be seen in the Figure, this error’s maximum value is around 0.4°. Nevertheless, since the error is being measured, it is used to correct the range data updating the pitch angle inclination at any given point by adding the error to the expected pitch inclination value computed using the uniform inclination speed.

Scanning speed

The regulation speed can be set but the most often used value for taking 3D images was 25 °/s, which corresponds to the 5s scanning duration of a 125° by 180° field of view. Such data sets contain approximately 67,000 points, though some of these are projected to the robot and inclination device’s construction. The size of the 3D data sets is also optimal especially concerning the registration algorithm’s computing time. This regulating speed gives a vertical resolution of approximately 0.33°. The horizontal resolution is given by the scanner's parameters and it is 0.5° interpolated (angular resolution in each scanned line is 1° and the lines are offset by 0.5°). The resolution is not uniform but the highest point density is around the inclining axis of rotation, though it still allows very detailed 3D images even for objects in the front of the scanner. Such detail is illustrated in Figure 20 where a standing human is being displayed. The surface of the human is formed by vertexes for the visualization purposes rather than displaying the point clouds. One can even distinguish such details as eye sockets or nose in the 3D image. The detail of the 3D images would therefore allow further development of methods for semantic analysis including automatic detection of humans or other objects.

The developed 3D platform proved to be suitable for the evaluation of the navigation method implemented in this thesis. The 3D images of different environments will be displayed further in this chapter in the subsections focused on the particular experimental environment.

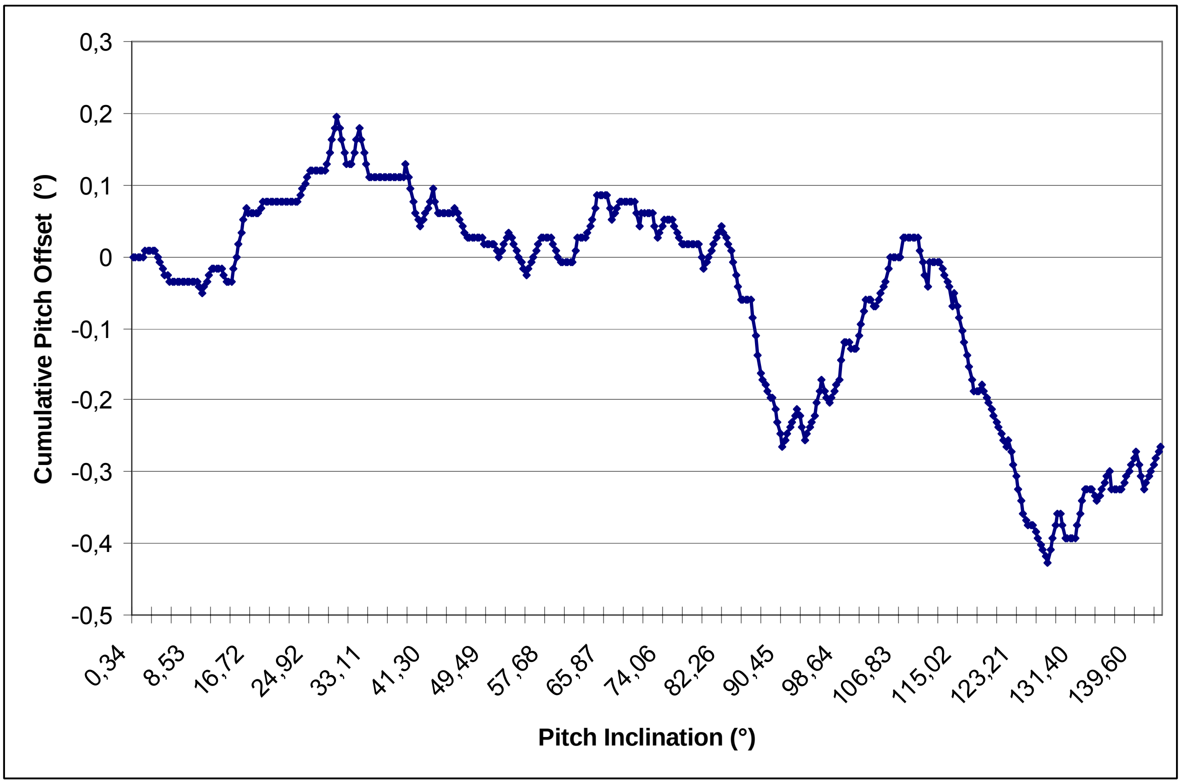


Figure 19: The cumulative pitch angle error as a function of the absolute pitch inclination; angular inclining velocity at 25.6 °/s

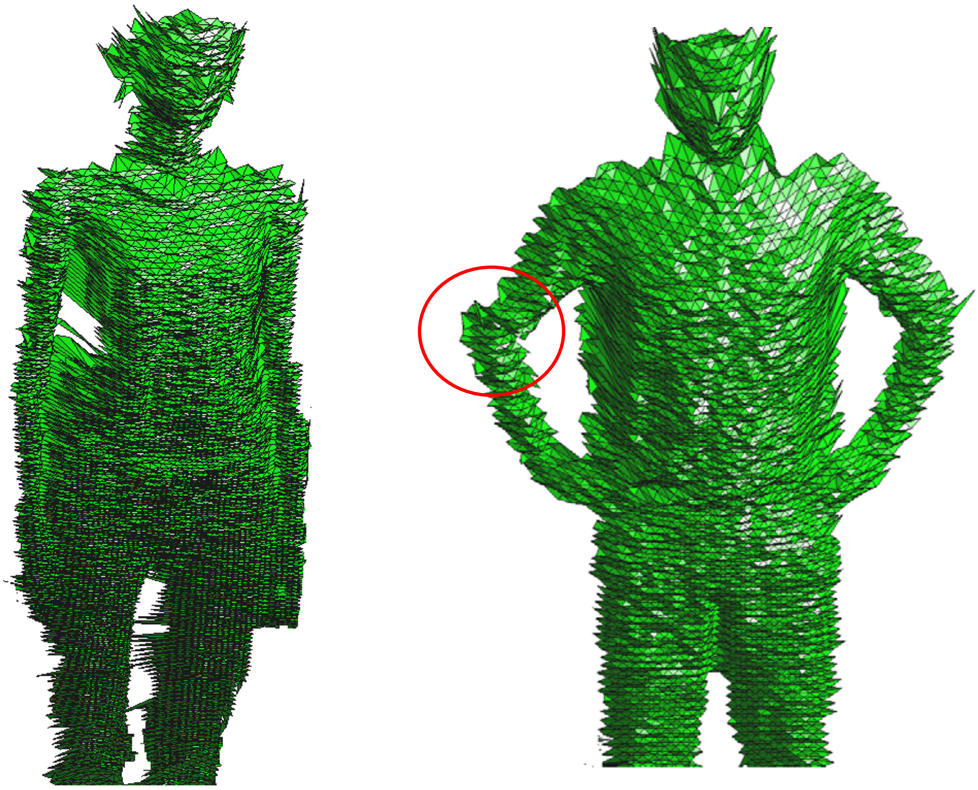


Figure 20: Demonstrating the details in the 3D image: a 3D scan of a woman (left) and a man (right); discontinuity area due to dynamics of the environment (movement of the person) is highlighted by a red circle

Dynamic environments

The limits of the applications of the developed platform are mainly given by the time required for the acquisition of a 3D image. With the above mentioned angular velocity settings, the platform takes approximately 5s to acquire the front view of the robot. This time requirement implies that highly dynamic environments would introduce significant registration errors in the proposed method as the objects would not be represented correctly in the point clouds. An example of such effect is demonstrated in Figure 20, where the man moved his right hand during the data acquisition. This resulted in a slight discontinuity in the image which can be seen in the visualization of the data. Such dynamic changes to the environment are introducing inconsistencies in the scenes acquired at a different time. Another limit that is imposed by the use of the scanner is the reflectivity criteria: the reflectivity at the angle of incidence in the direction of the angle of incidence has to be above the sensitivity threshold of the scanner otherwise multiple reflections occur (in case the object’s surface is not diffuse) or no objects are detected in the direction. The range limit of the platform is 80m which is more than sufficient for localization applications in experimental or rescue mobile robots.

6.2 Experiment #1: Close to Artificial Conditions: Ideal Match

Evaluation using artificial objects

The experiments usually start by testing of the implemented methods on artificially created objects. The reasoning behind this strategy is that for such artificial objects, one knows the correct solution (the ground truth) and therefore the performance of the method compared to the ground truth solution can be assessed. Since the objective of the 6DOF SLAM navigation method is to register multiple point clouds into a common frame of coordinates, the quality of the solution is given by the exactness of the registration of two sets of range points.

Focus of the first experiment

From this point of view, evaluating the implemented navigation method first requires an experiment in which the correct solution will be known and from which it will be easy to determine if the registration using the developed navigation method behaves correctly. One way to do this would be the above mentioned creation of an artificial point cloud with known characteristics, application of a homogenous transformation on this point cloud and registration of the two created point clouds. The disadvantage of such technique would be that the object would most likely not contain the characteristics of real environments in which the method would be expected to function in real operation. If a real 3D image was used instead of an artificial one, the homogenous transformation could also be applied to this point cloud and the created and original point cloud could then be registered using the SLAM algorithm, knowing the homogenous transformation between the point clouds (not having to measure it - depending on other measurements). The performance is expected to be the same as for the artificial object – the point clouds should be registered using the same but opposite homogenous transformation. Since such point sets contain equal number of points from the same measurement and since there are ideal corresponding pairs for all points of both sets, the mentioned ideal ICP application conditions are fulfilled and the method should perform as ideally. This type of evaluation was selected for the first experiment rather than using an artificial object especially because the real data contain characteristics of real environments while the correct solution is still known.

Phases of the experiment

The objective of the first experiment is therefore first to evaluate if the implemented 6DOF SLAM method matches two point clouds correctly in conditions given by the method's application constrains (existence of corresponding point pairs, homogenous transformation between sets). This initial evaluation will be divided into four parts: first, the original version based solely on the 6DOF ICP registration will be used, determining the correct functioning of the implemented ICP algorithm. Then, leveled maps will be created for the further described scene and finally, the match will be performed using the extended method: accelerating the registration by the factorization using 3DOF SLAM. Then, in the final part of this experiment, the time-wise performance of the algorithm depending on the desired transformation circumstances will be evaluated since the transformation of the two sets is artificial – user selected – and thus can be easily manipulated.

Data selection

An office environment was selected to form a testing real point cloud as it contains both vertical structures (walls, furniture) and also scattered object of various types. The data set for this experiment contained approximately 33 thousands of points after removing inclining construction from the 3D data. This data set contains an office room with open door and a small part of a hallway. The scattered objects are both furniture parts and office equipment; there are also glass windows in the sight of the scanner.

Applied transformation

The homogenous transformation generally consists of an arbitrary rotation and translation, transforming the data in 6DOF in total. Since the implemented method is based on the fact that the gravity vector is measured in the robotic platforms, the roll and pitch angles are available and the manipulation of these two degrees of freedom is additional from the evaluation point of view. Given the way the translation determination is decoupled from the iterative closest point method, this transformation also does not play any role in the measurements of the computing time performance of the algorithm. This is because the translation in the navigation method is evaluated as the difference between model centroid and rotated data centroid using the optimal rotation; therefore if we transpose the rotated set which is identical to the model set, the translation will be always correctly calculated as long as the rotation is determined correctly. This is though only the case of this experiment, since the data and model sets are identical in number of points and their representation of real objects. For the rotation, the remaining degree of freedom is the rotation around Z axis and in fact it is a major factor when evaluating the performance of the method (the two other rotations are given by the vector of gravity). Nevertheless, to make sure that the method is finding correct solution even if the gravity vector is measured incorrectly and also in order to test the above mentioned fact about the application of the translation, the homogenous transformation applied to the data in this initial experiment had the following parameters: the rotation around each axis was done by 40°, the translations were set to 1000 mm in all three axis directions.

Results of the initial match

In Figure 21, the solution for this registration with the above mentioned homogenous transformation applied to the model is shown. The method calculated the rotation and translation giving a perfectly matched resulting image. The average distance of data points from the closest model points for the matched solution is close to zero; therefore the ICP based registration method was verified in ideal conditions. The computing effort for this registration is expressed in the following data: there were 23 ICP iterations, in total 798,330 closest points' queries and on Pentium M 1.86 GHz machine, the registration took approximately 25 s.

Unique model points' use

In order to initially test the performance of the unique model points use enforcement, the same registration was performed with this setting enabled. Quality wise the method did not produce a more accurate solution since the classical ICP approach gives a perfect solution in this type of experiment. The improvement using this setting is expected in use with datasets with uneven number of points and occlusions present in the data. Regarding the computing effort, this setting resulted in the increase of required ICP iterations to 33. The number of closest point queries was 1,145,430 and the duration of the registration using the above mentioned machine was approx. 37 s.

