Spatial Representation and Reasoning for Robot Mapping
— A Shape-Based Approach —

von Diedrich Wolter

Dissertation

zur Erlangung des Grades eines Doktors der Naturwissenschaften
— Dr. rer. nat. —

Vorgelegt im Fachbereich 3 (Mathematik & Informatik)
der Universität Bremen
im Juni 2006

Zusammenfassung

Die vorliegende Arbeit behandelt die autonome Erstellung einer Repräsentation räumlicher Information durch mobile Roboter. Die Raumrepräsentation, häufig schlicht Karte genannt, ist die zentrale Informationsquelle des Roboters für die Planung räumlicher Aufgaben, wie etwa Wegplanung in der Navigation. Eine zuverlässige Karte ist unabdingbar für intelligente, zielgerichtete Bewältigung von Navigationsaufgaben. Die Fähigkeit eines Roboters, diese Karte autonom erstellen zu können, ist ein wichtiger Schritt zur Entwicklung weitgehend autonomer Roboter, denn diese müssen sich auch in neuen, ihnen zuvor nicht bekannten oder veränderlichen Umgebungen zurechtfinden können.

Die autonome Kartenerstellung ist jedoch ein in vielfacher Hinsicht schwieriges Problem, das vielfältige Fragen aufwirft. Im Rahmen dieser Arbeit widme ich mich der Analyse und Weiterentwicklung der einer Karte zugrundeliegenden Raumrepräsentation und der darauf operierenden Schlußfolgerungsmechanismen. Ein zusätzliches Augenmerk bei der Weiterentwicklung ist die prinzipielle Eignung der Techniken, die interne Raumrepräsentation zu externen Karten (etwa Raumplänen) in Bezug setzen zu können und so dem Roboter eine Kommunikation mit Menschen oder anderen Robotern mit Hilfe einer Karte als Medium zu ermöglichen.

Die in den Disziplinen Raumkognition, Robotik und visueller Objekterkennung entwickelten Ansätze zur Repräsentation und Verarbeitung räumlicher Information werden durch diese Arbeit auf der Ebene von Forminformation verbunden. Es wird gezeigt, daß Forminformation ein wichtiges Bindeglied darstellt. Im Rahmen der visuellen Objekterkennung wurde ein umfangreiches und zugleich leistungsfähiges Repertoire an Techniken zur Verarbeitung von Forminformation entwickelt, insbesondere in Bezug auf Erkennungsaufgaben. Ich leite eine Raumrepräsentation auf der Basis von Forminformation her; Repräsentations- und Schlußfolgerungstechniken, insbesondere für Erkennungsprobleme, werden aus den Bereichen der Raumkognition und der visuellen Objekterkennung übertragen und weiterentwickelt.

Ein wesentlicher Schlußfolgerunsprozeß in der Kartenerstellung ist die Bestimmung von Korrespondenzen zwischen verschiedenen Beobachtungen oder zwischen Beobachtung und Karte. Die Korrespondenzbestimmung dient der Identifikation beobachteter Objekte und der Positionsbestimmung des Roboters. Auf der Basis der entworfenen Raumrepräsentation entwickle ich Techniken, die eine robuste Bewältigung des Korrespondenzproblems ermöglichen. Hierzu werden Methoden der Formerkennung, des qualitativen räumlichen Schließens und mathematische Ansätze zu Korrespondenzproblemen entwickelt und kombiniert. Ich entwickle eine graphentheoretische Charakterisierung des Problems auf der Basis von Hypergraphen und formuliere ein sogenanntes „Matching“-Problem. Zur Lösung des Matching-Problems wird ein für den Anwendungsfall relevantes Teilproblem identifiziert, dessen Komplexität polynomial in

Bezug auf die Anzahl der zu korrelierenden Objekte und zugleich handhabbar ist; der geführte Beweis stellt eine Verbindung zu Verfahren der kombinatorischen Optimierung her und erlaubt die Ableitung eines Algorithmus.

Die entwickelte Korrespondenzbestimmung bildet die zentrale Komponente in einem Ansatz zur inkrementellen Kartierung, der beschrieben und algorithmisch umgesetzt wird. Abschließend widme ich mich einer Evaluation des Ansatzes durch praktische Untersuchungen und zeige auf, welche Verbesserungen sich durch den neuen, in dieser Arbeit entwickelten Ansatz zur Kartierung erzielen lassen. In einem Ausblick zeige ich auf, wie sich die erzielten Ergebnisse in reale Systeme umsetzen lassen und welche weiteren Fragestellungen dabei aufgeworfen werden.

Abstract

This thesis addresses spatial representations and reasoning techniques for mobile robot mapping. It provides an analysis of fundamental representations and processes involved. A spatial representation based on shape information is proposed and appropriate shape analysis techniques are developed. Robot mapping's core problem of determining correspondences between observation and map is tackled on the basis of shape similarity. An improved matching technique is described by a generalized mathematical formulation. Specifically, it addresses the matching of configurations of extended geometric primitives' configurations.

Robot mapping describes the process of a robot autonomously acquiring an internal spatial representation of its environment. This internal representation, commonly termed "map", provides the basis for planning future actions. A reliable map is essential for intelligent navigation. Henceforth, autonomous map acquisition is one of the most fundamental tasks for autonomous robots. Unfortunately, it is among the most challenging tasks as well.

To build a map, robots rely on observations, typically obtained from their own sensors, but sometimes information can also be obtained by communication. The map is constructed by integrating multiple views on the same spatial environment into a coherent whole. This requires the correlation of observations, i.e. to determine which observations refer to the same physical entity—this is the objective addressed by the so-called *correspondence problem*. Additionally, multiple, corresponding observations need to be integrated into a single model; this task is termed the *merging problem*.

There are several aspects contributing to the difficulty of robot mapping. Among them, first of all, is the development of a robust and efficient solution to the correspondence problem. Another aspect is the necessity to handle uncertain information; virtually all information available to the robot must be considered uncertain. For example, sensor readings suffer from noise and undetermined failure which results in uncertain information by interpretation. Besides the effects of uncertainty to observations, the environment may simply change between observations, complicating recognition. Despite the high complexity faced in the robot mapping problem, a real-time solution is indispensable in many applications.

My work is dedicated to improving spatial representation and reasoning techniques underlying the robot mapping task. I argue for utilizing a shape representation originating in the field of object recognition; a strong, yet underexploited connection between the research fields of shape recognition and robot mapping is explored. Distinctive shape similarity information facilitates an efficient and robust approach to the correspondence problem. The goal is

to design a representation that mediates between metric sensor data and an abstract level of object-centered information. By doing so, a solid basis for an analytical approach to the correspondence problem is formed. In a mathematical framework of generalized bipartite graph matching, I develop an analytical solution to the correspondence problem.

Notably, mapping is no self-contained application. Maps are acquired to be used for navigational tasks. So approaches to mapping must be discussed in the context of intended navigational tasks. Most approaches to robot mapping aim at providing the most accurate map given a set of sensor readings, whilst deliberating about the trade-off between accuracy and computational effort. My work also looks at navigational tasks which include external, map-based information.

In an experimental section, the applicability of my approach to real-world mapping tasks is evaluated. Additionally, exploitation of coarse map information in localization tasks is examined.

Acknowledgment—Danksagung

Many contribute to the development of a dissertation, but some of them stand out. Christian Freksa, I thank you for supporting and encouraging my work, for introducing me to interdisciplinary work, for giving me the freedom to develop this dissertation, and for providing an enjoyable atmosphere to work in. Longin Jan Latecki, thank you for countless in-depth discussions helping me to develop and to position my work, for the fruitful collaboration, and for making a research stay possible that has been very valuable to me. I thank the research groups in Bremen and Philadelphia for helpful discussions and feedback, in particular Jan Oliver Wallgrün. I also thank Kai-Florian Richter, Sven Bertel, and Lutz Frommberger for feedback on this work. Robert Ross, thank you for helping to proof-read this dissertation.

Ganz herzlichen Dank auch an Jan-Hinnerk für seine Bereitschaft, auch noch zu fortgeschrittener Stunde grundsätzliche Details zu ergründen. Ursel, mein herzlichster Dank für Deine Unterstützung und Motivation — und dafür, daß Du da bist. Ganz besonderer Dank gilt meinen Eltern für die Unterstützung über all die Jahre — das hier ist für Euch!

Für Paul Wolter

Contents

Notation

$\{P_1, P_2, \ldots, P_n\}$	Set of polylines
$P_{i:j}$	Subset of polylines $\{P_i, P_{i+1}, \ldots, P_j\}$
$P_{\overline{i:j}}$	Polyline obtained by concatenating $P_i, P_{i+1}, \ldots, P_j$
$\sim$	Correspondence relation
L_P	Curvature length of polyline P
$d(p, q)$	Euclidean distance of p and q
T_P	Normalized tangent space representation of a polyline P, i.e. $T_P : [0, 1] \rightarrow [-\pi, \pi)$
$(x)_t$	Value of x at time step t
$(x)_{s:t}$	Subsequence $(x)_{s,s+1,\ldots,t-1,t}$
ξ	Robot pose, $\xi \in \mathbb{R}^2 \times [-\pi, \pi)$

Chapter 1

Introduction

Autonomous mobile robots are a highly important field of interest, not only for science, but also for industry or developers of household appliances. First consumer products like autonomous lawn mowers or vacuum cleaners are already available and are likely to mark the beginning of an era of many robot applications yet to come. Within the near future, our world may be populated by service robots which could be capable of fulfilling surveillance or rescue tasks in dangerous environments or which simply assist us in our everyday life. Realizing such applications requires answers to several research topics. Devising sensible means to handle spatial information is among the most fundamental research questions to answer, since robots need to interact with space. In addition to raising research questions, mobile robots also provide a powerful tool to evaluate theories of spatial information processing under real-world conditions. Many technical challenges in building mobile robots have been mastered to a degree that reliable robot platforms suitable for indoor environments are available off-the-shelf. Prototypical realization of an indoor robot application is no longer in first matter an engineering task, but a challenge to intelligent information processing. In this sense, my work addresses the research area of spatial cognition rather than robotics, although dealing with mobile robots in terms of experimental evaluation. In particular, I investigate questions of spatial representation, reasoning, and mathematical modeling of matching tasks.

Central for any agent acting in the world—may it be a human or a robot—is the model of the world, i.e. the representation, that the agent employs. This representation covers universal, environment-independent knowledge such as geometric principles and specific, environment-dependent knowledge such as positions of specific objects. Most importantly, this representation serves as the basis for reasoning about future actions. Its quality and comprehensiveness are key to planning purposive actions. From all the various modalities of information the world displays, spatial information is of special importance: Intelligent use of spatial knowledge is indispensable to purposive navigation. Navigational tasks can be solved only and yet purely in consideration of spatial knowledge. My work investigates spatial knowledge in the form of representation and reasoning techniques for mobile robot navigation, in particular the autonomous acquisition of spatial information.

1.1 Robot mapping

Intelligent navigational behavior of a robot requires a representation of the surrounding that adequately resembles the spatial properties of the environment. This representation is termed a *map* in the literature on robot navigation. The term map potentially stands for any kind of representation of spatial information.

The autonomy of a mobile robot is an integral part of intelligent behavior. Robots must be enabled to find their way unaided in any environment encountered, including environments that are a priori unknown. Therefore, robots must be able to acquire a coherent representation—an overview map—of their surrounding. This task is referred to as *robot mapping*.

Typical robots of today employ a wheeled drive and sensors to scan their surroundings. The drive suits indoor environments and comprises two or more independently powered wheels that allow the robot to travel on planar ground and, depending on the type of drive at hand, even to turn on the spot. To learn about their environment, robots obtain and interpret sensor readings. The kind of information that can be derived from this depends on the type of sensor at hand. Among the variety of sensors developed, devices scanning for obstacles in the robot's surrounding are most valuable and are commonly employed. Range sensors such as sonars and laser range finders provide this functionality by sensing reflections of obstacles; ultrasonic sound or laser light, respectively, is emitted and the time of flight for the reflection to arrive is measured. Knowledge about distance to obstacles easily allows for collision-free motion. Range sensors provide purely spatial information, they capture relative positions to obstacles and, depending on the sensor's accuracy and resolution, size or shape information. Additionally, most robots also employ an odometer, i.e. a sensor measuring the motion with respect to the ground; these sensors are said to provide odometry information.

Often, the only source of information regarded in robot mapping is sensor data obtained from the robot's own sensors. Yet, there are other ways to learn about an environment. A robot could communicate about the environment with fellow agents, may they be humans or robots. In the case of multiple, distributed robots jointly gathering spatial knowledge about their environment, the task is referred to as *multi-robot mapping*. To learn about its surrounding, a robot could also refer to an external map providing—maybe coarse—overview knowledge. The ability to exploit external maps is one interesting option to learn about an environment, since floor plans are widely available, e.g. emergency escape plans are mandatory in public buildings. These maps could provide overview knowledge to robots as well as to humans. From a slightly abstracted point of view, the three variants, single robot mapping, multi-robot mapping, and single robot reading maps can all be gathered by the term robot mapping. Communication can be regarded as to provide abstract sensor information and maps serve as a medium in (single-ended) communication. In the following, I subsume the different flavors of mapping tasks simply by the term robot mapping. The main focus throughout this work will be put on single-robot mapping, though.

In most situations, the complete environment cannot be perceived at once. To obtain a complete view on the environment, several observations at different perspectives are required. To construct a complete map, the individual observations need to be combined. Most sensors provide information in an agent-centered manner, i.e. in a local frame of reference. For example, this is the case for range sensors. Maps abstract from individual observations and represent in an absolute frame of reference, i.e. they provide survey knowledge. To obtain a map, sensor information needs to be collected while the robot travels through the environment and it needs to be transformed to the absolute frame of reference employed by the map. In other words, integration of local knowledge to survey knowledge is the central objective of robot mapping. Thus, robot mapping is a task of data integration.

A key to integrating observations is the ability to correlate them, i.e. to infer that two observations correspond to the same physical entity. The literature refers to this correlation as the *correspondence problem* or as *data association*, although usually not the sensor data itself is correlated, but first interpreted into more abstract information. Therefore, the term correspondence problem seems more appropriate to me and I will adhere to it in the following. When referring to the process of solving an instance of the correspondence problem, I use the term *matching*—matching establishes a correspondence. Beyond correlation of individual observations, the correspondence problem addresses correlation between observation and the robot's map; in my generalized understanding of robot mapping, the correspondence problem also covers the task of relating the robot's observation to communicated information. The correspondence problem has substantial impact on robot navigation, in particular on the ability of a robot to localize itself with respect to its map. Recognizing observed entities in the map would allow the robot to infer its location by aligning observation and map. Unfortunately, a sensible solution to the correspondence problem is among the hardest problems in robot navigation (for example, see Leonard et al., 2001; Thrun, 2002).

Clearly, a robot's perceptual capabilities dictate the applicability of specific approaches to tackling the correspondence problem. For example, the ability to unequivocally recognize unique landmarks would render recognition a trivial task. Sometimes, industrial applications rely on artificial aids to accomplish this, e.g. by attaching unique tags to objects (for example, see Hähnel et al., 2004). However, the requirement to carefully prepare an environment would be a serious threat to achieving popularity of service robots. Service robots for home use are required to master unprepared and changing environments. Environmental features that are already present must be exploited here.

Besides identification of correspondences, mapping comprises another task of integration. Having identified distinct observations that present the same physical entity, these observations then need to be combined into a single piece of information, i.e. they are merged; I refer to this task as *merging*. Observing a scene from different perspectives and combining corresponding observations yields the complete map desired. Merging multiple observations of the same object can improve the representation of the object, as for example measurement noise can be canceled out. Merging reduces the degree of redundant information introduced by repeatedly observing the same physical entity. As new sensor data continuously arrives and aggregates, it is essential to eliminate redundant information in order not to let the map representation grow unlimitedly. Merging is a non-trivial task in its own right, inescapable uncertainty in sensor data (e.g. measurement noise) essentially shaping its difficulty. Multiple observations of the same entity can differ and resolving these differences can be ambiguous.

To sum up, robot mapping is a comprehensive task of integration: integration of local to survey knowledge by discovering correspondences in observations and merging them into a coherent whole. Before detailing difficulties faced in this enterprise and drawing motivation of challenges, I interrelate mapping and self-localization. These are both fundamental tasks for autonomous navigation and their interrelationship essentially characterizes robot mapping.

1.1.1 Localization, mapping, and the correspondence problem

Besides characterizing robot mapping as comprehensive task of integrating spatial knowledge, mapping can described by its relation to the task of self-localization. This is a widely adopted view and it highlights the central role of the correspondence problem in robot mapping.

Self-localization is the task of answering the question “Where am I?”. Knowledge of one’s location is a prerequisite to intelligent navigation, making self-localization a fundamental problem in navigation. Any answer to the stated question “Where am I?” must employ a reference system of some kind. In the following, some single global reference system is assumed to ease the description. Assume a robot capable of observing non-moving objects of some sort. Observations provide information about positions of objects in an agent-centered frame of reference, i.e. the robot’s current position and orientation (jointly referred to as *pose*) defines the origin and orientation of this reference system.

Self-localization can be performed in relation to the robot's starting pose by keeping track of the robot's movements (*dead reckoning*). This can be performed by the robot in an indirect manner by observing the relative displacement of static objects while it is moving. Objects are tracked by correlating successive observations; this procedure is termed *pose tracking*. An alternative approach to self-localization, applicable if a map of the surrounding is available, is to directly correlate the current observation with the map and to infer the robot's pose by congruently aligning local observations and absolute map. In both approaches outlined, determination of correspondences is the central objective.

By determining the robot's pose with respect to the absolute reference system of the map, self-localization induces a transformation from the local reference system of the robot to the global one. Positions of observed objects are transformed, too. This renders mapping a simple task, if self-localization is provided: observed objects sharing the same position in the global frame of reference are identical, so mapping can be performed by repeatedly performing self-localization and registering new objects in the global reference system. Roughly speaking, mapping is just little more than self-localization.

Analogously, self-localization can also be regarded a side-effect of mapping: If mapping is provided, the robot can construct an absolute map comprising all its observations, i.e. any observed object is registered in the map using the map's absolute frame of reference. This includes objects currently observed by the robot, and, considering where the mapping procedure registered them in the map, the robot can infer its location by reversing this registration.

The question which of these, self-localization or mapping, is the more fundamental task, resembles the chicken-and-egg paradox (cf. Thrun, 2002). Their inherent connection have given rise to the term *SLAM*, simultaneous localization and mapping (Dissanayake et al., 2001), or *CML*, concurrent mapping and localization (Leonard et al., 2001), how the robot mapping problem is often referred to. However, it is possible to break through this interdependency by focusing on the correspondence problem. In any of the two perspectives, the role of correlating information is central.

To conclude, the two central navigational objectives self-localization and mapping are intimately connected. Both tasks depend on a solution to the correspondence problem. The difficulty to achieve a sensible solution to the correspondence problem significantly contributes to the difficulty of robot mapping. Although researched for many years, robot mapping has raised research questions yet unanswered. In the following, I elaborate in detail on the dimensions that make robot mapping as challenging as it is.

1.1.2 Challenges in robot mapping

On a first note, one must acknowledge that mapping has no end in itself, but that map information is acquired exclusively for the purpose of mastering other—mainly navigational—tasks. Therefore, mapping needs to be regarded in the context of the tasks eventually faced. These tasks pose additional demands on the mapping procedure. In the context of mobile robots, navigation can be regarded as an umbrella term for any task involved with letting the robot move from one place to another; this includes path-planning, obstacle avoidance, exploration, etc. Suitable properties for a map need to be determined that supports all facets of mapping and navigation. Balancing desired properties and realizing suitable techniques to accomplish them is challenging, but robot mapping already presents challenges per se.

1. Design of a compact spatial representation

The spatial representation employed for the map builds the basis for all processes involved in mapping and navigation, thereby essentially shaping all navigational procedures (including mapping) in terms of efficiency and effectiveness. Compactness of a representation supports efficient information processing. Regarding navigation, path-planning is of special importance. Path-planning aims to compute the shortest collision-free path between two locations; path-planning operates directly on the map representation and its efficiency is determined by the underlying map representation.

A great variety of spatial representations have been suggested for use in robot navigation. They are extensively reviewed in Chapter 2. Yet alone the diversity of existing and actively pursued approaches demonstrates that design and selection of a representation formalism is far from being trivial.

2. Efficiently process large amounts of sensor data, possibly in real-time

Ideally, robots should be able to cope with environments of arbitrary size and complexity. When traveling, a robot must continuously observe its surroundings to check for obstacles and to localize itself. Thus, new observations are frequently available and may need to be integrated into the map. This requires the processing of a large amount of sensor data.

The more objects are registered in the map, the more potential correspondence partners need to be regarded for each object observed. Computational resources required to solve the correspondence problem depend on the size and complexity of an environment. To handle large environments, efficient algorithms are required that present good scaling characteristics.

Providing up-to-date map information allows a robot to immediately respond to new information. This is important to exploration, i.e. the navigation strategy that enables a robot to learn its surrounding. An exploration strategy determines places, such that the entire surrounding is covered by observations. To determine previously unseen parts of the environment, exploration needs to repeatedly refer to a map comprising all previous observations. This requires fast execution of the mapping procedure.

3. Cope with changing & dynamic environments
In an environment subject to changes, the appearance of a place may vary. To recognize that place, potential changes need to be taken into account. This adds another dimension to the aspects that require consideration in the correspondence problem. The difficulty is to balance between judging an observation to correspond to a place (by interpreting potential mismatches as change) and judging the observation to *not* correspond to that place, because it appears differently.

Dynamic environments display continuous changes, for example people moving by. The robot must account for dynamics of objects when relating to their position, but deciding which objects are moving and which are static can be difficult. Mapping in dynamic environments requires to add temporal reasoning to the time-invariant mapping in static environments.

4. Handle uncertainty
If observed independently, the same physical object may appear differently due to reasons unknown. For example, an unnoticed alteration of environmental factors influencing the observation can cause inexplainable differences. These factors include misalignment of the sensor, accidental change of view point, or perhaps changes to the object. Even if environmental factors do not vary, any observation is subject to measurement noise. No measurement can objectively reflect physical reality. Therefore, any information obtained by sensors must be considered uncertain.

Unfortunately, uncertainty cannot be engineered away thoroughly, e.g. by improving on sensors and interpretation techniques, etc. However, the existence of uncertainty, alone, does not constitute the true problem. If a model for uncertainty was known, knowledge about this model could be exploited to cancel out the effects caused by uncertainty. The difficulty lies in designing such a model and finding computationally tractable means to exploit it. Many authors (for an overview, see Thrun, 2002) believe in handling uncertainty to be at the core of challenges in robot mapping; I elaborate on this view in little more detail in the next section when motivating my approach.

To put in a nutshell, the importance assigned to handling uncertainty is documented in the multitude of literature on (mainly probabilistic) approaches, indicating that managing uncertainty is a very hard problem.

5. Define plausibility of data integration and determine the most plausible map given the observations available

Many applications rely on a sufficiently detailed, coherent map. Often it cannot be specified in advance what sufficient detail and coherency exactly means in terms of a decidable criterion. So, mapping should aim to produce the *best* map possible. Unfortunately, potential changes in the environment and uncertainty in sensor data introduce ambiguity when interpreting observations, e.g. detecting the same object in multiple observations at different positions can be explained by changes of object location or by distortions of the measurement. In other words, there exists no canonical interpretation of the computational goal in robot mapping. However, any approach to mapping must define its computational aim. I refer to this definition as the model of *plausibility* in data integration. Different approaches have been taken to capture plausibility in a computational model. For instance, an interpretation in the style of Occam's Razor has been suggested to determine the minimal map consistent with the observations made (cf. Remolina & Kuipers, 2004). Plausibility has also been addressed in terms of probability theory; the most plausible map in dependence of observation has been described as the most probable map under side conditions of the observations made (cf. Smith et al., 1990). I explicitly address the definition of plausibility as a challenge. Computational modeling of what constitutes plausibility is the very point at which spatial understanding of the world gets introduced. We, the ones engaged in the robot mapping endeavor, model what appears to be a sensitive map and what appears not to be one. Plausibility of data integration is addressed in more detail in my review of state-of-the-art approaches to robot mapping and I explicitly discuss my interpretation for the approach suggested in this work.

6. Support communication about the environment

In many robot applications (e.g. service robots), robots are no self-enclosed systems that operate purely autonomously. To become useful, they need to interact with their fellow inhabitants. Human instructors must be enabled to adequately interact with the robot. Even though it may be a desirable long-term goal to set up communication of humans with robots in natural language, this is a research topic in its own right, going well beyond pure natural language understanding. Communication by means of maps provides an alternative that may be realizable with less effort. External maps like floor plans could be employed to instruct a robot, e.g. by indicating a location on the map. Furthermore, a robot could refer to an external map to learn about an environment. To make use of an external map, the robot needs to relate the map to its internal representation. Maps commonly employed for providing overview knowledge present information on a comparatively coarse level of granularity and abstract from small obstacles. To make use of such maps, techniques are required to interrelate spatial information across different levels of granularity. In general, enabling robots to communicate about their environment is a very tough problem. For example, multi-robot mapping is considered an open research issue (cf. Thrun, 2002), although it only handles the restricted case of communicating (homogenous) robots. This flavor should be particularly easy, since communication partners can be designed to utilize the same computational models and the same sensors.

To sum up, challenges in robot mapping can be inherent to the task itself such as design of an adequate spatial representation and efficient techniques for integration of new information. Handling environments that change over time adds another dimension of aspects to take into account in the mapping procedure—in the presence uncertainty, ambiguities may arise. This makes it essential to devise a sensible model of plausibility in data integration and to determine appropriate means to compute the most plausible map. In the following, I elaborate in little more detail on models of plausible data integration and give an example of the facets fostered in my dissertation.

1.2 The spatial cognition perspective—motivation

Robot mapping is a complex problem, with many aspects contributing to its difficulty. Depending on which aspects are focused upon, different perspectives on robot mapping present themselves. In the following, I develop the view taken in my dissertation and illustrate key questions that are central to my work.

First, I examine the most popular perspective taken on robot mapping; it is to focus on handling uncertain information. One can argue that uncertainty in sensor data is the source of difficulties faced in robot mapping. If all information available was certain, it could be interpreted in a straight-forward manner. If further this information was sufficiently detailed and extensive, mapping could be easily realized. Assume the robot's initial pose was known and assume the robot had the ability to exactly sense its movement. Then, the robot could easily infer its location or, put differently, it could easily perform self-localization. As discussed, mapping is no longer difficult, given that self-localization is provided. In typical robots, odometry information is available to learn about the robot's movements. Odometry information is widely derived from shaft encoders that count the revolutions of the wheels. As these are real sensors, they suffer from distortions, for example caused by slipping wheels. Uninterpreted use of sensor information is not possible, since errors accumulate and have a gradually increasing prejorative effect on the mapping process. To make direct use of this information, techniques need to be devised that shift the information to a level of sufficient detail and reliability. As of today, statistical approaches dominate this task. Statistical methods are employed to model, to "explain", and to correct for sensor data—I present these techniques in more detail in Section

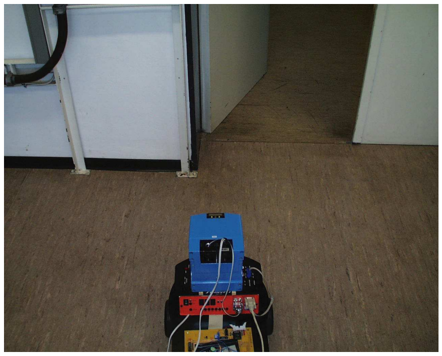
3.1.1. Put differently, aiming to engineer away uncertainty in sensor information would allow conquering robot mapping in a straight-forward manner, given that sufficiently detailed and extensive sensor information is available. In the context of reviewing robot mapping, I discuss this perspective in more detail (see Chapter 3).

An alternative perspective to robot mapping presents itself when focusing on the problems of spatial information processing faced in robot mapping. I argue that determining sophisticated techniques to process spatial information is also beneficial to mastering uncertainty. Returning to the aforementioned example of odometry sensors, I pose the following contemplation: suppose, robot mapping can be performed without the use of odometry information at all. In this case, no techniques are required to correct for sensor readings obtained from odometry sensors in the first place, saving computational effort. Failure of odometry sensors cannot disturb the mapping process. Furthermore, if odometry information is available it can be exploited in terms of an independent source of information that could also be used to check the mapping process. Suppose further, one can abstract from details in the robot's observation that are likely subject to measurement errors. The ability to abstract from such details adds to the robustness of mapping. As Freksa (2004) notes, uncertainty in spatio-temporal domains is almost always a function of a spatio-temporal vicinity. Thus, a suitable spatial abstraction can overcome uncertainty in spatial domains. Generally speaking, relaxing the requirements on the sensor data by abstracting from metric details yields an approach that is robust to uncertainty in the sensor information. Sensible spatial reasoning attacks the core of uncertainty, whereas stochastic frameworks address the effects of uninterpreted use of uncertain information.

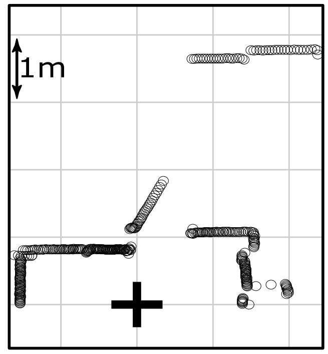
The two outlined perspectives are not conflicting, though. Advances in handling of uncertain information and advances in sensible processing of spatial information are beneficial to each other. Allowing to relax the requirements on sensor data requires less sophisticated (and computationally expensive) means to correct for it. Analogously, improving means to correct for sensor data provides more reliable information to the processing of spatial information.

In this dissertation I argue that sensor information available from widely employed laser range finders is very rich information that allows reliable robot navigation solely employing this source of information. Abstracting from raw sensor information to a level of shape information provides a solid basis to devise techniques for robot mapping that are per se robust to uncertain information. I will illustrate this in the following example.

In this example, I consider a robot equipped with a laser range finder—an illustration of such a robot is presented in Fig. 1.1 (a). Laser range finders provide today's richest range information by densely scanning the surrounding using time-of-flight measurements of a laser beam. In the case of the SICK



(a)



(b)

Figure 1.1: (a) Mobile robot sensing the environment. (b) Sensor information from the laser range finder, the cross demarks the robot's position.

LMS-200 laser range finder mounted to the robot presented in Fig. 1.1 (a), the sensor provides a field of view of 180° and is capable of sensing objects as far as 80 meters away with an error of ±15mm according to the technical data sheet. The sensor combines 361 individual range measurements to sense the 180° field of view. An exemplary sensor reading is depicted in Fig. 1.1 (b); it presents the view of the robot depicted in Fig. 1.1 (a). In an experimental setup, a robot is positioned in a hall at four different poses and sensor readings are obtained. In Fig. 1.3 (a) the robot is depicted in this environment, Fig. 1.2 (a) – (d) presents the four range scans obtained in the local perspective of the robot. In robot mapping, the task is to assemble individual observations to a coherent whole. As the reader may verify, humans are easily able to solve the task of congruently assembling the four range scans; the solution obtained by overlaying the four images is depicted in Fig. 1.3 (b). Note that humans may not have a detailed knowledge about the sensor, nor does he or she require any additional information, e.g. the relative position of the four view poses. Naturally, the question arises: what makes it possible to solve this task? In this example, one factor may be seen in identifying the salient object in the center of each of the range scans[1]. Recognizing this object provides sufficient information to congruently assemble the scans. Based on the identified correspondence, the scans can be easily aligned and the solution depicted in Fig. 1.3 (b) is obtained. This suggests that the sensor information is abstracted to an appropriate level that allows the identification of individual objects which can robustly be recognized.

In this dissertation, I tackle the question of which techniques of representing

[1] Grouping is required to identify an object based on individual range measurements that provide distinct points on the object boundary. In the context of human vision, these aspects are addressed by research on the so-called Gestalt laws (Wertheimer, 1925). However, in this example I abstract from the actual grouping

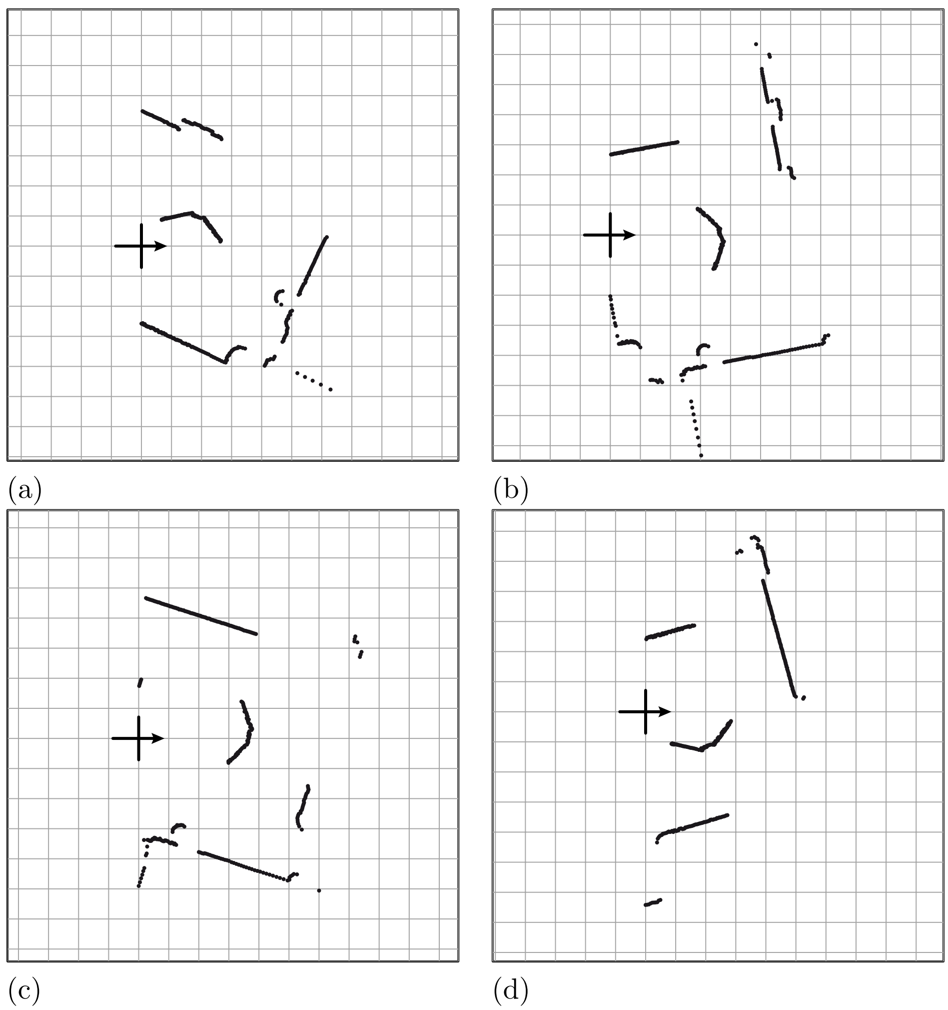
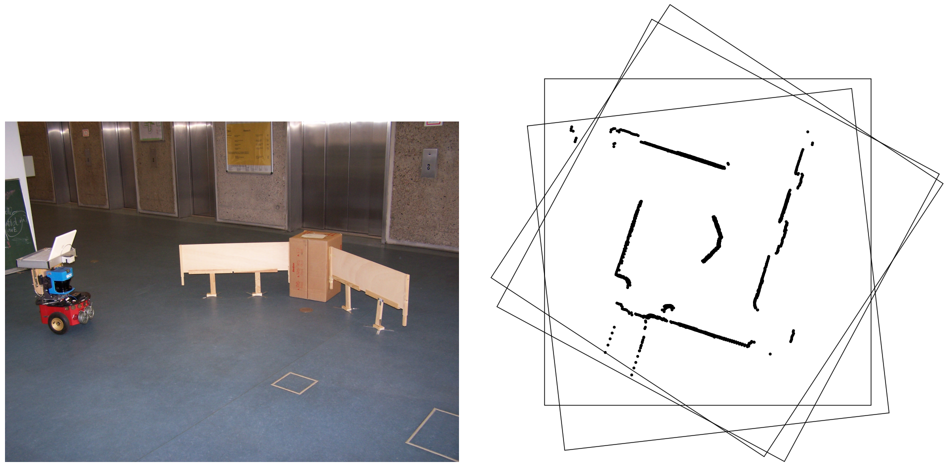


Figure 1.2: (a) – (d) Sensor readings of a laser range finder in a local coordinate system. The robot's view position is indicated by crosshairs, grids denote distances of 1 meter. The salient arc-shaped object in the middle of each of the scan makes it easy for humans to congruently assemble the scans.



(a) (b)

Figure 1.3: (a) Image of the robot in the test environment (b) The laser scans depicted in Fig. 1.2 are manually assembled to an overview map.

and of reasoning about spatial information provide such functionality, and of how this can be realized in robot mapping. To accomplish this, I aim at bringing together techniques originating in the fields of robotics, spatial cognition, and shape analysis as considered in visual object recognition. I argue that shape analysis provides a solid basis to tackle the outlined recognition task.

1.3 Shape analysis for robot mapping

Shape information, its representation, and, most importantly, recognition processes are of high importance to the field of computer vision, especially to object recognition. Shape is often considered *the* most important property of an object visually perceivable (for example, refer to Palmer (1999, p. 363)). Recognizing the shape of an object is often sufficient to recognize the object itself, as shape offers characteristic information. Shape information is rich and contributes more to recognition of objects than other object properties like color or texture. It is regarded to be the most relevant aspect to consider in object recognition and has been studied intensively.

Besides the great expressiveness of shape information, there are more aspects that contribute to the relevance of shape. First of all, shape information can be considered on various levels of abstraction. An illustration for this, following an example stated by Siddiqi et al. (1999b), is the ability to recognize a dog by briefly seeing its silhouette passing by. We may not have recognized the dog's specific breed on a glimpse, though.

To accomplish recognition across different levels in granularity, methods for changing that level have been developed, for example by means of shape simplification. Simplification can also allow for representing shape information in a compact manner by abstracting from irrelevant details.

Recognition processes examine shape similarity by means of a shape distance measure, a function which, given two shapes, determines a non-negative shape difference value. For example, shape distance measures allow for "retrieval by example" database queries (see Gottfried, 2005; Latecki & Lakämper, 2006a). Robustness and distinctiveness in recognition by means of shape similarity has been achieved to a comparative high level and is, for example, documented in the studies on MPEG-7 shape descriptors (Bober et al., 1999; Latecki et al., 2000b). Availability of sophisticated shape distance measures significantly contributes to the importance of shape information in object recognition. The key role of shape distance measures suggests a typical interpretation of a shape representation to comprehend the actual representation technique and the shape distance measure; this resembles the understanding of abstract data types in informatics where data representation and functions operating on the representation are coupled analogously.

In view of robot mapping, shape information offers an interesting perspective: shape information represents spatial information in the context of a complete object, it provides rich information and links to sophisticated recognition techniques originating in the research field of object recognition. These techniques could contribute to improving techniques addressing the correspondence problem in robot mapping. Representing a robot's internal map based on shape information, shape retrieval techniques may be applied to recognize places registered in the map. The ability to handle shapes on various levels of granularity gives rise to the expectation that sources of pictorial information such as external overview maps can also be integrated in the robot mapping or localization task.

The interrelation of robot mapping and object recognition, and shape representation in particular, has not been investigated deeply yet, though the link between the fields of robot mapping and computer vision is stated in the literature; Thrun (2002) claimed the connection between the two fields to be underexploited. One reason for this deficient exploitation may be the great advances in statistical frameworks for robot mapping that can even make up for shortcomings of the underlying spatial representation to a certain degree; much work conducted in robot mapping aims at improving statistical frameworks. In contrast, my work presents a different, somewhat cognitively motivated approach by focussing on the recognition of environmental features and their spatial configuration; I explicitly aim at bringing the research fields of robotics, spatial cognition, and objective recognition together to jointly attack the robot mapping problem.

1.4 Research question & thesis

My work approaches the challenges characterizing the robot mapping problem by developing techniques for spatial information processing. My research questions are as follows:

- What is an adequate map representation for robot mapping?
- How can a solution to the correspondence problem robustly yet efficiently be determined?
- Which properties allow a map representation to be related to (potentially coarse) external maps?

My aim is to develop a suitable spatial representation for robot maps that facilitates a robust solution to the correspondence problem. I pursue the development of an approach to the correspondence problem that is efficient and robust against uncertainty inherent to sensor information. My central thesis is: shape analysis provides a solid basis to accomplish this aim. Shape information provides means to mediate between metric sensor data and a more abstract object-centered representation.

To tackle the correspondence problem, I pursue to combine reasoning on the metric level (shape similarity) and on the object-centered level (qualitative reasoning about configuration of objects). My thesis is: improving on spatial representation and reasoning is a valuable contribution to robot mapping. First, by improving general robustness of matching techniques addressing the correspondence problem, less demanding means of tackling remaining uncertainty are required, as less alternative interpretations appear plausible from the suitably abstract point a shape-based map representation offers. Second, by relaxing the requirements for obtaining a sensible solution to the correspondence problem, the capabilities of mapping are advanced towards utilization of external, maybe coarse, map information or towards multi-robot mapping.

My work has been motivated by the assumption that there exists an underexploited interrelation between processing small-scale spatial information in vision applications and medium-scale spatial information in robot mapping. Additional motivation has been drawn from the observation that spatial reasoning available for object-centered representations could be exploited in approaches to the correspondence problem. These techniques enable adequate and yet computationally tractable modeling of plausible data integration in the mapping process. To substantiate the thesis:

1. A connection exists between mapping using range information and visual object recognition on the level of shape information.
2. A spatial representation based on shape information is well-suited to robot mapping and navigation and it allows utilizing external maps.

3. Sophisticated shape analysis originating from the field of computer vision can be transferred to the robot mapping domain.

4. An analytical, efficient, and robust approach to the correspondence problem can be designed on the basis of spatial reasoning and shape analysis.

5. Sophisticated matching strategies substantially attack the correspondence problem and allow for robust self-localization in context of relaxed requirements on input data. In particular, the absence of odometry information can be mastered.

1.5 Contribution of this dissertation

My dissertation contributes to current research questions in the field of spatial information processing as follows. I demonstrate that there exists a close connection of robotics, computer vision, and spatial cognition; all disciplines contribute to mastering robot mapping. In particular, visual object recognition is shown to be tightly linked to spatial information processing in robot mapping, allowing advances to be shared. I demonstrate that shape information, bridging the gap between low-level metric information and abstract object-centered knowledge, provides a solid basis to robot mapping. I adopt shape analysis techniques, tailor them to the robot mapping domain, and develop a new shape distance measure that is particularly suited to recognize polylines lacking of rich shape information in the presence of uncertainty. The advances achieved in the context of robot mapping are also beneficial to visual object recognition.

I introduce an efficient approach to correspondence problem that regards n-to-m-correspondences of objects. A graph-theoretic formulation is presented and a polynomial-time matching algorithm is derived. This technique is a well-suited theoretical foundation for tackling the correspondence problem. Besides application in robot mapping, the matching technique is a contribution to general matching tasks.

With respect to robot mapping, the developed matching technique provides means to base the map representation on extended geometric primitives, and to autonomously construct such map representation. Such object-centered representations are compact and appear to be a necessary foundation for abstract reasoning processes. On the basis of universal shape features, an incremental approach to robot mapping is developed. The developed spatial reasoning techniques are robust and capable of abstaining from odometry information in localization and mapping. The shape-based representation is capable of mediating between fine-grained sensor information and coarser information, such as represented in external maps commonly employed by humans. The developed techniques are a step towards more intelligent robots, which are able to communicate with humans or fellow robots, using a map as medium in communication.

1.6 Application scenario

To address my research questions, I regard typical wheeled robots equipped with a single laser range finder. Detailed knowledge about specific physical characteristics of the robot is not presumed, e.g. knowledge about motion characteristics such as accuracy of performed motion commands or reliability of odometry. The mapping procedure shall not require availability of odometry information, let alone assume a specific quality thereof.

The robot is assumed to sense its surrounding in a plane parallel to the ground using the laser range finder; the mapping process aims to acquire this two-dimensional perspective. No precise knowledge on the characteristics of the range sensor shall be required. In this work I assume a detailed range image to be available, abstracting from technical means to obtain it (e.g. choice of sensor, preprocessing of sensor data). In principle, other means than laser range measurements could be employed to obtain a detailed range image. Misalignment of the scanning device (for example caused by rough underground) is not explicitly addressed, but regarded in terms of general robustness. Generally, a robot with a statically mounted laser range finder meets these conditions if it operates in an environment with a planar ground surface. My work regards indoor environments to easily obtain suitable range images.

Different indoor environments are considered to foster a broad range of potential applications; preparation of the environment (e.g. by installing beacons) is not considered. Changing environments are considered in terms of robustness in recognition. The mapping procedure shall identify a local surrounding also if it is subject to moderate changes. Dynamic environments are not addressed. No specially tailored strategy for exploration or navigation shall be required for a robot to utilize the developed representation and reasoning techniques.

1.7 Research methodology

This work focuses on techniques to represent and reason about spatial information. In particular, literature from the research fields of robotics, shape analysis, object recognition, and qualitative spatial reasoning is investigated. My research proceeds on a theoretical level, on a computational modeling level, and on an experimental level.

On the theoretical level, a mathematical approach to the correspondence problem is formulated, using the techniques of graph-theory. I investigate graph-theoretic matching problems and adapt Dynamic Programming techniques connected to the field of operations research.

On the computational modeling level, I develop techniques for shape analysis. Besides their application to robot mapping, these techniques are strongly connected to the field of visual object recognition.

On the experimental level, I evaluate shape-based mapping. Using real as well as simulated sensor data, I examine my approach to robot mapping and relate it to other approaches developed in the field of robotics.

1.8 Structure of this dissertation

My dissertation is structured as follows: the second chapter is dedicated to a review of relevant approaches to representing spatial information in robot maps. I propose a classification for map representations and interrelate approaches from the field of robotics and shape representation. Approaches to representing spatial information are examined as regards their potential contribution to achieving my research aim. Finally, I select an appropriate representation. Clearly, the utility of a representation is determined in context of the processes that operate on it. In context of Chapter 2, I consider navigation and mapping processes on an abstract level.

In Chapter 3, mapping processes operating on the map representation are reviewed in detail. I approach robot mapping from an algorithmic perspective and investigate into the manifold techniques suggested for map construction. Means to tackle uncertainty and to process spatial information are analyzed. To structure robot mapping, I propose a decomposition into functional components which can be examined individually. Techniques for the interpretation of sensor information, for tackling the correspondence problem, for localization, for map update, and mapping architectures are discussed.

In Chapter 4, I elaborate on a mathematical foundation of matching to seize matching techniques theoretically. I devise a mathematical framework to approach the correspondence problem that suits the demands concluded from the reviews in Chapters 2 and 3. I formulate my approach to the correspondence problem in a framework of graph-theoretic matching problems. Finally, I propose an algorithm to solve such problems in the context of Dynamic Programming techniques and prove its correctness.

In Chapter 5, the computational modeling of my approach to robot mapping is presented. First, I specify the aim in computational terms, i.e. I present my interpretation of a plausible data integration. Then, the components of the computational model and their interrelationship are presented. Finally, components are presented in detail. I describe extraction of shape information from laser range scans, detail matching of observation and map on basis of the theoretical framework developed in Chapter 4, alignment of observation and map, and merging of corresponding shape information.

Chapter 6 is dedicated to an experimental evaluation of my approach. I evaluate the computational model in localization and mapping experiments. Examining self-localization, I investigate the capability of the developed techniques by comparing results obtained by my approach to state-of-the-art techniques employed in self-localization. In one experiment, the capability of utilizing external map information is investigated. In mapping experiments, I evaluate my approach using simulated and real-world sensor data.

My thesis concludes by summarizing the results with respect to the research questions and by giving an outlook. I present research questions raised through this work and discuss relevant research questions that benefit from the results achieved in this work.

Chapter 2

Spatial representations for mapping

This Chapter provides an overview of the multitude of spatial representations suggested for application in robot mapping. The goal is to outline a representation technique well-suited to my research goal. To accomplish this, a scheme for analyzing spatial representations in robot mapping is introduced which aims at deriving characteristic properties for classes of representations. Three aspects of a map representation are individually examined:

- Feature representation
- Configuration representation
- Map organization

On the level of feature representation, I discuss the selection of environmental features that can be registered in the map. On this level, a link to shape representation is established. After reviewing the robotics literature, I turn to shape representations employed in visual object recognition and discuss parallels in order to obtain an integrated view. With respect to the mapping task, I elaborate on the potential merits or implied complications of individual approaches to feature representation.

Configurations define the spatial relationship of features in the map. I outline which information is relevant to the mapping task and how it can adequately be represented.

Map organization finally specifies how representation techniques for feature and configuration knowledge can be composed to constitute the map representation. Possible combinations range from uniform coordinate-based configurations of landmarks to maps incorporating hierarchically organized, distinct layers of feature and configuration representations; implications on mapping and navigation processes are detailed.

To evaluate the approaches, I consider my research questions as regards design of the underlying spatial representation: Which representation is adequate for the mapping task and assists the involved reasoning? Which representation supports reasoning for intelligent navigation? Which properties of spatial representations support solving the correspondence problem robustly and efficiently? Which representations can be related to external map information? In the context of the three individual aspects, I will develop precise criteria.

2.1 Feature representation

Sensor data is interpreted in terms of environmental features. Features can range from hardly interpreted sensor patterns to complex objects that require sophisticated interpretation techniques for detection. To start with, a first distinction of map features is to differentiate between spatial properties (e.g. position, size, shape) and non-spatial properties (e.g. color, object category). My dissertation aims at advancing spatial representation and reasoning for map acquisition and therefore my review is restricted to spatial features. I concentrate on features suitable to describe unprepared environments that can be perceived by robots as well as by humans. Indeed, recognition processes could also benefit from exploitation of non-spatial properties that complement spatial information, e.g. regarding color to disambiguate objects. However, intelligent processing of spatial information is one fundamental and indispensable ingredient to successful self-localization, mapping, and navigation.

Several factors need to be regarded for feature selection. By exploring these factors, the adequacy of a specific approach to map feature representation can be evaluated with respect to my research aim. In detail, I examine the following questions:

- How many map features are required to model an environment?
- Are individual features distinguishable?
- Is the feature representation universal, i.e. are the features detectable in any potential working environment?
- How is partial visibility (occlusion) handled and what are the effects of view-point variations to feature appearance?
- Can the features also be extracted from external maps?
- Can the features be related to knowledge on differing levels of granularity?

By considering the effects implied by specific answers to these questions, desired properties for map features can be derived. Unfortunately, some desired properties can be conflicting. Therefore, advantages and disadvantages need to be carefully balanced.

Particularly a small amount of features and distinctiveness of features ease mastering the correspondence problem. Decreasing the number of features involved, decreases the search space of potential correspondences. This can be achieved by favoring a representation using few features over a representation using many features. However, reducing the number of features in a map also increases the adverse effect a single erroneously matched feature can cause.

Distinctive features allow correspondence search to restrict consideration to—typically few—similar features. Therefore, complex features that comprise rich attributes support a robust and efficient solution to the correspondence problem. Optimally, unequivocal feature recognition would allow for a straightforward assignment of corresponding features. Engineering away feature ambiguity is prominent amongst industrial applications, but relies on installation of artificial unambiguous features in form of tags in the environment (for example, see Djugash et al., 2005; Hähnel et al., 2004; Rafflin & Fournier, 1996).

By employing a universal feature representation, any environment can adequately be modeled. This ensures a wide range of applications. However, less universal but more specific features can be valuable, too. Features tailored to specific environments provide means to compactly represent these environments by grouping comprehensive observations to compact feature representations. Examples include ellipses in range images to model profiles of trees in parks (Forsman & Halme, 2004) or lines to represent straight walls in indoor office scenarios (e.g. Röfer, 2002). In both cases, points detected by a laser range finder (LRF) are grouped to few geometric primitives, allowing disregard for the comparatively large amount of points detected initially. Specific feature representations provide a compact approach for specific environments, but their application is limited to environments that present the specific features considered. So, universality vs. specificity of geometric primitives needs to be considered in the context of domain restrictions and compactness of the feature representation.

In any observation, parts of the environment are occluded; objects may appear differently depending on the view point. Therefore, it is important to regard partial occlusion and different appearances of features. This can be achieved by means of a representation that retains capability of feature recognition under varying conditions of observation.

With regards to integration of externally supplied map information that provides information on a different level of granularity than employed by the robot, granularity-invariant or granularity-adaptive feature representations are helpful for interrelating information.

To conclude, the criteria for feature selection can be competing and require careful balancing. In the following, I discuss specific approaches in detail. From the in-depth examination, I derive an adequately balanced approach.

2.1.1 Raw sensor patterns

Approaches memorizing sensor patterns are often biologically inspired (e.g. Bachelder & Waxman, 1994; Franz et al., 1998; Mallot et al., 1995), since there exists some evidence that animals memorize specific views (Cheng, 1986; Hermer, 1997; Margules & Gallistel, 1988; Schölkopf & Mallot, 1995) on the environment. In robot implementations, sensor patterns are transformed to views with hardly any interpretation. Approaches constructing a map by registering views have been termed *view-based* approaches[1] or are referred to as approaches constructing *robot-centric maps* (Thrun, 2002, p. 2).

In view-based approaches, sensor snapshots are stored for different discrete locations of the robot; a new location is encountered whenever the current view differs from the view obtained at the previous location by more than a fixed threshold with respect to some difference measure for views. For example, Franz et al. (1998) handle linear panoramic images acquired by a camera. Vector distance of a greyscale image vector serves as difference measure. Similarly, Matsumoto et al. (1999) memorize images from an omnidirectional camera.

2.1.2 Landmarks

In general context, landmarks are salient objects in space that are easy to identify. They can be represented by their position and, optionally, landmark signatures. Human navigation typically utilizes landmarks as environmental features for localization, particular with respect to route directions (see e.g. Denis, 1997). Humans employ a rich repertoire of landmarks, e.g. "the hardware store", "the lighthouse", etc.

In robotics, different kinds of landmarks have been employed. Forsman et al. proposed a tree detection that has been tailored to a park scenario (Forsman, 2001a,b; Forsman & Halme, 2004). Corners in an indoor environment have been suggested as landmarks and detection based on range data has been developed (Altermatt et al., 2004); Jefferies et al. (2004b) utilize salient object corners and regions of relatively constant color and texture as landmarks and present techniques to extract these landmarks from stereo vision.

[1] In computer vision, approaches to 3D object recognition that utilize collections of 2D views are also referred to as view-based approaches (Koenderink & van Doorn, 1979); similarly, approaches to object recognition utilizing separate views—for example, as obtained by wavelet transforms (e.g. Shokoufandeh et al., 1999)—are also regarded as view-based. This is an analogy to the interpretation of "view-based" in robotics where separate views on the same environment are jointly, yet independently represented.

2.1.3 Knowledge about navigability of space

Knowing which part of the environment is navigable is a key point in path planning. Consequently, many approaches represent knowledge about navigability of space, e.g. by representing the boundary of navigable space or by geometric features derived from it. Information about navigable space can also be obtained from maps that are used by humans. Floor plans, for instance, depict walls and outlines of rooms which restrict movements. Sensors like laser range finders or sonars measure the boundary of navigable space by sensing for obstacles. Thus, range sensors make information accessible that is required for navigation. In principle, information about navigability of space can also be derived from other sensors, e.g. vision. However, extracting this information from vision is a demanding and not yet fully understood problem. In the following, I will detail approaches for representing knowledge about navigable space in greater detail.

Cell occupancy

In cell occupancy representations, space is decomposed into discrete cells. For each of the cells, a degree of occupancy is represented. Thus, occupancy values constitute the map features. The gradual approach to represent occupancy responds to an property of cell decomposition—cells can be partially occupied; this is reflected in the degree of occupancy. Alternatively, occupancy values can be interpreted in a probabilistic manner: they are regarded to represent the probability that the cell is occupied by an obstacle (Hähnel et al., 2002; Thrun, 2002), or the probability distribution for gradual occupancy (Stachniss & Burgard, 2003a,b,c).

Usually, the spatial domain is partitioned into square-shaped cells of fixed size (e.g. 10cm $\times$ 10cm). The obtained map representation is an array of occupancy values, the so-called *occupancy grid* (Elfes, 1989; Moravec & Elfes, 1985).

Occupancy grids are a particularly popular representation when using range sensors (for example, see Baker et al., 2004; Fox et al., 1999; Thrun et al., 2000a). To interpret a single sensor reading in terms of an occupancy grid, cells are marked unoccupied, if they correspond to a distance smaller than the range measurement; cells indicated by the measurement are marked occupied. If cell occupancy is interpreted in terms of probability and potential sensor noise is taken into account, a non-zero probability of occupancy is assigned to cells in the surrounding of the cell indicated by the measurement (Thrun et al., 2005).

Boundary of navigable space

Points measured by a range finder coincide with the boundary of navigable space. To represent this boundary and to capture a wider context than single points, measured points can be grouped to geometric primitives. This allows abstraction from the raw measurements and reduces the amount of data to be handled.

In indoor environments, in particular in office-like environments, navigable space is often limited by obstacles that present a straight outline, e.g. a wall. For such environments grouping into line segments is especially popular (see for example Cox, 1990; Forsberg et al., 1995; Gutmann, 2000; Lu & Milios, 1997; Pfister et al., 2003; Röfer, 2002).

Besides representing the boundary of navigable space by individual line segments, polygonal lines (shortly: polylines) have been suggested. Utilizing polylines to constitute a geometrical model of the world is by no means new, but dates back to work such as (Chatila & Laumond, 1985; Laumond, 1983). However, these approaches address an application where the world model is known in advance and can be modeled manually. More recently, Veeck & Burgard (2004) and Latecki & Lakämper (2006b) proposed algorithms to autonomously extract polygonal line models from a map defined by a set of points. The input maps considered in their approaches can be obtained by accurately aligning range scans. González-Baños et al. fit polygonal lines to grouped clusters of range finder data to model regions in an exploration strategy (González-Baños & Latombe, 2001; González-Baños et al., 1999). However, the mapping task itself is not addressed here.

Similarly to modeling the boundary of navigable space by means of polylines, Austin & McCarragher (2001) suggest a fixed set of universal geometric primitives to model constraints that obstacles pose upon the robot's movement. In their work, the robot is assumed to be capable of following obstacle boundaries in constant distance, and of localizing itself in a global coordinate system with a bound error. The geometric primitives are identified by observing the trajectory of the robot traveling along the obstacles. Mapping is considered only with respect to detection and registration of geometric primitives, since localization is assumed to be provided.

Traversable routes

Representations focusing on routes in the environment are often referred to as *roadmap* approaches (cf. Choset et al., 2000, 2005; Wallgrün, 2005) or as *topological maps*. To computationally characterize routes, typically a variant of the Generalized Voronoi Diagram (GVD) (Lee & Drysdale, 1981) is utilized. The GVD represents the medial axis of free space ("skeleton"), the set of all points equally and maximally apart from the nearest obstacles. Each point of the GVD is the center of a circle inscribed in free space that touches at least two points of the obstacle outline (see Fig. 2.1 for illustration). A graph, the so-called Generalized Voronoi Graph (GVG), is then derived from the GVD; meet points and end points of the GVD constitute the nodes in the GVG. Nodes belonging to a GVG are identified by their degree. Roughly speaking, the degree corresponds to the number of different Voronoi paths emanating from a given point on the GVD.

GVGs offer abstract and compact means for representation (Thrun, 1998). Furthermore, routes that follow the GVD are maximally safe as they maintain maximum distance to obstacles. Approaches representing route information are especially popular for indoor office scenarios which provide a clear structure of routes (see e.g. Werner et al., 2000).

2.1.4 Discussion

For reviewing the variety of map features, I distinguished three main categories of map information: sensor patterns in robot-centric observations, landmarks, and the representation of knowledge about navigable space. In the following, I discuss characteristic properties of these categories.

Raw sensor patterns

Memorization of raw sensor patterns is possible in any environment; thus, view-based approaches to feature representation provide a universal approach. Moreover, maps can be built in a straightforward manner by memorizing views and interrelating them with, e.g. robot commands taking the robot from one place to another (Matsumoto et al., 1999) or directional knowledge (Franz et al., 1998). However, there are severe limitations which Thrun (2002) regards as the reasons why today's dominant approaches utilize different map representations. The main deficit of view-based approaches is that memorized views lack of a spatial interpretation, but are atomic representations. This has the effect that it not possible to extrapolate the appearance of a view obtained from one place to views at places nearby. Therefore, the robot might not be able to detect crossing a previously traversed route. This inhibits, for example, path-planning beyond previously taken routes as required for planning short cuts. Similarly, if two robots observe the same local surrounding from nearby positions, they might not be able to relate their observations. This makes it hard to relate robot-centric maps to external information supplied by fellow robots or external maps. Lacking knowledge of local spatial properties can hinder the recognition in environments that are affected by changes: changes locally alter the appearance which can cause undeterminable effects to a view that is interpreted atomically. Localization on the basis of views requires congruency of views in order to identify a specific view associated with a specific pose. Altogether, the lack of spatial interpretation complicates or even thoroughly hinders interrelating views taken under different observation conditions or by different robots. Furthermore, uninterpreted views are inextricably linked to the robot's perceptual apparatus and do not allow for integration of external information on a potentially different level of granularity.

To conclude, robot-centric maps provide no adequate means to tackle my research questions. In particular, they neither allow a robot to relate its internal map to external information, nor do they enable intelligent navigation.

Landmarks

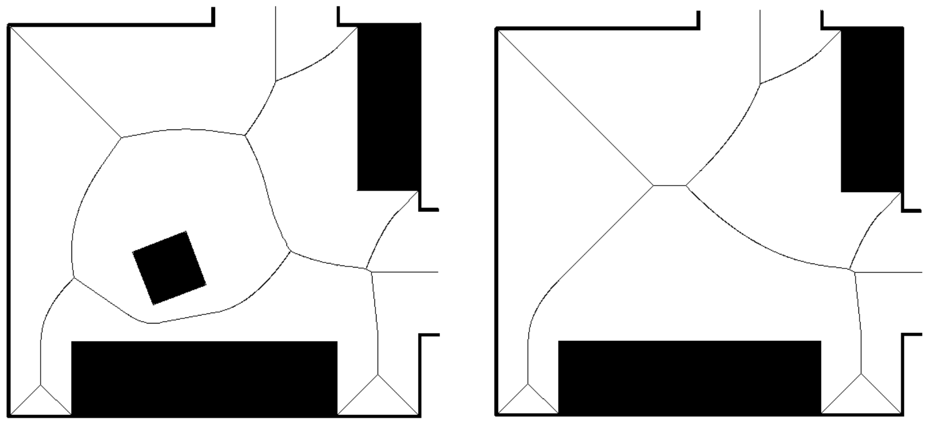
Information about landmarks is important to human navigation (Denis, 1997) and could also provide a basis for robot maps. Robots utilizing similar kinds of landmarks as employed by humans would certainly ease the interaction between robots and humans. To accomplish this interaction, a rich repertoire of landmark types is required. Techniques to identify the variety of landmarks used in route directions in human communication are yet unknown. I regard the gap between today's capabilities of landmark detection and the sophisticated techniques required to allow for interaction with humans as yet too wide to bridge.

In general, landmark representations expose the difficulty of determining a universally applicable and robustly detectable set of landmark types. To obtain a universal approach, a set of landmark types needs to be identified such that landmarks can be observed in any environment. Furthermore, navigation planning by pure consideration of landmarks is not possible, but any landmark-based representation would need to be complemented by a representation of knowledge about navigable space.

To put it in a nutshell, the utilization of landmarks is promising to support human-robot interaction, as it accords to human cognition, but unfortunately landmark detection depends on sophisticated object recognition beyond reach. Since landmarks alone provide no means to represent information required for navigation, I argue for a representation of navigable space.

Navigable space

Of the approaches representing knowledge about navigable space, GVGs present the most compact representation, as only distinct routes through an environment are represented. This makes roadmaps in general a popular basis for path planning, providing a straight-forward interface of continuous path-planning and discrete, efficient graph search. Roadmaps have been intensively studied in this regard (for an overview, see Latombe, 1991). Unfortunately, the graph structure of GVGs is susceptible to noise in input data, especially if the environment contains open places rather than a clear structure of routes; the problem of robust map acquisition on the basis of GVGs has not yet been solved—mainly, since sophisticated recognition techniques to correlate differing roadmaps are required (see also Section 2.2.3). It is not yet understood how to address robust recognition in context of changing environments (e.g. closing of a door).



(a) (b)

Figure 2.1: (a) Exemplary indoor environment and its corresponding GVD. (b) The resulting GVD after applying a change to the environment. Figures are taken from Wallgrün (2002).

Emerging of new objects or the disappearance of objects can result in significant differences in the GVG. For example, if a freestanding obstacle is located in a room, the GVD contains distinct routes passing by the obstacle, one at each side (see Fig. 2.1 (a)). If the obstacle is removed, the different routes are replaced by a single one (see Fig. 2.1 (b)). These changes of the graph structure complicate relating GVGs obtained in changing environments. The difficulty is anticipating the fundamental changes in the graph structure resulting from potential changes in the environment. As a complicating fact, even relatively small changes in the environment can imply fundamental changes in the graph structure. This effect can be caused by measurement noise, too. The applicability of GVGs to mastering the correspondence problem, to localization, and therefore to robot mapping in general, depends on improvements in handling skeleton-based recognition. These topics are currently under investigation (see Wallgrün, 2005).

There are two remaining general alternatives to consider, namely occupancy grids and representations of the boundary of navigable space. A clear strength of occupancy grids is their universality. They can be applied to arbitrary environments and have been applied to wide range of applications, for example mapping in static indoor environments (e.g. Thrun et al., 2000a), localization in dynamic indoor environments (Fox et al., 1999), and mapping in abandoned mines (Baker et al., 2004). Another advantage of occupancy grids is implied by the small spatial extent of single grid cells that makes it possible to avoid considering partial visibility.

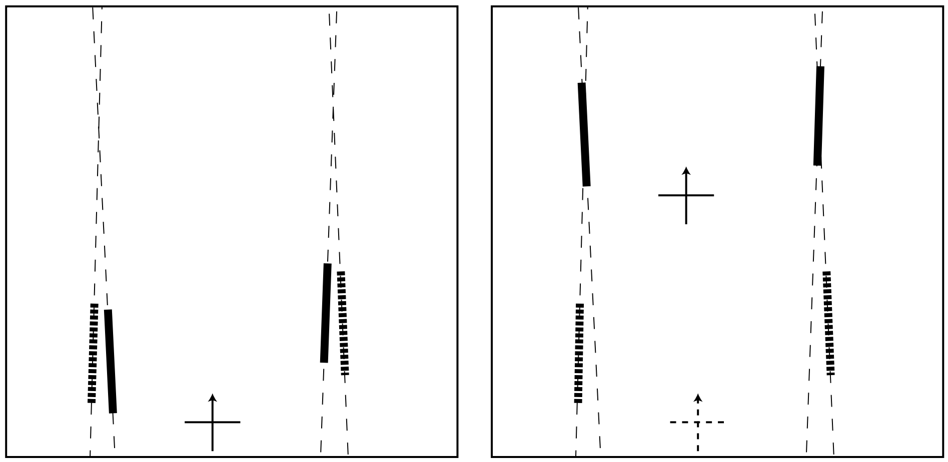
The pleasant simplicity of occupancy grids entails severe limitations though.

Occupancy grids are basically bitmap images that, if related to externally provided maps, would require complex image processing techniques for correlation. Communication on the basis of occupancy grids is limited to strongly constrained settings like multi-robot mapping involving identical robots and known start poses of all robots (see for example Fox et al., 1999; Thrun, 2001). Moreover, occupancy grids provide no compact means of representation. An environment requires the same amount of occupancy values to be represented, independent of its complexity. The large amount of data to be processed requires scan registration to rely on additional assumptions (pose estimates) to efficiently achieve solutions (cf. Section 3.5.3).

Representing the boundary of navigable space is based on extended geometric primitives, e.g. lines or polylines. Dealing with extended objects requires to address occlusion, since virtually all observations provide partial views on the features. Perceived line segments are sometimes interpreted as parts of infinite lines (Cox, 1990; Gutmann, 2000; Lu & Milios, 1997) to avoid explicitly handling occlusion. For example, line segments are regarded to be congruently aligned, if the induced infinite lines are aligned. However, this interpretation is not consistent with physical reality and, thus, can introduce artifacts. Erroneous results occur when aligning sets of nearly parallel lines as is illustrated in Fig. 2.2. In this example, a robot is assumed to perceive a pair of lines in two observations (Fig. 2.2 (a) thick solid and thick dashed line segments). Due to measurement noise, the lines can appear slightly rotated and at slightly different positions, although an identical set of parallel lines is observed and the robot has not moved between the observations. Robots can align consecutive observations to infer their movement, assuming the world is static. Interpreting the lines in the example as infinite lines, the robot finds a congruent alignment, i.e. a transformation moving one set of lines to minimize the distance of corresponding lines (Fig. 2.2 (b)). This congruent alignment erroneously indicates a movement of the robot.

Applicability of grouping into line segments is confined to environments whose boundaries present mostly straight obstacle outlines (Hähnel et al., 2003; Veeck & Burgard, 2004). Moreover, if many short line segments at nearby positions are observed, a robust line identification may not be possible. Mixups in determining the correspondence of lines can easily occur and derange the mapping process.

Polygonal line models provide means to overcome drawbacks of line-based models by (a) linking individual line segments to a more comprehensive spatial context, and (b) by allowing for good approximation of arbitrary (e.g. curved) contours. By linking contour segments, more complex obstacles can be modeled, e.g. corners or colums, using a single feature. Hence, less features need to be registered in the map and individual features bear more information, which facilitates an efficient solution of the correspondence problem.



(a) (b)

Figure 2.2: Aligning pairs of observed line segments which are nearly parallel leads to artefacts if the line segments are interpreted as infinite lines. (a) Two sets of nearly parallel line segments as can be detected by robots traversing hallways (solid and dashed thick lines). (b) Congruent alignment of the solid and dashed line segments if line segments are interpreted as parts of infinite lines.

Unfortunately, techniques developed so far address only the modeling of known environments (e.g. Chatila & Laumond, 1985), or the extraction of polylines from maps that have already been constructed by registering points (e.g. Latecki & Lakämper, 2006b). Techniques to construct maps from sensor data need yet to be devised.

To adopt the level of granularity of polygonal lines, simplification techniques have been developed (see Stein (2003) for an overview on generalization techniques). Such techniques can be helpful in relating fine-grained information to coarse information as is, for instance, represented in schematic maps (Barkowsky et al., 2000).

Despite a lack of techniques for obtaining polygonal maps from sensor information without constructing intermediate maps, polygonal line models appear to be an adequate basis for map representations. Polylines are also important to the field of shape representation in computer vision as detailed in Section 2.2.

Notably, approaches representing the boundary of navigable space by means of geometric primitives, or, in other words, the boundary of obstacles, can be related to landmark representations. Individual geometric primitives may be interpreted as landmarks. Thrun (2002), for example, subsumes both categories using the term *object maps*. He acknowledges four advantages of object maps over grid maps (cf. Thrun, 2002, p. 19):

1. Object maps are more compact than occupancy grids.
2. Object maps can be more accurate, if the objects in the approach are adequate to model the environment.
3. Object maps appear necessary to address dynamics, i.e. model objects which change position over time.
4. Object maps can closer resemble people's perception than occupancy grids.

In the following, I discuss these claims in more detail. Object maps achieve compactness by (a) grouping individual measurements to objects, and (b) by purely representing existent objects—occupancy grids represent cells regardless of their occupancy. In object maps, representations grow with the complexity of the environment, i.e. the number of objects and the amount of parameters required to represent them. In contrast, occupancy grids grow with the size of an environment. Additionally, grids are a discretization of space which are often significantly coarser than information initially provided by the sensor, e.g. grid cells of 5 or 10 cm length as opposed to LRFs with a resolution of approx. ± 15 mm. Grid sizes are a tradeoff between accuracy, efficiency, and compactness. In the case of an environment of 40×40 meter and a grid size of 10 cm, 160,000 cells (640,000 cells for 5cm resolution, respectively) need to be maintained. Assuming the degree of occupancy represented as real-valued probability value by means of a 32 Bit floating point number, this results in about 0.6 MB of data (2.4 MB respectively). By mainly representing positions and compact object signatures in object maps, these representations can be significantly smaller. For instance, if representing line segments using floating point values for coordinates, 0.6 MB correspond to 40,000 line segments (2.4 MB to 160,000, respectively)—many environments of the size 40×40 meter can likely be represented using significantly less lines. So, no need arises to coarsen the sensor data in object map approaches. Handling dynamics can also benefit from object maps, if moving objects in the physical world correspond to objects in the map. In this case it is possible to associate knowledge about movement to the map objects and thereby predict the future appearance of the environment. In contrast, when using a grid map, the knowledge about correspondence to real-world objects cannot be easily introduced and respected in map operations. Furthermore, if map objects correspond to the human interpretation of objects in the real world, the robot's internal map may even resemble a human's understanding of the represented environment.

The major disadvantage of object maps according to Thrun is that these approaches are confined to environments presenting the same geometric primitives as modeled by the specific approach at hand. This claim is illustrated by comparing line models and occupancy grids in Fig. 2.3. Scans obtained from a

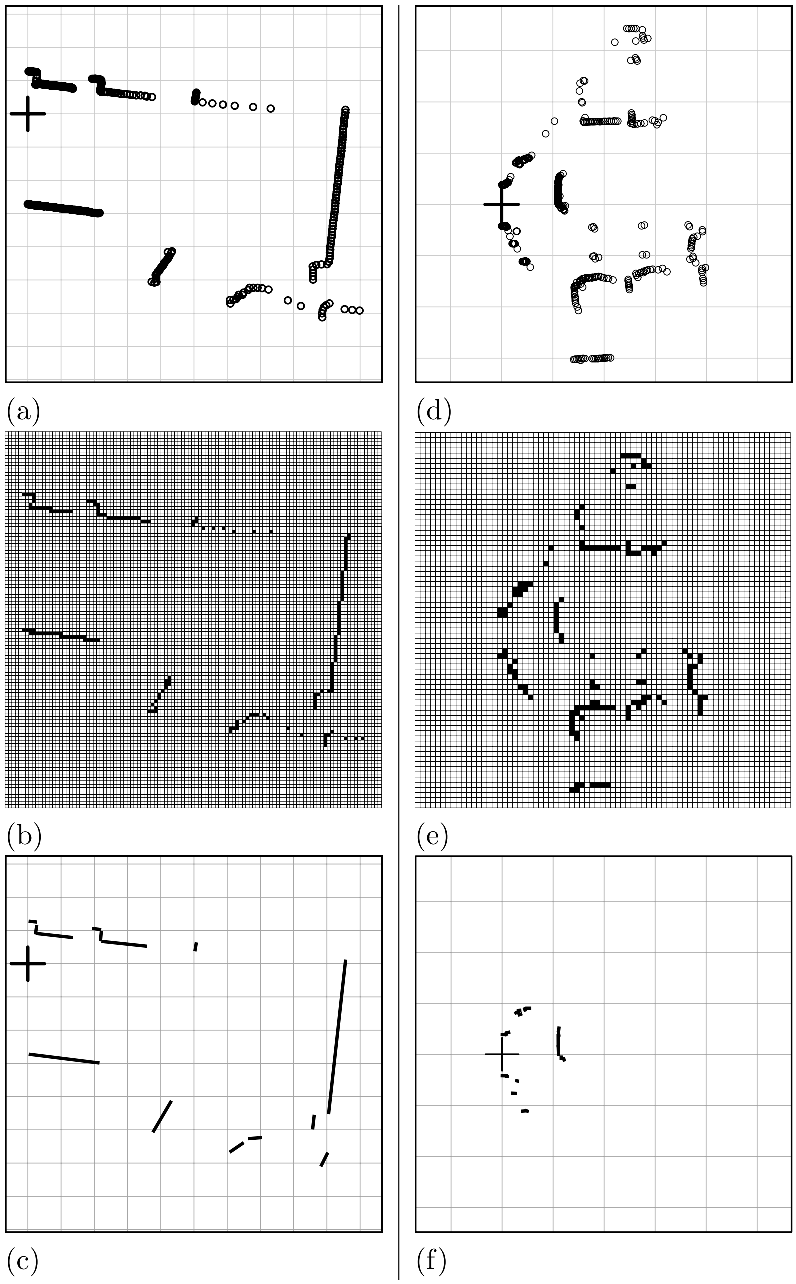


Figure 2.3: Feature extraction for exemplary environments. (a) Indoor office environment (MZH building at Universität Bremen), (b) corresponding occupancy grid, and (c) detectable line models. (d) Exemplary home environment (the robot is located in the doorway facing a living room), (e) corresponding occupancy grid, and (f) detectable line models. The depicted grid denotes 1 m distance; the robot is located as demarcated by the cross.

LRF in two different environments are interpreted to occupancy grids and line models, respectively. To obtain the occupancy grids, a cell size of 10 cm is chosen; the lines are extracted using the *split* algorithm (for example see Gutmann, 2000) with a setting of at least 3 points per line, maximal standard deviation of grouped lines of $\sigma = 20$, and unit size of 1 cm. As can be observed, line models provide no adequate means to represent the living room environment as most obstacles do not display straight boundaries. To restate Thrun's conclusion: if a universal approach to object maps existed, object maps would provide the superior approach to map representation for robots. My thesis addresses the development of such an approach employing universal shapes[2].