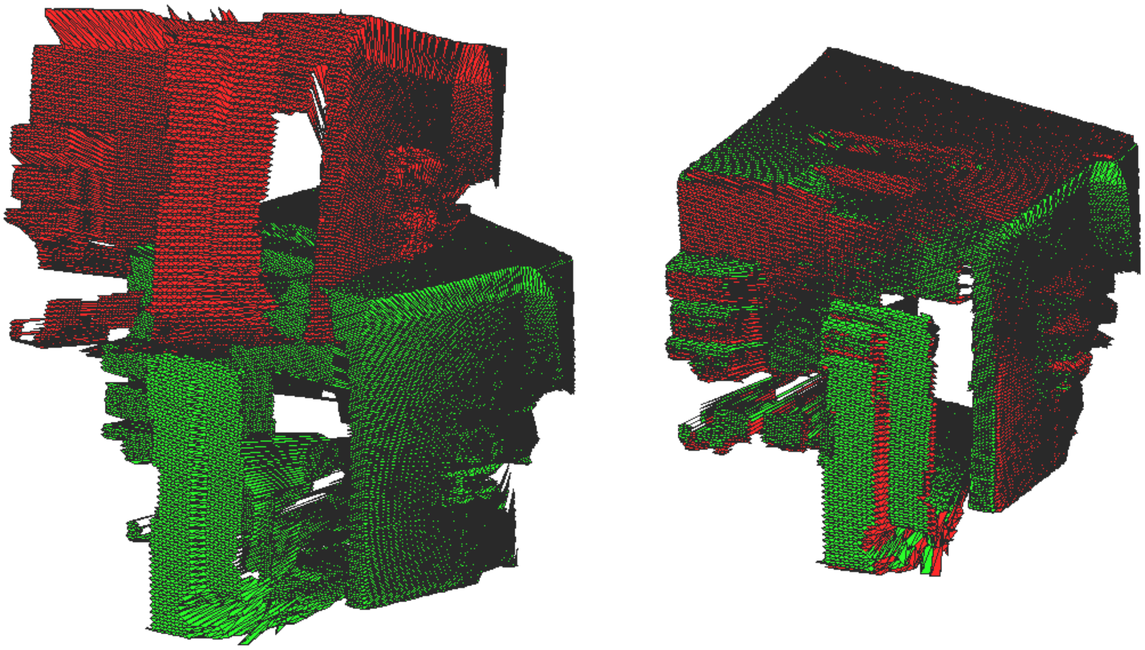


Figure 21: Registration of identical but rotated and translated point clouds from office environment: original sets (left); registered solution (right)

Leveled maps matching

The experimental evaluation of the leveled map factorization and 3DOF ICP based registration can only be performed on data transformed by rotation around z-axis (this time the gravity vector measurements have to be taken into account – giving the pitch and roll rotation angles). The same testing point cloud as in previous part of experiment was used, though this time the rotation was only applied in the z-axis sense. This ensures that both point clouds are matched in terms of horizon level of the environment which is the necessary condition for the leveled map extraction. The extracted leveled maps from the data and model set which was created by rotating the model around z-axis by 40° and translating the image by 1000 mm in x and y axes direction (z axis direction would not make sense since the data are leveled maps) are shown in Figure 22. There were 233 vertical structures detected in both data sets. The registered solution is also presented in Figure 22. The computing effort required for these procedures was the following: the required number of ICP iterations was 12, number of closest point queries was 2,784 and the total computing time using a Pentium M 1.86 GHz machine was 172 ms. The uniqueness of the model points' use output identical results, extending the required computing effort to 18 ICP registrations, requiring 4,176 closest point queries and taking 188 ms on the same machine. This increase in computing effort is not reflected as much in the overall computing time as in the case of 6DOF matching of data sets. This implies that creating the leveled model and data sets takes much longer than the 3DOF registration using ICP. This was actually expected, since when building the leveled set, the full 3D data set has to be sorted by the yaw angle which implies a demanding computing task.

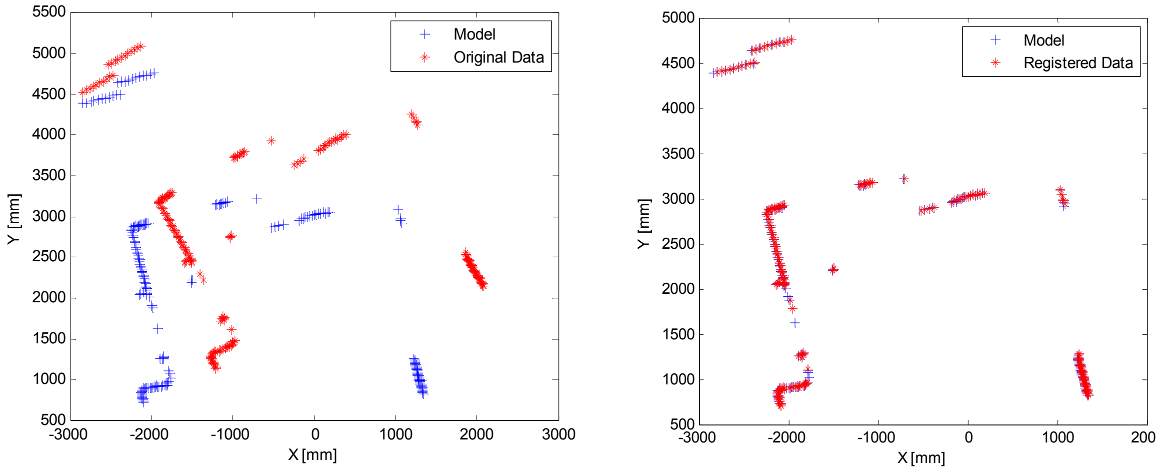


Figure 22: Leveled maps: extracted maps (left) and registered solution (right)

Sensitivity to the tranforma-tion between model and data

In the next part of the first experiment, the advantage of simple application of homogenous transformation on the 3D point cloud was used in order to measure the method's time-wise performance sensitivity on the scale of the homogenous transformation applied. Since the translation is decoupled from the ICP algorithm, the change of it would not introduce additional computing effort. The scale of the rotation angle by which the original – model - set is rotated to create the data set is in fact the key element in the computing-time requirements. Since the development of the acceleration method is based on the fact that we can measure two of the three possible rotations in the 6DOF, the applied rotation around z-axis direction is the key factor which will determine the performance of the method. Therefore in this experiment, this rotation angle in the homogenous transformation applied to the data before registration was the experimental factor. As a second though binary factor of the experiment, the uniqueness of the model points use enforcement was selected to measure its impact on the performance of the method. The measured dependent variables were the registration duration of both leveled maps matching and thereafter 6DOF SLAM matching. The experiment included registering the transformed data and model set using 6DOF ICP SLAM only and using the accelerated 6DOF ICP SLAM (accelerated by pre-aligning using the factorized solution - the leveled map based 3DOF SLAM). In both cases, the impact of the unique model points use enforcement was also observed.

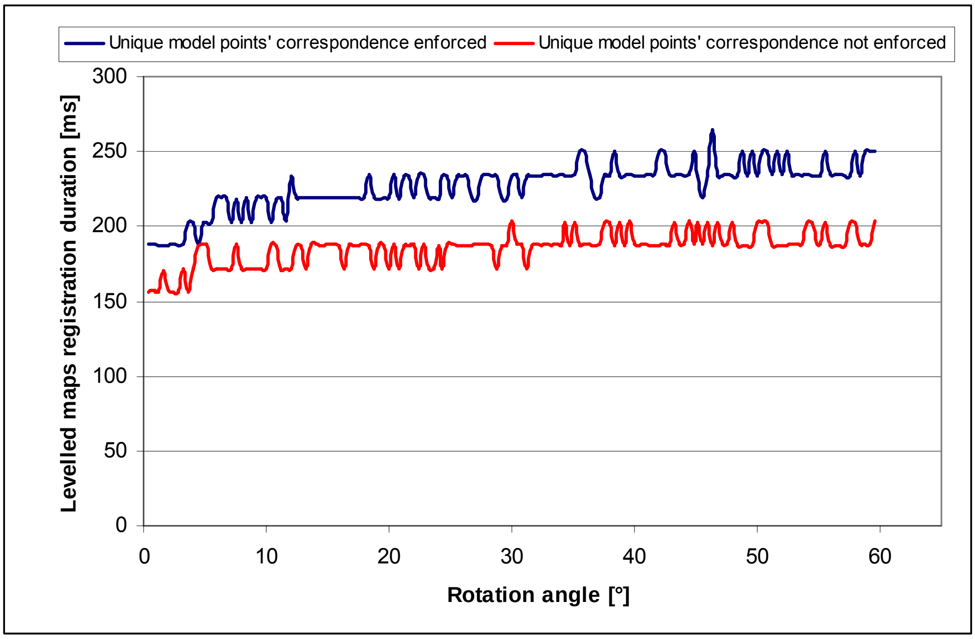


Figure 23: Duration of the Leveled maps registration as a function of the applied transformation's rotation angle

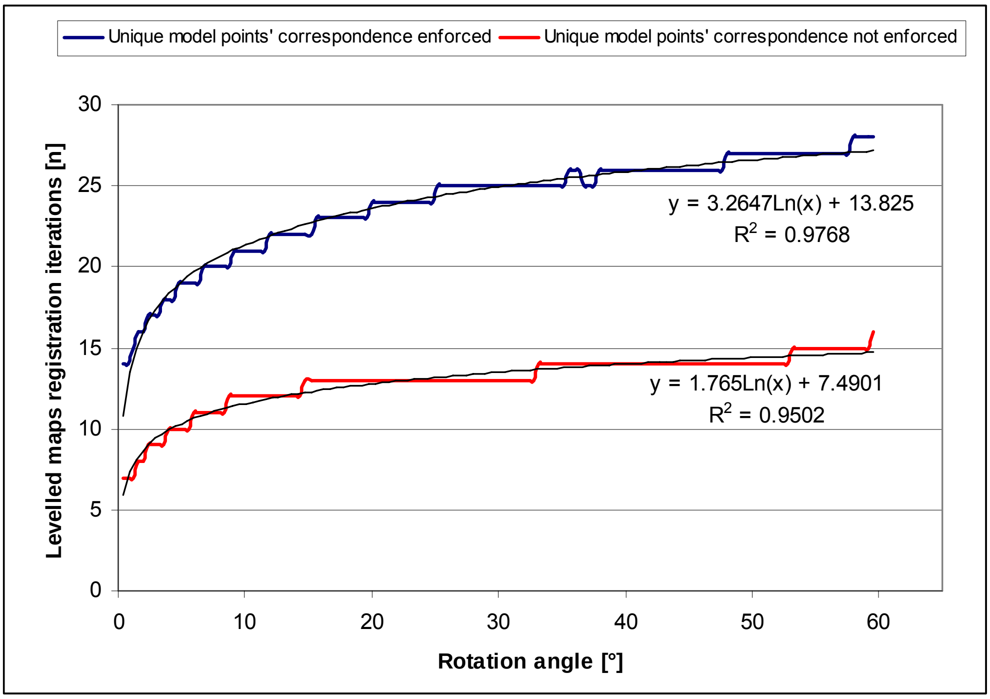


Figure 24: Number of ICP iterations in the Leveled maps registration as a function of the applied transformation's rotation angle

The registration was performed for 150 different values of the rotation angle for the rotation around z-axis direction which was applied on the model data to form the data set for the matching. This angle was ranging from 0° to 60° in 0.4° steps. This registration was performed for uniqueness of model points use enforcement enabled and disabled and also for the original ICP method and the method accelerated by the 3DOF leveled map ICP based SLAM.

3DOF factorized SLAM performance

The results of the experiment can be seen in Figure 23 through Figure 26. In Figure 23 the total duration of the leveled maps construction and the subsequent 3DOF ICP based match is shown as a function of the rotation applied on the data set. The computing time ranges from 150 to 250ms depending on the rotation scale and the uniqueness of model point enforcement feature. The duration though does not change significantly over the changing rotation scale; this confirms the fact that building of the leveled map takes much more computing effort than the matching of the created 2D sets. This can be confirmed by looking at Figure 24 where the number of iterations in the ICP based 3DOF leveled map registration is shown as a function of the rotation angle applied on the data set. This shows a significant rise in the number of iterations over the changing rotation applied on the data which though does not introduce such significant rise in computation time of the 3DOF SLAM (since the registration is time-wise only a small fraction of this procedure). The rise in the number of leveled map 3DOF SLAM ICP iterations is very interesting; one can observe that when the uniqueness of model points use in the closest point search is enforced, the number of required iterations approximately doubles. Also, the number of iterations which depends on the rotation angle representing the applied transformation between the model and data sets can be approximated by a logarithmic function. These regressions are demonstrated in the graph; the share of the variance in the data expressed by the regression function is over 95 %. The gradient of the number of iterations as a function of the applied rotation is much higher in the close to “no transformation between the model and data set” area (which corresponds to the origin of the graph – zero applied rotation).