2.2 Shape features in computer vision

Representation of areas like, for instance, navigable space is not exclusively considered in the domain of robotics. Approaches to computer vision also represent areas, usually extracted from camera images. Examples include areas of similar color or similar texture. Representation of an area—or its boundary, respectively—is addressed by shape representation. In the following, I relate shape representation approaches originating from the field of computer vision to feature representation for robot maps. The goal of my review is to discover parallels that can be exploited by transferring advanced shape analysis techniques to the robot mapping domain.

Shape representation and analysis plays an important role in computer vision, in particularly in object recognition. Shape has been studied intensively, resulting in an immense amount of literature in this field. Therefore, my review cannot aim at providing a comprehensive overview[3], but is restricted to relevant approaches that can be related to the domain of robot mapping. To approach this overview, I propose to classify shape representation techniques into three categories. Representations that describe the boundary of a shape (e.g. by means of polylines) constitute the category of *boundary-based* approaches. Representations describing the interior of a shape (e.g. by skeleton-based approaches) are subsumed by the category *area-based* representations. This distinction has been suggested by Pavlidis (1978); in addition, Loncaric (1998) distinguishes quantitative and qualitative approaches. Concerning my aim of comparing shape to feature representation, it is more appropriate to utilize a third top-level category covering all techniques which represent by so-called feature vectors of fixed size; this category is termed *shape characteristics*. Representation of shape characteristics include, for example, approaches to Fourier

[2]Comparative results to Fig. 2.3 obtained by my approach are presented in Fig. 5.3 on page 137.

[3]For more comprehensive reviews on shape analysis, see for example Kimia et al. (1990); Loncaric (1998); Pavlidis (1995)

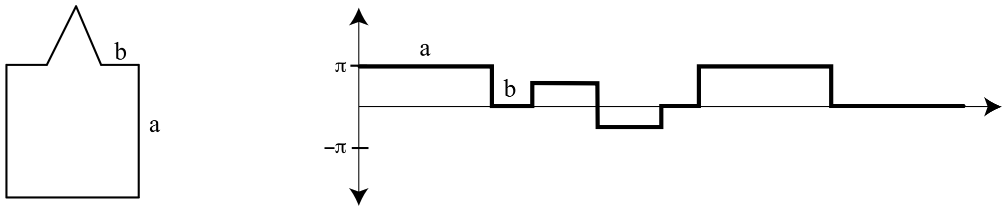


Figure 2.4: An exemplary curve in Euclidean space and its according tangent space representation.

transform, either based on boundary information (Zahn & Roskies, 1972) or area (Zhang & Lu, 2002). To judge the utility of specific approaches to map representation in the context of my work, the evaluation criteria introduced in the previous section are used and implications are derived from relating shape representation techniques to feature representation techniques discussed earlier.

2.2.1 Representation of shape characteristics

This category covers approaches that determine the characteristics of a shape with respect to certain properties, ranging from simple measures (e.g. diameter) or symmetry properties (cf. Jähne, 1997) to complex descriptors of a curvature transform. A fixed set of these properties is determined and compiled into a feature vector which characterizes the shape. The most important descriptors suggested for encoding the contour of a shape are Fourier coefficients (Zahn & Roskies, 1972). To represent a shape's interior, moments are especially popular.

In the case of Fourier descriptors for boundaries, the boundary is first mapped into the so-called tangent space, representing tangent angle vs. normalized curvature length (see Fig. 2.4 for an example). To be more precise, let L denote the curvature length of a shape's boundary b and let T_b denote the tangent space representation of b, i.e. $T_b : [0, L] \to [-\pi, \pi)$. The Fourier transform determines the coefficients $a_1, a_2, \ldots$ and $b_1, b_2, \ldots$ of a Fourier series:

$$T_b(t) = \mu + \sum_{n=0}^{\infty}(a_n \cos nt + b_n \sin nt) \tag{2.1}$$

To represent a shape, the first coefficients $(a_1, b_1, a_2, b_2, \ldots, a_i, b_i)$ are stored; the parameter μ depends solely on the chosen start point on the curve and is often omitted. The possibility of further compacting the data is given by the observation that phase angles are more important to recognition than amplitudes, which may even be omitted (see to Palmer (1999, p. 386) for illustration). In their original form, Fourier descriptors are defined for continuously differentiable curves only; they can however be adapted to non-differentiable, e.g. polygonal curves (Jähne, 1997; Zahn & Roskies, 1972).

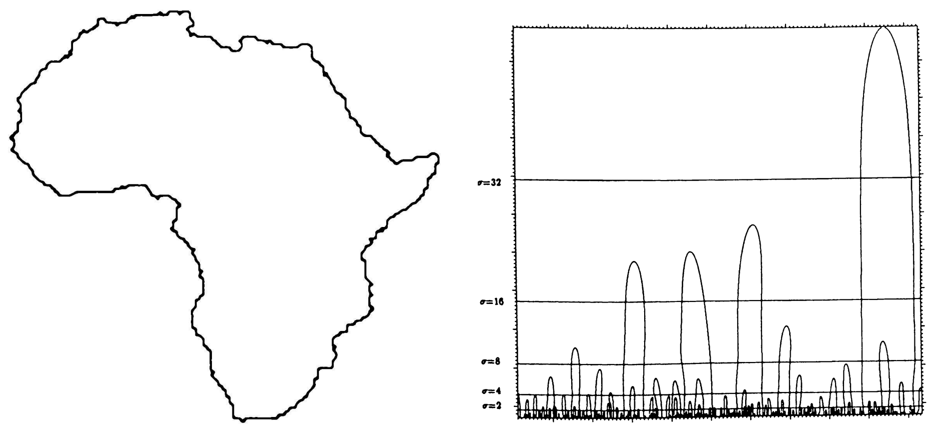


Figure 2.5: Exemplary shape and according curvature scale space representation; Figure is taken from Mokhtarian & Mackworth (1992)

In the case of area-based descriptors using moments, the input shape is interpreted as the characteristic function $\chi_{\text{shape}} : \mathbb{R}^2 \rightarrow \{0, 1\}$ or as a greyscale image (Jähne, 1997; Kim & Kim, 1998). Given a shape s represented as a greyscale image $g_s : \mathbb{R}^2 \rightarrow [0, 1]$, the non-zero support of g_s is interpreted as the interior of the shape. The two-dimensional Cartesian moments $m_{p,q}$ of order $p + q$ are defined as

$$m_{p,q} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q g_s(x, y) \, dx \, dy \tag{2.2}$$

According to Loncaric (1998), utilization of moments in computer vision has been initiated by Hu (1962) who introduced the techniques previously popular in mechanics. As of today, particularly the variants of Zernike moments are popular (Kim & Kim, 1998; Reiss, 1993). Generally speaking, variants of moments replace the kernel $x^p y^q$ in Eq. 2.2 by a generalized form using orthogonal polynomials, e.g. Zernike polynomials in the case of Zernike moments; see Teague (1980) for an overview of variations.

2.2.2 Boundary-based representations

A second category of shape representation techniques represents the boundary of a shape and encodes its course. Approaches include polylines or tangent space representations (e.g. Arkin et al., 1991; Latecki & Lakämper, 2000), curvature scale space (Mokhtarian & Mackworth, 1992), or qualitative description (e.g. Meathrel & Galton, 2000).

A complex shape descriptor derived from the boundary of a shape is the so-called "Curvature Scale Space" (CSS) (Mokhtarian & Mackworth, 1992). To construct a CSS representation, the contour is smoothened using a Gaussian convolution filter; the CSS represents the curvature of the contour vs. contour length for various stages σ of the smoothing filter application (see Fig. 2.5). By increasing the value of σ, i.e. by increasingly simplifying the shape, details disappear. Disappearance of inflection points is observed in the course of contour simplification. Parameter values of σ and position on the contour are determined for disappearing inflection points and constitute the CSS representation.

Polynomial curves are of importance to computer vision as they provide a discrete structure to which sensor information can easily be mapped. Consequently, many authors approached shape analysis, in particular shape recognition, on the basis of polylines (see also Section 3.5.2). Representation of partially visible shapes is accomplished by employing open polylines, whereas complete contours can be represented as closed polygons. Recognition has also been studied for closed contours (e.g. Basri et al., 1998) as well as for contour parts (e.g. Latecki et al., 2005b).

Aside from polygonal curves, *chain codes* are popular for constructing representations from pixel-based images. The chain code of a pixel-based contour captures the sequence of movements, e.g. "one pixel up" or "one pixel to the left", required to traverse the contour (Cortelazzo et al., 1994). Chain codes provide a very detailed representation of the contour, but are confined to pixel images.

An analog, but more abstract approach to boundary encoding is taken by Galton & Meathrel (1999); Meathrel & Galton (2000). They proposed a qualitative outline representation which distinguishes between seven types of curvature (see Fig. 2.6). The authors introduce a formal grammar to represent contours as sequence of curvature classes in a canonical manner. Analogously, Museros & Escrig (2004) approach shape representation tailored for describing mosaic tiles by qualitative means. In their approach, the course of a polygonal contour is described by qualitative angles ("right-angled", "acute", "obtuse"), distinction between convex and concave corners, and by classifying the relative length of consecutive line segments ("smaller", "equal", "longer"). Additionally, positions of holes are described using a qualitative spatial calculus.

To sum up, most boundary-based representations represent, by means of a linear data structure, a sequence of qualitative labels or quantitatively determined curve characteristics. To examine the similarity of two boundary-based representations, many authors suggest computation of a cost-minimizing correspondence of boundary fragments which can be computed using an elastic matching (for example, see Basri et al., 1998; Sebastian et al., 2003, or refer to Section 3.5.2). This is often accomplished by means of Dynamic Programming (cf. Section 4.3.1 or see for example Cortelazzo et al. (1994) for an overview of the application of string matching techniques).

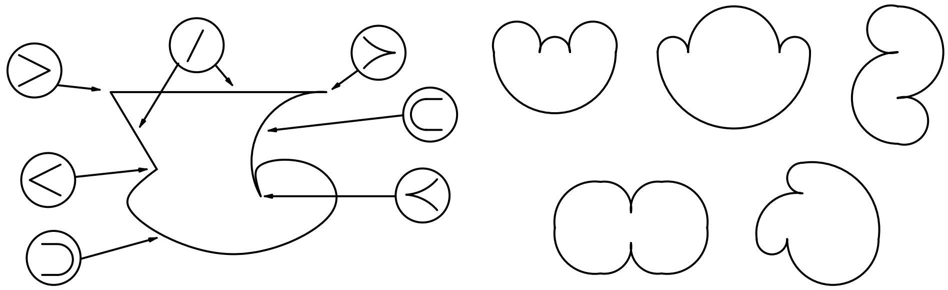

(a) (b)

Figure 2.6: (a) Curvature classes distinguished in the qualitative outline representation. (b) Different shapes with same qualitative description $\supset\prec\supset\prec$. Both Figures are taken from Galton & Meathrel (1999).

2.2.3 Area-based representations

A third category represents the inner structure of a shape by determining so-called skeletons (e.g. Blum, 1967) by decomposition into geometric primitives (e.g. Biedermann, 1987) or by means of pixel maps (e.g. Huttenlocher et al., 1993).

Skeletons introduced by Blum (1967) demarcate the beginning of shape representations deriving shape information from a medial axis transform (MAT). Roughly speaking, the medial axis or skeleton of a shape consists of the points equally and maximally apart from the boundary. In other words, the MAT is a kind of Generalized Voronoi Diagram (cf. Section 2.1.3); however, variants of MATs for shape representation are specifically designed for the discrete structure of pixel images, rather than for continuous positions of polygonal contours extracted from range data. Special consideration must be taken to compute a skeleton that is robust to effects of discretization in pixel images (cf. Dimitrov et al., 2000; Siddiqi et al., 1999a,b). Once the MAT has been computed, the actual representation is constructed, which is an attributed graph. Originally, Blum suggested a qualitative labeling (Blum, 1967), but quantitative information has later been introduced to provide sufficient discriminating information (Blum & Nagel, 1978). More recently, Siddiqi et al. (1999b) proposed to attribute the medial axis using quantitative descriptors alongside qualitative ones. Quantitative descriptors represent the course of axis fragments; qualitative labels are utilized to classify the distance of the axis to the contour. Labels correspond to local minima (constrictions), local maxima, parts of constant distance, and changing distance.

Biedermann (1987) proposed a cognitively motivated shape representation on the basis of a decomposition into shape primitives which are generalized cylinders and are termed *geons*. Representations by means of geons are prominent in the field of cognitive modeling, but do not include computational means to compute a representation and to recognize shapes using the structure of geons.

In their most elementary form, area-based shape representation makes direct, uninterpreted use of the (segmented) input image. Shapes are simply represented by their bitmap images (for example see Rucklidge, 1996).

2.2.4 Discussion

In the following, I discuss general properties of the three categories of shape representations and elaborate on the parallels to feature representations suggested for map representation.

Shape characteristics

Common to all approaches that encode shape characteristics is that they pursue the determination of a compact representation that allows large shape databases to be handled efficiently. Recognition of a query shape can be performed very quickly by computing a weighted vector difference of the query's feature vector and the feature vectors stored in the database. For example, Kim & Kim (1998) achieve remarkable results with feature vectors of only two dimensions[4].

Approaches to shape representation from this category are well-suited to recognition tasks in closed domains where the set of shapes to be discriminated is known a-priori and discriminating characteristics can be determined. In applications where arbitrary shapes are handled, it might not be possible, let alone feasible to determine a set of discriminating features for a-priori unknown shapes. As a complicating fact for application to robot mapping, shape characteristics are descriptors that are exclusively defined for complete shapes, i.e. they cannot handle a partially visible shape. Therefore, recognizing a shape in the presence of occlusion is often not possible. In the case of local changes to a shape, a global effect on the shape descriptor can occur as these descriptors are not local, but compile a complete shape to an atomic feature vector (Loncaric, 1998). Altogether, this renders approaches representing shape characteristics inadequate for application in robot mapping tasks.

Notably, there exists a—somewhat weak—relation between approaches to shape characteristic encoding and view-based approaches in robotics. In both domains a representation is constructed that provides an atomic representation by compiling the sensory input into a feature vector. In shape representations, compactness of the representation is particularly fostered by using complex transformations. Representations belonging to the category of shape characteristics are designed for efficient recognition given a fixed, discrete set of alter-

[4] The authors regard using that few moments as a naive approach and conducted this experiment purely for demonstration purposes—nevertheless, they yield a remarkable retrieval rate of more than 90% in a test employing 3000 company logos.

natives. In view-based representations, a discrete set of views is handled too. Due to the atomic character of shape characteristics, even local changes of the input may cause unpredictable global effects to the representation. In other words, representations of shape characteristics lack a spatial interpretation, or, as (Loncaric, 1998) puts it, shape characteristics provide no local representation. Extrapolation to anticipate the effects of changes (e.g. due to view point variations or changes in the environment, respectively) is not possible. Since feature recognition applicable to robot mapping requires robustness to view point variations, occlusion, or changes, the applicability of shape characteristics to feature representation in robot maps is doubtful.

Boundary-based representations

Several boundary-based shape descriptors have been developed that yield good results in shape recognition tasks. This suggests that, if these approaches can be transferred to the domain of robot mapping, they are beneficial for tackling the correspondence problem.

In particular, the representation on the basis of the CSS yields excellent results in the shape retrieval study conducted for MPEG-7 standardization (Bober, 2001; Bober et al., 1999). However, observe that a CSS representation is not defined for contours not displaying inflection points, since solely the disappearance of inflection points during curve smoothening is represented. Shapes that lack a CSS descriptor include circles, convex polygons, etc. Therefore, CSS descriptors contradict the goal of finding a universal feature representation; many environments display convex shapes like, e.g. columns or rectangular rooms. See Latecki et al. (2000b) for a discussion of additional difficulties faced in recognition on basis of the CSS.

The approach by Latecki & Lakämper (2000) achieves comparative results in the aforementioned shape retrieval study. It is based on a representation of polylines that can represent arbitrary shapes; to determine shape distance, polylines are mapped to tangent space representations.

Purely qualitative representations seem inadequate for representing spatial relations on a comparative low level of abstraction as is required to describe shapes distinctively, which might only differ in details. As Galton & Meathrel point out, *"it is in the nature of a qualitative representation system that one representation can correspond to many different objects"* (Galton & Meathrel, 1999, Section 3.8); they illustrate this by different shapes that are described by the same qualitative representation (cf. Fig. 2.6 (b)). For recognition tasks in open domains, one cannot decide in advance which spatial relations will be required to provide distinguishing information. In such cases, qualitative approaches are disadvantageous as it might be impossible to record all potentially relevant, i.e. discriminating, spatial relations.

In boundary-based approaches, representation and recognition of partially visible shapes can also be addressed. The representation of a shape's contour can easily be restricted to any subset of it. Shape similarity measures can also be applied to partial contours as has been demonstrated by Latecki et al. (2005b).

In summary, boundary-based representations provide a suitable basis for feature representation in robot mapping. Polylines provide a universal representation of obstacle boundaries, i.e. arbitrarily shaped contours can be modeled. For example, Veeck & Burgard (2004) argue from the perspective of robotics for the compactness and universality offered by polygonal line models over grid maps.

Representations based on polylines expose a strong analogy to boundary-based representations of navigable space used for robot maps (cf. Section 2.1.3), polylines being a fundamental representation in both domains. However, one cannot observe a high interconnection of the research fields of shape recognition and mobile robot localization and mapping—although a link between these fields has already been discussed by Grimson (1990). As Thrun noted, the connection between computer vision and robot mapping is underexploited (Thrun, 2002, Sec. 2). Lu & Milios even claim that a significant difference between the field of shape matching and scan registration (i.e. solving the correspondence problem on basis of range scans) exists, as mobile robots can only acquire noisy discrete points instead of high-quality models (Lu & Milios, 1997, Sec. 1)—this differentiation appears questionable to me as shape information extractable from a vision system is also discrete and robust handling of noisy data is one key issue for shape recognition too.

Area-based representations

From the category of area-based representations, skeleton-based techniques using attributed graphs stand out for their compact representation. They expose an inherently close connection to route-based representations suggested for mobile robots, utilizing similar techniques to construct the representation. Thus, skeleton-based shape representations expose similar characteristics as route-based representations (cf. Section 2.1.3), most importantly, the global structure of skeletons is affected by local changes to the shape, too. Graph matching techniques have been developed for recognition tasks in computer vision. Since even the graph structure of similar shapes is likely to differ, techniques for isomorphic graph matching are of little use and graph similarity measures need to be employed. Edit distances for graphs have been suggested for this purpose (Zhang & Shasha, 1989). To eclipse costly unrestricted graph matching, Bartoli et al. (2000); Siddiqi et al. (1999b) first transform the general graph representation to a tree. Nevertheless, graph matching remains computationally more expensive than linear elastic matching that is applicable to interrelating linear data structures of boundary-based representations. Though, the most severe complication faced in recognition is anticipating the changes to the graph structure caused by small changes of the input data. It is yet unknown how to master this problem. Difficulties of skeleton-based recognition find expression in the retrieval experiments that have been conducted in the context of MPEG-7 shape descriptor standardization: the participating skeleton-based descriptor showed the poorest performance (cf. Latecki et al., 2000b). As with the discussion of route-based representations, I conclude that skeleton-based representations provide no promising starting point to both address map representations, and to support mastering the correspondence problem.

The most direct approach to shape representation in vision applications is to utilize bitmap images. The tradeoff of this straightforward representation is the computationally expensive shape analysis, e.g. shape similarity and shape alignment (cf., e.g. Rucklidge, 1996). Bitmap images are interpreted as sets of points and are compared by means of the so-called Hausdorff distance H:

$$H(A,B) = \max\{h(A,B), h(B,A)\} \tag{2.3}$$

$$h(A,B) = \max_{x\in A} \min_{b\in B} d(a,b) \tag{2.4}$$

where d denotes a distance measure, typically Euclidean distance. As can be observed, the Hausdorff distance is a position and rotation dependent measure. In other words, to recognize rotated or shifted shapes (e.g. shapes described in a different frame of reference), all possible alignments (rotation and translation) need to be examined—depending on the potential alignments to consider, this can easily exceed feasibility. Today's successful approaches to shape recognition make use of more sophisticated representation techniques as documented in the reports on the MPEG-7 shape descriptor evaluation (Bober, 2001; Latecki et al., 2000b). Notably, shape recognition and alignment using uninterpreted pixel images is intimately connected to the occupancy grid representation employed in robotics. In analogy to computer vision, scan registration on the basis of occupancy grids determines the optimal alignment of two occupancy grids by minimizing the directed Hausdorff distance h (cf. Hähnel et al., 2002). However, references to research in computer vision are often missing. In order to meet requirements of efficiency (e.g. for online mapping), it is important to restrict variations of rotation and translation that need to be considered.

To conclude, the review of shape representations demonstrates the existence of parallels between feature representation in robot maps and shape representations employed in visual object recognition. Links between the two fields are apparent in several approaches. Skeleton-based techniques are applied to represent shape structure or routes, boundary-based representations describe the contour of a shape or the outline of navigable space, and greyscale images and occupancy grids resemble one another. Sophisticated shape retrieval techniques have been developed—the parallels suggest that shape retrieval techniques can be transferred to robot mapping and facilitate an efficient and robust approach to the correspondence problem. Since shape descriptors based on polylines support robust recognition and present means to represent arbitrary shapes, these approaches appear particularly suited to application in robot mapping. As a consequence, I develop a spatial representation that combines the fields of robotics and visual object recognition; in this respect, my approach differs from existing work.

2.3 Configuration representation

After reviewing feature representations in robotics and relating them to shape representation developed in computer vision, I now turn to representation of configuration information. A configuration describes the spatial arrangement of features in the map. Generally speaking, a configuration representation provides a formalism for representing spatial relations between atomic objects and a frame of reference. By linking individual map features, a comprehensive spatial context is established, which is particularly valuable for addressing the correspondence problem, since if provide rich information. To evaluate the utility of individual approaches with respect to mapping and navigation, I regard the following questions:

- What is the contribution of knowledge about configuration to distinguishing map features?
- How can matching efficiently exploit configuration knowledge?
- Which knowledge about configuration supports navigation?

Objects and their locations are intimately connected, so identifying an object and identifying its location are equivalent. Though features may be indistinguishable on their own, configuration knowledge can allow disambiguating features by distinguishing their locations. So, a representation of configuration knowledge should offer rich and robust information of feature locations.

The correspondence problem is tackled by matching the observation of the robot against the map, thereby interrelating features and interrelating configurations. A matching is plausible if only features are associated that present agreeable characteristics, and if the configuration of associated features is respected. For example, if two features are observed close to one another, they shall only be related to features in the map which are close to one another too. Configuration knowledge is particularly valuable if it can directly be utilized during the matching process, thereby restricting search space and facilitating an efficient approach (cf. Grimson, 1990).

A configuration specifies the relative positions of features to one another. If features in the map correspond to obstacles in the environment, knowledge about relative position is essential to path-planning, e.g. allowing a robot to determine whether it can pass between two obstacles or not.

Configuration knowledge is characterized by two aspects: the modality of spatial information that is represented, and the representation formalism employed. In the following, these aspects are discussed individually. It is remarkable that the variety of representations of configuration knowledge employed in robotics is much smaller than the variety of map features. In robotics, coordinate systems are most popular, representing positions of features in an absolute metric frame of reference. Outside of the field of robotics, representation of configuration knowledge is also studied in the field of spatial reasoning. This review covers both fields to provide a more comprehensive overview with the goal of identifying spatial reasoning techniques that are beneficial to robot mapping.

2.3.1 Modalities of spatial information

Following a classification by Freksa & Röhrig (1993), spatial information about object configuration can be divided into three distinctive modalities: directional information, distance information, and topological information.

Directional information represents information about the direction to an object. This may be done by expressing relative information (e.g. “left of”), or in an absolute manner (e.g. “32° North”).

Distance information describes the distance between two objects either in a relative manner by means of comparison (e.g. “farer than”) or in terms of an absolute scale.

Topological information denotes knowledge about connectivity. In the field of qualitative spatial reasoning, topological information is often considered to describe the relation of extended regions, the most prominent approach being the Region Connection Calculus (RCC) introduced by Randell et al. (1992). In robotics, topological information typically describes connectivity of places. Yeap & Jefferies (1999), for example, represent connectivity of local maps. Kuipers suggests in his Spatial Semantic Hierarchy (SSH) (Kuipers, 2000) to interpret connectivity of distinctive places as applicability of action primitives that allow moving the robot from one place to another. Representations of navigable routes (see Section 2.1.3) denote connectivity of decision points.

Additionally, ordering information plays a central role in many qualitative representations. Ordering information has been used to model the circular order of visibility (for example see Barkowsky et al., 1994; Schlieder, 1995) or to constitute directional knowledge (for example see Moratz et al., 2000; Skiadopoulos & Koubarakis, 2005). Spatial information can be ordered in various ways, even information across the modalities listed above can be captured in ordering information. Thus, ordering can be regarded a proper modality of spatial information and is added here as a separate modality.

Ordering information represents a sequence of locations according to their projection on an arbitrarily chosen linear reference, e.g. a coordinate axis or a circle. Ordering information has been employed in navigation too. Schlieder (1993) suggested a navigation strategy based on a panorama representation that has been recently improved by Wagner et al. (2004). Barkowsky et al. (1994) employs a navigation strategy based on the circular order of obstacles.

Information present in these modalities may be represented in different ways, e.g. individual by modality using distinct relations or cross-modal using coordinate-based geometry. There are two principally alternative representation techniques: quantitative approaches which measure with respect to an external scale and qualitative relations that specify selected (usually relative) properties. In the following, I contrast the two variants.

2.3.2 Qualitative representations

Qualitative representations employ a finite, typically small set of relations to model spatial information. Typically, the relations capture relative information obtained by comparison, for example, "north of" and "south of" can serve as qualitative relations acquired by comparing the geographic locations of two objects (for example see Frank, 1992). Technically speaking, qualitative representations abstract from fine-grained or continuous information to (typically few) discrete relations by means of equivalence classes.

Some authors confide the set of potential relations to a single connectivity relation expressing topology (for example see Choset et al., 2000; Kuipers & Byun, 1991; Yeap & Jefferies, 2000). Topological information captures connectivity of distinctive places and can be represented by a graph. Different types of places have been considered ranging from branching points in route-based representations (see Section 2.1.3) to (extended) areas (for example, see Yeap & Jefferies, 2000). Graph labeling is required to enable agents to identify individual edges that meet in a single node of the graph[5]. Yeap & Jefferies associate edges with specific spatial structures termed "exits"[6]). Kuipers (2000) labels directed edges with robot commands. The execution of an action associated

[5]Strictly speaking, such approaches are not truly topological, but incorporate additional spatial knowledge for distinguishing edges. However, literature in the field of robotics refers to representations that focus on connectivity information as topological approaches. Here, this generalized view is adapted.

[6]The approach by Yeap & Jefferies is discussed in greater detail as regards hybrid maps—see Section 2.4.2 on page 65.

with an edge moves the robot between the places associated to the nodes which are adjacent to the edge. The kind of information used to attribute the graph structure influences the matching process in important ways so that general statements about the properties of topological representations of configuration knowledge cannot be made.

Ordering information is also commonly addressed by qualitative representations, e.g. a binary ordering relation that relates two map features. Schlieder (1995) represents the cyclic order of point-like landmarks and Barkowsky et al. (1994) utilize cyclic order of extended landmarks in non-cyclic environments, i.e. environments without freestanding obstacles.

Qualitative representations have been claimed to provide adequate means for communicating spatial information. Moratz & Tenbrink (2006) utilize projective relations between objects in a robot instruction setting. A robot is instructed to move to a position described by qualitative information. As the robot needs to identify a certain place on the basis of its representation, this task is tightly connected to self-localization.

Qualitative relations can be the basis of a so-called qualitative calculus. Qualitative calculi extend qualitative relations by introducing means to "calculate with relations", e.g. to infer which relation can hold between A and C, if the relations holding between A and B as well as B and C are known (relation composition). To reason about qualitative spatial relations, relation composition & constraint propagation are commonly employed. Constraint-based reasoning has been suggested for pruning the search space in matching tasks (Grimson, 1990). A mapping of objects can be restricted by qualitative constraints posed on the objects. Thus, qualitative calculi can be employed to introduce hard constraints in correspondence computation (cf. Tsang, 1993). A review of the utility of geometric constraints in recognition tasks is presented by Grimson (1990). Besides constraint-based techniques, conceptual neighborhood structures (Freksa, 1991, 1992) have been introduced for qualitative reasoning. Conceptual neighborhoods are in particular valuable in resolving conflicts on the symbolic level by defining an interrelation on the level of relations.

2.3.3 Quantitative representations

Quantitative formalisms describe the world by means of absolute, often fine-grained, uniform scales. Quantitative representations employ no abstraction besides reduction of resolution. Henceforth, sensor data can directly be mapped to a quantitative representation. Virtually all approaches in robotics employ a quantitative representation. For example, Franz et al. (1998) represent the relative location of places in a plane using two-dimensional vectors. The most prominent form of quantitative representations is an absolute representation in the form of coordinate-based geometry; positions of map features are identified with locations in the Euclidean plane. Most approaches in robotics represent positions as coordinates in the absolute frame of reference defined by the global map (for example see Lu & Milios, 1997; Stachniss & Burgard, 2003b; Veeck & Burgard, 2004). Various navigation strategies have been developed for coordinate-based representations. In the case of representing obstacles by polylines, cell decomposition and route-based techniques are applicable (cf. Latombe, 1991). In the field of computational geometry, coordinate-based maps registering polylines are well-researched with respect to path-planning and exploration (cf. de Berg et al., 2000).

2.3.4 Discussion

As regards the four modalities of spatial information, the combination of directional and distance information is particularly useful, since it allows for an unequivocal description of the position of a map feature. Since objects and their positions in space are inextricably linked, these modalities are valuable to tackle the correspondence problem and should be made explicit in the map.

Topological and ordering information both provide abstract, coarse information. On their own, neither of the two knowledge sources may provide sufficient information to disambiguate features. For example, the bare knowledge of a place to be connected to another place is not sufficiently discriminating. Similarly, the cyclic order of points as suggested by Schlieder (1994) provides no adequate means of recognizing a particular configuration of points; see Fig. 2.7 for an illustration. However, the distinctions made by these coarse information sources may be well-suited to making knowledge explicit that is only implicitly contained in a representation. For example, if global coordinates are assigned to features, distance and direction knowledge is represented. The information that a feature is located between two other features is implicit in a coordinate-based representation. By making this information explicit in terms of a qualitative relation, it can be more efficiently assessed. This can support an efficient approach to the correspondence problem, since matchings should respect configuration knowledge.

The role of topological information in efficiently tackling the correspondence problem is not yet clear. Topological information is specified in route-based representations (cf. Section 2.1.3), but so far route-based approaches do not make substantial use of topological information when tackling the correspondence problem. For example, the approach by Choset et al. (1996, 2000) to route representation makes topological information explicit by means of a graph, but refrains from exploiting the topological graph structure in localization. Rather, the robot needs to cling to the paths in the environment that correspond to the graph structure. Motion primitives are employed that let the robot travel along Voronoi paths that correspond to edges in the graph. One reason for refraining from topological information can be seen in the high computational cost of unconstrained graph matching. Assuming the map and the robot's cur-

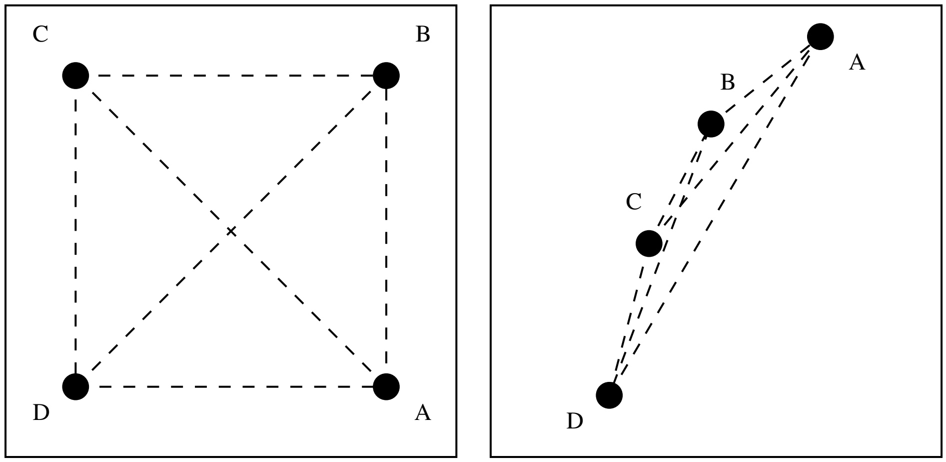


Figure 2.7: Ordering information as suggested by Schlieder (1994) for representation of configurations presents coarse information. The two configurations of points are represented by the same triangular ordering, i.e. any triple of points is equally oriented in both configurations. All configurations obtained by moving points without crossing the triangle outlines (dashed lines) fall in the same class.

rent observation to be represented by a graph, the correspondence problem is related to finding an isomorphic sub-graph of the map that corresponds to the observation—unfortunately, subgraph isomorphism tests are known to be NP-hard (Ullman, 1976). It remains an open problem how a feasible graph matching technique can be designed for topological representations. Recent advances are achieved by Huang & Beevers (2005) who approach map merging by graph matching on the basis of a roadmap.

Topological information provides very compact information and is well-suited to efficient path-planning. To determine a shortest path on the basis of a topological representation, discrete and efficient graph search can be employed.

Balancing the complications of utilizing topological information for solving the correspondence problem and the utility of efficient graph search in path planning, I conclude with respect to the scope of my work that it appears more appropriate not to include topological information into the map representation. Efficient navigation strategies exist for non-topological informations as well and techniques to substantially benefit from topological information in tackling the correspondence problem appear out of reach yet.

In contrast to topological knowledge, ordering knowledge can easily be exploited in matching processes. For example, contour-based shape matching respecting the order of boundary fragments is commonly approached by means of efficient Dynamic Programming techniques (Basri et al., 1998; Cortelazzo et al., 1994; Latecki & Lakämper, 2000; Sebastian et al., 2003). Indeed, these techniques may be adapted to matching techniques addressing the correspondence problem as is discussed in Chapter 4.

As regards representation techniques, I have elaborated on the fundamental alternatives of quantitative and qualitative representation. Qualitative approaches represent in an abstract manner, often by comparing objects with respect to specific spatial properties. Qualitative relations abstract from metric details. Such abstraction can be helpful in relating knowledge sources of different granularities. For example, when relating a fine-grained observation to a coarse map, congruency of observation and map may only be achievable by a sufficient abstraction. Reasoning by means of qualitative calculi can in principle be introduced to matching tasks (Grimson, 1990). However, the application of such techniques, e.g. by means of constraint propagation (see Section 3.5.3), has not yet been investigated thoroughly. A fundamental analysis of the utility of qualitative reasoning in matching is beyond the scope of my work, but presents an interesting starting point for further research.

Quantitative approaches present the most expressive formalism and can provide the most detailed information to differentiate features. In particular, coordinate-based representations are appropriate to link objects to their locations. Thus, coordinate-based representations are valuable for matching techniques tackling the correspondence problem. In quantitative representations all available information is maintained while in qualitative approaches some details are intentionally discarded. Put differently, in quantitative approaches all values are treated equally and no aspects are made explicit. In some situations, treating all values equally can be disadvantageous and can even hamper recognition, as a small example on coordinate-based geometry shows: consider a robot that observes two landmarks that are located close to one another. By measuring their position the robot determines two similar coordinates that are both subject to measurement errors. By evaluating the measurements and taking into account the error margins, the robot may not be able to decide which of the landmarks is located on the left and which is located on the right. The robot can, however, observe with certainty which of the two landmarks is left of the other. In a quantitative approach, this knowledge is shadowed by a representation that relates observations to an external scale rather than to one another. Notably, there are situations where one cannot decide in advance which spatial relations will be required later on. In such cases, quantitative approaches are more economical as it is impossible to record all potentially relevant spatial relations in an environment.

To conclude, qualitative approaches are valuable to handle spatial information on a coarse level of granularity and can explicitly describe distinguishing information. They abstain from metrics and, by doing so, avoid inescapable differences on the metrical level. In particular, when relating knowledge sources on different levels of granularity, a suitable abstraction is advantageous. Of the multitude of spatial relations available for capturing qualitative representation, ordering information appears especially promising. If a suitable ordering can be determined, efficient matching techniques may be applicable (see Chapter 4). Quantitative approaches are well-suited to presenting fine-grained information and can provide rich information to distinguish features. Efficient navigation techniques have been developed for coordinate-based representations that jointly represent information about direction and distance. In particular, coordinate-based maps representing obstacles by polygonal contours are well-researched (de Berg et al., 2000; Latombe, 1991). Qualitative representations and reasoning techniques can be valuable to interrelating information across different levels of granularity (e.g. as required in communication) and to restrict the search space in the correspondence problem. Therefore, combining a quantitative, coordinate-based representation of object configuration and qualitative descriptions appears well-suited to obtain a universal and expressive representation of knowledge about configuration.

2.4 Map organization

Map organization describes how feature and configuration representations are composed to an overall map representation. As regards the utility of a specific approach to robot mapping, I evaluate which technique is suited to model the information relevant to mapping and navigation. In particular, I consider two questions:

- Can information required for efficient navigation and mapping be represented by the approach at hand?
- Can information be accessed and updated efficiently?

Individual approaches to feature and configuration representation display different strengths and weaknesses. To obtain a well-suited map representation, different techniques may have to be combined. In the following I discuss different means of map organization and evaluate their utility.

I classify map organizations by the number of *views* they offer. Hereby, a view is defined by the type of information it represents and the granularity of this information. Each view provides a single frame of reference, i.e. locations of objects can be described with respect to the view. Therefore, each view utilizes one configuration representation and one kind of features. The characterization that a map offers individual views is related to the model of *aspect maps* presented by Berendt et al. (1998) for geographic maps. In that model, aspects are defined as properties of geographic entities and relations between them. Aspects cover a broader variety of representation formalisms than views considered here in the context of robot maps. For example, symbolic annotation in geographic maps is also covered by aspect maps. In order not to suggest the broad interpretation of aspects, the term view is used in the following.

Any map offers at least one view; maps registering locations of features like polylines in a global coordinate system can be regarded as an exemplary representative of this category (for example see Latecki & Lakämper, 2006b; Thrun et al., 1998a). In the following, I refer to maps offering one single view as *uniform maps*.

Maps can offer multiple views, for example, by linking distinct, independent uniform maps. For example, local maps, each covering a restricted area, can be linked by specifying connectivity of local maps. This is the case in a road atlas where maps on each page of the book bear information on which page the adjacent map is to be found. So, the atlas offers several individual views: one map per page and one additional view providing a linkage of maps, e.g., by connecting page 8 and page 16 in North-South direction.