6DOF SLAM performance

In Figure 25, the duration of the 6DOF SLAM using ICP method as a function of the rotation angle applied on the model to form the data for the alignment can be observed. In the following Figure 26, the number of iterations of this 6DOF registration is shown as a function of the same angle. There are four results in each graph, two measurement rows are the durations of the 6DOF ICP performed after applying the transformation obtained from the factorized 3DOF leveled map based ICP on the 3D data. It can be seen that the duration of the 6DOF ICP SLAM in this case remains constant over the whole range of the rotation angles. This is due to the fact that the already performed 3DOF SLAM prepares the 3D data to such state that only limited effort (constant to all applied transformation on the data) is required to finish the 3D image registration. Also, it can be seen that if uniqueness of model points use is enforced, the duration of the registration is approximately one third higher than if the uniqueness of the points use in the closest point search is not enforced. This impact is quite significant, though in this experiment we could not measure the impact on the registration quality because the data set is in fact rotated and translated model set and the original – classical ICP registration registers the 3D images just as correctly as when the uniqueness of the model points use is enforced. It should be noted here that all registrations which were the results of the cases shown in the graphs were near perfect – the average distance between the model and data sets closest points were close to zero.

Original 6DOF SLAM performance

Also, an interesting observation can be made when looking at the 6DOF ICP registration which is not accelerated by the leveled maps. When the uniqueness of model points is not enforced, the dependency of the ICP duration on the rotation angle can be approximated by a logarithmic function which is also presented in Figure 25. In case of the uniqueness of model points

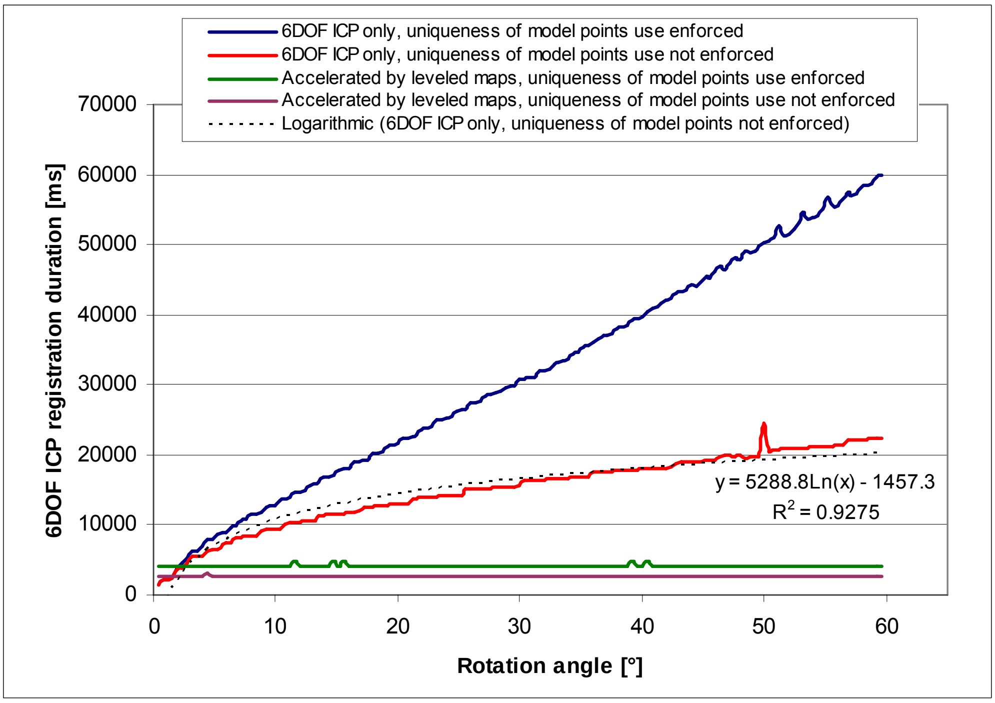


Figure 25: Duration of the 6DOF ICP registration as a function of the applied transformation's rotation angle

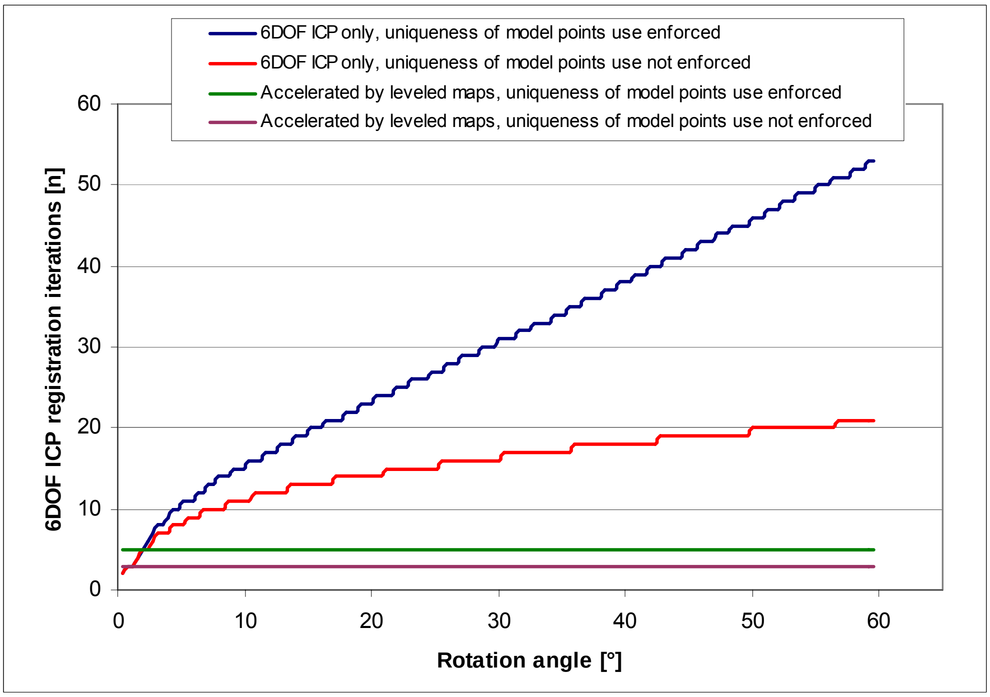


Figure 26: Number of ICP iterations in the 6DOF ICP registration as a function of the applied transformation's rotation angle

use in the closest point search, this logarithmic dependence changes to a rather linear one. This observation was not made when registering the leveled maps in 3DOF where both characteristics (with uniqueness applied and not applied) were approximated using logarithmic trend. Another important observation is that the leveled maps 3DOF ICP use accelerates the 6DOF ICP to a great extent: in case of this close-to-artificial conditions experiment, this acceleration is proportional to the applied transformation which was used to create the data set. Clearly, the larger the transformation between the model and the data sets, the bigger the impact of the implemented accelerated method. In case of the rotation around the *z* axis by 60°, the duration of the accelerated method's 6DOF ICP was almost ten times shorter than the original 6DOF ICP duration.

Factors of SLAM duration

Another aspect which can be observed from Figure 25 and Figure 26 is that the number of 6DOF ICP iterations is the main factor of the duration of the registration: the construction of the kd-tree, centering of the data and the application of the final rotation to the original data do not represent a significant computing task compared to the iterative closest point algorithm based registration. This contrasts with the previous step – registering the leveled maps, where the most significant computing effort is given to the construction of the leveled maps.

6.3 Experiment #2: Vertically Structured Environment, Simple Floor Projection

Aspect of real data

In the previous experiment, the data and the model sets originated from the same measurement, therefore containing identical though transformed points. In real operations of robots, this situation can be seen as artificial, since the point density and angular range of the measurement is limited and therefore the two registered sets will contain points which are placed differently on the same objects; and also the point densities will be different in different areas of the image.

Selection of environment

In this and also in the following experiments, all data sets used in registrations were acquired in real environments and also the model and data sets in all following registrations were acquired separately (in contrast to the previous experiment, where data sets were manipulated to form model and data pairs, allowing adjustable input transformation). The environment in this experiment was selected in indoor location such that the structuring of the environment is simple, vertical structures are present in the image and there are no disturbing scattered objects appearing in sight of the scanner. This environment is in fact an underground hallway of a multi-storey building where there are no windows, there aren't any glass objects which could introduce multiple reflections and also, where the floor projection is quite simple. The materials on the walls as well as on the ground are of a diffuse character; therefore multiple reflections should be rare. Such simple environment was selected so that the results can be compared to the following experiments in which the environment characteristics change, making the registration more difficult in experiment #3 and #4 and observing the changes in the method's performance.

Scanning procedure

There were 10 scans taken in the above specified environment, changing the pose of the scanning platform in both terms of rotation and translation. The pose of each scanning position was recorded and the consequent scans were then passed to the navigation method, registering the clusters of consequent scans. Certain issues were raised from the matching attempts. In the following paragraphs, first an unproblematic registration will be presented, showing both unregistered and transformed 3D data sets and leveled maps. This will be followed by presenting a registration which proved to be problematic and required additional modifications of the implemented method.

Well performing registraion

In Figure 27, two unregistered scans are displayed; the transformation between the scans is approximately 20° rotation-wise. These sets were matched using both leveled map based acceleration and also enabling and disabling the uniqueness of the model points' match enforcement. First the impact of the uniqueness of model points' use in the closest point search will be discussed. In Figure 28, both solutions after registering the data sets are presented. In the image on the left of the Figure, one can observe that the registration is not correct and the scans are overlapping only partially. This solution is achieved when the uniqueness of the model points' use is disabled. The explanation of the situation is the following: in each of the sets (model and data), there are points representing objects which aren't contained in just one image, since the rotation of the scanning platform allowed observation of new images in the new data set and view of some objects present in the previous scan was lost. These points do not have ideal corresponding points in the other data set. In the closest point search, the points close to the border of the area which is present in the other data set find incorrect correspondences, which are introducing error into the registration. This error can be seen in the image, the sum of the efforts reflected in the H matrix from these erroneous correspondences cause the incomplete rotation determination in the registration of the set. On the right side of Figure 28, one can see the solution which came out of the registration when the uniqueness of model points during the search was enabled. This feature actually prevented the points which are close to the overlap border but are present only in one of the sets from finding the erroneous corresponding points in the other data set. The quality of the solution is much better than in the previous experiment.

Factorization

The leveled maps which were created and subsequently registered for these two point sets are shown in Figure 29. The introduction of the factorization of the overall registration by performing the initial registration in 3DOF using leveled maps resulted in improvement of the overall computing time by approximately 46 %. The computation of the leveled maps and the registration of the created 2D data sets accounted for approximately 1.2 % of the overall computing time. In absolute number, the overall computing time was approximately 9s for the displayed data sets, while the number of points these consisted of was approximately 33,000 each.

As one can see, the two aspects of the implemented navigation method proved to be increasing the quality of the original ICP algorithm as well as accelerating the registration. The reason for this is that the factorization in this case is possible and the hallway environment contains perfect vertical structures in different directions (walls perpendicular to each other in the hallway). Such environment is perfect for the implemented method.