2.4.1 Uniform maps

Uniform maps present one single view and, thus, represent features of one kind using one representation of configuration. Hence, they utilize a single frame of reference to register map features. Typical representatives of uniform maps are global object maps registering the locations of specific features or uniform occupancy grids (cf. Section 2.1.2 or Section 2.1.3). The variety in uniform map representations has been presented in the preceding sections of this chapter by exploring feature and configuration representations. Thus, determining the utility of employing a uniform map can be evaluated by considering the utility of a single best combination of feature and configuration representation.

2.4.2 Hybrid maps

Hybrid maps provide multiple views and interrelate them tightly. Individual views can differ by the kind of information represented or by the granularity of the information represented. In the context of this work, hybrid maps are primarily analyzed with respect to their utility in robot mapping. See Buschka (2006) for a general in-depth analysis of hybrid maps. To classify hybrid maps, I regard two operators which interrelate individual views and thereby allow for the construction of hybrid maps from uniform maps; these operators are:

- Embed
- Link

In the following, I discuss the operators in more detail. The embed operator links views by embedding one in the frame of reference of the other. It is used

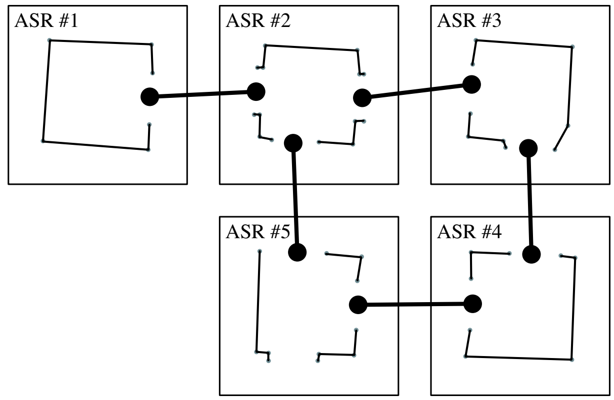


Figure 2.8: Hybrid map representation based on five independent local maps (termed ASRs). The individual ASRs are related by connectivity information captured in a graph representation. Edges link to so-called exits in the local maps. This map has been derived from log files supplied by Margaret Jefferies.

to construct hierarchical representations. Hybrid maps constructed by the embed operator have been termed *patchwork maps* (Buschka, 2006). The work by Jefferies & Yeap[7] can be regarded as a prototypical use of the embed operator. In their work, the map representation is organized in two distinct layers: On one layer, separate absolute metric maps are constructed for a local surrounding (termed ASR for Absolute Spatial Representation). An ASR is a local map that registers line segments in a coordinate-based reference system. The individual ASRs are embedded in a global frame of reference that is provided by a graph structure which resembles the interconnectivity of ASRs. Connectivity is defined by so-called *exits*, salient constrictions in the surroundings of the robot. By passing through an exit, a transition from one ASR to another occurs. Jefferies & Yeap's representation offers several views, it represents metric map information by the individual ASRs and represents topological information of connectivity of local spaces as graph structure—the ASRs are embedded in the topological map. Thus, an ASR-based map comprises two combinations of features and configuration representation: line segments and coordinate-systems on the level of individual ASRs, local maps and graph structure in the top-level map. Fig. 2.8 depicts one exemplary map consisting of five individual ASRs. A similar approach of local maps has been taken by (Bosse et al., 2003), who suggest an so-called atlas framework of local maps. Local maps maintaining their own coordinate system for registering map features maps are linked by specifying a mapping of coordinate systems in a graph structure. In the graph structure, vertices refer to distinct local maps and edges are labeled by coordi-

[7] Jefferies et al. (2001); Jefferies & Yeap (2001); Jefferies et al. (2003, 2004a); Yeap & Jefferies (1999, 2000)

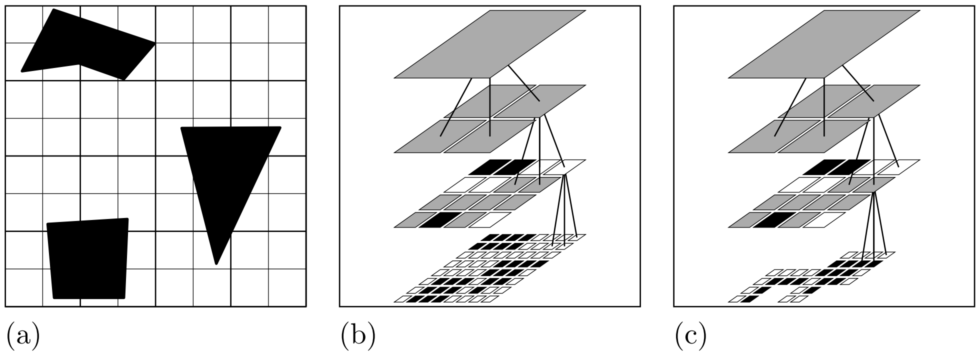


Figure 2.9: (a) Simple environment decomposed into discrete cells, (b) corresponding Quad Tree, (c) pruned Quad Tree.

nate transformations.

To combine two complementary spatial representations, the link operator is utilized. For example, Nieto et al. (2004) suggest combination of occupancy grids with a map registering positions of landmarks. Linking map representations that present different levels of granularity is a prominent approach to increase computation speed in path-planning. For example, Quad Trees (see Latombe (1991) for an overview) are popular for representing cell occupancy in different resolutions. Since every level of resolution provides an absolute global view on the environment, a Quad Tree representation is most adequately described as a linkage of global maps at different levels of resolution—such maps have been termed *parallel maps* by Buschka (2006). To construct a Quad Tree representation, four neighbored cells[8] in one layer are grouped and linked to a single cell in the next coarser layer. This results in a tree-like link structure of the overall representation (see Fig. 2.9). On the finest level, cells are classified to be either free or occupied. Abstracted cells can be either free (all children are free), occupied (all children are occupied), or mixed (some children are free and some children are occupied). Since neither children of free nor of occupied cells bear further information than their parents, they may be pruned, thereby compacting the representation. Path-planning on basis of a Quad Tree representation starts on the coarsest level which contains only a single cell. Only if a cell considered for path-planning is labeled "mixed", the finer levels of the map need to be considered; this speeds up computation in typical environments, as it is often not required to deeply descend into finer layers containing many cells.

[8]Depending on the cell topology, i.e. the definition of a neighborhood structure, different partition schemes are possible—see Latombe (1991) for details.

2.4.3 Discussion

In this Section, I have presented approaches to map organization, distinguishing uniform and hybrid maps. Uniform maps offer one view on the map representation by utilizing one set of map features and one representation of configuration knowledge. Hybrid maps offer several views that are either related by embedding or linking; in these maps distinct views are closely related.

In some situations, it can be difficult to classify an approach at hand. One example is the approach by Thrun (1998) to metric-topological mapping. Metric information is represented using an occupancy grid and route information is represented by means of a roadmap. This suggests that the representation is a hybrid map. However, the two distinct views remain decoupled. The mapping task is performed exclusively considering the occupancy grid. When mapping is finished, the route map is derived from the occupancy grid map to exploit the compactness of the GVG in path planning. In this approach, the additional view offered by the route map is exclusively considered for path planning and, henceforth, can adequately be regarded as a preprocessing step in path planning. Thus, Thrun's approach is most adequately regarded to utilize two distinct uniform maps. To transform this representation into a hybrid map, interrelation of route map and occupancy map need to be introduced, for example by propagating changes in one representation to the other.

Hybrid maps can be classified into parallel maps constructed by linking uniform maps and patchwork maps (hierarchical maps) constructed by embedding uniform maps in a superordinate frame of reference. Parallel maps allow the utilization of task-specific views, e.g. one view for mapping and one view for navigation. Since a single view representing navigable space by the outline of obstacles is well-suited to mapping and navigation (see Section 2.1.4 and Section 2.3.4), parallel maps offer no advantage over uniform maps as regards developing a suitable representation for robot mapping. Moreover, uniform maps are easier to maintain, since no coordination of different views is required.

In principal, hierarchical map representations could ease mapping as they allow decoupling information truly which may be vaguely related, e.g. two rooms at remote parts of a building. Decoupling local areas avoids difficulties of integrating vaguely related information in a single absolute representation. As regards the aforementioned remote rooms, detailed information about their spatial relation may not be known, since inferring their spatial relation requires a long reasoning chain relating the position of the rooms to several objects in-between first. Making vague relations explicit in an absolute map can easily introduce errors that may be difficult to detect and to resolve. In contrast, in the restricted area of a hypothetical local map, such errors are comparatively small and might even require no special treatment.

However, hierarchical representation also complicate robot mapping. Partitioning of space is fixed in present approaches, i.e. the area covered by a local map cannot be adapted. If multiple robots independently map their environment (maybe starting at different positions), the resulting partitions are likely to differ. Thus, local maps in hierarchical representations that have been acquired independently are often not congruent. Interrelating such representations is far from being trivial and has, to my knowledge, not been addressed so far. Developing adaptive partitioning schemes can be helpful to overcome these complications, but this is not in the scope of my work.

Path-planning on a hierarchical map can be decomposed in analogy to a single division step in divide-and-conquer algorithms. Path planning on the level of individual maps is performed as metric path planning and on the topological level by means of graph search. This approach can speed up computation, as costly metric path-planning is restricted to comparatively small local maps. On the topological level, efficient graph search on a comparatively small graph can be employed.

Evaluating the characteristics of uniform and hybrid maps, hybrid representations offer no advantage over uniform maps that would make up for the complex handling of hybrid maps. The most decisive aspect in designing a spatial representation in robot mapping is how information in the map is represented in terms of map features and configuration information and how it can be exploited in the determination of correspondences. In other words, for designing an advanced spatial representation and for developing reasoning techniques, it is appropriate to start on uniform maps. The results may then serve as starting point for future work on advancing techniques to construct and maintain a hierarchical map that makes path-planning more efficient and allows decoupling vaguely related places.

2.5 Summary & conclusion

This Chapter presented an overview on the multitude of spatial representations suggested for application to robot mapping. I have introduced a classification scheme to characterize map representations according to their selection of map features, their configuration representation, and their organization. The classification scheme is applied to derive strengths and weaknesses for classes of representations. On the level of map features, I have drawn parallels between map representations in robotics and shape representations in computer vision. Making this connection explicit, advanced techniques can be transferred among the research areas.

Map features can be classified into view-based representations, landmark-based representations, and representations of knowledge about navigability of space. I have argued for object-centered representations (sometimes called object maps) that represent the boundary of navigable space. The review of map features and shape representation demonstrates that shape feature representations originating from the field of computer vision are applicable to the field of map representations as well. There already exists a close connection between structural (skeleton-based) shape representations and the utilization of Generalized Voronoi Diagrams in route-based representations. Additionally, polylines representing the contour of objects are utilized in computer vision as well as in map representations. Boundary-based representations of navigable space using universal polylines are particular adequate for map representations in robots that need to be related to external map information, as the boundary of navigable space is represented in both. So far, polyline-based map representations have only been used for manual world modeling or compaction of point-based maps. My work aims at devising the techniques for mapping using polylines.

On the level of configuration information, I have elaborated on the utility of different modalities of spatial information and their representation. Configuration knowledge comprises the modalities of distance, direction, topology, and ordering. Representation formalisms can be classified into qualitative and quantitative approaches. Rich information about the position of features is valuable for distinguishing map features, as features and their location are intimately connected. The richest information is offered by distance and direction knowledge jointly represented in fine-grained coordinate-based geometry. I argue for registering polylines in a coordinate-based map. Such a representation is a suitable basis for adopting efficient navigation strategies.

Besides coordinate-based geometry, additional information can be valuable too. Determining a solution to the correspondence problem, configuration knowledge extracted from the view of the robot is matched against the map; plausible mappings respect configuration knowledge. The modality of ordering information can be valuable to facilitate efficient consideration of configuration knowledge by means of linear matching techniques. In general, qualitative relations describe at an abstract, coarse level. Thus, they are well-suited to make confident knowledge explicit in a configuration or to abstract from differing details, e.g. when relating knowledge sources of different granularities. Exploitation of qualitative knowledge can lead to a reduction of the search space in the correspondence problem, thus, qualitative knowledge is helpful for deriving an efficient solution.

Map configurations have been reviewed, distinguishing uniform and hybrid maps. Uniform maps offer a single view on the map representations, whereas hybrid maps offer multiple views. The operators 'embed' and 'link' have been described that allow distinct pairs of feature and configuration representations to be combined to a map. I have discussed that in the scope of this work a uniform amp is adequate. Such a map is easy to construct and maintain.

In summary, the review leads to the conclusion that a representation of navigable space is essential to mapping and navigation. In particular, polylines modeling the outlines of obstacles are adequate map features. Polylines can be interpreted as shape information and allow adopting shape analysis techniques to distinguish individual features. Coordinate-based geometry is well-suited for representing the configuration of individual polylines, but explicitly considering a qualitative arrangement with respect to a suitable ordering appears advantageous. Rich information of the quantitative representation can be combined with efficient matching techniques that are based on qualitative information.

Chapter 3

A functional analysis of robot mapping

Just hold me close
Then closer still
And you'll feel the probabilities pulling us apart.
Anne Clark, Poem For A Nuclear Romance

In this Chapter, I approach robot mapping from an algorithmic perspective. I argue for a functional perspective on robot mapping and provide a decomposition of it into distinct subtasks. I review approaches with respect to the individual subtasks. Methods addressing these subtasks are algorithms that operate on the underlying map representation. In my review, I acknowledge the conclusions drawn from my review of map representations by focusing the presentation on methods that are suitable for handling a representation of the boundary of navigable space.

To classify robot mapping from a computational perspective, Thrun (2002) examines the utilization of techniques to tackle uncertainty. In the following, I refer to this classification as the *uncertainty perspective*. Thrun argues that uncertainty in observations is the main reason that would make robot mapping a challenging endeavor—different methods to handle uncertain information would outline dimensions of potential approaches. Arguably, adequately handling uncertain information is among the key challenges of robot mapping, but there are other key challenges as well. In my introductory motivation, I argued that sensibly processing of spatial information is one of these key challenges, too. The uncertainty perspective takes a view that is independent of spatial representation techniques underlying all components of robot mapping, including the correspondence problem or the merging problem. The uncertainty perspective abstracts from processing of spatial information.

Addressing spatial representation and reasoning, the uncertainty perspective offers no adequate means to review techniques of spatial information processing. In context of this dissertation, it is most appropriate to identify distinct subtasks that take a functional role in processing spatial information. I propose a functional decomposition of robot mapping, which I refer to as the *functional perspective*. This perspective focuses on the utility of individual approaches with respect to specific spatial problems, such as the correspondence problem. The functional perspective offers a classification scheme that is orthogonal to the uncertainty perspective by focusing on spatial information processing.

In principle, several other views can be taken on robot mapping, depending on a specific aspect in focus. For example, primarily addressing dynamics and changes in the environment, a perspective on handling temporal information could be most appropriate. In current research, the uncertainty perspective is dominating. Virtually all approaches employ a stochastic modeling to represent uncertain information. Advances in probabilistic techniques demarcate today's state-of-the-art in handling uncertain information and have significantly contributed to the first milestones in autonomous service robots such as the museum tour-guide RHINO (Burgard et al., 1999a; Thrun et al., 1998a). Achievements in computational stochastic have given rise to the popularity of probabilistic reasoning, making these techniques an indispensable and fundamental component in many approaches. Therefore, I describe relevant techniques of probabilistic reasoning and discuss their principle characteristics and interaction with spatial information processing, before I turn to the functional analysis. Discussing basic properties of stochastic models, I derive limitations and demonstrate how these techniques can benefit from advanced processing of spatial information.

3.1 Addressing uncertainty with stochastic

Most of today's approaches employ some kind of stochastic model to handle uncertainty stemming from sensor data; probability distributions are represented instead of single values. In its ultimate interpretation, the map itself is interpreted as a probability value depending on the *side conditions* of observations made (cf., e.g. Montemerlo et al., 2002; Thrun et al., 1998b). In other words, a thoroughly stochastic model interprets plausibility of data integration in terms of probability and pursues computing the most probable map m dependent on data d obtained by observation, i.e. $\operatorname{argmax}_m p(m \mid d)$. Some approaches even aim for the full probability distribution $p(m \mid d)$ instead of the single, most likely map.

Though individual approaches differ, underlying stochastic foundations, models, and assumptions are the same. In the following I present a brief overview about capabilities, limitations, and implications of stochastic reasoning.

3.1.1 Stochastical foundations

The most import basis for stochastic reasoning in robot mapping is the Bayesian theorem. Given some observationss o and a hypothetical explanation (model) H, Bayes theorem expresses the so-called *posterior probability* of H under side condition o:

$$p\left(H \mid o\right) = \frac{p\left(o \mid H\right)\ p(H)}{p(o)} \tag{3.1}$$

Put differently, the rule allows for a reverse in computing the conditional probability of a hypothesis: to regard a *generative* conditional probability $p\left(o \mid H\right)$ of observing o in a hypothetical model H. The term $p(o|H)$ is called *likelihood* of the data and describes the probability to receive the observation o when the model is given. The term $p(H)$ is the *prior probability* of the model and reflects one's initial belief.

Any observation is affected by various distortions such as measurement noise. It is widely assumed that, even though each single influencing factor may be unknown, the overall distribution can be modeled using a normal distribution $\mathcal{N}(\mu, \Sigma)$ centered at the expectation μ and with covariance Σ.

For measuring in probabilistic domains the so-called Mahalanobis distance is defined as a quadric form of x, y in relation to a covariance Σ, $d_M(x, y) := (x - y)^T \Sigma^{-1}(x - y)$. In the case of an one-dimensional Gaussion $\mathcal{N}(\mu, \sigma)$, a Mahalanobis distance of 1 to the center μ marks a deviation by σ. An important application of Mahalanobis distances is gating, e.g. in correspondence determination: Candidates are pruned if the Mahalanobis between observation and estimation (e.g. regarding position) exceeds a given threshold, which is equivalent to the probability of correspondence falling below some confidence value. The Mahalanobis distance allows for a covariance-sensitive threshold.

3.1.2 Stochastic formulation of localization and mapping

Stochastic modeling of robot localization (and mapping) is based on Bayes theorem. The task of localization is posed as the task of computing the posterior probability of a pose ξ_t at time t, given all previous observation data $d_{1:t}$ (consisting of observations $s_{1:t}$ and odometry information $o_{1:t}$) and *known* map m:

$$p\left(\xi_t \mid d_{1:t}, m\right) = \eta_t\, p\left(d_t \mid d_{1:t-1}, \xi_t, m\right)\ p\left(\xi_t \mid d_{1:t-1}, m\right) \tag{3.2}$$

The denominator in Bayes theorem provides no helpful information in this formulation and is commonly replaced by a constant η which provides the scaling required to make the numerator a probability distribution (Thrun, 2000). Now, the *Markov assumption*, also referred to as *static world assumption* (Thrun, 2000), is applied to express that the robot's pose is the only state in this model, i.e. that distinct measurements are independent from one another except for their relation to the view pose. This allows simplification of the equation by removing $d_{1:t-1}$ in the estimation of d_t from the side conditions:

$$p\left(\xi_t \mid d_{1:t}, m\right) = \eta_t p\left(d_t \mid \xi_t, m\right) p\left(\xi_t \mid d_{1:t-1}\right) \tag{3.3}$$

Applying the rule of total probability, the second term on the right hand side of the equation is expanded:

$$p\left(\xi_t \mid d_{1:t}, m\right) = \eta_t\, p\left(d_t \mid \xi_t, m\right) \cdot \int p\left(\xi_t \mid d_{1:t-1}, \xi_{t-1}, m\right) p\left(\xi_{t-1} \mid d_{1:t-1}, m\right) \mathrm{d}\xi_{t-1} \tag{3.4}$$

This permits another application of the Markov assumption, since the pose ξ_t is only dependent on the last motion estimate o_{t-1} and pose ξ_{t-1}. This yields the recursive form of probabilistic localization (Thrun, 2002; Thrun et al., 2005):

$$p\left(\xi_t \mid d_{1:t}, m\right) = \eta_t \overbrace{p\left(s_t \mid \xi_t, m\right)}^{\text{perception model}} \cdot \int \overbrace{p\left(\xi_t \mid o_{t-1}, \xi_{t-1}, m\right)}^{\text{motion model}} p\left(\xi_{t-1} \mid d_{1:t-1}, m\right) \mathrm{d}\xi_{t-1} \tag{3.5}$$

To employ stochastic reasoning, a probabilistic motion model and probabilistic perception model need to be defined. Generally, they are time invariant so that the index t may be dropped. The perception model provides the probability value of receiving a measurement s_t given that the robot is located at pose ξ_t. Scan registration techniques (scan matching) can be applied to compute this value (Hähnel et al., 2002). Similarly, the motion model describes the probability of being at pose ξ_t given the odometry reading o_{t-1} for the movement from a previous pose ξ_{t-1}. Such model can be learned from empirical experiments with a robot (Thrun, 2000).

For simultaneous localization and mapping, equations that express the posterior probability of the robot being at pose ξ_t *and* the map being m_t can be derived in a similar way. Assuming that the map is independent of time and the robot motion is independent of the map, the resulting equation reads as follows (cf. Thrun, 2000, 2002)[1]:

$$p\left(\xi_t, m \mid d_{1:t}\right) = \eta_t'\, p\left(s_t \mid \xi_t, m\right) \int p\left(\xi_t \mid o_t, \xi_{t-1}\right) p\left(\xi_{t-1}, m \mid d_{1:t-1}\right) \mathrm{d}\xi_{t-1} \tag{3.6}$$

[1] As can be observed, the map does no longer appear in the side conditions of $p(\xi_t | \cdots)$, since it can be dropped by assuming independency of robot motion and map, which is a difference to the localization model of Eq. 3.5. The assumption can be regarded as a tribute to obtaining a computable formulation by escaping the interdependency of localization and mapping.

Besides the perception and motion model, the equation relates the joint pose and map estimate $p(\xi_t, m \mid d_{1:t})$ only to its preceding value $p(\xi_{t-1}, m \mid d_{1:t-1})$. The recursive characteristic of this formulation has let to the name *recursive Bayes filter*. This formulation underlies virtually all stochastic frameworks to robot mapping; approaches to obtaining a computational solution to it differ, though.

3.1.3 Kalman filter

The Kalman filter defines a stochastically sound method for combining measurements of the same physical entity as, for example, is performed during map update. It is assumed that the values o_1, o_2 are to be combined and both are represented by normal distributions $\mathcal{N}(\mu_i, \mathbf{\Sigma}_i),\ i \in \{1,2\}$ and that o_1, o_2 are stochastically independent. The combined measurement is also described by a normal distribution $\mathcal{N}(\mu, \mathbf{\Sigma})$ with expectation value according to the average of both expectations weighted with their respective covariance (for example see Bauer, 1991):

$$\begin{aligned} \mu &= \left(\mathbf{\Sigma}_1^{-1} + \mathbf{\Sigma}_2^{-1}\right)^{-1} \left(\mathbf{\Sigma}_1^{-1}\mu_1 + \mathbf{\Sigma}_2^{-1}\mu_2\right) \\ \mathbf{\Sigma} &= \left(\mathbf{\Sigma}_1^{-1} + \mathbf{\Sigma}_2^{-1}\right)^{-1} \end{aligned} \tag{3.7}$$

This builds the basis for the Kalman Filter. Let x_k denote the true state of some system and $\hat{x}_k$ an estimation thereof. The Kalman filter is applicable, if the system can be described using linear functions. Kalman filtering can then be used to stochastically propagate the estimate state $\hat{x}$ according to a system model

$$\tilde{x}_k = \mathbf{A}\hat{x}_{k-1} + \mathbf{B}u_{k-1} \tag{3.8}$$

where $\mathbf{A}$ is the state transition function and $\mathbf{B}u_{k-1}$ describes the effect of an added control u_{k-1} in the previous time step; the covariance $\widetilde{\mathbf{\Sigma}}_k$ according to $\tilde{x}_k$ gets propagated, too. Now, a measurement z_k of the current state is obtained which is assumed to be linearly dependent on the state with added Gaussian noise, i.e. $z_k = \mathbf{C}x_{k-1} + \epsilon_{k-1}$; the covariance $\mathbf{\Sigma}_{\mathbf{A},\mathbf{B}}$ of z_k is required to be known, too. This allows refinement of the propagated estimation $\tilde{x}_k$ in combination with the measurement z_k to a new estimation of $\hat{x}_{k+1}$ (Kalman, 1960):

$$
\begin{aligned}
\hat{x}_{k+1} &= \tilde{x}_k + \mathbf{K}\left(z_k - \mathbf{C}\tilde{x}_k\right) \\
\mathbf{\Sigma}_{k+1} &= \left(\mathrm{Id} - \mathbf{K}\mathbf{C}\widetilde{\mathbf{\Sigma}}_{k+1}\right) \\
\text{whereby} & \\
\mathbf{K} &= \widetilde{\mathbf{\Sigma}}_k \mathbf{C}^T (\mathbf{C}\widetilde{\mathbf{\Sigma}}_k \mathbf{C}^T + \mathbf{\Sigma}_{\mathbf{z_k}})^{-1} \\
\widetilde{\mathbf{\Sigma}}_{k+1} &= \mathbf{A}\mathbf{\Sigma}_k\mathbf{A}^T + \mathbf{\Sigma}_{\mathbf{A},\mathbf{B}}
\end{aligned}
\tag{3.9}
$$

Kalman filters provide a popular technique to construct a map that consists of locations of landmarks and that incorporates uncertainty in landmark position given that the correspondence of observations is *known* (Neira & Tardós, 2001). Gaussians are used to model the posterior $p\left(\xi_t, m \mid d_{1:t}\right)$. The probabilistic state ξ_t, m is a high-dimensional vector comprising, in the case of planar maps, robot pose (x, y, ϕ) and $x-$, and $y-$coordinates for all landmarks. However, the necessity to specify linear models for perception and motion is widely regarded as an inadequate restriction (Thrun, 2000; Thrun et al., 2005).

To allow for non-linear models, the Extended Kalman Filter (EKF) has been introduced which approximates the model's non-linearities by means of a Taylor series approximation (see Smith & Cheeseman, 1986). The overall procedure is similar to standard Kalman filters. Though the EKF advances stochastic reasoning towards less restricted system models, prerequisites for sound application of Kalman filters in general may still not be fulfillable, in particular, it may not be adequate to model uncertainty as Gaussian noise superimposed on a measured value (Thrun et al., 2005). Since Kalman filters bear quadratic computational complexity in the number of landmarks for individual map updates, research has concentrated on speeding up computation of the posterior $p\left(\xi_t, m \mid d_{1:t}\right)$, e.g. by means of efficient data structures based on a decomposition of map (Frese, 2005) or particle filters (e.g. Hähnel et al., 2003; Montemerlo et al., 2002, 2003).

3.1.4 Particle filter

Particle filters provide an alternative approach to stochastic propagation. Each particle represents an individual hypothesis which, in the case of simultaneous localization and mapping, consists of a robot pose and a map. Particles represent distinct hypotheses that, if taken together, approximate a probability distribution such as, e.g. the posterior introduced in Eq. 3.6. There are several variants of particle filter algorithms which, roughly speaking, operate on individual particles by integrating new information into the particle's hypothesis and updating an importance weight. To avoid degeneration of importance weights, i.e. convergence of all but one weight to zero, a resampling is performed to re-select particles from the distribution—see Arulampalam et al. (2001) for details.

Particle filters allow for approximation of arbitrary distributions and extend Kalman techniques in that respect. For some distributions, an adequate approximation may require many particles, though. Since the required amount of data can easily outgrow feasibility when either large sets of particles are required[2] or large environments are mapped, input data needs to compacted.

Hähnel (2004, Chapter 5) proposes application of incremental scan matching techniques to interpret a sequence of range finder measurements as pose measurements of the view pose similar to odometry information. Individual particles only need to represent a sequence of poses rather than a complete map; the sequence of scans can be stored once and shared by all particles to enable rendering of a point-based map.

Particle Filters are acknowledged for their capability of handling distributions other than Gaussians. Moreover, they are claimed to improve efficiency (Thrun, 2002). Migrating from an analytical representation of uncertainty (e.g. parameterization of Gaussians) to a discretized one can introduce severe complications, though. Most notably, the so-called *particle depletion problem* (van der Merwe et al., 2000) causes—as a side-effect of the resampling step—the set of particles to converge to the maximum likelihood of the posterior rather than truly approximating the distribution. To handle this effect, parameters for the number of particles and resampling need to be chosen carefully. A significant problem in practical application is discussed by Stachniss et al. (2004): if a robot travels through an environment, visiting one part of the environment significantly more often than another, the variation in trajectory hypotheses depletes for less frequently visited areas. This can easily result in mapping errors, if critical uncertainty information for some poses vanishes and poses previously exhibiting uncertainty are now considered certain information (see Fig. 3.1 for an illustration). Technically speaking, the represented distribution has partially converged to the maximum likelihood. This is a general problem of particle filters as has been shown by Bailey et al. (2006). So, application of particle filters to represent uncertain pose histories along a non-cyclic path with constant estimation quality would require an ever-growing set of particles to maintain approximation quality for remote poses, since pose approximations are propagated from predecessors—inherent approximation errors get propagated as well and accumulate. As of today, this effect can only be avoided by appropriately directing the robot during the exploration phase (cf. Stachniss et al., 2004). During regular robot operation with occasional map updates to respond to changes, such a technique may not be applicable, though. In summary, particle filters represent arbitrary distributions, but require careful

[2] Hähnel (2004, Chapter 5) reports employing 100 to 500 particles to successfully map environments from 12m × 50m to 232×198m, for example.

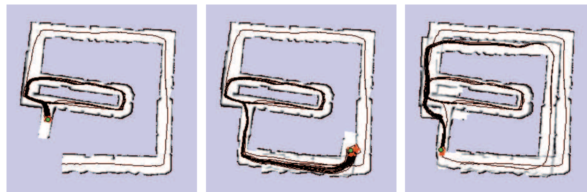


Figure 3.1: Illustration of the particle depletion problem by Stachniss et al. (2004). Traversing the left loop for some time, the uncertainty in the path, which took the robot to the start of the loop, decreases (left image). Moving on to close the outer loop, no fitting hypothesis for the remaining part of the cyclic trajectory remains represented and subsequent mapping fails.

adaption to the specific environment and may still not be applicable to derive a map from sensor information along arbitrary paths.

3.1.5 Multi-hypothesis tracking

Approaches to multi-hypothesis tracking (see e.g. Cox & Leonard, 1994; Jensfelt & Kristensen, 1999; Roumeliotis & Bekey, 2000) allow for combining the advantage of the Kalman filters, namely its sound technique for Gaussian distributions in observations with known correspondence, with the ability of particle filters to approximate arbitrary distributions. In this category of approaches, the uncertainty in correspondence determination is represented by particles, i.e. single particle represents the hypothesis of a specific history of correspondence determination. This allows utilization of Kalman techniques for propagating position uncertainty, as the correspondence relation in context of individual particles is fixed. Technically speaking, the full posterior $p(m, \xi_t | d_{1:t})$ is decomposed into individual Gaussians representing landmark positions and "glued together" by an analytically intracktable distribution over correspondences which is modeled using particle filters. Methods of decomposing a stochastic distribution are referred to as *Rao-Blackwellization* (Doucet et al., 2000). The particles are weighted by a plausibility estimated for the complete sequence of correspondences. As Thrun (2002) points out, the main problem in these approaches is that ambiguities in correspondence determination can cause the amount of required particles to grow exponentially as new observations arrive.

3.1.6 Discussion

Stochastic reasoning techniques provide a sound framework to calculate with uncertain information, given that uncertainty can be modeled by a probability distribution. Various techniques have been developed that allow propagation of stochastic information, i.e. the accumulated uncertainty of landmark position can be inferred, when the landmark is observed in relation to the uncertain pose of the robot.

To apply stochastic reasoning, several assumptions need to be made. Most importantly, individual measurements need to be stochastically independent from one another. However, in practical applications this is hardly the case. Measurements are affected by systematic distortions, like those resulting from misalignment of the robot's sensory device at certain places in the environment (cf. Section 3.4.3 for discussion and illustration). Furthermore, object surfaces can display characteristic noise. Complications of applying stochastic reasoning techniques also arise, if correspondence determination is subject to errors; for example, the Kalman filter will not converge under these conditions (Neira & Tardós, 2001).

The example depicted in Fig. 3.1 demonstrated that a purely stochastic formulation is limited: subsequent mapping (middle and right image) introduced overlapping, self-intersecting corridors in the map. A more sensible processing of spatial information should have rejected any attempt to register spatially conflicting knowledge. A sensible approach to the correspondence problem should have registered a correspondence despite a moderate position error. In the case of the middle map in Fig. 3.1, the position estimate deviates by just about half the corridor's width. Nevertheless, no correspondence is established, but the conflicting knowledge sources are simply overlaid.

To conclude, efficient and sound techniques exist for reasoning about information suffering from measurement noise. However, handling the full bandwidth of uncertainty inherent in robot mapping tasks is beyond today's capabilities. Furthermore, it is questionable if a stochastic framework can provide the required means at all, since application of stochastic models requires many simplifying assumptions to be made in order to fit to a computationally tractable model; these assumptions introduce effects not thoroughly understood. Advancing intelligence in spatial information processing in terms of relaxed requirements on estimates can help to overcome limitations faced in current approaches.

3.2 Functional components of robot mapping

In the following, I present my functional approach to analyzing algorithms in robot mapping. I distinguish four distinctive tasks; in one or the other way, any approach to robot mapping handles these tasks:

- View acquisition from sensor data
- Correspondence determination between view and map
- Alignment of the reference systems of view and map

- Merging view information and map

View acquisition describes the process of mapping sensor information to a spatial representation representing the robot's view. By sensor information, I subsume information obtained by the robot's own sensors and externally supplied information, such as maps or information obtained by communication with a fellow robot (cf. Section 1.1). In that sense, external information provides an abstract sensor reading. View extraction comprises two steps: feature detection and construction of a representation of feature configuration. Feature detection interprets sensor readings according to the types of features employed in the robot's map representation, e.g. a grouping into geometric primitives occurs. Feature extraction is a complex process that is discussed in detail in Section 3.4. To construct the configuration resembling the robot's view, spatial information interrelating the extracted features is made explicit according to the representation formalism at hand; this step is straightforward and is not explicitly covered.

Correspondence determination relates spatial knowledge from the observation and the map. To achieve this, matching algorithms are employed which compute a plausible correspondence given the uncertain and partial information retrieved from the observation and the map. In stochastic frameworks, correspondence determination rates hypothetical correlations of observations and map in terms of a probabilistic measure, i.e. the perception model. Put differently, all possible associations are rated by a matching plausibility.

Alignment procedures determine the mapping from the observation's local frame of reference to the map's absolute frame of reference on basis of a matching of observation and map. In other words, the robot is localized.

Merging is used to update the internal map on the basis of the current observation. At this point, observation and map are already aligned. Merging includes registration of newly explored features and refinement of repeatedly observed features.

These four functional components may be combined in different ways. Interconnection of components characterizes the overall functional architecture. In the following, I discuss two alternatives and their implications.

3.3 Mapping architectures

Mapping architectures are defined by the interrelation of the individual functional components. Two principle mapping architectures are commonly employed. First, the sensor data may be processed sequentially as it arrives. Features are extracted from the sensor data and matched against the map. Finally, observation and map are aligned on the basis of the correspondences determined and the map is updated. This category of architectures aims at providing up-to-date map information as new sensor information arrives. Such step-by-step map construction is commonly referred to as *incremental mapping* (cf. Thrun, 2002). A second category of approaches processes all sensor data in parallel. First, the robot collects sensor data which is then interpreted to features, correlated, and integrated to a map representation. I refer to such all-at-once approach as *closed mapping*. In the following, I discuss these alternatives in more detail.

3.3.1 Incremental mapping

Incremental mapping approaches compute an up-to-date map in an any-time fashion as new information arrives. The current map $(m)_t$ at time step t contains all information which, in combination with the new observation, is required to update it to $(m)_{t+1}$. The new observation is used to refine the information represented in the map and new features get registered when they emerge. Given that involved computations can be carried out fast enough, incremental mapping allows for real-time map construction and is, henceforth, the most commonly pursued approach (cf. Thrun, 2002).

Dissanayake et al. (2001) point out the importance of correct correspondence determination: no information besides the current map is retained so that recovering from past errors is often not possible. In presence of erroneous feature association, a statistical refinement procedure may not converge.

The key challenge in incremental mapping is to handle cyclic environments. Accumulating localization errors in the course of ongoing mapping can lead to large errors in position estimation. Fig. 3.2 illustrates this using an example of Gutmann & Konolige (1999): a robot has traveled a cyclic path and the accumulated error caused the estimated robot pose to significantly differ from the true pose. As a result, the robot did not realize that it re-entered a previously visited area. Due to not discovering a correspondence of places, subsequent mapping obfuscates the map (see Fig. 3.2 or middle and right map in Fig. 3.1 on p. 80).

3.3.2 Closed mapping by Expectation Maximization (EM)

An alternative to incremental mapping algorithms are closed mapping algorithms which consider all sensor data in parallel after the robot has finished exploration. Simultaneous evaluation of sensor data helps to handle the pejorative effect of accumulating mapping errors in incremental approaches. The key algorithm utilized in closed mapping is the statistical Expectation Maximization (EM) algorithm (Thrun et al., 2005). It determines the maximum likelihood map by optimizing its parameters. In robotics, EM has also been applied to various other tasks, including feature extraction (cf. Section 3.4.2). I

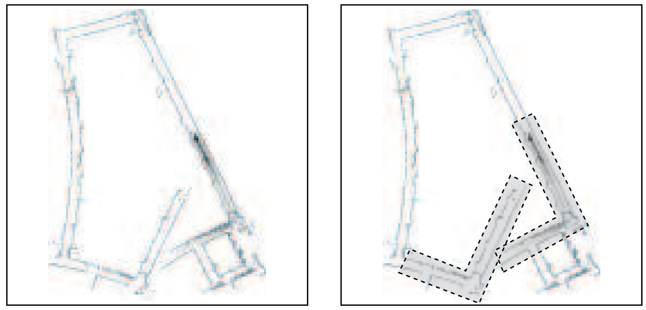


Figure 3.2: (a) Accumulation of localization errors is a side-effect of purely incremental mapping, straight hallways are bend in the map. The robot may not notice that it has re-entered a previously visited area (marked in (b)). The map computed from sensor data is taken from (Gutmann & Konolige, 1999).

outline EM in greater detail here and will reference back when reviewing feature extraction.

EM algorithms have been applied to mapping by congruently aligning sets of range measurements (Burgard et al., 1999b; Thrun et al., 1998b) and to consolidate a map of points to a more compact representation, e.g. by fitting polylines (Latecki & Lakämper, 2006b; Veeck & Burgard, 2004). Additionally, EM has been used to feature extraction (Sack & Burgard, 2003) from single observations similar to interpreting an overall map into geometric primitives.