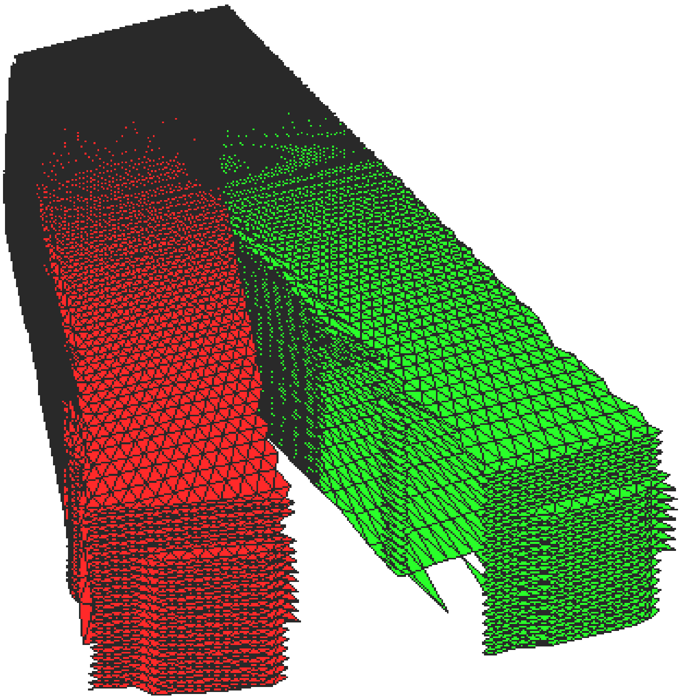


Figure 27: Unregistered model and data sets in the experiment #2

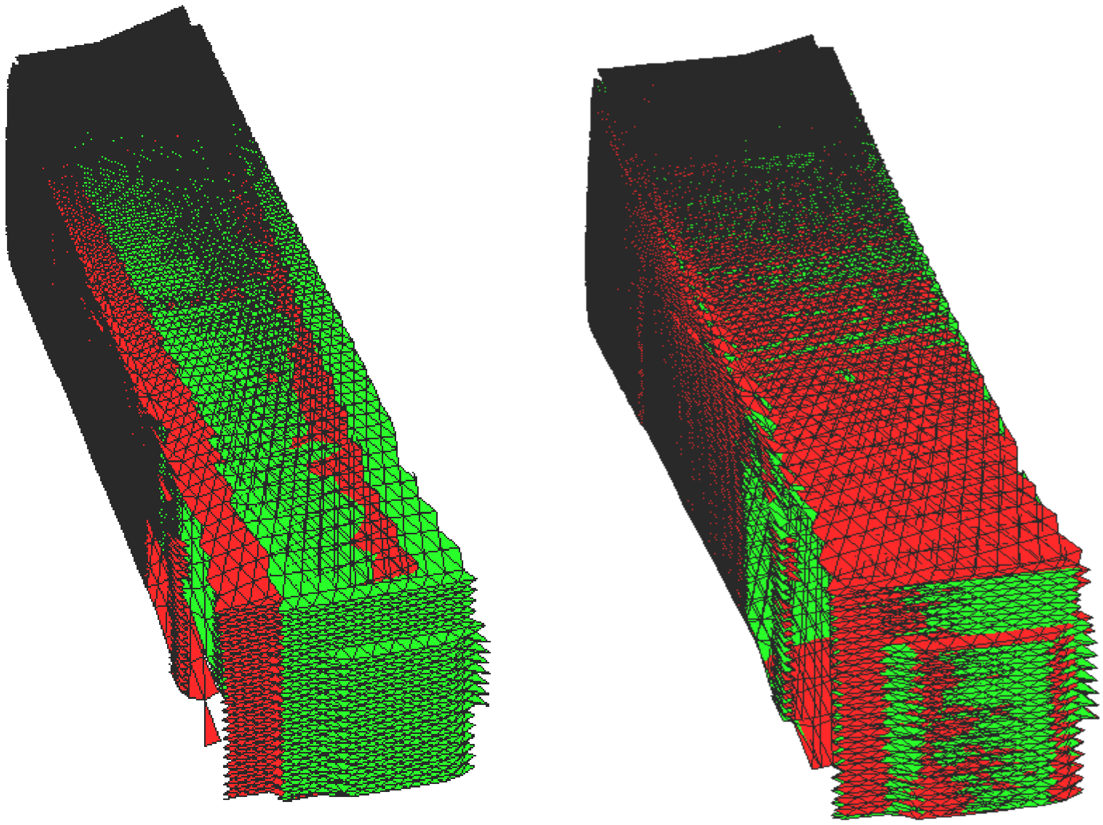


Figure 28: Registered solution: uniqueness of model points use in closest point queries not enforced (left) and enforced (right)

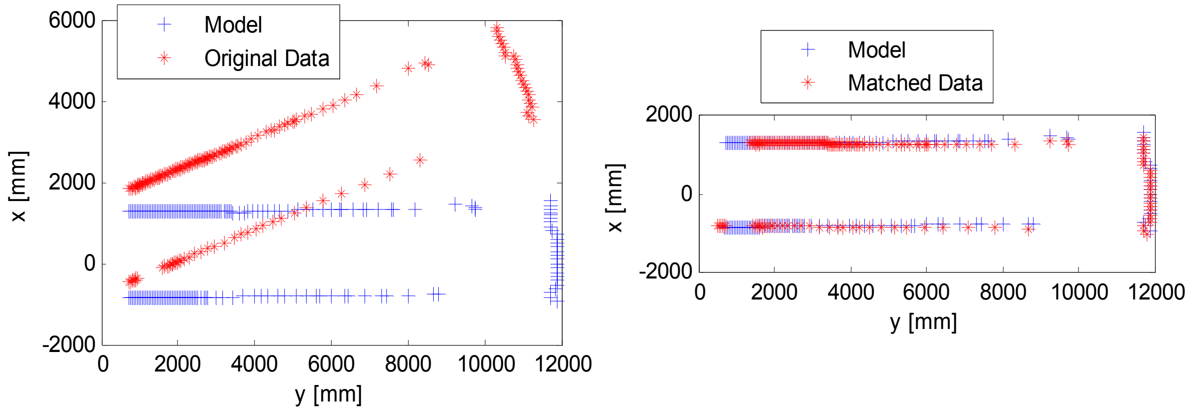


Figure 29: Unregistered (left) and registered (right, enforcing uniqueness of model points use) leveled model and data sets in the experiment #2

Complicated point clouds

When scanning in the hallway environment, there also appeared to be problematic data sets the registration of which pointed to the expected quality issues of the ICP algorithm. The ICP method is very robust especially in determining the rotation matrix, but problems can appear when determining the translation using the rotation matrix and the centroid points of the sets. This has proved to be the problem in registration shown in Figure 31. The data set (red) contains points – vertical objects detected on "the left hand side wall" which are not present in the blue model set; it also contains measurement points of objects which were not visible in the model set. Although the acquisition pose's position change is about 50cm (the error which normally occurs when determining the pose using odometry), the distance between the model centroid and the rotated data centroid is much higher than the real displacement. If the registration is performed and the translation is subsequently determined using the original method, this results in the translation offset presence in the solution which is shown in Figure 32 in the left image. This registration was obtained enforcing the unique use of model points in the closest point search; the result when this feature was disabled was similar – the translation offset was also present though the rotation was not determined as correctly as in the presented solution. This situation is quite problematic since the problem is in fact the Catch 22 situation as it was described earlier in this thesis.

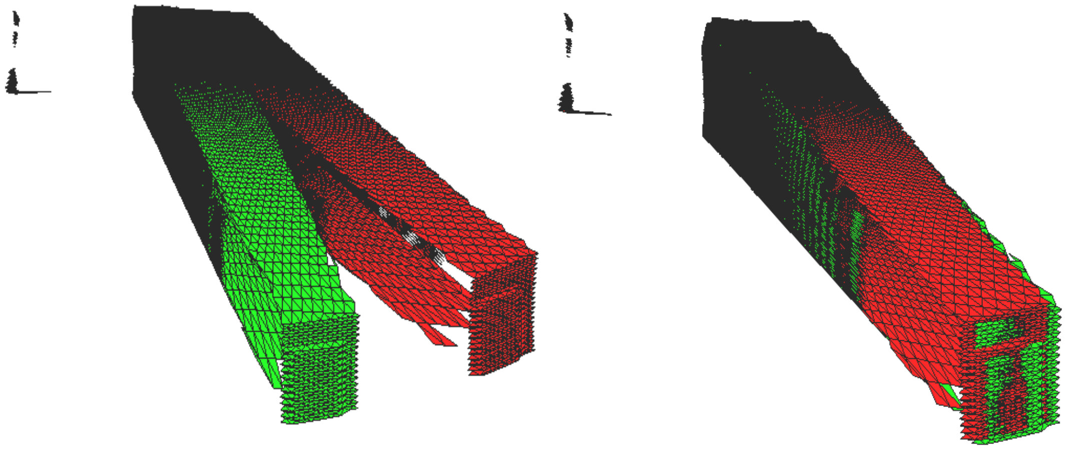


Figure 30: Registering a problematic 3D pair: other objects are in the view of the data set; unregistered sets (left) and registered (right)

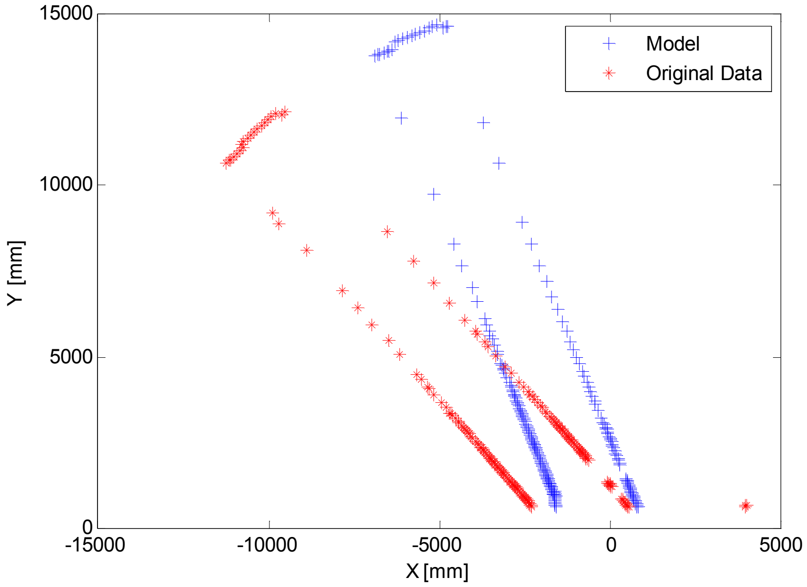


Figure 31: Data and model leveled maps to be matched (problematic solution due to variable point density and appearance of new objects in data set)

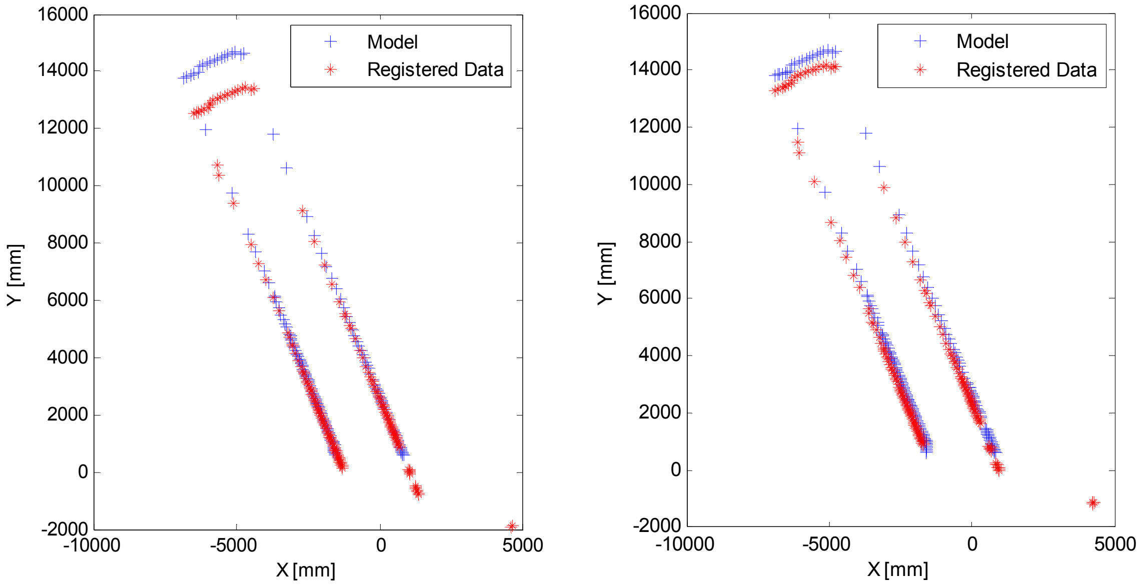


Figure 32: Registered leveled maps in 3DOF: using a single pass method (left) and using two passes and updated centroids (right)

Two pass extension of the 3DOF SLAM

In order to include only the points which are concurrently present in both data sets, the transformation between the scanning poses has to be known. This is though the product of the registration. Therefore the only way to provide the method with the corrected data set is running the whole registration twice and expecting the first solution to be close to the ground truth. In the first pass, the rotation between the sets is estimated and using the odometry information and this rotation value, the leveled data and model sets are cut so that the points which are not expected to be in both sets based on the first registration are removed. This was actually implemented in the method after encountering the above shown problem, in Figure 32 in the right image the registration after two 3DOF SLAM passes with the data and model filtration is shown. One can observe that the translation obtained using the modified centroids (since data and model points were left out) and the second pass rotation results are much more accurate especially in the y axis direction (the original – first pass result was translation of approx -1 m in the y-axis direction, after second pass this was around -20 cm which is close to the ground truth – this was approx. -15 cm). The points which were left out for the second pass are still shown in the image since these are otherwise valid measurements. Changing the number of 3DOF SLAM passes to two introduced around 25 % rise in the computing effort of the first phase of the navigation (leveled map matching) including the data filtration and centroid estimation, which increased the computing time from 100 ms to approx. 125 ms for the particular – displayed data and model set and using a Pentium M 1.8GHz machine. The subsequent 6DOF registration was significantly accelerated by the pre-estimate from the leveled map based 3DOF SLAM.

Time-wise performance

As a part of this experiment, six pairs of data sets were matched using both accelerated and original SLAM methods. The data sets so far presented are pair#1 (the problematic set) and pair#5 (the unproblematic set). The facts for the data sets and the transformation between them can be seen in Table 4. The impact of the implemented acceleration and the performance of the factorized 3DOF SLAM is shown in Table 5. As can be seen in the data, the implemented extended method significantly accelerates the matching of most of the pair data sets which were acquired in real environment, applying transformation on the scanning platform which is specified in Table 4. In this real experiment, only one pair match using the extended method’s acceleration by factorizing the SLAM using 3DOF leveled map pre-alignment has shown a deceleration compared to the original 6DOF ICP based SLAM. This case was actually acquired by simply moving the scanning platform (the robot) in the translation sense only, not applying any rotation. The effect of the 3DOF factorized SLAM in such situation proves to be ineffective as the iterative part of the ICP algorithm is targeting to calculate rotation and the translation is derived using the obtained rotation matrix at the end of the registration procedure. Hence if only translation is applied on the scanning platform when acquiring the 3D data sets, the time-wise performance of the overall algorithm will not be accelerated since the leveled maps are from the computing performance point of view effective in accelerating the iterative part of the 6DOF ICP algorithm by pre-aligning the model and data set. The overhead caused by the building of the leveled sets and matching them together with difference in optima of the solution in 3DOF SLAM and 6DOF SLAM actually introduces the -8 % deceleration compared to the original method (this deceleration is most likely variant, leveled maps building overhead should range between 1 and 2 %). On the other hand, for other pairs where rotation was the main factor in the transformation applied to the scanning platform between acquiring the images in the pair[3], the performance of the modified method was much better than the performance of the original 6DOF ICP SLAM. The acceleration varied for different pairs between approximately 14 % and 46 %. One factor which stands behind this variance and which could not be easily quantified is the structuring of the environment; one can see that the overall duration of the pair-wise registration of sets where transformation included rotation ranged from 9 to 45 s when using the extended navigation method. This variance can be partly explained by the scale of the rotation (the registration whose scale of transformation was largest in the selected data – rotation of approx. 40° - took the longest time to register). Also, if one compares the registration of pair #1 and pair #5, the scale of the rotation is the same, number of points in the data and model set is approximately same and the duration difference is very large for both original and extended methods. It is difficult to analyze the characteristics of the environment which cause such disproportion and it is beyond the focus of this research to do so, though in future research this area would be very interesting to explore.

Quality of the registration

In Table 6, the results from the registration are displayed. The rotation angles were determined from the rotation matrices. The rotation angles determined by the 6DOF SLAM are reasonably accurate, taking into account that the absolute

[3] Please note that the accuracy of the translation in the Table is limited since the odometry and absolute distance measurements were not allowing precise measurements. The accuracy is similar to the real performance of odometry based dead reckoning systems – in units of centimeters.

measurements of rotation angles on-site when scanning and moving the scanning platform were quite inaccurate. From the visualizations, the overall quality of rotation determination was very good. The weakness of the method which can be seen in Table 6 is the determination of the translation between the data sets. This is caused by the already mentioned problem in the different number of points in the data and model sets and also in different density distribution along the scanned surfaces. These aspects influence the position of data and model centroids which are further used to calculate the overall translation. These cetroids in case of the uneven distribution and incomplete environment representation do not correspond to the geometrical centroids of the environment. The registration of pair #4 when only translation was applied on the scanning platform performed reasonably taking into account the preconditions which are not fulfilled for the ICP method (even point distribution, presence of same objects in the data). The error for this registration was approximately 14cm in the x-axis direction and 5 cm in other axes directions. The translation resolving inaccuracy was mainly appearing in registrations of sets when the scanning platform was rotated between acquiring data and model sets. This issue could be solved by an introduction of other navigation methods which do not share the problem of varying distribution density (such as inertial navigation systems, for range measurement based methods cross-correlation methods etc); some of these are being developed at our workplace. Another solution would be in changing the density of points, which will be considered in the continuing research. As already stated, the objective of this thesis was not to develop an independent navigation method but rather to create a subsystem which would significantly contribute to the overall navigation capabilities of the navigation system.

Table 4 : Point cloud sets selected for the experiment #2

Data sets	Points in model set [n]	Points in data set [n]	Z-axis transformation [°]	X-axis translation [cm]	Y-axis translation [cm]
Pair #1	36016	35540	-20	16,5	-2,1
Pair #2	34741	32216	-20	16,5	-2,1
Pair #3	32247	27449	-20	16,5	-2,1
Pair #4	35136	32144	0	60,0	0,0
Pair #5	34533	31900	-20	16,5	-2,1
Pair #6	31930	36352	40	-28,9	8,1

Table 5 : Impact on the 6DOF SLAM when accelerating the method using the leveled map factorization

	6DOF SLAM ICP only		6DOF factorized SLAM accelerated by leveled maps				Comparison
Data sets	ICP duration [ms]	ICP iterations [n]	3DOF SLAM duration [ms]	3DOF SLAM iterations [n]	ICP duration [ms]	ICP iterations [n]	Leveled map acceleration [%]
Pair #1	31703	32	140	14	27156	25	13,90%
Pair #2	13937	29	141	13	9063	17	33,96%
Pair #3	14328	28	110	9	10328	18	27,15%
Pair #4	7312	15	93	1	7828	13	-8,33%
Pair #5	16969	31	125	10	9031	13	46,04%
Pair #6	44796	49	234	14	35125	36	21,07%

Table 6 : Results of the pairwise registration in experiment #2

	3DOF SLAM			6DOF SLAM			
Data sets	rot(z) [°]	x [cm]	y [cm]	rot(z) [°]	x [cm]	y [cm]	z [cm]
Pair #1	-18,9	-24	-45	21,8	18	-13	-1
Pair #2	-21,0	-10	-44	22,5	62	-17	-3
Pair #3	-18,3	-21	-27	22,1	-11	-42	-2
Pair #4	0,0	-1161	508	0,0	-73,6	-5,3	-5,1
Pair #5	-22,3	-8	-34	22,4	8	-17	-2
Pair #6	35,0	19	17	53,1	-20	36	-1

6.4 Experiment #3: Vertically Structured Environment with Scattered Objects

Ideal conditions

In the previous experiment, the point clouds were acquired in an “empty” environment with clear geometrical structuring. The underground hallway also did not contain any glass or other highly reflective objects which would introduce multiple reflections depending on the angle of incidence with the objects’ surface. Such environment represents ideal conditions for the operation of the implemented navigation method, though real indoor environments are often more complex in terms of geometrical characteristics and more complicated to observe with a ranging device such as optical time of flight sensors.

Scope of this experiment

To explore and verify the behavior of the method in different and more complicated real environments, in the third experiment range data were acquired in an office and laboratory environment. The environment was selected in such way that it contained various objects of different shapes scattered in the view of the sensor. This was aiming to test the method under conditions where the original ICP algorithm encounters difficulties when finding the closest points – scattered objects are complicating the closest point search especially if the scale of the transformation between the two sets is larger than the structuring of the environment. The structuring of the environment with scattered objects is much more complicated than of the environment analyzed in the previous experiment #2.

Scene selection

There were two scenes selected for this experiment: the first scene was already used as the close-to-artificial data set in the first experiment. In experiment #1, only one measurement was used to form both the model and the data set. Here we are testing performance of the method under real conditions, therefore the method is tested with separate measurements for model and data sets. The second scene is in fact a laboratory with many scattered objects in it.

Office environment scans

In the first part of the experiment, the office environment data sets were acquired: there were four data sets acquired in a single office room (with door open to the hallway). The office was equipped with built-in furniture which made the detection of leveled structures more difficult. There were also various objects scattered in the view of the scanner such as ventilator, computer, various objects in the shelves behind glass doors etc. The scene was highly structured compared to the hallway in the previous experiment. Four acquired data sets formed two pairs of model / data sets: the transformation between the model and the data set was rotation of 20° around the z-axis direction for the first pair, and rotation of 45° around z-axis direction for the second model and data pair, while the translation was close to zero (the scale of translation could be compared to a general odometry error). The result of the first registration (the scale of rotation approximately 20°) is shown in Figure 33 and Figure 34. This registration took approximately 57 s when accelerated by leveled maps, compared to 78s when using the original method – 6DOF ICP based SLAM only. The result seems inaccurate though the rotation was very precisely determined in the registration process; the error is introduced again by the erroneous translation calculation resulting from the miscalculation of centroids (or in other terms difference in model and data point's geometrical centroids). A much more challenging was the situation where the scale of rotation was approximately 45°, this registration is presented in Figure 35 and Figure 36. The quality of the registration is in this case very limited, since the share of common points (points representing physical objects present in both data sets) in the combined point cloud is very low. The obtained solution was only achievable by using the extended method; the original method resulted in erroneous registration: the determined rotation matrix rotated the data set in the opposite direction. Therefore in this case it is impossible to judge the "acceleration" of the extended method compared to the original method since the registration quality was unequal.

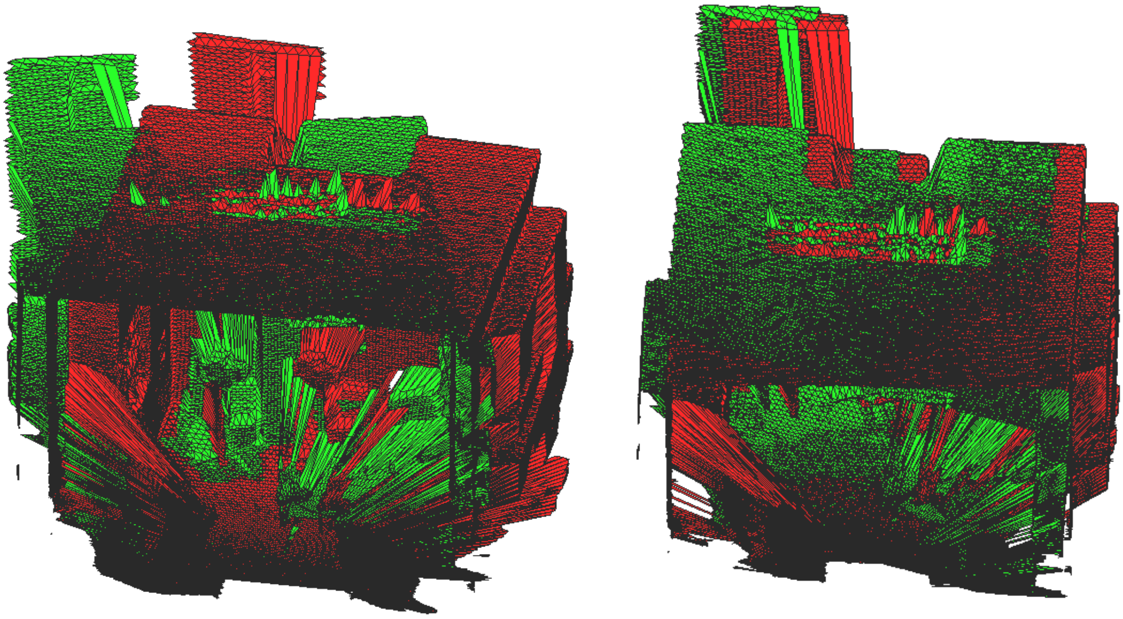


Figure 33: Registration of data sets acquired in office environment: scale of rotation approximately 20° (left: unregistered; right: registered)

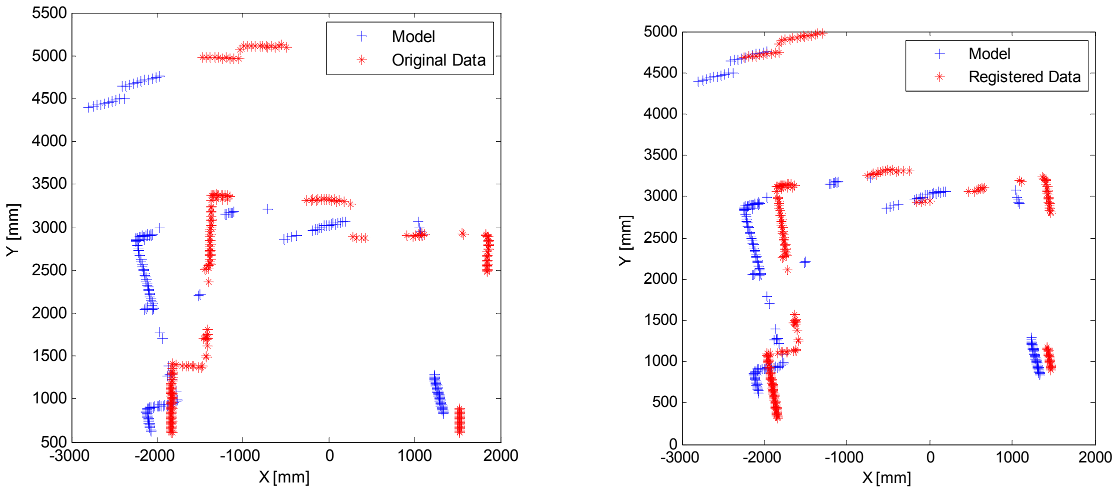


Figure 34: Leveled model and data sets from the office environment (scale of rotation 20°; left: unregistered; right: registered)