EM is applied to determine the optimal parameter vector $\vec{x}$ with respect to some quality measure given some data d. In the case of constructing a point-based map or grid map, the data d constitutes from the robot's local observations, e.g. range scans. Poses at which the data has been recorded define the parameters $\vec{x}$ (Burgard et al., 1999b; Thrun et al., 1998b). When applying EM to construct a map of landmark positions, $\vec{x}$ describes the positions of landmarks and the observation poses (Koenig & Simmons, 1996; Thrun et al., 1998b).

The EM algorithm is iteratively performed as a two-step process: In the expectation step (E-step), the probabilistic support of the current step's choice of parameters $(\vec{x})_s$ is evaluated, i.e. it is determined how well the data accords with the expectation induced by $(\vec{x})_s$. In the successive maximization step (M-step), $(\vec{x})_s$ is optimally fit to the expectation, yielding a new parameter vector $(\vec{x})_{s+1}$. In the case of EM applied to maps of landmarks, position of landmarks and poses are selected to most congruently fit the observation using a probabilistic support measure as weighting, i.e. highly probable correspondences have a stronger effect than less probable correspondences.

EM can be viewed as breaking up the interdependency of localization and mapping (cf. Section 1.1.1) by alternately developing a hypothesis of localization for all poses (while retaining the map) and improving the map (while retaining the poses). In this sense, the E-step can be related to localization and the M-step to mapping (cf. Burgard et al. (1999b)).

Two problematical observations can be made about EM: first, the amount of model parameters must be known in advance. In mapping, this is the case when poses constitute the parameters; a fixed set of view poses corresponds to the individual observation. However, if EM is applied to compute a map of landmarks, the overall number of different landmarks must be known in advance (cf. Koenig & Simmons, 1996; Thrun et al., 1998b). Typically, this is not the case. Similarly, when fitting geometric models to data, the number of models is usually not known in advance. Thus, the number of parameters is initially unknown. Adaptively determining the amount of model parameters has been proposed for handling this complication. Some authors (Bennewitz et al., 2002; Sack & Burgard, 2003) apply EM using an estimated set of parameters and inspect the outcome of EM. If too many data points remain unmapped to the primitives described by the parameters, EM is restarted with an increased number of primitives and according parameters. Analogously, the number of primitives is decreased if a primitive does not receive sufficient support by the data. In the case of determining line models, this is indicated by lines that are closest to few data points only; roughly speaking, too many lines are too densely arranged. The outlined technique to adapt the amount of parameters may require careful fine-tuning of parameters. If EM needs to be restarted often, this can become computationally expensive.

In an alternative approach to adapting the set of parameters, Latecki & Lakämper (2006b) modify the EM framework itself. Their objective is to fit polylines to aligned range data. To adapt the set of parameters, the EM framework is extended by introducing line splitting and line joining steps. These steps are motivated by cognitive principles of perceptual grouping, i.e. Gestalt laws (Wertheimer, 1925).

The second problem in EM stems from EM being an iterative optimization technique: EM is susceptible to getting stuck at locally optimal parameter vectors. Therefore, a good initial estimation for the parameter vector $\vec{x}$ is required. Burgard et al. (1999b), for example, deduce this estimation by first running an incremental mapping algorithm on the data and thereby localizing the robot considerably well. Latecki & Lakämper (2006b) present nice illustrations of local minima for the task of line fitting; they claim their approach to master local minima purely by adapting the number of model components.

3.3.3 Discussion

Although Thrun (2002) regards EM algorithms as providing today's most accurate maps, the capabilities of EM are quite restricted. First of all, a good initial estimation of model parameters is required, so that sensor information needs actually to be preprocessed by a more robust mapping algorithm. Second, the accuracy of EM algorithms requires consideration of all sensor information in parallel and can, consequently, not be performed in on-line applications. The computational cost of EM in mapping applications scales with the operating time of the robot, since it directly depends on the amount of scans to be processed. Regardless of the robot's working environment, the amount of data collected will exceed the boundary of computational feasibility at some point. Moreover, current EM algorithms are not able to handle changes in the environment, i.e. to handle the presence or absence of an environmental feature from one time step on.

In contrast, incremental approaches enable online map acquisition and, therefore, allow the robot to respond to new observations and changes in the environment. However, incremental approaches suffer from accumulating errors along paths traversing previously unvisited parts of the environment. However, accumulating errors are not the core of the problem: whenever the robot reenters known terrain *and* is able to recognize the part it is re-entering, the robot correctly solves the correspondence problem. In such situations the accumulated error can be canceled out (Gutmann & Konolige, 1999). Loop detection requires tackling the correspondence problem in presence of high pose uncertainty. In other words, advancing correspondence determination by reducing the influence of pose estimates makes incremental mapping the superior approach. Aiming at advances in correspondence determination, I adopt the incremental mapping paradigm in my approach.

3.4 Feature extraction

Feature extraction is the process of interpreting sensor information with respect to a parametric model of the environment. When, for example, range data is interpreted to line segments, position, orientation, and number of line segments are the parameters to determine. Parameters are chosen to model the sensor data adequately. Often, adequacy of parameters is evaluated by means of statistical error analysis. The distance between the points obtained from range data and the extracted boundaries determines the quality of feature extraction. To interpret range measurements to a representation of the boundary of navigable space, approaches that fit lines or polygons to sensor data are particularly relevant. These approaches build the foundation of the shape extraction I develop and which is presented in Section 5.4. In principle, other models can be employed to describe the boundary of navigable space—for example, fitting of parametric curves like splines which are widely employed in computer graphics. However, such techniques appear difficult to combine with shape distance measures developed for comparing contours. This work focuses on extraction and handling of polygonal curves, which are a solid basis to shape analysis and robot mapping. Exploring alternatives and developing appropriate shape analysis is beyond the scope of this work.

3.4.1 Line fitting

Utilization of lines to represent obstacle boundaries is a popular approach (cf. Section 2.1.3) and different approaches for interpreting range data to lines have been proposed. Geometric world models that solely employ straight lines display shortcomings (cf. Section 2.1.4) which can, in principle, be overcome by grouping line segments to more universal polygonal curves. This makes the process of line extraction relevant to extracting polygonal curves.

If a set of data points is known to constitute a line, line parameters can easily be determined using linear regression. Therefore, the true objective in line fitting is to determine the grouping, i.e. to determine the set of points supporting the individual lines. Hereby, approaches in robotics can rely on the circular order of points obtained from a LRF. Obstacle surfaces (or parts thereof) correspond to connected subsets of the data points. Grouping points in a laser range measurement to distinct surfaces is equivalent to determining object transitions in the sequence of data points. Approaches to line fitting described in the literature can be classified into three groups: generalization, iterative splitting, and Hough Transform.

Generalization techniques are sequential algorithms. They have been comprehensively reviewed by Stein (2003) in the context of interpreting sequences of positions to trajectories. Generalization to extract lines is utilized by Röfer (2002), who employs the variant developed by Musto et al. (1999). Points retrieved from a LRF are sequentially processed in the order provided by the sensor. The first two points are used to initialize the line parameters for the first line. Subsequent points are tested whether they fit to the line or not; the distance of a point to the line is determined and compared against a fixed threshold. If the point is close enough, it is integrated in the line by updating the line parameters. If not, a new line is started using the current point as start point. Generalization is an efficient algorithm of linear complexity with respect to the number of points.

Iterative splitting (Duda & Hart, 1973) is performed recursively. Gutmann (2000) extensively describes its application in the context of robots equipped with a LRF. A single line is fit to all points using linear regression. While line fitting yields a standard deviation exceeding a fixed threshold, the set of points is split at the point farthest away from the fitted line. The algorithm is recursively invoked on the two resulting subsets. This results in a computational complexity of $O(n^2)$, where n denotes the number of points. The extraction is parameterized by the threshold for maximal deviation in line fitting and the minimum number of points required to constitute a line.

Forsberg et al. (1995) suggest an adaption of the Hough Transform known from computer vision (Duda & Hart, 1972). Approaches based on the Hough Transform map Euclidean space of data points to parameter space (here: line parameters). Clusters in the parameter space are interpreted as support for a specific set of parameters. The Hough Transform is particularly valuable to detect lines in fragmentary observations, e.g. if scattered points constitute a line (Duda & Hart, 1972). Pfister et al. (2003) extend the Hough Transform by including a statistical error estimate into the line extraction. Line parameters determined by Hough Transform are adapted optimally to fit a statistical model of measurement errors using numerical optimization. Unfortunately, details on how optimization can be performed are omitted and no empirical run times are presented.

3.4.2 Polygonal line fittting

The task of interpreting a contour described by points to a polygonal representation is an important task in the context of computer vision and has been intensively studied in this context – see Kolesnikov & Fränti (2005) for an overview. In the context of robotics, only few approaches deal with extraction of polylines.

Fitting polylines to points requires balancing between the amount of corner points to use and the desired proximity of the points to the polyline. In principle, a set of points may be interpreted as a single polyline by using each point as corner point. This yields the closest fit to the data, as each point is contained in the polyline. However, such approach is not useful. First, the resulting polyline does not improve compactness of the input data in terms of, as all data is retained. This counteracts to the slenderness geometric primitives can provide in map representations (cf. Section 2.1.4). Second, as sensor data is affected by noise, the polyline is fit not only to the data, but to the noise as well—this is known as *overfitting*.

For balancing between proximity to the input data and compactness of the polyline, i.e. the amount of corner points, a parameter ϵ can be introduced to specify the maximum distance between data points and the resulting polyline. The parameter can be chosen in accordance with a noise model of the sensor data. Larger noise in the data is countered by larger tolerances ϵ. Determination of the curve using the fewest points to approximate the data with required tolerance ϵ can be performed in $O(n^2)$ time (cf. Kolesnikov & Fränti, 2005), where n denotes the number of data points.

In robot mapping, an observation may cover several independent objects, which each require a polyline to be modeled. The aforementioned approaches to approximating points by polylines assume that the points are grouped to individual contours. So, sensor data needs to be grouped first. González-Baños et al. (1999) suggest a fixed threshold to cluster points (the threshold shall be chosen to obtain one cluster per obstacle). Each of the clusters is then approximated by a polyline. The number of vertices is minimized such that the distance between point and polyline does not exceed a distance threshold.

Determining a suitable threshold for grouping sensor data may be difficult. Depending on the distance of the sensor to the obstacle, the distance of consecutive points obtained from the sensor varies, as LRFs sample with a fixed angular resolution[3]. Latecki & Lakämper (2006b) suggest a dynamical adaption of the amount of model components by extending the EM algorithm[4] in their application of interpreting a set of point in terms of a generalized definition of polylines. They apply perceptual grouping techniques motivated by the Gestalt Laws (Wertheimer, 1925) to determine whether individual line segments can be combined to a polyline. Although intended for interpretation of comprehensive maps composed out of aligned LRF scans, the approach can also be applied to individual scans. It is parameterized by the desired approximation accuracy. A related approach is suggested by Veeck & Burgard (2004), which requires several restarts of the algorithm to adapt the model components though.

In the context of computer vision, a comprehensive discussion of recovering polygonal shape information from noisy data is presented in Latecki & Rosenfeld (2002). The utility of an alternative to model fitting is described, which builds on cognitively motivated principles of shape similarity consideration. On the basis of a local vertex relevance measure, a discrete curve evolution (DCE) (Latecki & Lakämper, 1999) is performed to cancel out the effects of noise.

3.4.3 Discussion

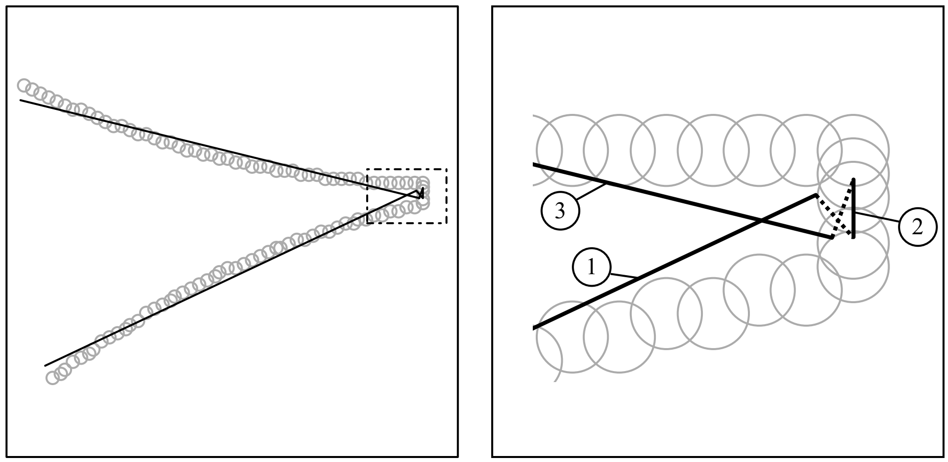
Extraction of polygonal obstacle boundaries from range data is confronted with the dilemma an unknown model to the data. Observations comprise an unknown amount of (freestanding) objects that need to be modeled by individual polylines. Even fitting a polyline to a single model component is non-trivial and requires carefully balancing proximity to the data of fitting vs. generalization. Line fitting techniques approximate contours by lines—likewise, a parameter for balancing proximity to the data vs. generalization is required. In the following, I discuss characteristic properties of line fitting and polygonal line fitting, how to handle the grouping problem, and how to assess the appropriateness of extracted features.

Line fitting versus polygonal approximation

Line fitting is well-suited to interpret straight obstacle surfaces to lines and, by combining individual line segments, it can be applied to extract polygonal

[3]Using a SICK LMS-200, the angular resolution is up to 0.5°—this yields a distance of approx. 0.4 cm between consecutive points sensing an obstacle in range of 1 meter and approx. 4 cm in range of 10 m, respectively.

[4]See also Section 3.3.2



(a) (b)

Figure 3.3: Interpreting curved obstacle boundaries (circles denote points on the contour) to linear segments can introduce self-intersections, if line segments are independently fitted to the data (solid lines); (b) shows a magnification of the marked area in (a).

models. Virtually no environment purely exhibits obstacles with straight outlines. Applied to curved contours, line fitting can introduce systematical errors, as an inappropriate model is fit to the data. As illustrated in Fig. 3.3, this can result in self-intersection of extracted boundaries. In the featured example, a flare-shaped boundary is sampled to points and is used as input for a recursive-split line extraction; this results in three individual line segments marked 1 – 3. Retaining the order of points on the contour, the corresponding line segments are joined (dashed lines in Fig. 3.3 (b)). Unfortunately, such process can result in self-intersections, as shown in the illustration. Self-intersecting lines do not adequately model real world spatial properties and need to be avoided.

By considering a wider spatial context than individual lines, polygonal line extraction can avoid self-intersection (see Fig. 5.4 on page 138 for the outcome of the shape extraction developed in this work). Of the approaches to fitting polylines to range data, EM approaches appear to be the computationally most expensive. In general, the computational complexity is difficult to determine as EM approaches iterate until convergence is reached and it may be impossible to estimate the number of iterations required. Computing times are likely to exceed feasibility for use in online mapping systems[5]. Thus, the more efficient approaches to polygonal approximation are better suited. Standard approxima-

[5]Veeck & Burgard (2004) state a compute time of about 100 minutes for 268,640 laser beams. In the case of a LRF providing 361 beams per half-plane at a rate of 50 scans a second (e.g. SICK LMS 200 LRF), this is an equivalent of nearly 15 seconds real-time.

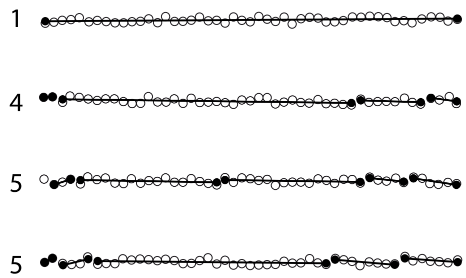


Figure 3.4: Four sets of points are obtained by simulated sensing of a straight surface; Gaussian noise is superimposed on the measurements. As a result of a fixed grouping threshold in presence of measurement noise, a single line-like object may be interpreted as multiple lines; the four exemplary outcomes yield a grouping into 1–5 line segments.

tion techniques require to specify a maximum tolerance, whereas the approach by Latecki & Rosenfeld (2002) utilizes a vertex evolution process that allows for various stop criteria.

Grouping in feature extraction

The main challenge in determining a representation of the boundary is to find a suitable grouping, i.e. a mapping of data points to model components (polylines), and, in the case of line fitting, grouping individual object surfaces to linear patches. One can distinguish two approaches: *conservative strategies* that only group local clusters and *progressive strategies* that also aim at finding clusters constituted by distributed points. Any approach that sequentially processes the sensor data and indicates an object transition, whenever the distance of consecutive points exceeds a threshold, pursues a conservative strategy. Line fitting by generalization and splitting in polyline extraction (González-Baños et al., 1999) are representatives of this category. These approaches ensure that new groups are formed if an object transition is possible—this may result in more model components than physical entities. For example, if an obstacle is partially occluded, visible fragments are interpreted as distinct groups. Besides, if thresholds are used to determine object transitions, grouping is affected by small differences on the input data. This can result in an *alias problem*, detection of multiple, distinct features instead of a single one (see Fig. 3.4 for illustration).

Line fitting by Hough Transform and the EM fitting developed by Sack & Burgard (2003) are progressive approaches as they are capable of determining clusters constituted by distributed points. Especially the Hough Transform addresses grouping into comprehensive clusters, even if the observation is fragmentary. In the above example of some partially occluded object, line fitting by means of the Hough Transform enables to group the individual fragments that are observable at separated positions. However, such an approach gives rise to serious complications. Whenever two object surfaces of distinct, freestanding obstacles are (nearly) collinear, they are grouped together, interpreting the unobserved link between them as an occluded obstacle boundary. The resulting feature representation does not agree with physical reality. Erroneous links declare truly passable space to be blocked, possibly leading to complications in path-planning. Moreover, erroneously blocked space has a severe effect on correspondence determination. Erroneously inferring a link between separate obstacles ignores an opening between obstacles through which arbitrarily extensive parts of the environment can be observed. An observation can significantly differ from the view that can be inferred according to the map. This can hinder matching observation and map which inhibits robot localization.

In contrast, conservative groupings may miss connectivity of observed feature fragments, declaring truly blocked space to be passable. If the gaps between feature fragments are smaller than would be required for the robot to pass through, these gaps have no effect on path-planning. So, erroneous linkage of distinct obstacles implied by progressive groupings has a worse effect on the map than erroneously missed links in conservative groupings. Thus, conservative grouping appears to be more appropriate for robot mapping.

Notably, the grouping that truly resembles real world object transitions is unknown. Therefore, any approach for extracting extended features remains a heuristic approach, possibly determining an inadequate grouping. Besides, determined groupings are subject to uncertainty in sensor data. Thus, it can be necessary to consider variations of the determined grouping.

Assessing quality in feature extraction

Sensor data is affected by measurement noise. If a straight obstacle surface is scanned by a LRF, linear regression can be employed to fit the line parameters optimally to the model, i.e. to determine the line receiving the largest support by the sensor data. Statistical error analysis can then be applied to keep track of the residual uncertainty in line parameters (Pfister et al., 2003). This suggests that a suitable goal in feature extraction is to aim at an optimal fit. However, there are several complications that inhibit achieving desired optimality in feature extraction.

First, the aforementioned optimal fitting of line parameters assumes that the surface is known to be straight and the sensor data only to be affected by measurement noise. Many object surfaces are not straight and it appears impossible to reliably detect straightness. Interpreting sensor data in terms of not sufficiently adequate models introduces systematical errors, i.e. stochastically dependent, intractable errors. This is particularly the case of fitting straight line segments to curved contours, but, unfortunately, the problem remains for

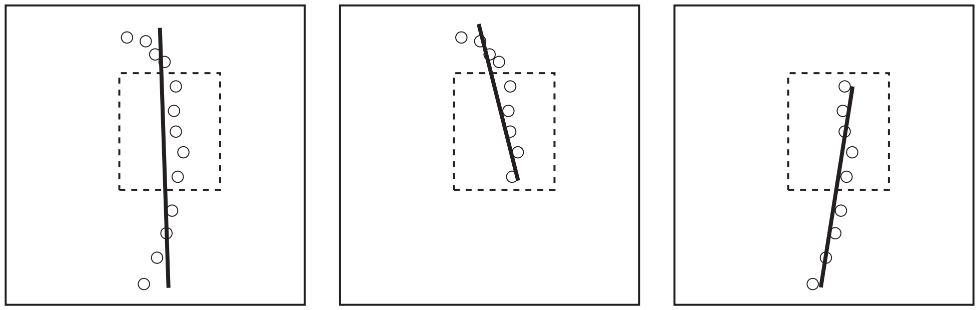


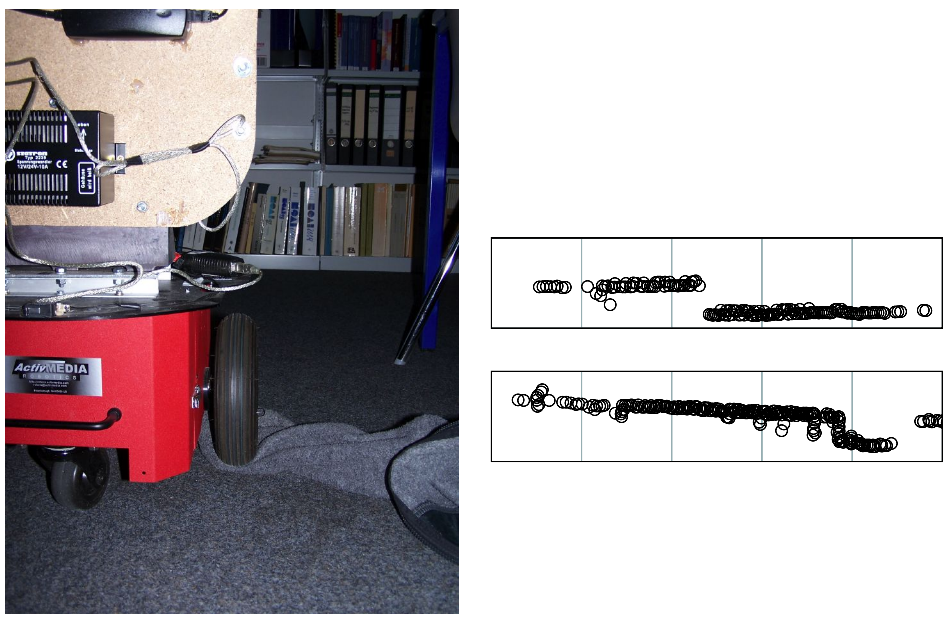
Figure 3.5: Interpreting surfaces to line segments introduces errors dependent on the boundary's visible part. In each of the three images, the boundary fragment marked by the box is approximated differently. Differences solely depend on the visible fragment.

polygonal curve approximation as well. For example, by fitting a polygonal curve to a nearly straight obstacle boundary can be dependent on the visible fragment of that boundary. This is illustrated in Fig. 3.5. In the illustration, the visibility of both the upper and lower end of the contour influences the contour orientation of the middle part. The quality of feature extraction or, equivalently, the approximation error can only be determined with respect to the observation at hand, but not with respect to the true environment.

Furthermore, if error characteristics are dependent on unknown properties such as the kind of surface material sensed, it is questionable if a sound error model can be designed at all. By empirical analysis of LRF data, I discovered greater distortions scanning semi-translucent obstacles (toned glass doors with wire frame) than opaque, white walls.

As a complicating fact, the robot and the sensor are typically not exactly leveled out. Hence, the LRF scans obstacles at varying scan heights and orientations, depending on the orientation of the sensor. This problem is inherent to any kind of features—including direct uninterpreted use of measured points. For example, such distortions can be caused by bumps in the ground. Misalignment of the scan line occurs when the robot drives on the bump. Thus, view pose and distortion are stochastically dependent errors. A possible effect of misalignment is depicted in Fig. 3.6. Sensor information can differ from objectively true data in ways that cannot be described in known statistical uncertainty frameworks, since these deviations are strongly dependent on the view pose and on the type of environment sensed (cf. Section 3.1.2 and Eq. 3.6). In other words, even if the boundary could precisely be extracted from the data, the interpretation of sensed information to obstacle boundary is affected by distortions.

To sum up, one can find a variety of line extraction techniques ranging from heuristic approaches with linear complexity to computational expensive



(a) (b)

Figure 3.6: Misalignment of the scanning device results in deviations in the observable boundary. (a) A robot misaligned by a jacket on the floor faces a shelf. (b) Scan fragments corresponding to the shelf illustrate the difference of a robot equally leveled out (top) and a misaligned robot (bottom).

approximations which seek to compute an optimum fit. The core problem is to determine the grouping. Unfortunately, it remains unknown which points constitute an obstacle or a single straight surface. This means that any approach is heuristic, as it estimates the grouping.

The full spectrum of residual uncertainty in extracted features cannot be modeled easily. Thus, some distortions in feature extraction cannot be engineered away. The utility of approaches aiming at optimally fitting models to sensor data appears difficult to comprehend, as statistical criteria for assessing the quality of model fitting rely on assumptions, e.g. the grouping is assumed to be known or surfaces are assumed to be straight. Notably, difficulties in feature extraction have an impact on stochastic frameworks that reason about uncertainty. Generative models that stochastically evaluate likelihood of sensor readings constitutes one fundamental component in stochastic frameworks (cf. Section 3.1). If no sound model of uncertainty in feature extraction is provided, erroneous conclusions may be drawn. This should, however, not be interpreted as a consent to completely drop statistical analysis, since results documented in the literature illustrate its utility (Thrun et al., 2005). However, as argued by Latecki et al., cognitively motivated approaches to interpret uncertain spatial information provide an equally sound basis to feature extraction (Latecki & Lakämper, 2006b; Latecki & Rosenfeld, 2002).

3.5 Correspondence determination

Determining correspondences between an observation and the map is fundamental to integrating the observation and the map. Mastering the correspondence problem is among the most challenging endeavors in robot mapping and it is challenging in many regards: obtaining a plausible solution, obtaining it efficiently, responding to changes of the environment or novel views on it, and handling conflicting or uncertain knowledge. Unfortunately, plausibility of matching is hard to define. Generally speaking, one desires that only similar features are associated and that spatial configurations are respected, i.e. the mapping between observation and map induced by a correspondence should be homomorphic with respect to the spatial structure. Changes, measurement noise, and uncertainty make it necessary to balance all contributing factors. As a result, it is doubtful if an indisputable definition of plausible matching exists at all. In the following, I review existing approaches to correspondence determination, describe key techniques, and elaborate on my interpretation of a plausible correspondence.

3.5.1 Achieving feasibility in correspondence computation

Correspondence determination may be posed as a search problem in the space of feature-feature correspondences. Considering a map containing m features and an observation comprising n features, there are

$$\sum_{i=0}^{n} \binom{n}{i} \cdot \binom{m}{i} \cdot i! \qquad (3.10)$$

potential correspondences, if observed features are not necessarily represented in the map, i.e. only $i = 0, 1, \ldots, n$ features get matched, and only correspondences of type 1-to-1 are taken into account. Even this restricted case is infeasibly complex, so additional knowledge must be exploited to reduce the search space and computation time. If a pose estimate is available, the *projection filter* (Lu & Milios, 1997) can be employed to disregard map features that are assumed to be hidden to the robot. Likewise, observed features are filtered. The pose estimate must be of high quality in order not to disregard features erroneously classified as invisible, which would disturb the matching. Similarly, the Mahalanobis distance (cf. Section 3.1.1) can be applied to pruning candidates located outside a confidence area (Neira & Tardós, 2001).

Computational complexity can be further reduced if features are distinguishable by restricting the search space of potential correspondence partners to similar candidates. Additionally, respecting spatial configurations of observed features in relation to the configuration of map features can also allow for a reduction of search space. Confident knowledge, for example, can be exploited in terms of hard constraints restricting the search space.

3.5.2 Recognizing individual features

Disambiguating features is one key to efficient matching. As most map representations employ simple features like points or lines, no distinctive feature similarity has yet been utilized in the context of robot mapping based on range scans. Matching on basis of point-based maps is therefore highly susceptible to wrong assignments on the level of points and is commonly performed by iterative alignment (see Section 3.5.4).

My work utilizes polygonal contours, so extensive research on shape similarity originating in the field of computer vision moves into focus. To measure the similarity of features (including shapes), feature distance measures are applied. In the context of shape analysis as performed in object recognition, shape similarity is determined by shape similarity measures (cf. Veltkamp, 2001, for an overview), which are in fact shape distance measures as identical shapes yield 0, the lowest possible value. In the following, I will therefore use the term *shape distance measure*.

Shape distance measures

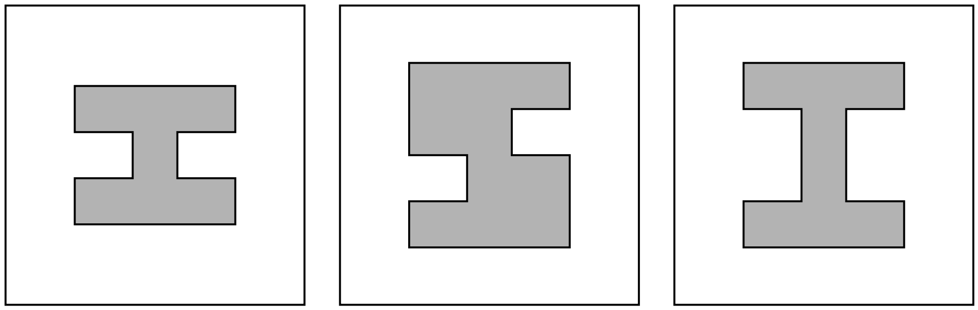
To evaluate the benefit of a shape distance measure in robot mapping, one must consider the special requirements of this domain. First, perceivable contours like nearly straight walls with only small protrusions are of relative simple kind as compared to contours of complex objects in shape retrieval. This requires a highly sensitive measure. Second, complicating the simplicity of contours, uncertainty in measurements calls for techniques that can handle this uncertainty in extracted features. Uncertainty in perception may be challenging with the total amount of shape information available, making it difficult to single out shape information. Third, in most observations objects are partially occluded; this has been discussed in Section 2.2 and I argued for polygonal shape representations to allow for representation of partially visible features. This review on shape distance measures is therefore restricted to measures operating on polygonal shape representations—for a general overview of shape distance measures, see to Loncaric (1998); Veltkamp (2001).

Any shape distance measure suitable to object recognition robustly responds to distortions or noise, as shapes need to be recognized in presence of distorted models. Distortions can be caused by a multitude of ways, e.g. by change of view pose, observation conditions, or segmentation errors—its effects on the shape cannot easily be modeled. Instead, shape distance measures address robustness by aiming at a cognitively motivated visual similarity (cf. Latecki & Lakämper, 2000; Siddiqi et al., 1999b). In this respect, object recognition approaches differ from robotics.

Some shape distance measures approach modeling of shape distance as a metric, which eases design of shape retrieval database (see Basri et al. (1998) for a discussion). However, properties of metrics like symmetry in arguments or the triangular inequality do not agree with human perception (cf. Tversky, 1977; Veltkamp, 2001). To measure shape distance, Arkin et al. (1991) utilize a L_2 measure in tangent space[6]; by pairing contours in a straight-forward manner, the approach handles distortions that are uniformly distributed over the contour. Basri et al. (1998) incorporate a matching of discrete contour points to compute a matching of points that minimizes deformation energy. Shape distance is defined by summing up local differences. Sebastian et al. (2003) design a shape distance as locally computable edit distance in a contour transformation process. Besides shape distance computation, the approach addresses shape morphing in computer graphics. Sebastian et al. report a slightly improved performance in the MPEG-7 similarity-based retrieval experiment (see Latecki et al., 2000b) over the best performance by correspondence of visual parts (Latecki & Lakämper, 2000)[7]. Curve transformation as proposed by Sebastian

[6] See also Section 2.2.1.

[7] Sebastian et al. report a retrieval rate of 78.17% whereas shape similarity based on correspondence of visual parts yields 76.45%.



(a) (b) (c)

Figure 3.7: Shape (a) is modified in two ways by doubling the length of two segments (shapes (b), (c)). Shape distance measures accumulating a local distance measure such as deformation energy (Basri et al., 1998) determine (b) and (c) to be equally similar to (a); human perception, as the reader may verify, tends to attest (c) a higher similarity to (a).

et al. requires balancing of penalty measures for curvature and position mismatch. Approaches based on local distance measures widely employ Dynamic Programming techniques which are governed by a local distance measure to compute an elastic matching of discrete contour points. Purely local determination of shape distance may in some situations yield counter-intuitive results, which is exemplary illustrated in Fig. 3.7. To overcome limitations of purely local distance measures, Latecki & Lakämper (2000) suggest a combination of a local measure and consideration of the shape's structure of arcs. A cost-optimal matching of maximal convex and concave arcs is performed, whereby the cost of arc correspondence is locally determined by a L_2 distance measure in tangent space. This approach can directly utilize the discrete structure of polygonal curves. In this respect, it is suitable for polygonal lines extracted from LRF data. Furthermore, the approach performed well in the MPEG-7 shape retrieval experiment on basis of shape similarity (as noted above) and in the experiment testing for robustness to effects of digital rotation and scaling which introduce distortions to the contour (Latecki et al., 2000b).

To design the shape distance measure in my approach, I adapt the shape distance measure by correspondence of visual parts.

3.5.3 Respecting spatial configurations

Configuration information links together individual features. Therefore, it provides a larger context than a single feature. This makes configuration information valuable for matching—in the case of features lacking distinctive properties, configuration knowledge is the only clue to feature identification. Configuration representations are often coordinate-based and Euclidean distance measures are widely applied: The positional difference of associated features is used as a measure of configuration congruency by most authors[8]. Lu & Milios (1997) proposed an additional rule to estimate the likelihood that points detected in different scans correspond to the same physical entity by considering the orientation of a fitted tangent. Similarly, Gutmann (2000) suggests a line-based alignment measure combining differences of lines with respect to position and orientation. Difference of positions can only be determined if the observation is already aligned with the map. Henceforth, all of these methods perform matching in an iterative framework, which repeatedly alternates a matching and an alignment phase. Notice, that any configurational knowledge to be exploited without prior alignment is of relative nature, e.g. as captured by qualitative representations. Relative positional knowledge is acknowledged in approaches to object recognition (cf. Grimson, 1990).

Not handling configuration knowledge on a relative basis can introduce complications though. As discussed by Neira & Tardós (2001), configuration knowledge must be jointly exploited to fully acknowledge it; a simple example—similar to the one of Neira and Tardós—illustrates this. An observer detects two features A, B which—for the sake of simplicity—are assumed to be indistinguishable (cf. Fig. 3.8). In a second observation, the observer detects three features A', B', C'. The task of determining the most plausible correspondence should yield the correspondences $A \sim A'$ and $B \sim B'$. However, if individually considering differences of position, the result may be counter-intuitive as illustrated in Fig. 3.8 (c): since the spurious detection of C is closer to the expected position of A, the correspondence $A \sim C$ is erroneously determined. Simultaneously considering the mapping $A \sim C$ and $B \sim B'$ indicates that A and B are indeed mapped to features much closer to one another than agreeable with the initial observation. Neira and Tardós propose an algorithm that jointly considers differences in position in a probabilistic model. To conclude, handling of spatial configuration information must not restrict to relationships of individual feature correspondences.

3.5.4 Matching techniques

Approaches to the correspondence problem can be classified into two main categories. Approaches that explore the discrete search space of potential correspondences constitute the first category. Their main focus is the determination of a mapping between features in observation and map. Simultaneously to relating features, spatial relations are also related, since features are embedded in a configuration representation. Matching needs to mediate between congruency of assigned features and congruency of spatial relations in the presence of

[8] Among others, see Besl & McKay (1992); Cox (1990); Gutmann et al. (2001); Hähnel et al. (2002); Lu & Milios (1997); Thrun et al. (2000b).

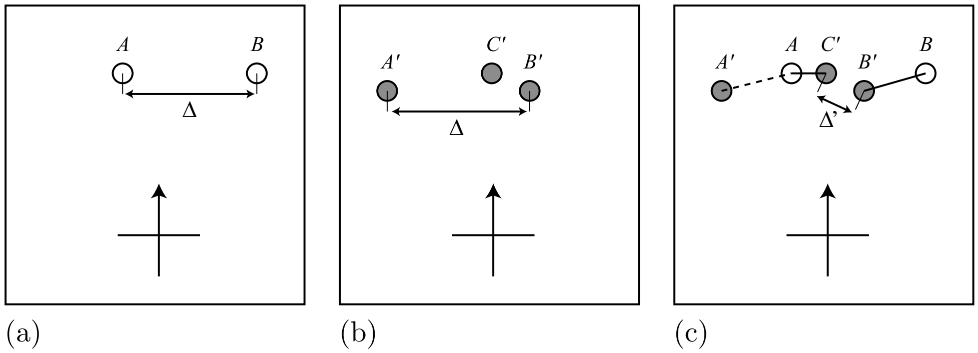


Figure 3.8: (a) An observer located as marked by the cross detects two features, A and B. (b) In a second observation from the same pose, three features are detected at different positions. This can be the result of measurement noise. (c) Both observations are matched using the compatible local reference system. Individually considering the distances of associated features (nearest neighbor) can result in a counter-intuitive matching (solid lines). Considering the relative position of features (Δ) allows handling this situation (dashes lines).

uncertain, sometimes even conflicting knowledge. Evaluation of both aspects is combined to a discrete matching algorithm that computes the most plausible correspondence, given two (maybe not fully agreeable) configurations of features. I refer to this matching technique as *discrete matching*.

The second category covers all approaches that, exploiting the inherent connection of objects and their location, pose correspondence determination as the problem of searching for the alignment which aligns observation and map. Thus, the functional components correspondence determination and alignment are jointly tackled. I refer to such techniques by the term *iterative alignment*[9].

Discrete matching

Lingemann et al. (2004) employ the greedy nearest neighbor (NN) algorithm to associate sets of feature points in a pose tracking application. The error function to be minimized is composed of feature distance (features belonging to the same class have zero distance and features of different classes infinitely high distance) and a distance measure based on the Euclidean distance of the features in a local coordinate system.[10] After association, ambiguous correspondences

[9] Iterative alignment has been termed *non-matching* by Grimson (1990); since approaches in robot mapping that abstain from explicit correspondence determination by searching for a plausible alignment employ at least some simple matching, the term *iterative alignment* is used here for clarity.