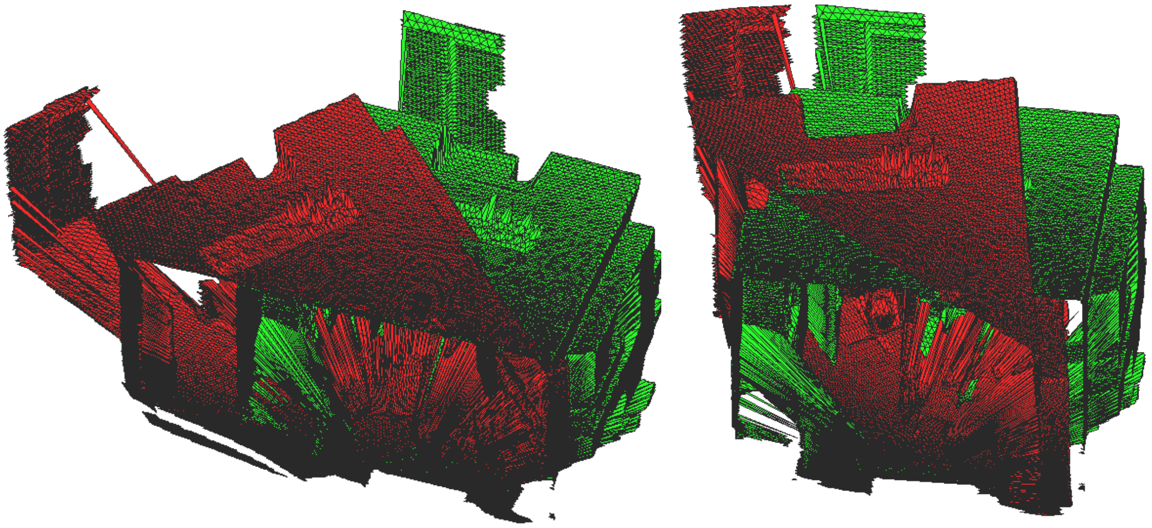


Figure 35: Registration from office environment: scale of rotation approximately 45° (left: unregistered; right: registered)

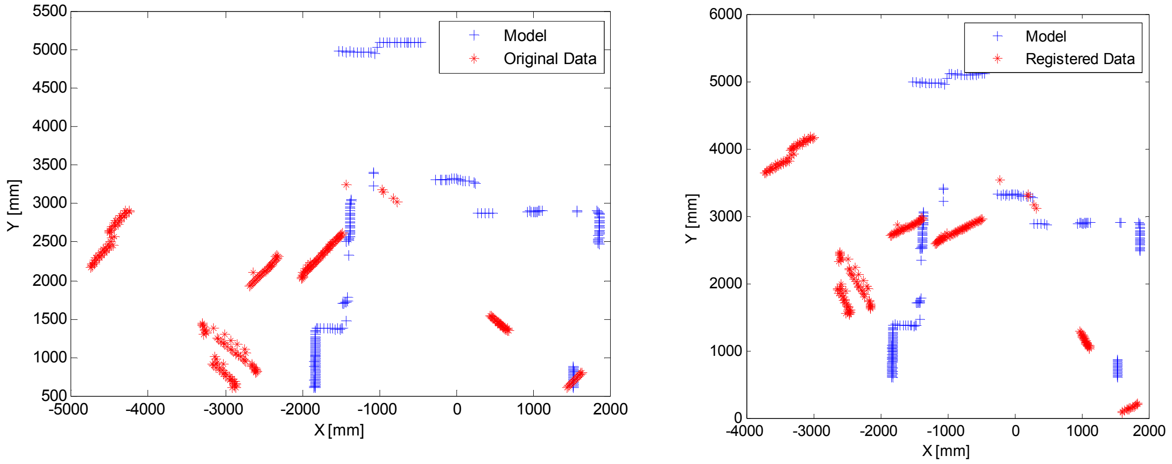


Figure 36: Leveled model and data sets from the office environment (scale of rotation 45°; left: unregistered; right: registered)

Laboratory environment with scattered objects

The second measurement was done in a laboratory environment where the number of scattered objects was highly above normal conditions. The scattered objects were very close to the scanning platform and the transformation between the acquisition of model and data sets was rotation around the z-axis direction and translation in x and y-axis directions. Also, the space in front of the scanner was not extending too far, so the scattered objects blocked a significant share of the sensor's field of view. The registration was performed for two scanned pairs, one contained rotation of scale 35°, the other rotation of 15°. Both registration solutions shared some characteristics: the leveled map 3DOF SLAM did not perform as expected. For the less rotated sets, the rotation around z-axis was determined to only half of the scale which was applied to the scanning platform, although the scattered objects were “filtered” and not appeared in the 2D leveled maps. For the other sets which were acquired applying larger transformation on the platform, the leveled map matching encountered a local minimum even earlier than at half of the transformation scale and did not help to pre-align the data and model sets. The robust 6DOF SLAM also encountered difficulties since the sets were only partly overlapping and there were other objects than vertical and horizontal structures; the final calculated transformation was also rotating the data set around y-axis direction. The scale of rotation around the z-axis from the 6DOF SLAM was determined correctly for both pairs, one of the registrations has shown erroneous translation determination as in some of the previous experiments. One of the set pairs with the smaller transformation between the model and data sets is presented in Figure 38 and Figure 39. Overall, the extended method did not accelerate the registration compared to the original method for these set pairs. For example for the shown data set pairs, the registration using the extended method took slightly longer in spite of a little lower number of iterations in the 6DOF ICP SLAM phase (112 iterations taking 183 s in extended method vs. 116 iterations taking 163 s in the original method), meaning that the searches for closest points must have been more effective in the original method. The long duration of the registration can be partly explained by the size of the data sets which was nearly double compared to some other presented experiments (data size approx. 57,000).

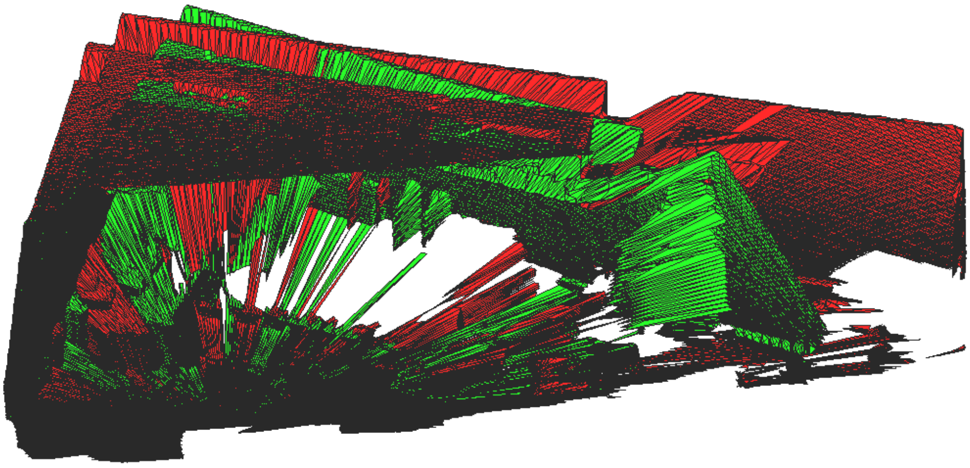


Figure 37: Unregistered scans taken in a laboratory environment with large amount of scattered objects

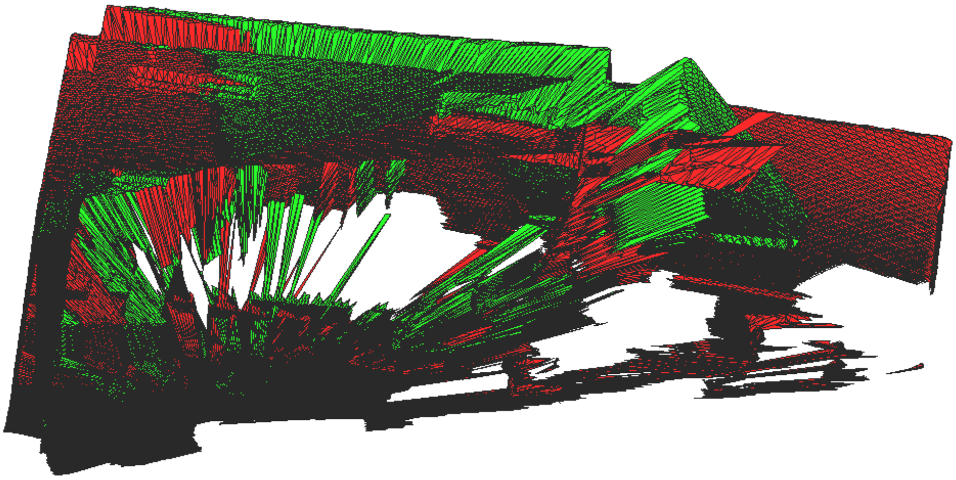


Figure 38: Registered scans taken in a laboratory environment with large amount of scattered objects

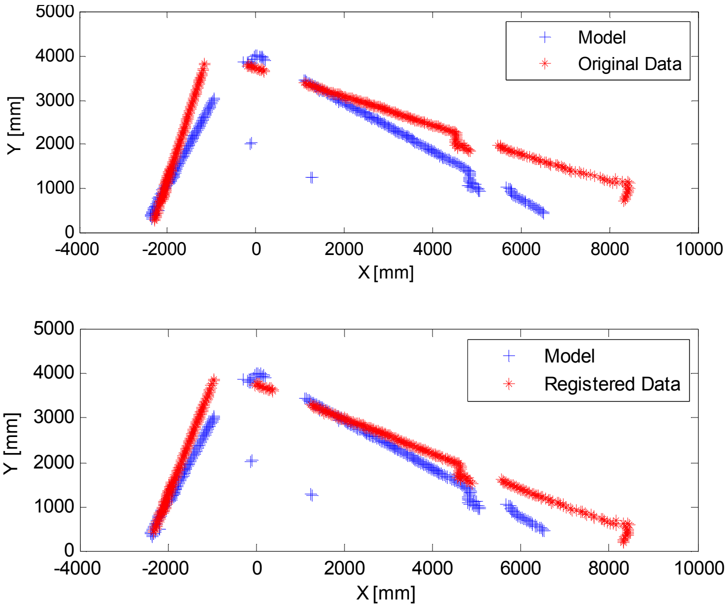


Figure 39: Leveled model and data sets from the laboratory environment (scale of rotation 15°; top: unregistered; bottom: registered)

6.5 Experiment #4: Vertically Structured Environment, Scattered Objects, Multi-Floor Vertical Structuring

Scope of the experiment

In the last experiment of this thesis, 3D range data were acquired in an environment which was expected to be extremely difficult for the registration using both the original and the extended method: a multi-storey area with stairway and various objects of irregular shapes in the view of the scanning platform. Also, the stairway was lined with a glass border, possibly introducing multiple reflections depending on the angle of attack of the laser beam. Another feature of the environment was a presence of bars which enclosed the scattered objects in the bottom part of the scans, resulting in regular structuring of the 3D data sets in this part of the image. In this experiment, the objective was only to demonstrate the behavior of the leveled maps generation and 6DOF robustness of the ICP registration under extreme conditions.

Experimental results

In the first experimental data acquisition, the transformation applied on the platform between acquiring the model and data sets was a rotation around z-axis direction of approximately 15°. The unregistered data and model sets are shown in Figure 41. The registration when using leveled maps took approx. 203ms, performing 10 ICP iterations in the 3DOF SLAM. The rotation determined by this 3DOF SLAM was of approximately 18°, the data set was therefore slightly “over-rotated”. The generated and matched leveled maps are shown in Figure 40. The following 6DOF SLAM took approx. 46 s, completing 33 iterations in the ICP algorithm and determining the rotation around z-axis of approx. 21°. The solution is therefore “over-rotated”; this could be explained by the appearance of new objects in the data set, which are marked in Figure 42. This Figure shows the registered data and model set. When testing the original method using only 6DOF ICP, the registration has given the same results but it took approx. 100 s while completing 72 ICP iterations. Regardless of the quality issues, the extended method therefore accelerated the original method by more than 50 % even for such a complicated environment. The determined translation which in the ground truth was close to zero did not show any significant error in the 6DOF SLAM (the calculated translation vector (1.6 cm, 1.2 cm, -1.3 cm)). In contrast the 3DOF registration of leveled maps calculated an erroneous translation of (-65 cm, 56 cm, 0 cm), which can be seen in Figure 40. The generated leveled maps contain detected vertical structures from multiple levels of the environment, supporting the presumption about improving the z-axis invariance by using the vertical objects in the leveled map building.

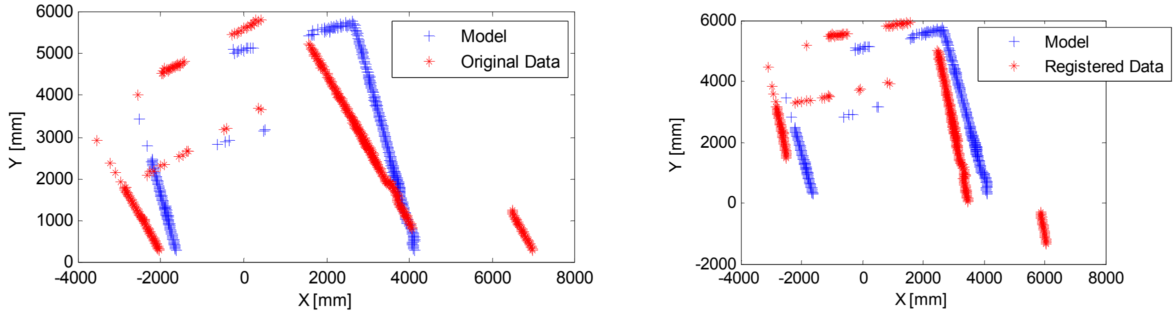


Figure 40: Leveled maps of multi-storey data and model sets: unregistered (left) and registered (right)

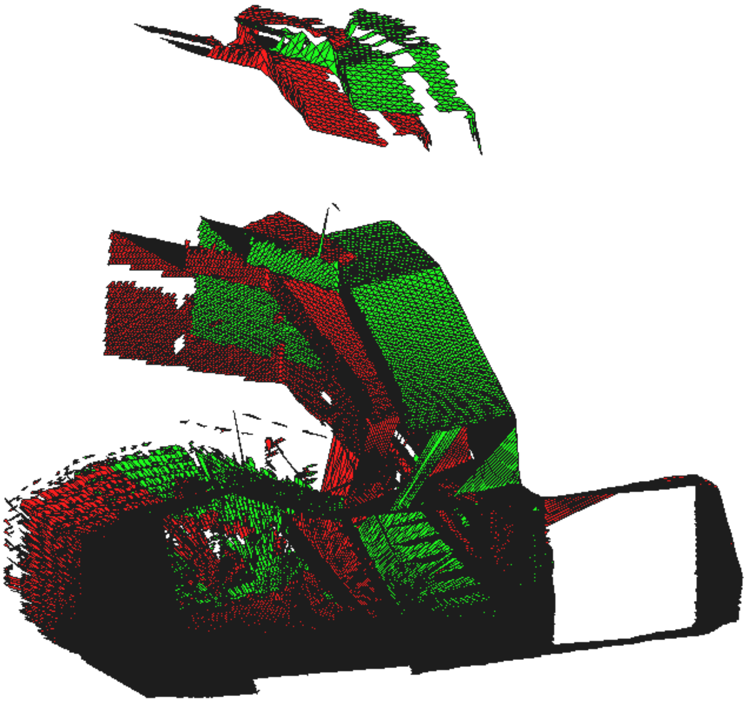


Figure 41: Multi-storey unregistered data and model sets: transformation approx. 15° rotation around z-axis direction

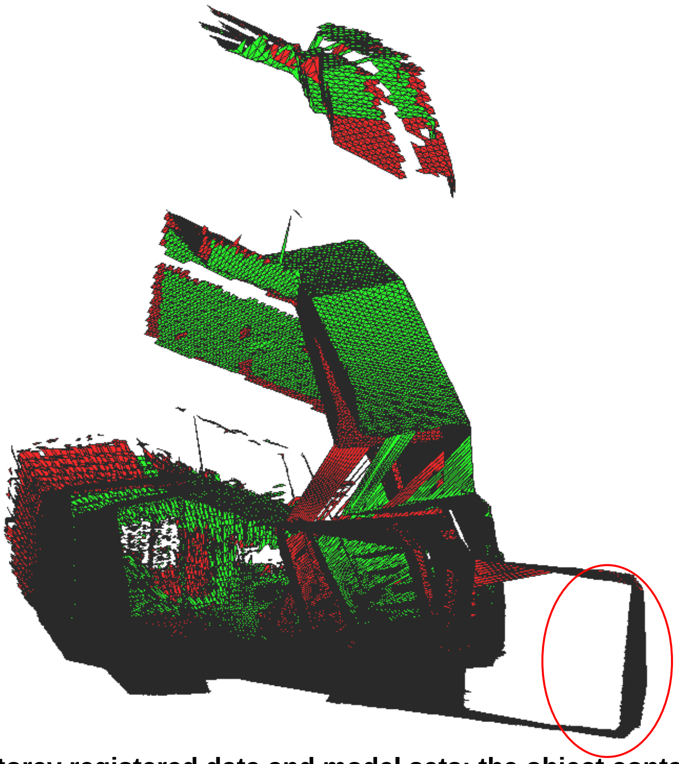


Figure 42: Multi-storey registered data and model sets; the object contained only in data is marked with a red circle

Second measurement

Another measurement was carried out in the same environment, applying different transformation on the scanning platform. This time the scale of rotation was 45°, translation vector was (0 cm, 30 cm, 0 cm). In the first phase of the extended method, the leveled maps were registered incorrectly since the rotation determined from the ICP algorithm was in the opposite direction than the correct solution, leading to a local minimum in the rotation of approx -60°. If the original method was run on the original data and model sets, the direction in which the sets were transformed in the 6DOF ICP was correct, though the method encountered local minimum before reaching the correct solution. The tall side walls in the environment also introduced rotation around y-axis. This can be seen in Figure 43. Also, one can notice the fractionalism in the bottom part of both model and data sets. This is due to a bar which enclosed many of the scattered objects (facility equipment). This difficult structuring makes the registration task very difficult since it can introduce many local minimums in which regular structures appear to be matched.

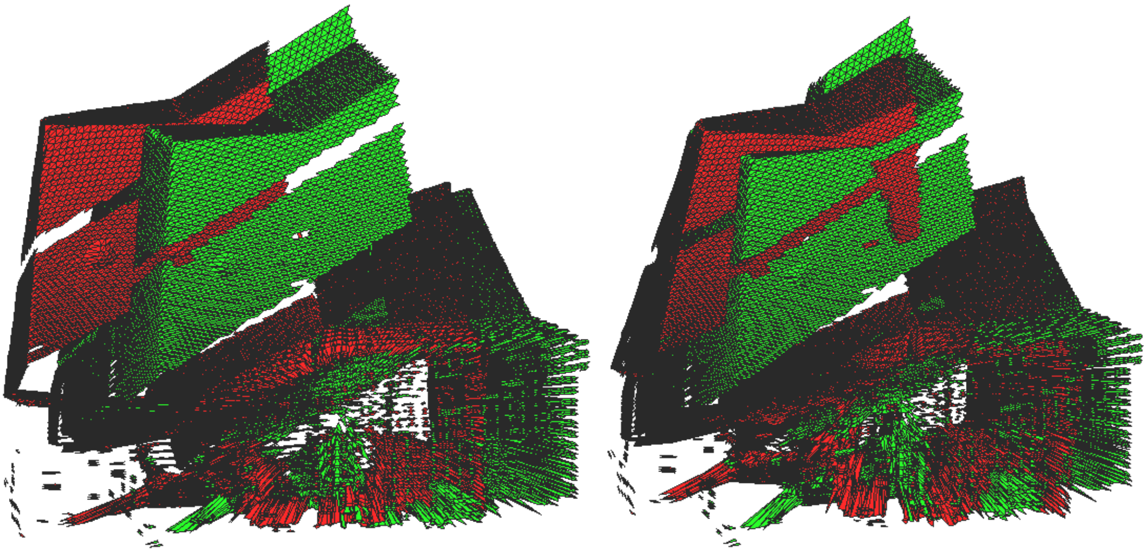


Figure 43: Second experiment in the multi-storey environment: unregistered model and data (left) and registered solution (right)

7 Conclusions and Future Prospects

Implemented navigation method

The implemented navigation method is in fact a hybrid two stage solution which uses factorization to accelerate the method for simultaneous localization and mapping in 6DOF. This is done by extracting 3DOF invariant vertical structures (landmarks) from the 3D data, creating a 2D map of vertical objects and registering these in 3DOF prior to the robust and computing time demanding registration in 6DOF. The aim of such modification was to pre-align the 3D data in 3DOF and therefore to reduce the time required for the 6DOF SLAM algorithm in this way while preserving the robustness of the original 6DOF ICP based method. The practical implementation of such a method requires use of accurate and inexpensive technologies of gravity vector measurement, initially aligning the data with a horizontal plane by hardware means. Selecting vertical structures for extraction ensures maximum height invariance of the created 2D data for the 3DOF registration. Features focusing on the quality of the final registration were also implemented as part of the method, such as the enforcement of uniqueness of model points' use in the closest point search of the ICP algorithm. The aim of this was to partly eliminate the errors introduced by occlusions and different fields of view of acquired images. The closest point search was also accelerated using kd-trees.

Evaluating by manipulating real data

Series of experiments was performed to asses the proposed method. In the first experiment, the overall behavior of the ICP method was evaluated. First, the ICP method in general was verified on real data, manipulating a real 3D image to form both data and model point cloud in order to provide ideal conditions for the ICP algorithm (existing correspondences for all data and model points). Then the performance of the method was observed manipulating the transformation between the data and model sets, concluding that the computing time (or number of iterations) dependency on the scale of the rotation in the transformation between the model and data sets can be approximated by a logarithmic trend. The introduction of factorization resulted in a significant improvement of the method's computing time requirements since the leveled maps pre-aligned the data saving many ICP iterations in

6DOF ICP. The scale of improvements was depending on the transformation between data and model, the higher the scale of rotation the higher improvement was achieved. The quality of matches in this experiment was perfect due to the existence of ideal closest model and data point pairs.

Evaluating in a hallway

Second experiment was performed to verify the behavior of the implemented navigation method in real indoor environment: an underground hallway. This time the scans for model and data were acquired separately as in real mapping situation. The results are very promising: not only the method was able to correctly register most of the acquired data pairs but the use of the extended method with implemented factorization has shown the improvement in average 30 % of the overall computing time. Nevertheless, a weakness of the method has also appeared in this experiment: this seems to be the inaccuracy of translation determination which can be explained by the fact that the centroids which are used to calculate the translation do not represent the identical physical part of the environment. Another cause of this error is the uneven distribution of measured points over the surfaces in the environment. In this experiment though, the introduction of enforcement of uniqueness of model points use in the closest point search proved to be significantly improving the quality of the match, partly eliminating the influence of the uneven point density and occlusions in the point clouds. The cost of this improvement is though increase in the computing-time required for the registration.

Laboratory environment experiment

In the third experiment, a laboratory and office environment with scattered objects were used to verify the behavior of the method in a randomly structured environment. The experimental results from two point pairs have shown that the scattered objects introduce significant error in the registration. Also, the use of factorized 3DOF SLAM in the registration of point clouds acquired in this type of environment did not appear to be accelerating the overall 6DOF registration. This was the only experiment in which the extended method did not accelerate the solution. A significant decrease in quality of the match also appeared in this registration, the ICP based 6DOF registration has output such rotation matrix that the point clouds were not correctly registered, “over-rotating” the data sets around wrong axis of rotation. The cause of this appears to be in the occlusions and also in the different fields of view when acquiring the model and the data range images, resulting in unique measurements of objects being present only in one of the two registered sets.

Multi-floor enviroment experiment

Originally expected to be the most difficult environment for the operation of the 6DOF SLAM method, a multi-floor environment with scattered objects enclosed behind metal bars and with staircase surrounded by glass banisters was selected for the third experiment. In this environment, two data pairs were acquired, applying different transformations on the scanning platform. One of the registrations performed reasonably well, registering the two pairs and slightly over-rotating the solution. Taking into account the difficult structuring of the environment, this was a positive result. The presence of multiple levels in the environment did not have a negative impact on the extraction of leveled maps, these were also registered coherently with the 6DOF final registration, significantly accelerating the overall computing time by more than 50 %. In case of the second pair where there was larger scale of transformation between the point sets, the registration did not succeed although the calculated transformation registered the sets approximately “half-way”. Unfortunately the 3DOF SLAM did not perform optimally for such a large transformation and actually proved to be decelerating the overall registration process.

Benefits and issues of the implemented method

Overall, one can conclude that the implemented method gives very promising results in terms of the expected ICP behavior. This means that when factorization described in this thesis is introduced in the SLAM method, the quality and robustness of the 6DOF ICP based SLAM is preserved and often the overall registration process is accelerated. In ideal conditions and in vertically structured environments, the implemented method is of a great benefit in terms of the computation power requirements. Also, the leveled maps can be used as a separate mapping output since these maps depict the vertical structuring of the environment and may be viewed as reduced 2D maps. The difficulties of the implemented 6DOF SLAM are tightly connected to the general issues of ICP use in the SLAM algorithms. On one hand the navigation method preserves the robustness of a general 6DOF ICP based SLAM but on the other hand it also shares the disadvantages of the ICP. The issues seem to be appearing when registering images of randomly and finely structured environments and also in registrations where the scale of transformation between the data and model sets is too large. In both mentioned situations occlusions are present in the point clouds and there are also objects appearing only in one of the registered sets, sometimes introducing error in the rotation determination but much more often in the translation calculation. The calculation of translation between the registered sets seems to be the most important issue of the ICP based method as resolving it within the ICP framework appears to be difficult due to the Catch 22 situation described in this thesis.

Future improvements

In the continuation of this research, this issue could be resolved by introducing other than ICP based registration method into the navigation system solely to help to determine the correct translation. Other option would be the development of a data reduction method which would initially determine common objects in data and model set.

Prospects of further factorization

Also an interesting extension of this research would be broadening of the conception of landmarks in unknown environments, extending the leveled maps generation. An option for complete factorization would be the introduction of horizontal objects extraction and further factorization of the 6DOF SLAM by adding a second 3DOF registration, adding one phase in the overall SLAM process. Such maps containing horizontal structures could be extracted based on the output from registering the leveled maps containing vertical objects. The extracted horizontal objects would then ideally be yaw invariant and the extraction could be performed based on the projection of horizontal object detection in the 3D data to X-Z (resp. Y-Z) plane. The created maps could then pre-register the 3D data in x and z (resp. y and z) axes translation and roll (resp. pitch) rotation degrees of freedom thus adding one yet unregistered degree of freedom (z-axis translation) and verifying the two already evaluated: x (resp. y) axis translation previously calculated in vertical object based leveled maps match, roll (resp. pitch) rotation (previously acquired by HW means – gravity vector).