[10] The application is intended to high-speed tracking. The robot's movement between consecutive scans is small, so the relative displacement of the robot is estimated to be close to zero. Therefore, two consecutive observations can be related using the same local coordinate

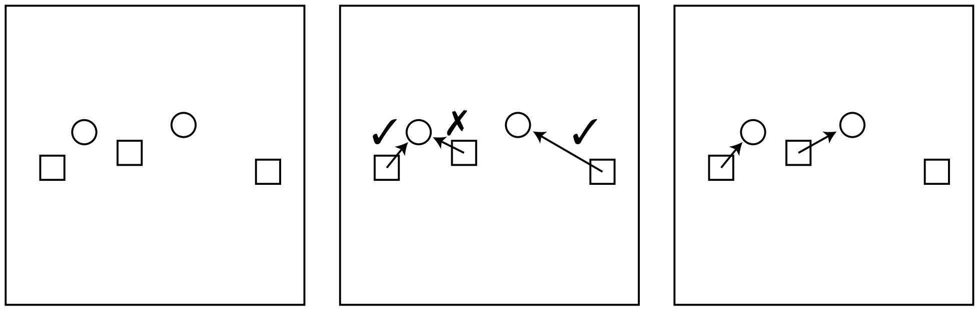


Figure 3.9: (a) Two sets of features (denoted by circles and boxes, respectively) are to be matched. (b) Using the nearest neighbor algorithm, a 2:1-correspondence is established which may be unacceptable in some applications. By resolving ambiguity, the middle feature is no longer associated. a posteriori consideration of the alldifferent constraint leads to a different result than including the constraint as side condition into the computation (c).

are resolved, i.e. whenever two or more features of one scan correspond to a single feature in the other scan, all but the closest match are removed since two distinct features must not correspond to a single one. This pruning of multiple associations can be interpreted as the application of the *alldifferent* constraint (van Hoeve, 2001). However, this approach is not equivalent to seeking the optimal association under the side condition of alldifferent as is illustrated in Fig. 3.9. Truly determining the cost-minimizing, disjunct association of features is a combinatorial optimization problem. If two sets of features of equal size are to be matched, the problem can be be posed as a maximum weight bipartite graph matching (see Chapter 4) and it can be solved in $O(n^3)$ time complexity using the Hungarian method (Kuhn, 1955), whereby n denotes the number of features in one set. Put differently, any greedy approach to computing one-to-one correspondences is heuristic and not guaranteed to find the cost-optimal assignment. But optimal assignments are desired since feature distance measures capture plausibility of feature correspondence.

Neira & Tardós (2001) formulate matching as a search in the *interpretation tree* to acknowledge the spatial configuration in the matching process. The interpretation tree is the decision tree of all possible correspondences (cf. Grimson, 1990). Each feature to be matched corresponds to a specific level of the tree; the root does not correspond to any feature. Nodes on one level represent possible correspondences of the associated feature (cf. Fig. 4.1 on page 114 for an illustration). Matchings correspond to paths in the interpretation tree from the root to some leaf. If mutually exclusive feature assignments need to

system.

be respected (e.g. the alldifferent constraint), the interpretation tree is pruned during search to remove inadmissible alternatives. In the absence of mutually exclusive assignments, the full search tree of exponential size according to Eq. 3.10 needs to be explored. Neira and Tardós note that their matching algorithm may, despite of its exponential worst-case complexity, provide a feasible approach to real applications. Applicability depends on the existence of sufficiently many, mutually exclusive assignments and a (heuristic) technique to apply branch and bound most effectively and prune the search space.

Iterative alignment

Iterative alignment techniques have been suggested to eclipse explicit, complex computation of a jointly compatible association, but to improve over NN matching. This family of approaches shifts the problem from a search in correspondence space to a problem in pose space: the goal is to compute the observation pose directly which most congruently aligns perceived features and map. In other words, it is a joint solution to correspondence determination and alignment. Iterative alignment is a continuous optimization problem, which is commonly tackled by a hillclimbing search. NN matching is iteratively performed in alternation with an alignment step. The steps are repeated until convergence is detected. The underlying idea is that the matching (and alignment, simultaneously) converges to a sensible solution if sufficiently many features are correctly assigned. In other words, erroneous associations of the NN algorithm may be recoverable in a subsequent step, i.e. after the alignment has been updated. In the case of point-based iterative matching, the algorithm is commonly referred to as Iterative Closest Point (ICP), which has its origin in computer vision (Besl & McKay, 1992). ICP has, for example, been employed by Hähnel et al. (2002); Lu & Milios (1997); Stachniss (2006). Gutmann & Schlegel (1996) complement the point-based iterative alignment by a line-based approach, which proceeds analogously.

Besides robot mapping, matching has been applied to several other domains, most notably object recognition, and is related to matching theory in graph theory (cf. Chapter 4). In his survey book, Grimson (1990) reviews several approaches to visual object recognition that make use of branch and bound techniques similar to Neira and Tardós. From the field of biological applications, Wang et al. (2004) extend the bipartite graph matching to acknowledge an order of protein's structural elements. By respecting a total linear order in the structure, the assignment problem can be solved efficiently.

3.5.5 Discussion

Shape distance measures provide a promising foundation for attacking feature similarity on the basis of polylines that represent the boundary of navigable space. These measures disambiguate features and, thereby, ease efficient correspondence determination. Complementary means to increase efficiency are offered by exploitation of configurational knowledge. In principle, this can be achieved similar to constraint propagation (cf. Tsang, 1993), treating relative positions of features as constraints. If, for example, feature A is observed north of feature B, then, by assigning A to some map feature, the set of candidates for B can be pruned. Unfortunately, uncertainty inherent in map and observation requires a careful selection of hard constraints that are applied to prune the search space and, hence, can be interpreted to model confident knowledge. Grimson (1990) notes in the context of object recognition that constraints are particularly valuable if they can directly be implemented in the formulation of the matching algorithm, i.e. respecting constraints is an integral part of the algorithm and requires no active reasoning at runtime. To give an example, elastic matching in shape distance measures by means of Dynamic Programming respects the sequence of points on the contour by algorithm design. Grimson adds that constraints should capture global configurational knowledge such that single feature assignments possibly allow for extensive pruning.

Notably, the application of the Mahalanobis distance for pruning candidates located outside some confidence area can be interpreted as an elementary application of constraint handling. To my knowledge, constraint propagation has not been further exploited in robot mapping. Instead, correspondences are sometimes pruned in a successive step. For example, Gutmann (2000) suggests to disregard already assigned features if feature correspondence entails a transformation from the agent-centered to the absolute frame of reference that significantly deviates from the overall average. Similarly, Lingemann et al. (2004) handles the alldifferent constraint in a successive step. These techniques can yield sub-optimal solutions though.

Joint consideration of configurational knowledge is important—the probabilistic, quantitative framework suggested by Neira & Tardós (2001) may not cover all aspects though. In quantitative representations, all positional information is treated equally. It is not possible to make confident knowledge explicit. This can indeed hamper recognition as a small example demonstrates: consider the observer in Fig. 3.8 detecting the features A and B and assume the distance from A to B being small with respect to measurement uncertainty. By evaluating the measurements and taking into account the error margins, one may not be able to decide which of the landmarks is located on the left and which is located on the right. The observer has, however, observed with certainty which of the two landmarks is left of the other. In any quantitative approach, this knowledge is shadowed by a representation that relates observations to an external scale rather than to one another (cf. Section 2.3.2, Section 2.3.4). I suggest to incorporate confident knowledge explicitly in terms of qualitative relations.

To sum up, approaches to matching in the field of robot mapping typically employ very efficient greedy strategies, NN for instance. Application of NN is susceptible to erroneous assignments, in particularly if the set of individually agreeable correspondence partners is high, e.g. if features are located densely. Furthermore, NN strategies require an a-priori estimate for the mapping of spatial configurations (cf. Section 3.5.3) to express a measure of joint compatibility of configuration as on the level of individual feature correspondence, i.e. to consider the relative position of features. If such an estimate is not available, branch and bound techniques can be applied to prune the search space, but such approach yields an exponential worst time complexity.

Alternatively, matching can be shifted from discrete correspondence space to continuous pose space. Iterative matching algorithms iterate matching and alignment by means of optimization, which principally allows erroneous feature assignments to be retracted. This approach displays two major drawbacks: first, it requires a pose estimate to start with. In the case of relating the robot's observation to an external map no such estimate may be available. Second, optimization algorithms are susceptible to getting stuck in local minima. This can easily happen when the optimal alignment of perception and map is of poor quality, e.g. when features identified in the sensor information are not registered in the map or vice versa as can, e.g. be caused by changes in the environment.

Statistical propagation of uncertainty in mapping is a computationally demanding task. It is often granted many computational resources by restricting matching to highly efficient, though error-prone greedy processes in order to meet time constraints for online mapping. I argue for favoring correspondence determination in allocation of computational resources—simultaneously the need for extensive stochastic computation may decrease as fewer alternatives appear "plausible" to a more sensible algorithm.

I suggest focusing on the correspondence problem in order to compute an optimal mapping from perceived features to the map with respect to a plausibility measure. The computational goal in matching should be modeled as simultaneous minimization of the feature distance measure and the spatial distance between assigned features under the side condition of confident knowledge. Doing so enables formulation of matching as a discrete optimization problem that can be solved analytically, i.e. without the risk of getting stuck in a local minimum. Confident information can be explicitly introduced by means of qualitative information, its exploitation facilitating an efficient approach. In summary, by incorporating shape distance measures resembling visual similarity and simultaneously relaxing requirements on estimates (e.g. expected position of features), on the one hand, and by explicitly handling confident knowledge, on the other hand, one can significantly reduce the number of plausible alternatives that must be taken into consideration while at the same time fostering plausibility of determined correspondences. Such approach may be regarded to resemble knowledge-based hypothesis matching in natural cognitive systems more closely and considerably cuts down the computational complexity.

3.6 Alignment

Alignment describes the task of determining a mapping between two frames of reference such that corresponding locations line up. Knowledge about correspondence of observed features and features registered in the map allows the observation's local frame of reference to be aligned to the map's global frame of reference, thereby localizing the robot.

Due to distortions (e.g. noisy feature detection), usually there exists no fully congruent alignment of observation and map. Thus, alignment is most adequately posed as the optimization task of determining the *most congruent* mapping of reference frames. In the context of 2D maps, a mapping is defined by a rotation ϕ and a translation $\vec{t}$. The most commonly used formulation of the alignment task is to determine $(\phi, \vec{t}\,)$ such that the the distance of corresponding objects is minimized with respect to the sum of least squares, i.e. the L_2 metric. Using the L_2 metric, closed form solutions exists for sets of corresponding points (see e.g. Lingemann et al., 2004) and sets of corresponding lines (see e.g. Gutmann, 2000, pp. 42). An efficient heuristic alternative for aligning sets of lines has been developed by Röfer (2002), who suggests a histogram-based correlation.

3.6.1 Alignment of contours

Aligning complex objects like sets of corresponding polylines is more difficult. One difficulty is to define a suitable distance measure for extended objects. A suitable distance measure can be derived from the distance of one or several anchor point(s) belonging to the extended objects at hand. Assuming that a correspondence of anchor points is known, Larsen & Eiriksson (2001) address alignment of polylines by linear regression and evaluate the effects of various metrics. In the case of the L_2 metric, they pose alignment as a problem of linear programming and apply the simplex algorithm. Unfortunately, it is non-trivial to determine suitable sets of anchor points and to select an appropriate mapping between them. Observe, that aligning two polylines is equivalent to aligning two observations of point-like landmarks: each observation can be interpreted as a polyline by connecting the locations of the landmarks observed in some well-defined manner.

In the context of computer vision, the Hausdorff distance (cf. Eq. 2.4) has been applied to measure the distance of pixel images (Rucklidge, 1997). Each pixel serves as anchor point and the closest pixel on the other image determines the distance (cf. Section 2.2.4). Alignment can then be performed as minimization of the Hausdorff distance. If the minimization is pursued by hill-climbing search, it is equivalent to ICP (cf. Section 3.5.4), which is a very popular method of alignment on basis of occupancy grids (Hähnel, 2004). The idea underlying iterative approaches like ICP is that no robust correspondence of anchor points needs to computed, but a simple greedy assignment stabilizes itself during the optimization process (see discussion in Section 3.5.4). Optimization techniques such as ICP depend on an initial estimate $(\phi_0, \vec{t_0})$ close enough to the globally optimal solution in order not to get stuck at a locally optimal solution. Different techniques have been suggested to determine a suitable start estimate. In robotics, odometers measure the movement of the robot and, by relating the measured movement to the last pose at which localization has been performed, the alignment can be estimated. In the context of aligning contours in computer vision, Marques & Abrantes (1997) propose utilization of the Discrete Fourier Transform to estimate alignment parameters.

3.6.2 Discussion

Alignment has a broader field of applications than just robotics, in particular applications related to computer vision require knowledge about congruent mappings of reference frames. The task is well understood if a correspondence of points or lines is provided. If no correspondence is known, iterative alignment strategies may be applicable. These strategies avoid explicit determination of a robust, sensible matching, but rely on the stabilizing effect of repeatedly performing greedy matching and alignment (cf. Besl & McKay, 1992; Gutmann, 2000). A good estimate close to the globally optimal solution is required to start the process (see Fig. 5.15 on page 160 for illustration of difficulties in iterative alignment).

Aligning sets of polylines is particularly difficult as a suitable distance needs to be defined, e.g. based on correspondence of anchor points. In context of shape distance measures, some shape matching techniques determine a correspondence of anchor points (e.g. Basri et al., 1998; Sebastian et al., 2003) or arcs (Latecki & Lakämper, 2000) as a side-effect. Thus, alignment of polylines can benefit from shape analysis. Using these anchor points, an alignment can be determined. However, since shape distance considers two polylines at a time and alignment considers two configurations of polylines simultaneously, this solution may not be optimal. Starting with this solution, iterative strategies may now be applied that allow re-consideration of anchor point assignment in a global context. This can improve the initial solution towards global optimality. In other words, required start estimates for the aligning shape features can be derived from shape analysis; no estimates need to be derived from odometry.

3.7 Merging

Merging describes the task of updating the map when correspondence of observed features and map features is known and the observation has been aligned to the map. The main objective in merging is to integrate multiple observations of the same map features into a coherent observation. Merging is applied at two levels: in the configuration representation to integrate observed configuration information and in the feature representation to integrate multiple observations of feature appearance.

3.7.1 Configuration merging

On the configuration level using coordinate-based geometry, merging determines a feature's position in agreement with multiple position measurements. In principle, this can be approached by weighted averaging. Modeling uncertainty in position stochastically by Gaussians, the Kalman filter (cf. Section 3.1.3) determines a weighted average accordingly that respects uncertainty in positions—this approach is popular if object maps (i.e. landmarks of geometric primitives) are constructed (e.g. Castellanos & Tardós, 2000; Dissanayake et al., 2001; Leonard et al., 1992; Se et al., 2002). To my knowledge, merging configuration information represented in terms of qualitative relations has not been addressed so far. It appears a challenging problem to mediate between different relations on a symbolic level, in particular if conflicting knowledge needs to be resolved. Moratz & Freksa (1998) indicates that conceptual neighborhoods (cf. Section 2.3.2) and relaxation techniques can provide a point of departure.

3.7.2 Feature merging

As regards the feature level, the feature representation needs to be updated. Only few approaches explicitly address merging of feature representations. In stochastically interpreted occupancy grids, merging is performed by stochastically combining the distinct measurements of an occupancy value—usually, ad hoc models for uncertainty in measurements build the basis (cf. Stachniss, 2006, pp. 37). Pfister et al. (2003) develop a stochastic reasoning to update line parameters in a map registering lines. Latecki & Lakämper (2006b) draw motivation from human principles of visual grouping to determine which distinct line segments are to be merged into a single one. Criteria of parallelism and proximity are evaluated. If the line segments are sufficiently close and parallel, they are merged by fitting a single line segment to the data points constituting the individual line segments.

Merging of two contours can be interpreted as determining a specific contour in the space that continuously transforms one contour into the other. Besides robot mapping, techniques to continuously transform images into one another are investigated in the field of computer graphics. As regards transformation of contours, the approach by Sebastian et al. (2003) addresses shape distance computation (see Section 3.5.2) by a transformation process that morphs one polygonal contour into another. Elastic matching by means of Dynamic Programming is performed to determine an anchor point correspondence that governs the morphing process.

3.7.3 Discussion

Merging combines distinct observations into a coherent whole. If uncertainty in feature position or feature appearance can stochastically be seized, stochastic reasoning techniques (e.g. the Kalman filter) are applicable. The spatial processes underlying merging can be regarded as weighted averaging: average positions are computed when merging landmark positions by the Kalman filter and observed feature and map feature are merged by averaging lines (Latecki & Lakämper, 2006b; Pfister et al., 2003) or contours (Sebastian et al., 2003). The approach by Latecki & Lakämper (2006b) and the approach by Sebastian et al. (2003) both address polylines, but Latecki & Lakämper only handle merging of distinct line segments in their application. Besides, the data points need to be retained to perform the merging. In an application of incremental mapping, this is disadvantageous as all data points need to be stored alongside the map, which results in a map representation of infeasible size. Thus, the approach by Sebastian et al. (2003) applicable to curve morphing appears promising. It is based on a correspondence of anchor points whose computation requires a balancing of parameters that control the matching—in context of shape distance measures comparing unaligned curves. I regarded this a shortcoming (see Section 3.5.2). However, in the context of corresponding polylines that are already aligned, demands on the anchor point matching may be relaxed. My approach utilizes a curve transformation process similar to the one suggested by Sebastian et al.

3.8 Summary & conclusion

In this Chapter, I have reviewed techniques for robot mapping and proposed a functional analysis scheme of robot mapping that identifies the components view acquisition (feature extraction), correspondence determination (matching), aligning (localization), and merging (map update).

Probabilistic techniques have been presented that demarcate today's state-of-the-art in uncertainty handling. I have demonstrated limitations of stochastic modeling in terms of restricting assumptions like state transition models for Kalman filters, in terms of mathematical limitations as particle depletion in particle filters, and fundamentally by discussing if computationally tractable probabilistic models can offer adequate means to model the full bandwidth of uncertainty at all. I have argued not to subordinate spatial information processing to stochastic frameworks, but to make uncertainty handling less challenging by improving on spatial information processing.

Two alternative mapping architectures have been contrasted. Closed mapping processes data that has previously been collected and incremental mapping aims at providing up-to-date map information in immediate response to observations. Incremental mapping appears necessary for enabling intelligent exploration strategies and for allowing a robot to respond to changes in the environment. The main deficit of incremental mapping is that position errors accumulate during the mapping process—this leads to the challenging loop detection problem. Mastering loop detection requires recognition of a place in presence of strongly uncertain pose estimates. Developing techniques to address the correspondence problem with relaxed requirements on position estimates as is addressed in this dissertation provides means to master the challenges of incremental mapping.

The crucial task in view acquisition is feature extraction. As regards features extraction, the problem of defining an optimal model fitting to uncertain observations has been discussed and an alternative approach originating in the field of computer vision has been presented. Cognitively motivated principles of visual perception provide a solid basis to feature extraction too.

Matching techniques for tackling the correspondence problem have been described in consideration of approaches to robot mapping and to object recognition. Matching techniques in robot mapping are often very efficient at cost of limited robustness; computational resources are widely spent on expensive stochastic propagation rather than on carefully handling the correspondence problem. I have argued for re-balancing these components in favor of improved correspondence determination. Improving correspondence determination simultaneously decreases the necessity of extensive stochastic propagation. Advanced matching that allows for incorporation of constraints describing confident knowledge has already been proven valuable in object recognition and appears to be adequate in robot mapping too.

Alignment techniques for aligning observation and map and, thereby, localizing the robot have been described. A transformation is computed to most congruently align corresponding features. If extended features like polylines are aligned to one another, sensible anchor points on the polylines need to be determined and a correspondence between these anchor points is required. Shape analysis provides appropriate means to derive this knowledge. Alignment of polylines can jointly be achieved by shape analysis and iterative alignment techniques.

Finally, merging techniques to update the robot's internal map on basis of aligned observations with known correspondence has been described. In the context of polygonal curves, curve morphing is applicable.

To put it in a nutshell, I have analyzed the processes involved in robot mapping and derived suitable means to realize them. Open problems have been detailed which are attacked by my approach to robot mapping presented in the following. First, Chapter 4 describes a new graph-theoretic formulation of the correspondence problem from a theoretical point of view and presents a new matching algorithm. Second, Chapter 5 describes realizations of all functional components which constitute my approach.

Chapter 4

Homomorphic matching in balanced hypergraphs

Licht aus, Spot an!
Ilja Richter

This chapter elaborates on the mathematical background of describing correspondence determination. I develop a generalized graph-theoretic formulation and relate it to research on combinatorial optimization. In the previous chapter, in particular in Section 3.5, existing algorithms to tackle the correspondence problem have been reviewed; these techniques are fairly straight-forward like greedy nearest-neighbor association and often lack of a theoretic classification. By investigating into matching problems in graph theory, a sound basis for correspondence determination can be established.

Responding to properties of perceivable features, their extraction from sensor data, and requirements for sensible matching discussed in the previous Chapters, I develop a theoretical framework for correspondence determination and introduce a new matching technique tailored to correspondence determination on the basis of extended geometric primitives. From the review I derive the following characteristics for a theoretic framework underlying matching observation and map:

- Map and observation are likely incongruent, so some features may not get associated; matchings may not be of maximum cardinality

Visibility of a feature can change for various reasons, e.g. due to occlusion, changes to the environment, or the feature may unexpectedly emerge in the field of view, to name but a few. Matchings that need to leave some features unassociated may very well be a typical case rather than an exception. However, a suitable overall goal is to associate as many features as sensible.

- Correspondences are not restricted to the type one-to-one

The alias problem in feature detection (see Section 3.4.3 and Fig. 3.4 on page 91) may cause several features to be detected for a single obstacle and, similarly, a single object may be perceived as two separate ones when a missing joint is hidden behind another obstacle. The alias problem in grouping can lead to a different amount of features detected for a single physical obstacle—features only correspond to some fragment of a physical object. Recognition of a feature may require consideration all features corresponding to the same physical entity. Thus, correspondences that cover multiple entities need to be expressed to handle the alias problem and to address partial occlusion.

- Matchings shall maximize some plausibility measure of features to correspond

Associated features should provide a plausible solution in terms of respecting additional measures of likelihood for their correspondence, such as feature similarity or configuration information. In this theoretical contemplation I abstract from concrete, potentially contributing factors. To enable determining the *most* plausible matching of features—independent of any interpretation of what plausibility could mean and how to compute it—plausibility of feature correspondence must be expressible by ordered values to enable comparison of alternative solutions.

- Side conditions to introduce confident knowledge shall be expressible

Explicit consideration of side conditions in the matching process allows to introduce confident knowledge that must not be violated by a correspondence of features (cf. Section 3.5.3). Simultaneously, side conditions reduce the search space, cutting down the overall computational cost.

In the following, I discuss relevant mathematical frameworks to express the matching problem and examine their suitability to express the outlined matching properties or presence of starting-points for a generalization.

4.1 Mathematical characterizations

Correspondence determination by means of matching can be posed in a multitude of ways. In the following, I discuss the most popular approaches and examine which comes closest to the outlined characteristics and provides a solid basis for generalization to suit the demands. In forthcoming examples a simple setup using the two sets of features $\{F_1, F_2, F_3\}$ and $\{G_1, G_2, G_3\}$ is assumed. In more general contexts, the sets $F = \{F_1, F_2, \ldots, F_f\}$ and $G = \{G_1, G_2, \ldots, G_g\}$ will be regarded.

Combinatorial optimization problems are one natural form of characterizing the matching task, as they aim at maximizing some similarity function—let S

denote this function. The problem to determine the optimal matching $\sim^\star \subseteq 2^F \times 2^G$ can be posed as follows:

$$\sim^\star = \underset{\sim \in \subseteq 2^F \times 2^G}{\arg\max} \sum_{(f,g)\in\sim} S(f,g) \tag{4.1}$$

However, difficulties in such an approach lie in the integration of side conditions and in an unclear relationship of the problem formulation to other combinatorial problems. The actual combinatorial problem structure is somewhat shadowed, making it difficult to derive efficient algorithms. To make the combinatorial aspect first matter, graph-theoretic approaches are more appropriate.

Definition 1 (Weighted graphs)**.** An *edge weighted graph* $\mathcal{G} = (V, E, \omega)$ is a graph (V, E) with a set of vertices V, a set of edges $E \subseteq \{\{v_1, v_2\} | v_1 \in V, v_2 \in V\}$, and a weighting function $\omega : E \to \mathbb{R}^+ \cup \{0\}$. Analogously, the *vertex weighted graph* $\mathcal{G} = (V, E, \sigma)$ is a graph (V, E) with a weighting function $\sigma : V \to \mathbb{R}^+ \cup \{0\}$.

The interpretation tree already mentioned in Section 3.5.4 is a decision tree in which nodes at depth d represent all possible associations of features $F_1, \ldots, F_d$ (see Fig. 4.1). It can be modeled as vertex weighted graph using feature similarity as vertex weight.

To determine a maximum weight matching, a path of maximal weight from the root $\emptyset$ to some leave is searched, e.g. using branch and bound techniques (Neira & Tardós, 2001). Interpretation trees comprise exponentially many nodes against the number of features. Generalization to other than one-to-one matches would grow the set of nodes at each level but the root by a factor of $2^g - 1$, as any node represents a potential correspondence and there are $2^g - 1$ non-empty subsets in G, i.e. potential correspondence partners. This results in a double-exponential tree size, making it doubtful that an efficient algorithm for exploration of this tree structure can be designed. There is yet another complication involved: assume, F_i is associated to $G' \subset G$ and F_j is associated to $G'' \subset G$ with $G' \cap G'' \neq \emptyset$; in such cases there seems no proper interpretation to the matching of the simultaneous yet independent association of features $G' \cap G''$.

Association graphs introduced by Ambler et al. (1973) extend expressiveness of matching by incorporating observance of relational structures, i.e. potential correspondences can be constrained by a binary relation. Utilization of association graphs transforms the matching problem into a maximum weight clique problem. Roughly speaking, nodes represent potential correspondences and the largest set of mutually agreeable correspondences is searched for. In the context of robot mapping this has been detailed by e.g. Bailey (2002, Chapter 3).

Definition 2 (Maximum weight cliques)**.** Let $\mathcal{G} = (V, E, \sigma)$ be a vertex weighted graph. A subset $V' \subseteq V$ is called a clique, if the induced subgraph $\mathcal{G}' = (V', E')$,

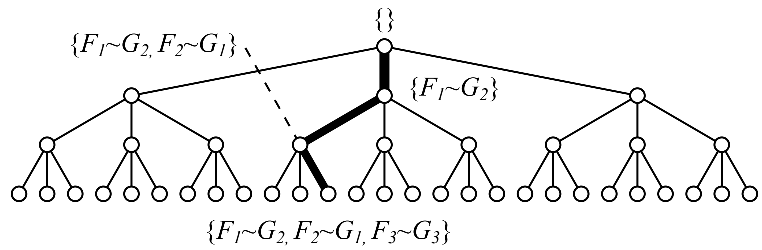


Figure 4.1: Interpretation tree and iterative construction of $\{F_1 \sim G_2, F_2 \sim G_1, F_3 \sim G_3\}$.

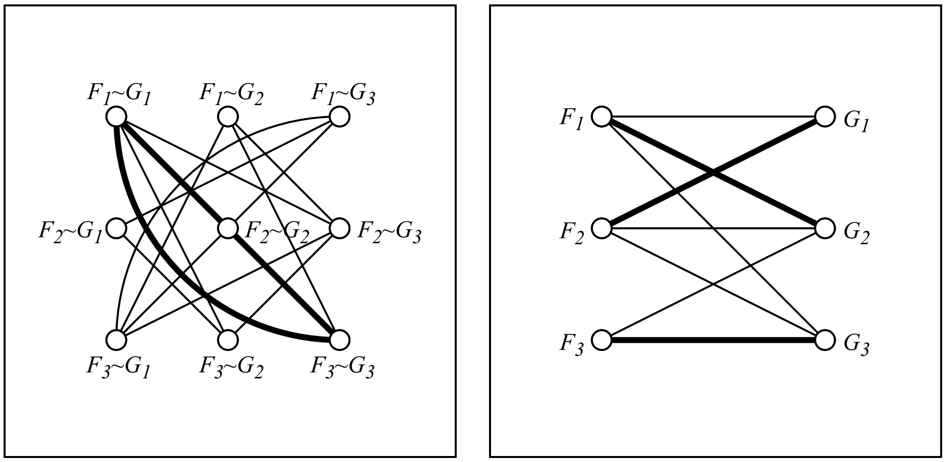
$E' = \{\{v_1, v_2\} | \{v_1, v_2\} \in E,\ v_1 \in V',\ v_2 \in V'\}$ is complete, i.e. $\forall v_1, v_2 \in V'$: $\{v_1, v_2\} \in E'$. A clique is *maximum* if it is not a proper subset of another clique and *maximal* if it is the largest clique in the graph. V' is the *maximum (vertex) weight clique* if $V' = \text{argmax}\{\sum_{v \in V'} \sigma(v) | V' \text{ is clique of } \mathcal{G}\}$.

Given that the weighting is positive, maximum weight cliques are maximum cliques. They may not be maximal, though.

Definition 3 (Association graph)**.** Let F, G denote disjunct sets of features with feature distance measure δ. Let further $R \subseteq (F \times G) \times (F \times G)$ be a (binary) relation over correspondences, i.e. R describes the set of mutually compatible assignments between F and G. The association graph $\mathcal{G}_A = (V, E, \sigma)$ defines a vertex weighted graph whereby $V := F \times G$, $E := R$, and $\sigma := \delta$.

Consider an example of matching the sets F and G whereby the alldifferent constraint shall be obeyed, i.e. the desired correspondence relation $\sim$ needs to be bijective. Assume $F_1 \sim G_2$ and $F_2 \sim G_1$ are mutually exclusive. The resulting association graph is depicted in Fig. 4.2 (a).

Unfortunately, the general maximum clique problem is known to be NP-hard. There exist, however, tractable subclasses and equivalent formulations to maximum clique problems, e.g. maximum clique computation can be mapped to a continuous optimization problem in the n-dimensional Euclidean simplex $\{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^{n} x_i = 1\}$ (cf. Pelillo, 1998) which allows to adopt heuristic optimization algorithms and thereby elegantly links the discrete and continuous domain. This is just one result from the extensive research on the maximum clique problem; refer to Bomze et al. (1999) for an extensive review. In-depth analysis of tractability for a maximum clique problem at hand is difficult and exceeds the scope of this work. As regards computational modeling of matching, association graphs already offer means to incorporate constraints to the matching and could be generalized to n-to-m-matches in a straight-forward way. One simply adds nodes for all potentially corresponding subsets $F' \sim G'$, $F' \subseteq F$, $G' \subseteq G$. This increases the set of nodes exponentially. Since the original prob-



(a) (b)

Figure 4.2: (a) Association graph and a maximal clique (bold edges) (b) Bipartite graph and a perfect matching (bold edges)

lem is already NP-hard it appears questionable if a tractable subproblem can be identified if the set of nodes is increased further.

The number of different graphs on a fixed set of nodes is exponential against the number of nodes, as there are exponentially many ways to select edges. Therefore, I argue for a formulation modeling correspondences by means of edges rather than nodes, as are used in interpretation trees and association graphs. Such framework is provided by bipartite graph matching.

Definition 4 (Bipartite graph)**.** A graph $\mathcal{G} = (V, E)$ is bipartite, if its set of vertices V can be divided into exhaustive, disjunctive sets V', V'' such that $\forall e \in E : e \cap V' \neq \emptyset \wedge e \cap V'' \neq \emptyset$. Bipartite graphs will also be denoted $\mathcal{G} = (V', V'', E)$.

Definition 5 (Biparite matching)**.** Let $\mathcal{G} = (V', V'', E)$ be a bipartite graph. A *matching* is a subset of the edges $E' \subseteq E$ such that a vertex is adjacent to at most one edge. The matching is *maximum*, if it is not properly contained in another matching; it is *maximal*, if it is the largest. A matching E' is *perfect*, if $\forall v \in V' \cup V'' : \exists E_v \in E' : v \in E_v$. A *maximal weight matching* is a matching E' in an edge weighted bipartite graph $\mathcal{G} = (V', V'', E, \omega)$ such that $\sum_{e \in E'} \omega(e)$ is maximal.

Thus, perfect matchings are *vertex covers* of the graph, i.e. every vertex is adjacent to an edge. In Fig. 4.2 (b) the example used to illustrate the problem formulation with association graphs is presented using bipartite graphs. Note that mutual exclusiveness of certain associations cannot be modeled in the framework of bipartite graph matching.

Several theoretical and algorithmic results exist on matchings in bipartite graphs, e.g. Hall's marriage theorem about existence of perfect matchings (for example, see Lovasz, 1986) or the Hungarian method for computing maximum weight matches in complete, symmetric bipartite graphs (Kuhn, 1955). Generalizing matching to include n-to-m-correspondences, edges can be introduced which are adjacent to multiple nodes in V' or V'', respectively. This generalizes the graph to a hypergraph. Edges adjacent to a single vertex only can analogously be introduced—they are commonly termed *loops*. Loops allow modeling that a feature is not associated to another feature, while at the same time allowing for maintenance of the notion of perfect matches, as all vertices are covered by an edge. In the context of correspondence determination in robot mapping, loops model loss of visibility.

To conclude, several alternatives exist to approach mathematical modeling of matching problems. Taken the desired generalization into account, bipartite graphs can provide a suitable basis to generalizing matching problems, in particular as the introduction of n-to-m-matches does not increase the set of nodes. Unfortunately, bipartite graphs do not provide inherent means to incorporate side conditions as association graphs do. In summary, bipartite graphs provide a promising start to find a computationally tractable approach to generalized matching.

4.2 Generalizing matching to hypergraphs

Definition 6 (Hypergraph)**.** A graph $\mathcal{G} = (V, E)$ with a finite set of vertices V and set of hyperedges $E \subseteq 2^V$ which is a subset of the powerset of vertices is called hypergraph.

An illustration of a hypergraph is shown in Fig. 4.3 which depicts a first challenge in generalizing matching: there is no sensible interpretation of the attribute bipartite in hypergraphs. Generally, one wants to view edges constituting a matching as edges linking the two distinct parts in a graph. However, in the depicted hypergraph the edge $\{F, G_1, G_2\}$ intended to model a one-to-two correspondence simultaneously links two nodes in the set G. From the definition alone one is unable to deduce the intended partition of the graph's vertices. Balanced hypergraphs (Berge, 1970) have been suggested to generalize bipartite graphs.

Definition 7. A hypergraph $\mathcal{G}$ is balanced if each odd cycle in $\mathcal{G}$ has an edge containing at least three vertices of the cycle. Hereby, a cycle is defined as closed path $v_i \overset{e_i}{\rightarrow} v_{i+1} \overset{e_{i+1}}{\rightarrow} \ldots \overset{e_j}{\rightarrow} v_{j+1} = v_i$, whereby $v_i \in e_i \wedge v_{i+1} \in e_i$. Analogously to bipartite graphs, $\mathcal{G} = (V', V'', E)$ is used as notation, whereby V', V'' model two disjunct sets of vertices to be associated.

Hypergraphs allow for a straightforward generalization of matching.

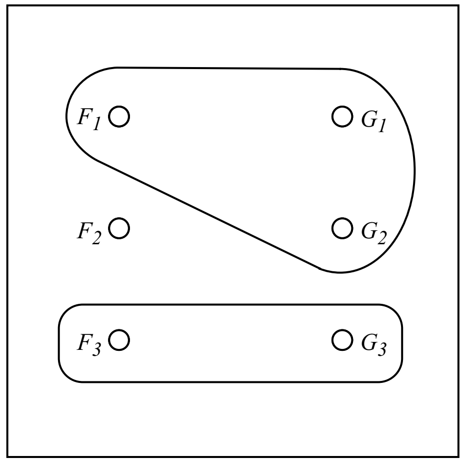


Figure 4.3: Hypergraph modeling a matching that comprises a two-to-one correspondence; edges in hypergraphs are depicted to enclose adjacent vertices.

Definition 8 (Balanced hypergraph matching)**.** Let $\mathcal{G} = (V, E)$ be a balanced hypergraph; then, $M \subseteq E$ is a matching if M consists of pairwise disjoint edges. If M is a vertex cover of $\mathcal{G}$ then M is called *perfect*.

Notably, this definition already includes the case of not-associated vertices by allowing for hyperedges comprising single vertices. Therefore, the generalized matching task can simply be formulated as the maximal weight vertex cover. The formulation using vertex covers ensures that all features are regarded, and it allows for a transformation into an equivalent minimization problem, given that a distance measure is to be minimized, instead of similarity measure to be maximized.

To incorporate side conditions to the matching, I introduce the notion of homomorphic matchings that obey constraints. Since constraints are usually defined for tuples of individual objects, but multiple correspondence partners are allowed in the desired correspondence relation $\sim \subseteq 2^F \times 2^G$, it is useful to first define an elementary correspondence relation $\approx \subseteq F \times G$ with respect to the correspondence relation $\sim$:

$$x \approx y :\Leftrightarrow \exists X, Y \,:\, X \sim Y \wedge x \in X \wedge y \in Y \tag{4.2}$$

This relation maps n-to-m-correspondences to 1-to-1 correspondences and eases the definition of a homomorphic matching.

Definition 9 (Homomorphic matching)**.** Let $\mathcal{G} = (V', V'', E)$ be a balanced hypergraph and $\sim$ a matching. Let further C be an n-ary constraint on V', i.e. a n-ary relation over the vertices. A matching defined by its correspondence relation $\sim$ and induced elementary correspondence relation $\approx$ is homomorphic, if the mapping from V' to V'' (and vice versa) respects the given constraint:

$V' \to V''$ $\forall v'_1, \ldots, v'_n \in V' : C(v'_1, \ldots, v'_n) \Rightarrow \exists v''_1, \ldots, v''_n \in V'' : v'_1 \approx v''_1 \wedge \ldots \wedge v'_n \approx v''_n \wedge C(v''_1, \ldots, v''_n)$

$V'' \to V'$ $\forall v''_1, \ldots, v''_n \in V'' : C(v''_1, \ldots, v''_n) \Rightarrow \exists v'_1, \ldots, v'_n \in V' : v'_1 \approx v''_1 \wedge \ldots \wedge v'_n \approx v''_n \wedge C(v'_1, \ldots, v'_n)$

Homomorphic matching with respect to a set of constraints is defined analogously.

In this definition, existence of some correspondence partners that satisfy the constraint is sufficient; it could hence be regarded as weak homomorphic matching as opposed to a strong homomorphic matching requiring all correspondence partners to satisfy the constraint. However, such distinction is not required in the following. In-depth evaluation of matching in balanced hypergraphs and algorithmic bounds of its computation would exceed the scope of this work by far; I will restrict myself to a special case of homomorphic matching relying on a total ordering relation. As is demonstrated in the following chapters, total ordering is valuable to capture confident information in the robot mapping task. Now, I elaborate on efficient computability of matchings that respect a total ordering of features.

4.3 Algorithmic solution

Since the utilized framework of bipartite graphs—or balanced hypergraphs, respectively—does not include methods for expressing constraints such as the association graph, constraints need to be directly accounted for in the determination of the matching. However, this might not be a deficit. As has been argued by Grimson (1990), constraints directly introduced into matching computation are especially suited to improve computation speed as they already limit the search space in the algorithm's formulation and do not require an active pruning during runtime, as is the case with, e.g. branch and bound techniques. In the following, I demonstrate that the Dynamic Programming (DP) paradigm introduced by Bellman (1957) can be used to tackle the outlined matching task.