Required reliability and modular deployment

The implemented navigation method was not expected to be operating as a standalone module on which the robot would be completely reliant. The research in the Laboratory of Telepresence and Robotics builds on the development of modular systems applicable on different robots allowing multiple configurations and levels of autonomy and teleoperation support. This is mainly due to the fact that teleoperation is still one of the most practical ways in which the robot can fulfill complicated tasks yet unsuitable for autonomous regimes. The current state of this research now allows the implementation of the developed and experimentally verified navigation method in the existing robot prototypes, supporting teleoperation and mapping of unknown environments. The module is expected to provide 3D images for the HUD displays of the operator’s console. The localization information will be combined with other pose calculations e.g. coming from the inertial navigation subsystem and GPS module and will inform the operator about the robot’s position. The mentioned errors in translation determination can be corrected by means of other measuring systems such as the 3DOF real-time SLAM, GPS and inertial navigation. This is in fact the major advantage of the interconnected modular approach to navigation.

Fitting on robots and future research

The system can be fitted on U.T.A.R., HERMES and Orpheus robotic platforms, allowing multiple teams to further enhance the navigation capabilities of the robots, e.g. by implementing path planning algorithms. It will also be possible to evaluate the method in outdoor environments using the mobile robots. The fitting of the method into the existing ARGOS control and monitoring software framework is the next step in the customization of the implemented method. Also, in near future the method will be tested with the new CCD based MESA-2000 sensor which can be integrated to considerably smaller robots, allowing further application of the navigation algorithm. This research was a “pioneer” work in the 3D range measurements based navigation at our department and as the development of the method progressed, it has helped to broaden the perspectives in the Laboratory of Telepresence and Robotics and opened new areas to explore.

References

[1] ALLEN, P., STAMOS, I., GUEORGUIEV, A., GOLD, E., BLAER, P. AVENUE: Automated Site Modeling in Urban Environments. In *Proceedings of the third International Conference on 3D Digital Imaging and Modeling (3DIM '01)*, Quebec City, Canada, May 2001.

[2] ALLEN, P.K.; STAMOS, I.; TROCCOLI, A.; SMITH, B.; LEORDEANU, M.; HSU, Y.C. 3D modeling of historic sites using range and image data. In *Proceedings of the IEEE International Conference on Robotics and Automation*. Vol. 1, ISBN 0-7803-7736-2. Pp. 145-150

[3] ARUN, K. S.; HUANG, T.; BLOSTEIN, S. D. Least-squares fitting of two 3-d point sets. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.9, no.5, ISSN 0162-8828. pp 698-700

[4] BALTZAKIS, H.; ARGYROS, A.; TRAHANIAS, P. Fusion of laser and visual data for robot motion planning and collision avoidance. In *Machine Vision and Applications*, Vol. 15, Springer, 2003. pp. 92-100

[5] BENTLEY, J. L. *K-d Trees for Semidynamic Point Sets*. In Proceedings of the 6th Annual Symposium on Computational Geometry, Berkeley, CA, 1990. Pp. 187–197

[6] BESL, P. J.; MCKAY, N. D. A method for registration of 3-d shapes. In *Proceedings of IEEE Transactions on Pattern Analalysis and Machine Intelligence* 14(2), Feb 1992. Pp 239-256.

[7] BIBER, P; ANDREASSON, H.; DUCKETT, T.; SCHILLING, A. 3D Modeling of Indoor Environments by a Mobile Robot with a Laser Scanner and Panoramic Camera. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'04),* Sendai, Japan, September 2004.

[8] BORENSTEIN, J. EVERETT, H. R. FENG, L. *“Where am I?” Sensors and Methods for Mobile Robot Positioning.* University of Michigan, Oak Ridge National Lab D&D Program, United States Department of Energy, USA, 1996

[9] *Comb Sort: Simplistic Sorting Algorithm*; online http://en.wikipedia.org/wiki/Comb_sort (14/10/2007)

[10] CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L. *Introduction to Algorithms*. McGraw-Hill, 1996: ISBN 0-07-013143-0

[11] EVERETT, H. R. *Sensors for Mobile Robots.* A.K. Peters Ltd., Natick, USA 1995.

[12] GIANCOLI, D. C. *Physics (Principles with Applications)*. Pearson Prentice Hall 2005. ISBN 0131846612

[13] Hokuyo Automatic Co. - Scanning Range Finder (SOUKUIKI Sensor); online http://www.hokuyo-aut.jp/02sensor/07scanner/urg.html (15/10/2007)

[14] LEBEDA, O. *Převodník USB-UART/FIFO – project report*. VUT Brno 2005

[15] MARTIN, C. M.; MORAVEC, H. P. *Robot Evidence Grids.* The Robotics Institute Carnagie Melon University, Pittsburg, 1996

[16] MESSA Swiss Ranger SR-3000. online http://www.mesa-imaging.ch/ (15/10/2007)

[17] MOLER, C.: *Numerical Computing with Matlab.* Society for Industrial and Applied Mathematics, 2004, ISBN-13: 978-0-898715-60-6

[18] MOORE, A. *Efficient Memory-based Learning for Robot Control.* PhD. thesis. Technical Report No. 209, Computer Laboratory, University of Cambridge, 2001.

[19] MOUNT, D.; ARYA, S.: *ANN: Approximate Nearest Neighbour Searching. Programming Manual.* online http://www.cs.umd.edu/~mount/ANN/ (1/10/2007)

[20] NUCHTER, A.; LINGEMANN, K.; HERTZBERG, J.; SURMANN, H. 6D SLAM with Approximate Data Association, in *Proceedings of the 12th International Conference on Advanced Robotics (ICAR '05)*, ISBN 0-7803-9178-0, Seattle, USA, July 2005a. Pp 242 – 249.

[21] NUCHTER, A.; SURMANN, H.; LINGEMANN, K.; HERTZBERG, J. Semantic Scene Analysis of Scanned 3D Indoor Environments, in *Proceedings of the 8th International Fall Workshop Vision, Modeling, and Visualization 2003 (VMV '03)*, IOS Press, ISBN 3-89838-048-3, Munich, Germany, November 2003. Pp 215 – 222.

[22] NUCHTER, A.; WULF, O.; LINGEMANN, K.; HERTZBERG, J.; WAGNER, B.; HARTMUT, S. 3D Mapping with Semantic Knowledge, in *Proceedings of the RoboCup International Symposium 2005*, Osaka, Japan, July 2005b.

[23] NUCHTER, A. LINGEMAN, K. SURMANN, H.; HERTZBERG, J. 6D SLAM – Preliminary Report on Closing the Loop in Six Dimensions. In *Proceedings of the IFAC/EURON Symposium, Portugal, 5 - 7 July 2004*. Elsevier, Amsterdam 2005. Pp. 203- 208

[24] PULLI, K. Multiview Registration for Large Data Sets. In *Proceedings of the 2nd International Conference on 3D Digital Imaging and Modeling (3DIM '99)*, Ottawa, Canada, October 1999. Pp 160 – 168.

[25] ROBERTSON, C.; FISHER, B. *Parallel Evolutionary Registration of Range Data*. Computer Vision and Image Understanding, Vol. 87, Elsevier Science Inc., New York, USA, 2002. Pp. 39 – 50

[26] SE, S.; LOWE, D.; LITTLE, J. Local and Global Localization for Mobile Robots using Visual Landmarks. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '01)*, Hawaii, USA, October 2001.

[27] SEQUEIRA,V.;NG, K.; WOLFART, E.; GONCALVES, J.; HOGG, D. Automated 3D reconstruction of interiors with multiple scan-views. In *Proceedings of SPIE, Electronic Imaging '99, The Society for Imaging Science and Technology /SPIE's 11th Annual Symposium*, San Jose, CA, USA, January 1999.

[28] SICK Technical Description: *LMS 200 / LMS 211 / LMS220 / LMS 221 / LMS 291 Laser Measurement Systems.* SICK AG – Division Auto Ident Germany. 8008970/06-2003

[29] SICK Telegram Listing: *Telegrams for Operating / Configuring the LMS 2XX Laser Measurement Systems.* SICK AG – Division Auto Ident Germany. 8007954/04-04-2003

[30] SOLC, F. ZALUD, L.: *Robotika*. Studying text. Brno University of Technology, 2006.

[31] STRANG, G. *Introduction to Linear Algebra*. 3rd ed., Wellesley-Cambridge Press, 1998

[32] SURMANN, H.; LINGEMAN, K.; NUCHTER, A.; HERZBERG, J.: A 3D laser range finder for autonomous mobile robots. In *Proceedings of the 32nd International Symposium on Robotics*, Seoul, Korea, April 2001. pp. 153 - 158

[33] THRUN, S.; HAHNEL, D.; BURGARD, W. Learning Compact 3D Models of Indoor and Outdoor Environments with a Mobile Robot. In *Proceedings of the fourth European workshop on advanced mobile robots (EUROBOT '01)*, Lund, Sweden, September 2001.

[34] WULF, O.; BRENNEKE, C.; WAGNER, B.. : Colored 2D Maps for Robot Navigation with 3D Sensor Data. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, September 28-October 2, 2004, Sendai, Japan.

[35] WULF, O.; WAGNER, B.. Fast 3D-Scanning Methods for Laser Measurement Systems. In *Proceedings of the International Conference on Control Systems and Computer Science (CSCS14)*, Bucharest, Romania, July 2-5, 2003.

[36] ZALUD, L.; KOPECNY, L.; NEUZIL, T.. Laser Proximity Scanner Correlation Based Method for Cooperative Localization and Map Building. In *Proc. of the 7th International Workshop on Advanced Motion Control,* Maribor, Slovenia, 2002. Pp 480-486

[37] ZALUD, L. RoboCup 2004: *Robot Soccer World Cup VIII. Chapter*: *Orpheus – Universal Reconnaissance Teleoperated Robot*. Springer-Verlag GmbH, Berlin, 2005, ISBN 978-3-540-25046-3. Pp. 491-498

[38] ZALUD, L.; KOPECNY, L., NEUZIL, T. 3D Proximity Scanner Integration to Rescue Robotic System. In *WSEAS Transactions on Systems*, ISSN 1109-2777, 2005. Pp. 43 - 48.

[39] ZHAO, H.; SHIBASAKI, R. Reconstructing Textured CAD Model of Urban Environment Using Vehicle-Borne Laser Range Scanners and Line Cameras. In *Second International Workshop on Computer Vision System (ICVS '01)*, Vancouver, Canada, July 2001. Pp 284 – 295

List of Author’s Publications

[1] JEŽ, O. Detekce spor v mikroskopickém obraze. In *Sborník prací konference a soutěže Student EEICT 2004*. Ing. Zdeněk Novotný CSc., Ondráčkova 105, Brno, 2004. Pp. 49-51. ISBN: 80-214-2634-9.

[2] JEŽ, O. Dynamic Modelling of a Multisensorial Nanomanipulator. In *Proceedings of the 11th Conference and Competition Student EEICT 2005*, Volume 2. Ing. Zdenek Novotny CSc., Ondrackova 105, Brno, 2005. Pp. 22-26. ISBN: 80-214-2889-9.

[3] JEŽ, O. Modelling and Simulation of a Nanometer Scale Manipulator. In *Proceedings Honeywell EMI 2005*. Ing. Zdenek Novotny CSc., Ondrackova 105, Brno, 2005. Pp. 124-128.

[4] JEŽ, O. VODA-BESANCON, A. BESANCON, G. MARLIERE, S. Dynamic Modelling of a Nanomanipulator Chain. In *Proceedings of the 16th IFAC World Congress, Prague 2005*. Elsevier Science, Oxford, GB, 2005. ISBN: 0-08-144130.

[5] JEŽ, O. Graph Cuts in 3D Magnetic Resonance Data Segmentation. In *Sborník prací konference a soutěže Student EEICT 2007*. Ing. Zdeněk Novotný CSc., Ondráčkova 105, Brno, 2007. Pp. 63-67.

[6] JEŽ, O.; GRAHAM, J. Interactive Segmentation of 3D MRI Images. In *Proceedings of the 8th International Carpathian Control Conference, Košice*. TU v Košiciach, Slovakia 2007, ISBN 80-8073-805-1

[7] JEŽ, O. Navigation of Mobile Robots Using 6DOF SLAM Accelerated by Leveled Maps. In *Proceedings of IETA 2007*. Published by Springer Verlag, 2008. (In press)

[8] NEUŽIL, T.; JEŽ, O. Data processing for mapping in mobile robotics. In *Proceedings of IETA 2007*. Published by Springer Verlag, 2008. (In press)

[9] JEŽ, O. 3D Mapping and Localization Using Leveled Map Accelerated ICP. In European Robotics Symposium 2008. *Springer Tracts in Advanced Robotics*. Springer, 2008. Pp. 343-353. ISBN: 978-3-540-78315-2.