4.3.1 Dynamic Programming

In today's understanding the term Dynamic Programming can be misleading since programming refers to a tabular structure employed in the computation, rather than to programming in the sense of informatics.

DP can be applied to problems presenting a structure of overlapping subproblems which form an optimal substructure. In other words, DP can be applied to a problem Π, if Π can be decomposed into subproblems $\Pi_1 \subset \Pi_2 \subset$

$\ldots \subset \Pi_n$ such that Π_i is a subproblem of Π_{i+1} and the solution to Π_i can be extended to a solution to Π_{i+1}. Graph search serves as an example: to determine the shortest path from a to b, the subproblems of determining the shortest path from a to all of b's neighbors can be regarded. The shortest path obtained by linking the paths to a neighbor of b and the edge between the neighbor and a extends the solution of the subproblem to the overall solution.

Lemma 1. Let $F = [F_1, \ldots, F_f]$ and $G = [G_1, \ldots, G_g]$ be two sequences of features and let δ be a feature distance; let further τ denote a penalty measure for not associating a feature. The cost-minimizing assignment of objects $F_i \sim G_j$ that is homomorphic with respect to an order $\prec$, $F_i \prec F_j$, $G_i \prec G_j \Leftrightarrow i < j$, i.e. $(i < i' \wedge F_i \sim G_j \wedge F_{i'} \sim G_{j'}) \rightarrow j < j'$, is of optimal overlapping substructure.

Proof. To show that the problem is of optimal overlapping structure, I introduce the subproblems $\Pi_{i,j}$ which address computation of the cost-minimizing assignments of the sequences $[F_1, \ldots, F_i]$ and $[G_1, \ldots, G_j]$. Thus, $\Pi_{f,g}$ is the original problem. First, observe that $\Pi_{i,1}$ can easily be solved for $i = 1, \ldots, f$ by

$$\min \left\{ \overbrace{\tau(G_1) + \sum_{i'=1}^{i} \tau(F_{i'})}^{\sim = \emptyset} \, , \; \overbrace{\min_{k \in \{1,\ldots,i\}} \left(\delta(F_k, G_1) + \sum_{i' \neq k} \tau(F_{i'}) \right)}^{F_k \sim G_1} \right\}$$

The first term expresses the option of not associating G_1 ($\sim = \emptyset$), whereas the second term expresses the association of G_1 and F_k. Analogously, $\Pi_{1,j}$ can be determined for $j = 1, \ldots, g$. Second, a solution to $\Pi_{i,j}$ for $i, j > 1$ can be reduced to one of the exhaustive cases:

- F_i is associated to G_j; in this case, the problem can be reduced to $\Pi_{i-1,j-1}$
- F_i is not associated to any feature in G, reducing the problem to $\Pi_{i-1,j}$
- F_i is associated to a feature preceding G_j, i.e. G_j is unassociated in the context of $[F_1, \ldots, F_i]$. The problem can be reduced to $\Pi_{i,j-1}$.

The cost-optimal choice provides the optimal solution to $\Pi_{i,j}$ □

Corollary 1. The computational complexity of this assignment problem is $O(f \cdot g)$.

Proof. Follows directly from the definition of subproblems $\Pi_{i,j}$ in the proof of Lemma 1; there are $f \cdot g$ problems which can be reduced to subproblems in constant time by evaluating the three alternatives. □

There are many application of DP, for example the elastic matching of contours in determination of shape distance mentioned earlier.

4.3.2 Matching in bipartite graphs

Theorem 1. Let $\mathcal{G} = (V, V', E, \omega)$ be an edge weighted bipartite graph and let $\prec$ be a total order on V and $\prec'$ on V', respectively. The homomorphic maximum weight match can be computed using Dynamic Programming.

Proof. The proof follows directly from Lemma 1, constructing the sequences F and G according to the order on V and V' respectively; the edge weighting ω defines the distance measure δ to be minimized: $\delta := -\omega$. The penalty τ can be set to $\max \omega(\{v', v''\}) + 1$. □

Corollary 2. Theorem 1 can be extended to include unmatched features by introducing a vertex weight σ that defines a penalty for not matching a vertex. In the DP scheme, an additional step is introduced that determines if skipping a vertex F_i (or G_j respectively) is advantageous when advancing from $\Pi_{i-1,j}$ to $\Pi_{i,j}$ (or from $\Pi_{i,j-1}$ to $\Pi_{i,j}$ respectively).

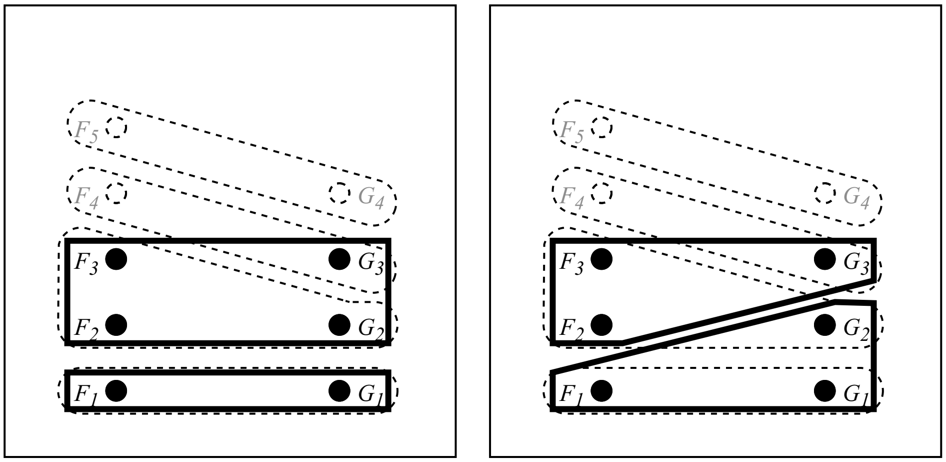
4.3.3 Matching in balanced hypergraphs

By generalizing matching to multiple correspondences, the set of cases to consider grows further as compared to Eq. 3.10. However, with one for practical applications mild assumption and some extra computational effort, the result of Theorem 1 can be adapted to hypergraph matching. In other words, it is possible to derive an algorithm to compute the optimal matching in polynomial time. The remainder of this section describes which assumptions allow to derive a modified DP scheme and prove its correctness.

As a first restriction, I restrict the variety of associable subsets to partitions with respect to a total order of features. This means that only connected subsets are considered, i.e. if $F_i \sim G_j$ and $F_{i'} \sim G_j$, $i < i'$, then $F_k \sim G_j$ holds for every $k = i, i+1, \ldots, i'$. This restriction resembles the observation that the alias problem only introduces neighbored features. Nevertheless, the ability to determine a cost-minimizing matching in polynomial time—with just one further, quite mild assumption—is a remarkable result. Naive, unconstrained matching would involve consideration of all potential correspondences of connected subsets. Note that there are exponentially many pairings of partitions with respect to the number of elements, even if obeying homomorphic associations with respect to order, since there are 2^{n-1} connected subsets for n features[1]. Put differently, naive matching cannot be solved in polynomial time.

The assumption made to squeeze computation to polynomial time is as follows. Let $\Pi_{i,j}$ denote the subproblems introduced in the proof of Lemma 1 and let $\sim^\star$ denote the desired globally optimal matching of the complete problem,

[1]To determine the number of connected subsets, all possible positions for boundaries between subsets are counted. In a sequence of n features there are $n-1$ possible boundary positions. Each of the $n-1$ potential boundaries can independently split the overall sequence.



(a) (b)

Figure 4.4: Illustration to the assumption of local optimality. Exemplary subproblem $\Pi_{3,3}$ of the overall problem $\Pi_{5,4}$; the globally optimal correspondence is depicted by dashed outlines. The assumption of local optimality is fulfilled in (a) since the second preceding, fully contained correspondence, $F_1 \sim G_1$ is contained in the solution of $\Pi_{3,3}$. (b) Example of a violation of the assumption, $F_1 \not\sim G_1$.

i.e. $\Pi_{f,g}$. The matching $\sim^\star$ consists of a sequence of individual assignments (hyperedges) $\sim^\star = [\sim_1^\star, \sim_2^\star, \ldots, \sim_k^\star]$. This sequence is ordered according to a partial order induced by features' order: $\sim_l^\star \leq \sim_k^\star :\Leftrightarrow \exists \{F_i, G_j\} \subseteq \sim_l^\star : \forall \{F_{i'}, G_{j'}\} \subseteq \sim_k^\star : F_i \prec F_{i'} \wedge G_j \prec G_{j'}$. Thus, $[\sim_1^\star, \ldots, \sim_k^\star]$ is a serialization of the partial order on $\sim^\star$ induced by the total order $\prec$ of features.

Put differently, correspondences are ordered as induced by the indices of features i, j in the sets $\{F_1, \ldots, F_f\}$ and $\{G_1, \ldots, G_g\}$. Then, I assume that the second maximum correspondence of $\sim^\star$ fully contained in the subproblem $\Pi_{i,j}$ must also be optimal with respect to $\Pi_{i,j}$—see Fig. 4.4. I refer to this assumption as *local optimality.*

In the context of the correspondence problem in robotics, local optimality can be rephrased as follows: given a set of features to associate, the cost-optimal association of features must not only be cost-optimal with respect to the complete set of features to be associated, but to a local context as well. The formulation using the local context of complete matches is considerably less restrictive as would be required for application of greedy matching strategies (the association would be required to be optimal in *any* context (Lovasz, 1986)). Local optimality appears to be a reasonable assumption in practical application: recognizing an object should not be affected by potential associations of remote objects.

Theorem 2. Let $\mathcal{G} = (V, V', E, \omega)$ be an edge weighted balanced hypergraph. If ω is *locally optimal*, then the homomorphic maximum weight matching w.r.t. $\prec$ can be computed using an extended DP scheme.

Proof. The idea underlying the proof is to inductively show that a local optimal correspondence can be propagated while starting at subproblem $\Pi_{1,1}$ and advancing towards the overall solution. Let $\sim^\star$ denote the globally optimal solution as regards $\Pi_{f,g}$ and let $[\sim_1^\star, \sim_2^\star, \ldots, \sim_k^\star]$ denote an ordered sequence of associations with respect to the order induced by the features. Since $\sim^\star$ is a vertex cover, it is non-empty. Let $\Pi_{[\sim_i^\star]}$ denote the minimal subproblem which contains all features associated in $\sim_i^\star$.

Basis $\Pi_{[\sim_1^\star]} =: \Pi_{i,j}$ can be computed by regarding all $i' < i$ and $j' < j$ to determine the optimal association $\{F_{i'}, \ldots, F_i\} \sim \{G_{j'}, \ldots, G_j\}$ as regards the edge weighting ω. This association is optimal with respect to $\Pi_{[\sim_1^\star]}$ as well as to $\Pi_{f,g}$.

Inductive step To solve $\Pi_{[\sim_{k+1}^\star]} =: \Pi_{i,j}$, all $i' < i$ and $j' < j$ need to be regarded in a similar manner. Determine i' and j' such that the association $\{F_{i'}, \ldots, F_i\} \sim \{G_{j'}, \ldots, G_j\}$ as regards the edge weighting ω is optimal in conjunction with the overall edge weighting of the association determined by $\Pi_{i'-1,j'-1}$. According to the assumption of local optimality, $\Pi_{i-1,j-1} = \Pi_{[\sim_k^\star}$ is optimal in $\Pi_{i,j}$ and $\Pi_{f,g}$. Hence, the association $\{F_{i'}, \ldots, F_i\} \sim \{G_{j'}, \ldots, G_j\}$ is optimal as regards $\Pi_{i,j}$ as well as $\Pi_{f,g}$.

Observe, that it is required to have knowledge about the subproblems $\Pi_{i,j}$, $\Pi_{[\sim_i^\star]}$ refers to. It is sufficient to determine a solution to all $\Pi_{i,j}$ using the single method used in the basis and inductive step. □

Corollary 3. The homomorphic maximum weight match in Theorem 2 can be computed in $O(f^2 \cdot g^2)$.

Proof. From the proof it follows that in each step of the computation the best association of the topmost i' features of sequence F and topmost j' features of sequence G need to be regarded; i' can be assessed with an upper bound of f and j' with g respectively. □

Observe that the Theorem provides an upper bound assessment—it remains an open question, whether there exist more efficient means of computation.

4.3.4 Generalizing the matching

It is possible to relax the requirement of local optimality and thereby gain a more general formulation of Theorem 2—at the cost of extra computational complexity, though. The matching as described above is already appropriate for correspondence determination in my approach to robot mapping. Therefore, the generalization is only discussed briefly here. In-depth investigation of the generalized matching is an interesting starting point for further research, though.

To generalize the matching, the requirement of local optimality is relaxed. In the given definition of local optimality, the second maximum correspondence that is fully contained in a subproblem must be optimal to both the subproblem and the global problem—here, I abbreviate this condition as LO_2. A relaxed condition LO_3 can be defined analogously to LO_2 as optimality in the context of the *third* maximum correspondence that is fully contained in a subproblem, i.e. the local context is enlarged in which local optimality entails global optimality of a correspondence. This definition is relaxed, since any edge weighting that meets the condition LO_2 meets LO_3 too. Matching on the basis of an edge weighting that meets the LO_3 condition can be performed similar to matching with an edge weighting conforming to LO_2, but instead of determining the single maximum correspondence when advancing the subproblems $\Pi_{i,j}$ (see proof for Theorem 2) the two maximum correspondences may need to be determined simultaneously. When solving $\Pi_{i,j}$ this requires evaluating an additional case of establishing two correspondences by selecting variables i', i'' with $1 \leq i' < i'' \leq i$ and j', j'' with $1 \leq j' < j'' \leq j$ such that the three components $F_{i'':i} \sim G_{j'':j}$ (maximum correspondence), $F_{i':i''-1} \sim G_{j':j''-1}$ (second maximum correspondence), and the solution[2] to $\Pi_{i'-1,j'-1}$are cost-optimal with respect to the edge-weighting. Thus, solving the subproblem $\Pi_{i,j}$ requires an extra effort of $i \cdot (i-1) \cdot j \cdot (j-1)$ alternatives to evaluate (as compared to Theorem 2). This results in an overall complexity of $O(f^3 \cdot g^3)$, since the extra effort is necessary for each of the $i \cdot j$ subproblems. The exemplified generalization from LO_2 to LO_3 suggests that these steps can be carried out for analogous definitions of LO_4, $LO_5, \ldots$ as well and a general theorem can be shown that the matching problem for an edge weighting that meets the LO_i condition can be solved in $O(f^i \cdot g^i)$ time.

4.4 Summary & conclusion

In this Chapter, I have investigated mathematical frameworks for matching tasks and devised an adequate theoretical foundation of matching for correspondence determination. Graph-theory provides appropriate means to formulate the matching task. I have proposed a new mathematical formulation of the correspondence problem based on balanced hypergraphs, a generalization of bipartite graphs. My formulation generalizes matching in bipartite graphs to hypergraphs. It augments existent approaches to correspondence determination

[2]For simplicity, I assume $\Pi_{j,k}$ to be defined for negative j, k as well. The subproblem is empty then, i.e. its solution is the empty correspondence relation.

in robot mapping with the capability to express n-to-m correspondences of features, i.e. joint associations of multiple features. This extension is particularly suited to address different results of the grouping in the feature extraction (alias problem). Advancing matching techniques to multiple feature associations is one important step for grounding robot mapping on extended features, as detection of extended features is subject to grouping differences. I have proposed the problem formulation of homomorphic matchings which combine qualitative and quantitative knowledge in the matching. A quantitative measure is employed in terms of an edge weighting to evaluate the plausibility of individual feature correspondences and to allow for differentiating between alternative solutions. Qualitative knowledge is employed as constraints that model side conditions, i.e. confident knowledge about feature configuration that needs to be respected by the matching.

In general, correspondence determination considering n-to-m correspondences comprises a search space of non-polynomial size, but (qualitative) knowledge about ordering allows for an efficient approach. I have developed a new algorithm for determining a cost-optimal matching that is homomorphic with respect to an ordering of features. The developed algorithm is an extended Dynamic Programming scheme and it efficiently computes the cost-optimal matching in $O(f^2 \cdot g^2)$ time where f, g denote the number of features to be associated.

The algorithm proposed in this Chapter presents two characteristics that need to be evaluated with respect to the concrete task of matching configurations of polylines. First, n-to-m-correspondences are restricted to connected sets with respect to the order of features. Connected subsets are suitable to address the alias problem, given that an appropriate ordering is applied, as individual fragments are neighbored. However, connected subsets cannot model all facets of perceiving a single physical entity as multiple features (see Section 7.3.4). For example, visibility of an object partially intercepted due to occlusion, results in an unconnected set of features: one fragment to the left of the interceptor, one fragment to the right of it. To include such aspect in the matching, correspondence of arbitrary subsets of the set of features would need to be examined. However, this requires an extension of the matching framework that increases the computational cost as well, most likely beyond polynomial time. Thus, the developed technique is a compromise between addressing the full scope of potential correspondences of extended objects and computational feasibility.

The second characteristic is the an edge weighting is required that meets the condition of local optimality. Local optimality of a measure means to correctly determine a globally optimal association in a restricted context. Local optimality can be regarded an fulfillable requirement: identification of feature associations should not be affected by potential associations of remote features.

However, in order to meet the requirement a decisive and robust feature similarity measure must be employed.

Chapter 5

Shape-based incremental mapping

Bring a quarter-liter Riesling to boil and reduce to half; add 200g of cream and let it reduce further. Add 200g peeled, diced tomatoes and cook for 10 minutes. Add 100g cooked shrimps and freshly chopped dill; let it heat up. Season to taste with salt, pepper, nutmeg, sugar, and lemon. Serve with spaghetti.

In this Chapter, I detail my approach to robot mapping. At a glance, I present an incremental mapping based on shape features extracted from range finder data. Being a SHape-based approach to Robot Mapping, my approach is named SHRIMP and its prototypical implementation is named SHRIMPS, short for shape-based robot mapping system.[1] First, I develop my characterization of plausible mapping data integration and explain how this can be implemented in terms of computational modeling. Thereafter, SHRIMP and its functional components are presented in detail.

5.1 Characteristics of plausible mapping

The central component in my approach is correspondence determination. Characterizing plausibility as regards correspondence determination characterizes mapping as a whole. To start with, I give an overview of my requirements to plausibility in correspondence determination:

1. Only similar features may be matched.

[1]In the era of iPods, iThis, and iThats one should be allowed to talk about incremental robot mapping in terms of robot i-mapping as well and, by doing so, finding a vowel at the right place to add some melodiousness to an otherwise unpronounceable acronym.

2. Grouping differences must be addressed.

3. Confident knowledge about feature configuration must be obeyed.

4. Feature associations must be mutually compatible with respect to the mapping of configurations induced.

In the following, I explain these requirements in more detail and discuss, how they can be realized in terms of computational modeling.

Restrict matching to mutually similar features Feature distance measures support correspondence determination by quantifying a feature similarity that can be interpreted as likelihood of feature correspondence. From reviewing spatial representations, I concluded to employ polylines as features; a shape distance measure compares individual polylines. Feature appearance can vary due to a multitude of factors, some of which cannot be anticipated, e.g. measurement noise. Under these conditions, feature similarity is subject to undetermined change, even if the features compared relate to the same physical object. Therefore, I suggest implementing consideration of feature similarity in terms of an optimization: the aim is to minimize the summed up feature distances of all features associated. If feature distance between two features exceeds an upper limit, matching of these features is inhibited, i.e. unsimilar features may not be matched. The outlined modeling reflects the claim in my thesis that shape information is valuable to improve robustness of correspondence determination and to disregard odometry (cf. Section 1.4).

Addressing grouping differences Extended objects may be only partially visible, for example due to occlusion. In some situations, this may lead to detection of multiple features, if middle parts remain hidden. Due to the alias problem in grouping, extended objects may be interpreted to multiple features, even though they are visible in their full extent. These effects need to be addressed. Eclipsing explicit consideration of partial visibility, e.g. by interpreting perceived line segments as infinite lines (cf. Section 2.1.3 and Section 3.4.3), introduces artifacts, i.e. counter-intuitive results. Ignoring the alias effect would discard valuable information, as individual feature fragments may be uninformative on their own, but provide rich information, if interpreted as a whole. This results in the necessity to handle multiple features potentially corresponding to a single feature, or even to handle two sets of corresponding features. If possible, multiple correspondences should be accounted for as early as in the matching. The developed theory of hypergraph matching provides a suitable tool for this task.

Respecting confident knowledge Discovering confident knowledge and respecting confident knowledge provides a way to improve on matching plausibility and efficiency at the same time. The difficulty is to discover confident knowledge in robot mapping, as this domain is characterized by inherently uncertain information. In the context of extended geometric primitives like polylines, one characteristic property of a local surrounding is the circular order of visibility, in which individual features are arranged. I regard the circular order of polylines as confident knowledge for recognizing a specific place. Extended features maintain their circular sequence in observations, if moderately varying the view pose. If the view pose is continuously changed, two features that are neighbored in the sequence of circular order do not interchange positions before one feature (partially) occludes the other. In the case of total occlusion, viewpoint variation leads to the disappearance of one feature. In the case of partial occlusion, viewpoint variation leads to a splitting of one feature; this single feature is no longer detected, but two distinctive features are observable instead: one before the feature blocking sight, and one after it. These changes can be interpreted in terms of conceptual neighborhoods (see Section 2.3): disappearance or appearance of features marks a transition to a conceptually neighbored configuration with respect to circular order. Hence, configurations that differ by two features interchanging positions are not conceptually neighbored. Put differently, obeying circular order in correspondence determination limits to relating configurations that are either isomorphic or conceptually neighbored. In terms of homomorphic matching (cf. Section 4.2 and Definition 9, respectively) this means to determine a matching that is homomorphic with respect to the circular order.

Ordering plays a double role in my approach. First, it delimits acceptable deviations between configurations to relate. In this way, it overcomes problems of unlikely associations discussed in Section 2.3.4. Second, by making unacceptable configurations explicit in terms of hard constraints, the search space is trimmed and the matching techniques derived in Section 4.3 are applicable. Technically speaking, correspondence determination can be formulated as a combinatorial optimization problem using balanced hypergraphs.

In respecting confident knowledge, I restrict myself to confident information about configuration—there may be alternative sources of confident information valuable to exploit, though. Exploring further alternatives is beyond the scope of my work; it could serve as starting point for future research.

Ensuring mutual compatibility of feature correspondences Any single feature correspondence of observed feature and map influences the overall alignment, i.e. the mapping of the local frame of reference of observation to the absolute frame of reference of the map. Metric positional information allows for a fine-grained, gradual assessment of compatibility in feature association.

My argumentation for jointly compatible feature assignments conforms to the discussion by Neira & Tardós (2001) (see Section 3.5). Positional information can provide valuable information to disambiguate potentially corresponding features. Since metric information of positions is subject to uncertainty, it is most adequate to address mutual compatibility in terms of an optimization: determine the assignment of two sets of features that minimizes the summed up deviations from true compatibility. Association of features is inhibited, if the induced deviation from true compatibility is too large.

As regards computational modeling, addressing mutual compatibility of induced alignment is similar to addressing feature similarity; in both cases, handling of uncertain, fine-grained metric information is tackled by optimization. Both goals are independent, though. A different feature association may be optimal with respect to feature similarity, as is optimal with respect to compatibility in alignments. This requires mediating between both goals, which can be done by weighting the two goals and computing the jointly optimal solution. In my approach, I utilize a weighting which favors consideration of feature similarity over alignment.

To sum up, I detailed characteristics for correspondence determination and formulated four individual goals. My characterization of plausibility is arranged at an abstract level of spatial reasoning. The presented goals contrast to purely stochastic mapping architectures which "only require" specification of probability models for involved components like sensors. Computational characteristics in stochastic approaches are derived from probabilistic reasoning. In this way, plausibility of matching is derived along a deep probabilistic reasoning chain from sensor models to the map update processes, thereby potentially propagating shortcomings in the initial models as well. I objected to this approach in Section 3.1, since I do not fully agree to the underlying probabilistic modeling in first place. Therefore, the present characterization of plausibility is anchored at a more abstract level. My demands extend existent approaches in that I introduce feature-intrinsic similarity for geometric primitives and focus on it. Explicit incorporation of hard constraints and implications of extended features, namely alias effect and partial feature visibility, have to my knowledge not been addressed before explicitly. However, my list of demands is by no means exhaustive; it expresses key demands I derive and which are realized in my computational model. I address some potential extensions and implied research questions in the outlook of this thesis.

In the following, I present my computational model to robot mapping which reflects my characterization of plausibility. Starting with an overview, I then descend to an in-depth description of the individual components of SHRIMP.

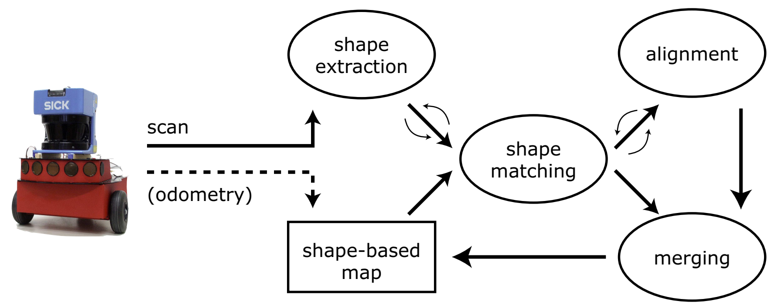


Figure 5.1: Architectural overview of the SHRIMP architecture

5.2 Overview of SHRIMP

SHRIMP's architecture interrelates the functional components view acquisition, correspondence determination, alignment, and merging. They constitute an incremental mapping architecture that is is based on a shape-based map representation. A diagram of the mapping architecture is depicted in Fig. 5.1, and in the following, I present the interconnection of the individual components in some detail, before descending to the in-depth description of the individual components.

Regarding the architecture from the perspective of information flow, the first building block—besides the sensors—is view acquisition by means of shape extraction. Range information is retrieved from the LRF and is transferred into a shape representation that is based on polylines. Sensor data is grouped to individual polylines, and compacted by a curve evolution. Shape extraction does not rely on fixed thresholds, but it adapts in response to shape information present in the map. During the matching, when shape information extracted from sensor data is related to the map, the extraction is finalized, determining a configuration of polylines that agrees to both sensor data and map. So, shape extraction operates in a data- and model-driven manner. In the diagram, this adaption is indicated as circular arcs along shape extraction and matching.

Simultaneous to extracting shape information from sensor data, shape information is also retrieved from the map. The map registers polylines as features, so that shape information can directly be retrieved without shape extraction. View extraction from the map aims at retrieving a view, i.e. a configuration of polylines that significantly overlaps with the observation of the robot. This is achieved by retrieving the view according to an estimated pose of the robot. Since no full congruency of map view and observation is required, two ways to obtain an estimate of the robot pose present themselves: first, the robot pose that has been determined by the most recent localization of the robot can be used. If the robot has not traveled so far that the view on its surroundings substantially changes (e.g. less than 1 meter), this option is applicable. Second, if odometry information is available, it can be exploited to derive an estimate of the robot's pose. Odometers measure the movement of a robot, i.e. the pose of the robot is determined in relation to a reference pose. Relating odometry information to the pose determined by the previous localization, the desired pose estimate is obtained. One asset of SHRIMP is that precise pose estimates are not required. Matching will be successfully performed if a significant overlap of observation and the map view exists.

Matching observation and map is the central component in SHRIMP, which is characterized by two aspects: feature similarity in terms of a shape distance measure and a matching algorithm based on Theorem 2. The shape distance measure I developed is especially tailored to the domain of identifying polylines that are extracted from range data. The measure evaluates the agreement of perceived shape information and shape information represented in the map by determining the most agreeable interpretation of the sensor data. This results in a noise-sensitive shape extraction that is particularly valuable in situations where the amount of noise is challenging with the overall amount of shape information. In the matching algorithm distinct polylines in observation and or map can be joined, thereby adapting feature extraction and map information. In other words, shape extraction is finalized in the context of computing the shape distance of corresponding polylines and matching observation and map. This yields a parameter-free approach to feature extraction.

Besides considering plausibility of matching in the context of alignment, joint compatibility of feature associations is evaluated. Circular arrows in the diagram Fig. 5.1 indicate that alignment is evaluated in the matching to ensure a mutually agreeable alignment of corresponding features. This linkage of matching and alignment implements the demand for jointly compatible associations.

Based on the matching that determines a correspondence on the level of individual polylines, perceived shape information is aligned with the map. Alignment induces a two-way mapping between the robot's local coordinate system to the global coordinate system of the map. This localizes the robot.

After the alignment has been determined, the map can be updated. Observed features that correspond to map features are integrated into one feature representation. Features that are not matched to the map are interpreted as newly emerged featured; they are added to the map.

The rest of this Chapter presents an in-depth description of the components.

5.3 Map representation

SHRIMP is based on a uniform object map registering features in a single, absolute coordinate system. On the feature representation layer, polylines are employed to represent the boundary of navigable space, or, equally, the outline of obstacle boundaries. Polylines are universal geometric primitives representing boundary information in an extended context such as the outline of an entire obstacle. Ideally, the complete outline of an obstacle is represented by single polyline.

Configuration information is represented metrically, the vertices of polylines are anchored in the map's global coordinate system, which is once initialized at startup time, i.e. the robot is interpreted to start from the origin of the coordinate system. When required, qualitative information can also be retrieved from such fine-grained metric information.

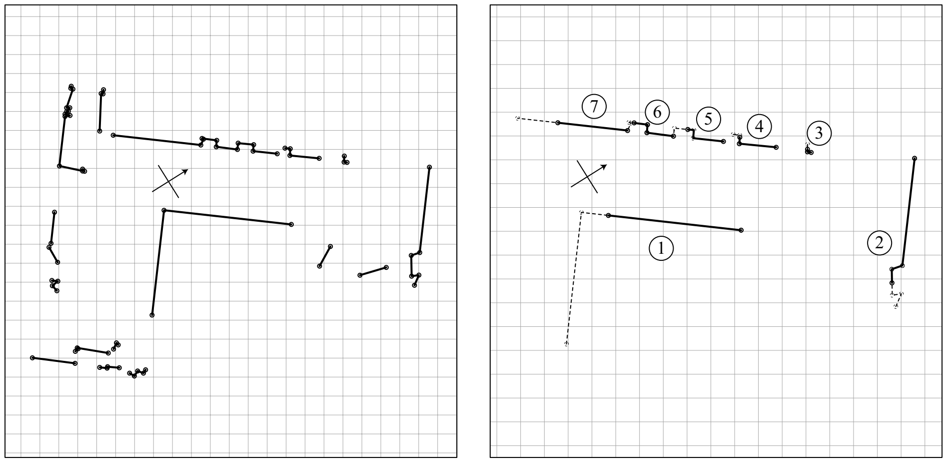
For retrieving the view from the map that associated to a pose ξ, a radial sweep line algorithm can be applied. This technique has been developed in the field of computational geometry and yields a computational complexity of $O(n \log n)$, where n denotes the total number of vertices stored in the map (de Berg et al., 2000). The retrieval process (see Fig. 5.2 for an example) computes the visible fragments of the polylines stored in the map and places them in the a local coordinate system, using the view pose ξ as origin. Extracted polylines maintain a link to the original map features that are required in the alignment and merging process (cf. Section 5.7 and Section 5.8). For example, these links allow one to determine if multiple visible fragments emanate from the same map object. Simultaneously to the local metric information about polylines, circular order of polylines is determined, which is regarded as confident information in correspondence determination.

In summary, map retrieval computes a sequence of polylines. The polylines are assumed visible according to the map and a pose estimate ξ; they are ordered in a circular, counter-clockwise manner.

5.4 Extracting polylines from range finder data

Sensor data is interpreted to features by extracting polylines from the sensor data. The polylines are represented in a local coordinate system, and their circular order is determined. Laser range finders scan the surrounding in a circular manner by letting the laser beam sense the obstacles from right to left, i.e. counterclockwise. So, the sequence of range measurements already provides the circular order.

In a preparation step, range values are mapped to points in the Euclidean plane. Shape extraction comprises two separate steps: grouping to individual polylines, and consolidating shape information represented by the polylines. Feature extraction adapts to map information and is finalized in the matching, making this process independent of parameters.



(a) (b)

Figure 5.2: (a) Exemplary map representation in SHRIMPS comprising 16 polylines. (b) View retrieved from the map according to the view pose marked in (a); 7 polylines are determined, they correspond to 5 polylines in the map (denoted by dashed lines). Numbers denote the sequence order of circular visibility.

5.4.1 Grouping to polylines

Grouping is performed to assign measured points to individual features. Like any grouping, it is a heuristic process, since the true assignment of points to real world objects is unknown (cf. Section 3.4.3). The rule employed in SHRIMP assumes that neighbored points belong to the same object, if their distance does not exceed a threshold. Analogously, an object transition is assumed present wherever two consecutive points are farer apart than specified by the threshold. By sequentially processing the data, grouping is performed in linear time with respect to the number of points provided by the sensor.

To select the grouping threshold, two effects need to be balanced: if the threshold is decreased, more object transitions are assumed. This intensifies the effect of the alias problem, i.e. the problem of multiple registrations of the same physical entity. Different outcomes of the grouping process are handled in the matching by adequately linking distinct features, since any crisp grouping—independent of a potential choice of parameters—can yield different outcomes depending on the view pose or noise superimposing on the data. However, the more individual features are detected in an observation, the more computational resources are required in the matching, possibly slowing down the mapping system to a degree unsuitable to online application.

On the contrary, if the threshold is increased, object transitions are more likely to be ignored for separate objects located close to one another. Joining boundaries across gaps can introduce severe complications as has been discussed in Section 3.4.3. Overlooking object transitions has significantly worse effect than surplus detections of object transitions.

So, the threshold is chosen to reliably detect object transitions. In experiments a threshold of up to 10cm seems adequate for typical indoor environments. If using a LRF with an angular resolution of 0.5°, a threshold of 10cm will group range measurements corresponding to a wall opposing the robot in a distance of as far as about 10 meter into a single feature.

5.4.2 Consolidating shape information by curve evolution

Shape consolidation describes the process of reducing the bare amount of data stored for polylines, while maintaining their visual appearance, which is examined in correspondence determination. Curve evolution can filter out noise present in sensor data that superimposes on relevant shape information. In contrast to approaches aiming at optimally fitting of a model to the data, SHRIMP pursues an adaptive approach based on a discrete curve evolution process. This avoids the difficult definition of an optimality criterion in model fitting (cf. Section 3.4.3).

In SHRIMP, the Discrete Curve Evolution (DCE) developed by Latecki & Lakämper[2] is employed. Applicability of DCE has already been demonstrated in applications in computer vision (e.g. Latecki & Lakämper, 2000; Latecki & Rosenfeld, 2002) and appears well-suited to interpreting range information (cf. Section 3.4.2; Section 3.4.3). DCE describes a context-sensitive process of evolving a polyline by vertex removal. It proceeds in a straightforward manner by removing the vertex with least *relevance* until the lowest relevance value remaining exceeds a chosen stop threshold. To compute the relevance of a vertex v_i as regards visual appearance of the complete polyline, DCE employs a local measure. Let v_{i-1} and v_{i+1} refer to vertices adjacent to v_i, and let $\measuredangle$ denote the angular difference of line segments, then the measure reads as follows:[3]

$$K(v_{i-1}, v_i, v_{i+1}) := \frac{\measuredangle(\overline{v_{i-1}v_i}, \overline{v_i v_{i+1}})\, d(v_{i-1}, v_1)\, d(v_i, v_{i+1})}{d(v_{i-1}, v_i) + d(v_i, v_{i+1})} \tag{5.1}$$

The relevance measure K grows monotonously with respect to the overhang of v_i over $\overline{v_{i-1}v_{i+1}}$ as well as with respect to the length of the segments adjacent to v_i. The evolution process is stopped, when all remaining vertices present a

[2]Latecki & Lakämper (2000); Latecki et al. (2000a)

[3]Earlier (Wolter & Latecki, 2004; Wolter et al., 2004), I utilized a computationally less expensive relevance measure $K'(v_i) := d(v_{i-1}, v_i) + d(v_i, v_{i+1}) - d(v_{i-1}, v_{i+1})$. The measure avoids computation of angles and performs comparatively well in environments presenting mostly polygonal contours, but it appears more difficult to determine a suitable intermediate threshold (see Fig. 5.5). Therefore, I returned to the original DCE measure.

relevance value that exceeds a pre-defined threshold. Due to the changing vertex neighborhoods during the removal process, the spatial context considered by the relevance measure grows.

No fixed stop criterion is utilized in SHRIMP. Rather, the process continues in negotiation with shape similarity (cf. Section 5.5). The reason to delay the shape extraction until correspondence determination is given by the overall small amount of shape information perceivable in some environments compared to the amount of noise. When removing anything than *can* be caused by noise, much valuable shape information would also be removed. DCE makes vertex removal decisions in the context of a single polyline. A better noise identification can be made in the context of comparing corresponding polylines. Shape properties caused by noise are less likely to receive support from an independent polyline representing the same physical entity.

SHRIMP utilizes an intermediate threshold that halts DCE well before reaching the final end. This allows benefiting from the small computational cost of solely performing DCE. In the experiments an intermediate stop threshold of 5.0 has proven adequate for an unit size of one centimeter. The concrete choice is not critical. The relevance of the least relevant vertex in a polyline grows increasingly fast, the more vertices are removed (see Fig. 5.5). This reduces the effect variations of the intermediate threshold cause.

Implementation of DCE can benefit from the observation that polylines can simultaneously be represented as double-linked lists and as self-balancing trees ordered by vertex relevance. Setting up this data structure and computing the initial relevance measures for n vertices can be done in $O(n \cdot \log n)$ time. Each simplification step consists of identifying the least relevant vertex, removing it, and updating its neighboring vertices' relevance measures. This can be performed in $O(\log n)$ time. Since there are at most n vertices to remove, the overall time complexity for DCE is $O(n \cdot \log n)$.

Shape extraction by DCE describes polylines using vertices already contained in the sensor data. In environments known to only display polygonal surfaces a more accurate approximation is possible by fitting line segments to known groups of scan points. In such cases, polylines obtained through DCE can be adapted by applying linear regression to the original scan points constituting the individual segments of the polyline. This approach improves on pure line fitting (e.g. relying on the recursive split algorithm—see Section 3.4.1), as the comprehensive spatial context of a complete boundary is regarded to determine the grouping into line segments. It allows robust detection of short line segments.

Exemplary results of the shape extraction process are shown in Fig. 5.3–Fig. 5.6. Fig. 5.3 picks up the example of Fig. 2.3 on page 47 which demonstrated that line segments are not adequate to represent arbitrary indoor environments, but polylines are indeed. As can be observed, shape extraction is well-suited

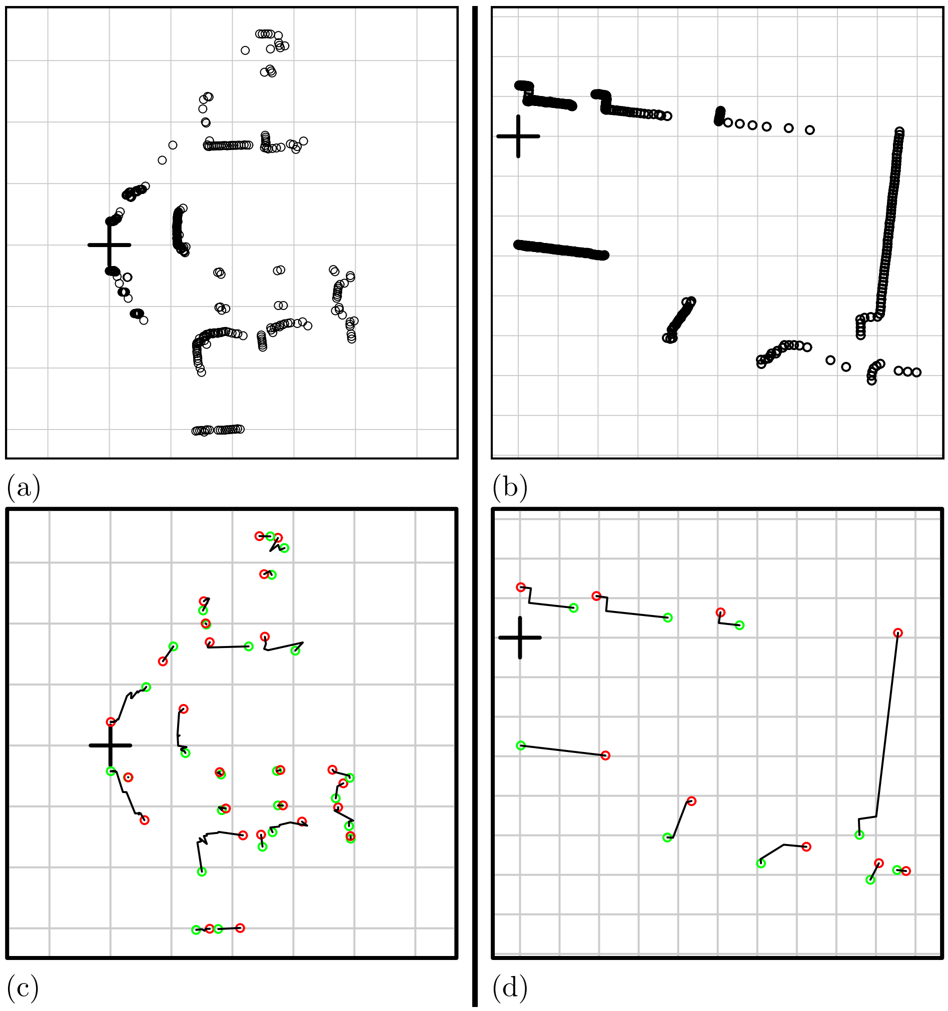


Figure 5.3: Exemplary results of shape extraction. (a), (b) depict two exemplary scans obtained in indoor environments and (c), (d) show polylines extracted from them, using a grouping threshold of 10 cm and a DCE threshold of 5. Compare to Fig. 2.3 on page 47. Grid lines denote 1 meter distance.

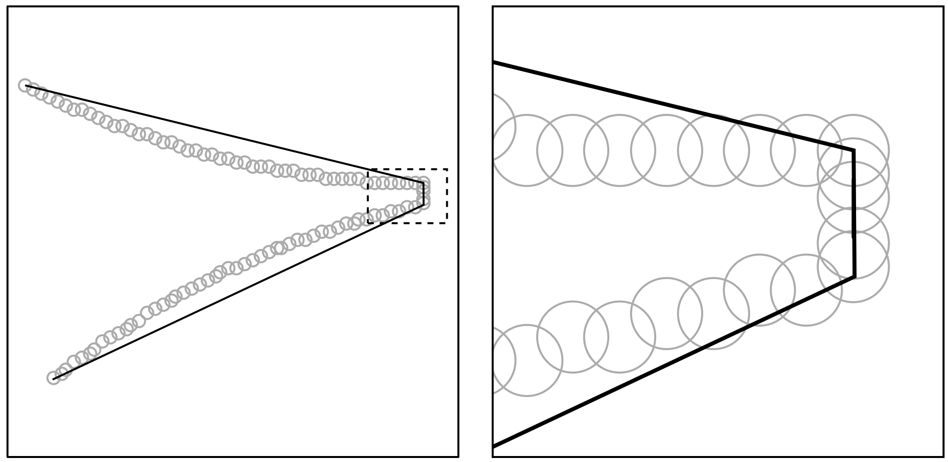


Figure 5.4: Shape extraction by means of DCE (solid line) using points on a flare-shaped contour as input data (circles); self-intersections do not occur (b) presents an enlargement of the area marked in (a). Compare with the discussion related to Fig. 3.3 on page 90.

to both examples. The example illustrated in Fig. 5.4 demonstrates that polyline extraction by grouping and consecutive DCE avoids the risk of topological violations faced in individually fitting line segments (cf. Fig. 3.3 on page 90). Adequacy of DCE-based polyline extraction for representing curved contours is presented in Fig. 5.5. In Fig. 5.6, an exemplary development of the grouping and DCE process using different intermediate stop thresholds is shown. These examples clearly indicate that DCE-based polyline extraction is also applicable to range data. In a context of interpreting trajectories, an extensive review of generalization techniques can be found in Stein (2003). His exemplary comparison of different relevance measures also indicates that the relevance measure of Eq. 5.1 is well-suited to consolidate shape information stemming from non-polygonal contours (see Stein (2003, p. 204)).

5.5 Shape similarity

Shape similarity operates on shape features that are either extracted from perception or represented in the map. Shape similarity consideration provides a decisive input to the central matching algorithm. More precisely, a shape distance measure defines a feature distance measure that is interpreted as plausibility measure for correspondence of features.

To briefly recap the discussion on suitable shape distance measures (cf. Section 3.5.2), measures need to be decisive even for comparatively simple shapes, and they need to account for uncertainty in the shape information. In the

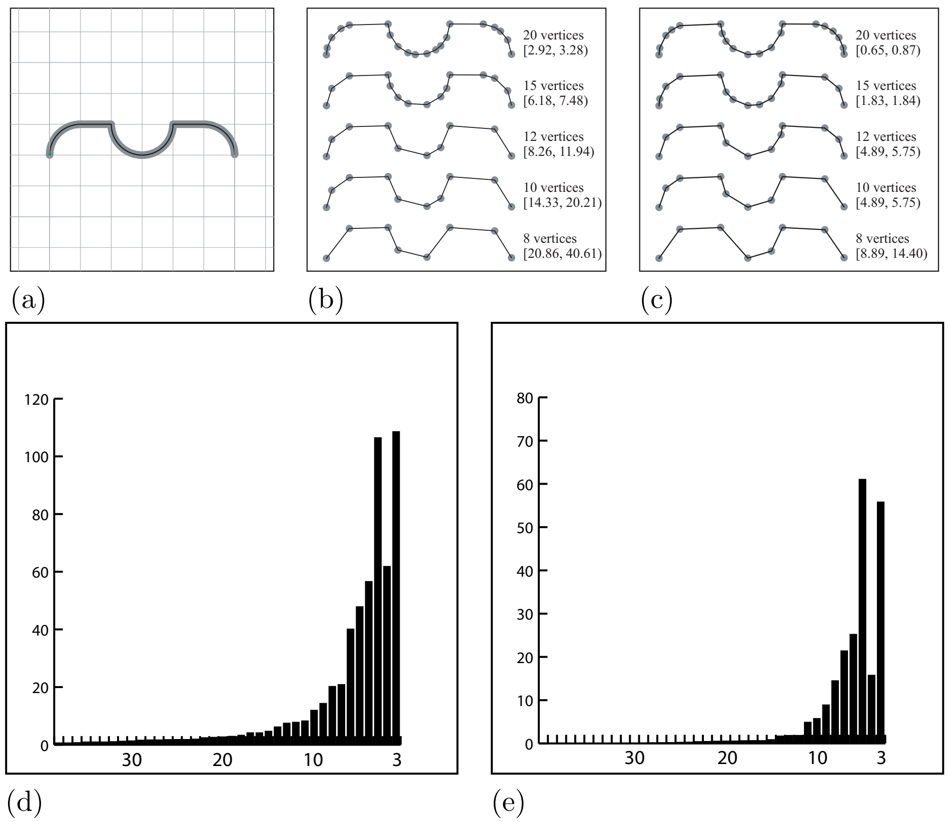


Figure 5.5: Applicability of DCE to curved contours. (a) Original synthetic data. (b) Some DCE stages using the original measure. The interval of relevance thresholds is denoted, which yields the depicted polylines. (c) The plot analogous to (b), but using the distance-based relevance measure. The intervals are smaller than in (b), making choice of the threshold more difficult. (d) Relevance of the least relevant vertex against number of remaining vertices for original DCE measure. (e) The plot analogous to (d), but using the purely distance-based relevance measure, the decreased differentiation can be observed.

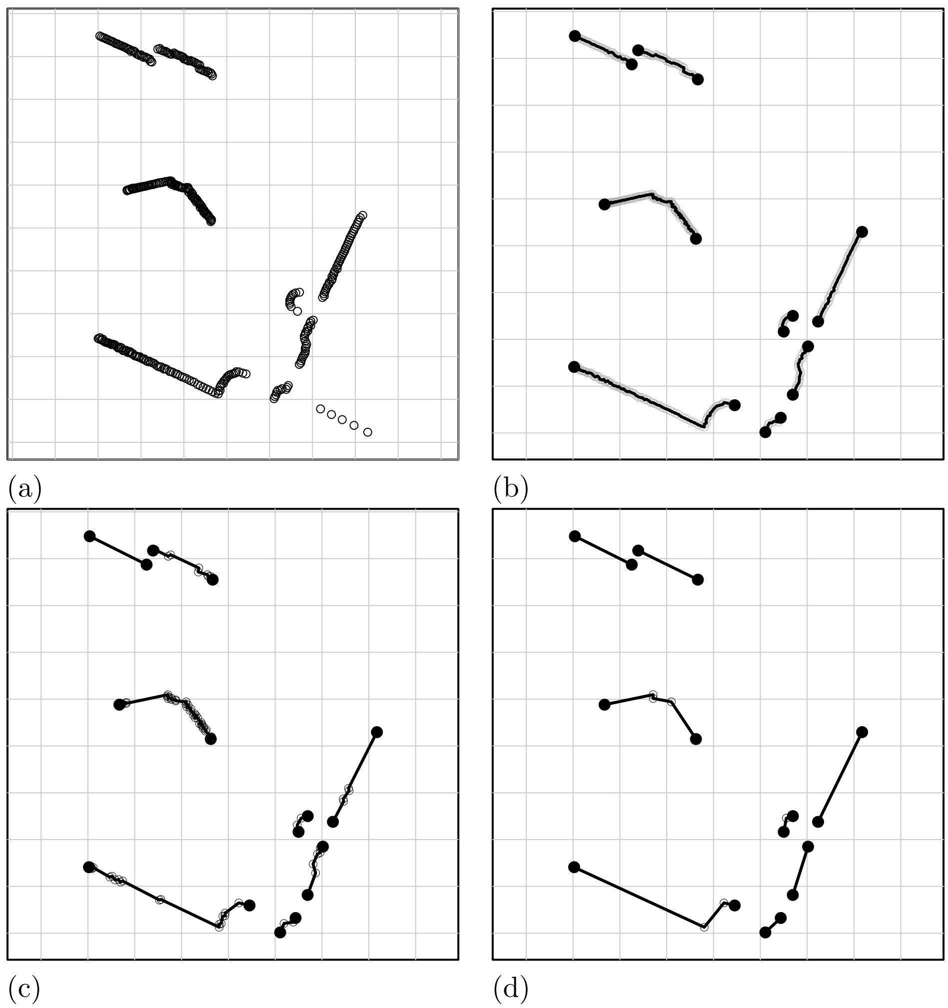


Figure 5.6: Extracting polylines from a scan. (a) Raw scan points in local coordinates. (b) Grouping to individual poylines (start- and endpoints are highlighted). (c), (d) Shape simplification using DCE. The threshold used to obtain (c) is 1.0 (resulting in 88 points) and 5.0 in (d) (resulting in 22 points). Grid lines denote 1 meter distance.

context of range information the level of noise may nearly conceal the overall amount of shape information present in obstacle boundaries. Shape distance measures employed in object recognition tasks can usually rely on significant shape information, e.g. when comparing the silhouettes of complex objects. From an experimental analysis I concluded that a more sensitive measure is required for recognition of polylines in robot mapping than is urged by object recognition tasks—see the forthcoming Fig. 5.8 for an exemplary comparison of shape distance measures.

Combining the DCE-based polyline extraction with an existing shape distance measure, I derived a new shape distance measure. The measure focuses shape comparison to shape properties exhibiting a maximal similarity. The measure presented in the following has been generalized for application in visual object recognition (Latecki et al., 2005b) and is termed partial optimal similarity, shortly POS. The approach can be regarded as a meta-approach to shape distance measures, as it relies on a basic shape distance measure and encapsulates this measure in an iterative evolution process.

The two following sections deal with the two components of this shape distance measure. First, the basic shape distance measure is presented. Thereafter, the integration of the basic shape distance measure into the iterative search process is detailed.

5.5.1 Basic shape similarity

As basic shape distance measure, shape similarity consideration based on correspondence of visual parts (Latecki & Lakämper, 2000) is utilized in SHRIMP; the utility of this measure has been demonstrated in context of the MPEG-7 shape descriptor experiments (see Latecki et al., 2000b). Shape distance based on correspondence of visual parts can be regarded an extension of the L_2-distance measure in tangent space proposed by Arkin et al. (1991). To fully acknowledge the properties of the robot mapping domain, I slightly adapted the measure as is presented in the following.

To determine a L_2-distance for polygonal curves P, Q in tangent space, the tangent space representations T_P and T_Q are normalized to the same curvature length of 1. Tangent space representations are then aligned to minimize the global difference in orientation before summing up the local differences. In contrast to the original approach by Latecki & Lakämper that utilizes a penalty for difference of *relative* length, difference of *absolute* length is regarded in SHRIMP. Since object boundaries are always present in the same scale, and it is not desired to achieve scale invariance, as is the case in many object recognition applications, consideration of change in absolute length is more appropriate in robot mapping:

$$
\begin{aligned}
S_a(P,Q) &:= \left(1+\lambda_1\left(L_P-L_Q\right)^2\right)\cdot\int_0^1\left(T_P(s)-T_Q(s)+\Theta_{P,Q}\right)^2 ds \\
\Theta_{P,Q} &:= \int_0^1 (T_P(s)-T_Q(s))ds
\end{aligned}
\tag{5.2}
$$

Since discrete polylines are compared in SHRIMP, the above integrals reduce to sums in the implementation. $\Theta_{P,Q}$ provides the alignment of orientation (refer to Arkin et al. (1991) for derivation). The deficit of the L_2-measure in tangent space is its sensitivity to local occurrence of noise. By shifting vertices from the contour, local noise elongates the contour locally, thereby deforming the proportions of the polyline with respect to the curvature length, e.g. the parts not affected by noise become shorter as regards relative curvature length. Fixed scaling and alignment only with respect to curvature length aligns the curves unfittingly (an illustration appears in Fig. 5.7, which is discussed later on). Shape distance computation as described by correspondence of visual parts overcomes this deficit by encapsulating the measure in an elastic matching of contour fragments that locally adapts curvature length.

Polygonal curves are decomposed into sequences of maximal left- or right-arcuated arcs, i.e. parts of positive and negative curvature. Based on this decomposition, the correspondence allowing for one-to-one, one-to-n, and n-to-one associations is computed that minimizes the shape distance of corresponding arcs. The main idea here is that at least on one of the contours a maximal arc corresponds to a part of the other contour, which is composed of adjacent maximal arcs (cf. Latecki & Lakämper, 2000). To determine shape distance of corresponding arcs, the tangent space measure of Eq. 5.2 is applied; the optimal rotation Θ is once determined globally. Cost-minimizing correspondence of arcs can be computed using Dynamic Programming (cf. Latecki & Lakämper, 2000). In the following, shape distance by correspondence of maximal arcs is denoted as S.

The computation of the basic shape distance measure is illustrated in Fig. 5.7. In the example, two polylines, P and Q, are compared, whereby Q differs from P only in the occurrence of local noise. When comparing the polylines by means of the L_2, a larger difference is determined than for correspondence of maximal arcs (shaded area in Fig. 5.7 (b)). In this example, the correspondence of maximal arcs can handle non-uniformly distributed noise. However, a considerable small change to a contour can change the structure of maximal arcs. In such a situation, the shape distance might be overestimated dramatically (see Fig. 5.8, which is discussed later on). As is demonstrated in the following, the developed extension to shape distance computation overcomes this deficit.

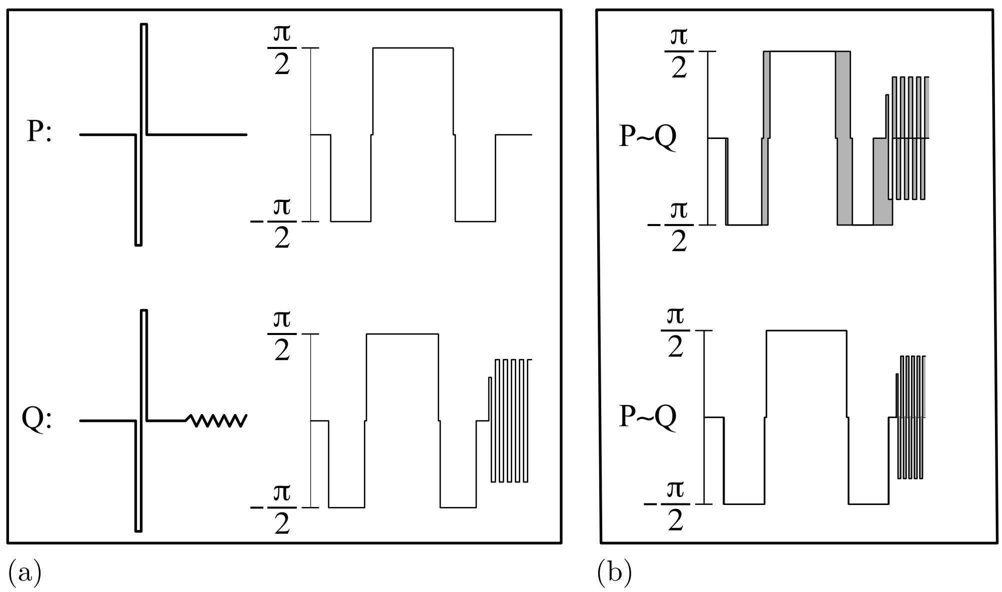


Figure 5.7: (a) Two polylines and their corresponding tangent-space representation. (b) Difference in tangent space for comparing $P \sim Q$ using L_2 measure in tangent space (top), the non-uniform distortions stretch the contour, thereby altering the relative position of the two salient peaks in the contour. As a result, the difference between the tangent-space representations is overestimated (shaded area). Comparing $P \sim Q$ using correspondence of maximal arcs (bottom), shape differences remain local to the actual distortion.

5.5.2 Partial optimal shape similarity

Polylines resembling laser range data suffer from sensor noise that cannot be removed easily. This is due to the relative size of the noise, which is challenging with the amount of shape information present. As a concrete example, the typical noise of a SICK LMS series laser range finder used in the experimental evaluation of this work is about $\pm$ 2cm, which is just about the size a door frame protrudes a straight wall. Thus, removing noise up to the magnitude of $\pm$ 2cm would also remove shape properties. However, losing any shape feature cannot be afforded due to the lack of salient shape characteristics.

To overcome this dilemma, noise identification and noise removal is delayed until shape distance of corresponding polylines is determined. In the context of corresponding polylines, it can more easily be decided whether a small shape property is more likely caused by noise in the sensor data or it contributes to the shape information. The proposed approach is to restrict determination of shape distance to the parts of a polyline that present a maximal similarity. Clearly, the ability to mask out parts of a polyline before determining shape distance requires a counter-weight to prevent masking out all differing shape properties.

In this context of robot mapping, polylines bearing sensor noise are compared with polylines represented in the map. Polylines in the map are considered to be already freed of noise.[4] Only vertices in the perceived polyline are examined, whether they are more likely caused by noise or they contribute to shape information. The challenge in designing an algorithm that resembles the outlined idea of restricting a polyline to a subset displaying high similarity lies in finding a computationally tractable solution. When considering a part of a polyline, i.e. a subset of vertices, combinatorial explosion can easily occur, as there are possibly 2^n subsets of n vertices to consider. Any naive implementation is infeasible in terms of required computational resources. To avoid combinatorial explosion, I propose an iterative algorithm that minimizes shape distance by iteratively removing vertices. Discrete Curve Evolution employed in the feature extraction is restarted until the minimum shape distance is reached. Iterative algorithms are efficient, but introduce sensitivity to local minima. A small lookahead can help to overcome most minima. In the implementation SHRIMPS, a lookahead of three steps is employed. DCE is continued for three more steps, after a local minimum has been detected. If shape distance decreases during the lookahead steps, minimization is continued. If shape

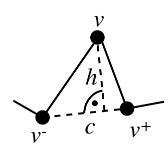
[4]Strictly speaking, this assumption is not fully valid, since any polyline extracted from the map has been added to the map at the moment it came first into sight. Newly added polylines suffer from noise just like any observation. However, repeated observation of the feature results in a gradually decreasing influence of noise, and, after a few observations, the map polyline suffers from little noise only, which is negligible in comparison with polylines extracted from sensor data. Later on, I will discuss a variation of partial optimal similarity that adapts to noise in both, observed polylines and map polylines.

distance does not decrease further during the lookahead steps, the process is ultimately stopped, yielding the determined minimum.

Vertex removal needs to be balanced with a counter-weight in order not to remove all differing shape information. The counter-weight is defined as a removal cost for vertices. One somewhat natural choice for defining the vertex removal cost is to reuse the vertex relevance measure: the larger the contribution of a vertex to the overall shape information, the higher the penalty for its removal. If a noise model is available for the range sensor at hand, it can be exploited for designing a noise-sensitive counter-weight. For example, the aforementioned residual noise of $\pm$ 2cm in range data can be transformed into a relevance measure that assigns a very low penalty for shifts in contour below 2cm, but grows rapidly for larger deviations. A large removal cost eventually inhibits vertex removal, as the removal cost finally dominates over any decrease of shape distance. Models defined in accordance with noise models appear well-suited to cancel out noise in polylines extracted from LRF data. SHRIMPS utilizes a removal cost that is based on typical noise in LRF data. The removal cost R for removing a set of vertices from a polyline P is defined on basis of a removal cost r that evaluates the cost for removing a single vertex v from a polyline. Denoting the sequence of vertices to be removed as $[v_1, \ldots, v_n]$, the measures are defined as follows:

$$R_P([v_1, \ldots, v_n]) := \sum_{i=1}^{n} r_{P \setminus \{v_1, \ldots, v_{i-1}\}} v_i$$

$$r_Q(v) := \lambda_2 \left(\frac{h}{c}\right)^2 \qquad (5.3)$$



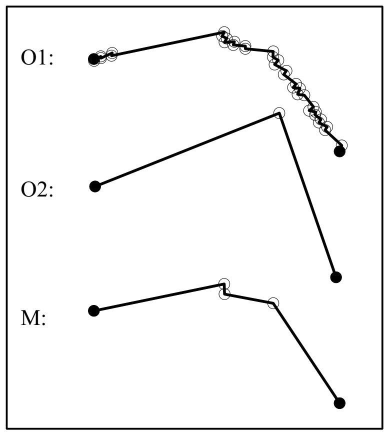
Hereby, h denotes the height and c the denotes the base of the triangle $\triangle v^+ v v^-$; λ_2 is a parameter balancing removal cost and shape distance[5]. The overall shape distance measure, $S^\star$, is then defined on the basis of the basic similarity measure S as follows:

$$S^\star(P, Q) := \min_{\substack{P^\star \subset P, \\ P^\star = [p_{i_1}, p_{i_2}, \ldots, p_{i_j}]}} \left(S(P \setminus P^\star, Q) + R_P(P^\star)\right) \qquad (5.4)$$

An illustration of the process of computing $S^\star$ is presented in Fig. 5.9; a noisy polyline is compared against a model in the course of curve evolution.

The presented definition of partial optimal similarity can be generalized to accomodate for a wider range of applications, such as a "key part" shape retrieval in analogy to keyword text search. In Latecki et al. (2005b) we presented

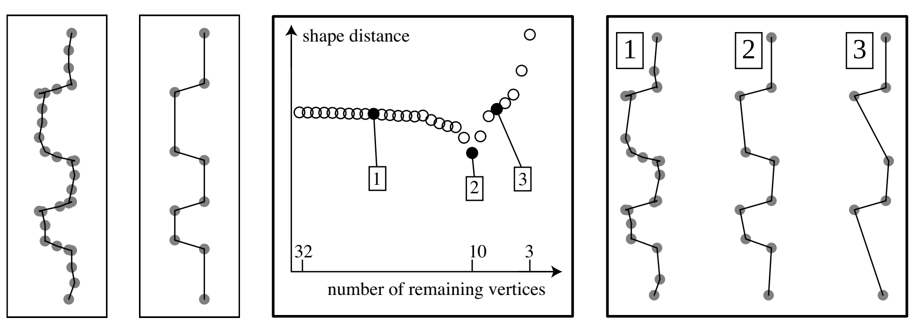
[5]In the experimental evaluation, a parameter value of $\lambda_2 := 2.5 \cdot 10^{-5}$ yields good results—see Section 6.1.



Measure	O1∼M	O2∼M
L_2-measure	1.85	1.51
Corr. of max. arcs	8.93	1.62
POS	0.10	73.97

(a) (b)

Figure 5.8: Comparison of shape distance measures. (a) Polylines O1 and O2 denote observations and M denotes a polyline represented in the map. O1 is a distorted variation of M and should result in a smaller shape distance than O2 to M. (b) Results obtained for different shape similarity measures; only the proposed partial optimal similarity (POS) determines a smaller shape distance for O1∼M than for O2∼M. Both, the L_2-measure by Arkin et al. (1991) and the correspondence of maximal arcs arcs by Latecki & Lakämper (2000), judge O2 to be more similar to M as O1 to M. The example indicates that partial optimal similarity is well-suited to recognizing noisy polylines.



(a) (b) (c) (d)

Figure 5.9: Computation of the model-based similarity measure. (a) An exemplary, distorted polyline as could be perceived. (b) A polyline as could be represented in the robot's internal map. (c) The development of the basic shape distance measure against number of remaining vertices while the vertex deletion advances; for illustration purposes, the process is not stopped when no further improvement has been found (marked '2'). (d) Intermediate stages in the vertex deletion process marked in (c).

variations of partial optimal similarity and their respective applications. For example, the process I developed for comparing polylines extracted from LRF data on basis of a removal cost according to a noise model can be adapted to recognizing shapes suffering from digitization noise.

As regards application to robot mapping, it should be noted that a symmetric vertex removal for comparing polylines similarly suffering from noise can be performed by alternatively removing a vertex from one or the other polyline. The symmetric formulation is a direct extension of Eq. 5.4 and reads as follows:

$$S^{\star}_{sim}(P,Q) := \min_{\substack{P^{\star}=[p_{i_1},p_{i_2},\ldots,p_{i_j}],P^{\star}\subset P\\ Q^{\star}=[q_{i_1},q_{i_2},\ldots,q_{i_k}],Q^{\star}\subset Q}} \{S(P\setminus P^{\star}, Q\setminus Q^{\star}) + R_P(P^{\star}) + R_Q(Q^{\star})\} \tag{5.5}$$

One option to derive a computational tractable approach that avoids combinatorial explosion in determining suitable subsets $P^{\star}$, $Q^{\star}$ is to approach computation of $S^{\star}_{sim}$ in an iterative manner by alternating evolution of both polylines as in computation of $S^{\star}$. The removal cost R inhibits that both polylines are evolved to straight lines which would present minimum shape distance. In principle, application of the symmetric variant allows dropping the assumption of only observed polylines substantially suffering from noise. However, an adequate removal measure for map polylines would be required that accounts for an estimated noise level in a map polyline, e.g. by considering the history of merging steps. Determining an appropriate noise model and deriving removal measures appear to be a challenging task, exceeding the scope of this work, but providing a starting point for further work and for potential improvements. Notably, experiments indicate that the asymmetric shape distance defined in Eq. 5.4 does not introduce any complications.

It should be noted that this shape distance measure is based on shape information only, neither the position nor the orientation of polylines is considered, i.e. shape distance is a purely feature-based measure not regarding configuration information. Obtaining decisive feature information is possible due to the large context information captured in polylines. The ability to employ distinctive feature distance measures is a significant difference between a shape-based map representation and a map representation based on less informative features, e.g. lines or points.

5.6 Matching

Provided two configurations of features, the task of the matching algorithm is to determine a plausible correspondence on the level of individual polylines. In SHRIMP, the two configurations stem from observation and map, respectively.

Qualitative ordering information about the circular sequence of perceivable objects is made explicit in the configuration, resembling that ordering information of extended objects can be regarded as confident information. This paves the way to applying the matching techniques described in Chapter 4. Put differently, the matching is formulated as a combinatorial optimization problem. Picking up the demands developed for a plausible correspondence in Section 5.1, they can now be stated precisely:

- The summed up shape distance of associated polylines shall be minimized.
- Circular order of visibility is regarded as a side condition.
- Polylines may be disregarded at the cost of a penalty measure.
- n-to-m correspondences of consecutive polylines need to be considered, concatenating simultaneously associated polylines to a single one.
- Each potential correspondence of two polylines induces an alignment that would adjust the complete configuration of polylines; a penalty for deviating from the common alignment is introduced.

In the following, I describe the computational modeling of these requirements in detail and describe how the matching can be computed using the matching technique presented in Section 4.3.3. For applying the matching technique developed in Chapter 4 polylines extracted from observation and map serve as the vertices of the hypergraph. Edges represent the correspondence relation. An edge weighting needs to be specified that assigns a non-negative measure to any potential correspondence of features, and to any single feature (if the feature remains unmatched). Before detailing the modeling, I briefly introduce the notation used in the following.

Let $S^\star$: polyline $\times$ polyline $\to \mathbb{R}^+ \cup \{0\}$ denote the shape distance measure. Sequences of polylines ordered in a circular manner are denoted by $P = [P_1, P_2, \ldots, P_n]$ and $Q = [Q_1, Q_2, \ldots, Q_m]$ respectively; a sub-sequence $[P_i, P_{i+1}, \ldots, P_j]$ will be abbreviated $P_{i,j}$ and $P_{i,i}$ will be abbreviated P_i. Subvectors represent a single polyline composed by concatenating a sequence of polylines. Polylines are concatenated by joining vertices in the order as obtained by the sensor. Furthermore, let $\sim$ denote the relation of correspondence which pins polylines from map and observation together. The aim is to compute the *optimal* correspondence relation $\sim$ with respect to the circular order of polylines denoted by $\prec$.

Since matching is a combinatorial optimization problem, a penalty for not associating a polyline needs to be introduced, as otherwise, the empty correspondence relation yields zero shape distance, the lowest possible, i.e. optimal choice. This ignorance penalty measure is defined for individual polylines. It defines the edge weighting for hyperedges covering a single polyline. I propose a counterweight function I : polyline $\to \mathbb{R}^+ \cup \{0\}$ that grows linearly with the

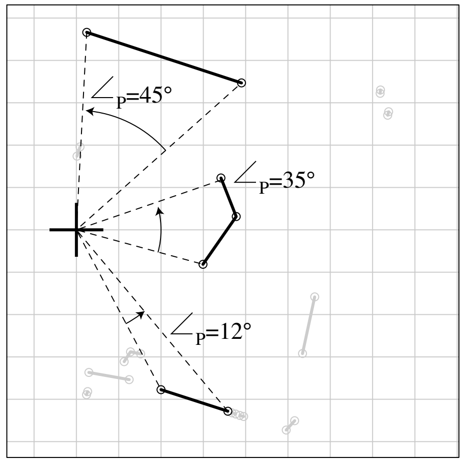


Figure 5.10: Measuring the angular field of view to determine the ignorance penalty for polylines

polyline's angular size in the field of view (see Fig. 5.10). Denoting the angular field of view covered by a polyline by $\angle_P$, the penalty is defined as follows:

$$I(P) = \lambda_3 \cdot \angle_P \tag{5.6}$$

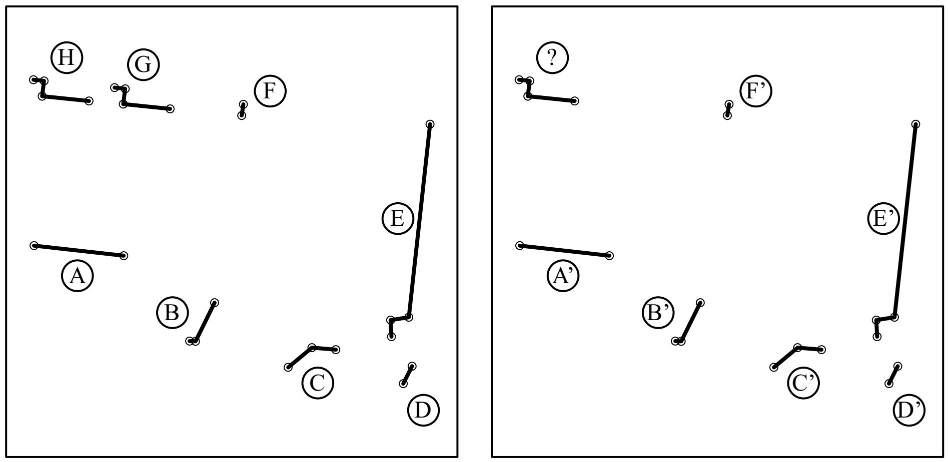
A linearly growing penalty reflects the observation that the shape distance of two polylines, which differ only by independent noise, grows linearly too. In the developed shape distance measure, differences of curvature are accumulated. Assuming distortions to be equally distributed over two corresponding contours, a relative change of curvature lengths implies the same relative change of shape distance. The proposed object-dependent penalty that increases with the size of objects enforces correspondences for salient environmental features to be established. As a result, the matching procedure more easily agrees with many small changes in the environment as compared to fewer large ones, which is a reasonable characteristic. This model vaguely echoes human intuition in recognizing local surroundings, since small objects tend to be moved around more frequently (cf. Dong, 2005).

The necessity to allow for n-to-m correspondences has been discussed, but a naive realization would introduce complications—a simple example illustrates this. Assume, two identical sequences of polylines $[P_1, \ldots, P_n]$ are matched. The sequence of one-to-one correspondences $P_1 \sim P_1, \ldots, P_n \sim P_n$ would not be preferred over the matching using a single n-to-n correspondence $P_{1:n} \sim P_{1:n}$. Noise superimposing on one of the involved polylines may even cause the concatenated polylines to be the cost-optimal solution. Inadequate concatenation of polylines spans (and closes) arbitrary large gaps between map features that could not possibly stem from grouping differences; such erroneous links easily cut off parts of the environment in the map, thereby introducing the problem of extra grouping discussed in Section 3.4.3. Thus, a constraint needs to be introduced to avoid inadequate grouping. I propose employing a combination of soft and hard constraints. One one hand, hard constraints inhibit grouping of polylines farther apart than a fixed threshold, but do not help avoiding groupings spanning smaller gaps, if reasonable. On the other hand, soft constraints by means of gap-dependent penalties avoid grouping, but cannot reliably suppress grouping, e.g. inhibiting linkage of gaps that are regarded as too wide. In SHRIMPS, I utilize a small penalty of $0.5d^2$ as soft constraint, whereby d denotes the gap distance. Additionally, a hard constraint suppresses re-grouping of polylines farther apart than 0.5 meter. This ad-hoc solution proofed reasonable in the experiments. The penalty for concatenating two polylines is extended for concatenation of a sequence of polylines, hereby d_i denotes the gap between polylines P_i and P_{i+1}:

$$\begin{aligned} g(d) &:= \begin{cases} 0.5d^2, & \text{if } d < 0.5 \\ \infty, & \text{otherwise} \end{cases} \\ G([P_i, P_{i+1}, \ldots, P_j]) &:= \sum_{k=i}^{j-1} g(d_i) \end{aligned} \tag{5.7}$$

My interpretation of plausible data integration includes the demand to only associate polylines which induce a compatible alignment (see Section 5.1)—this is the most delicate part in computation of the matching, since a common ground for induced alignments is not known a-priori. This mens, no misalignment penalty can directly be assigned to pairs of polylines in terms of an edge weighting required for application of the developed hypergraph matching technique. One option of addressing this difficulty is to start associating polylines without consideration of alignment, and consider alignment, not before it can be derived from the features already assigned. Truly to determine the maximal set of mutually compatible assignments, different alternatives for the first assignments need to be examined in order not to select an unfavourable assignment in first place. Examination of alternative associations increases the computational burden (cf. Neira & Tardós, 2001). Therefore, I propose a different solution. In the following, I discuss that in shape-based mapping, one can easily obtain a sufficiently reliable estimate of the alignment, which then allows definition of an edge weighting.

Assume that an estimate of the induced alignment would already exist. In this case, a penalty for incompatibility of induced alignments can be designed by comparison with this estimate. Alignment consistency, shape distance, and ignorance penalty could be combined to a single edge weighting in hypergraph matching. Note that in the context of a decisive feature distance measure such as shape distance, positional information takes a subordinate role. Significant shape distance dominates over positional information, so precise evaluation of



(a) (b)

Figure 5.11: (a) Exemplary map view comprising two features of identical shape (G and H). (b) Observation of the surrounding depicted in (a) which misses one of the two features (G); only knowledge about congruent alignment allows identifications of the observation marked "?" and the map feature H.

alignment is rarely necessary. Many polylines can be identified purely by considering shape information. Indeed, in some situations alignment needs to be regarded: in Fig. 5.11 an exemplary environment is depicted that comprises two features of identical shape. Assume that in an observation of this surrounding, only one of these features is observed. In the depicted example (image (b)), only positional knowledge allows for disambiguation, since both map features G and H are potential correspondence partners in terms of circular order, and both are of identical shape. In practical investigations, this phenomenon occurs when identifying small, simple polylines that lack of informative shape information and can easily be overlooked. I conclude that the alignment estimate needs to provide sufficiently detailed information to disambiguate simple shapes, but does not necessarily need to provide very detailed information.

Two alternatives present themselves for obtaining a suitable alignment: first, if odometry information is available, it provides a sufficiently reliable estimate. In this case, the map view has been retrieved according to odometry and the observation and the map view can be regard to be aligned already, i.e. the estimated alignment is zero. Second, if odometry is not available, the estimate can be computed by shape analysis—the next section details this process. On that account, I assume in the following that such an estimate, i.e. a translation vector $\vec{t}$ and a rotation by Φ exists. I denote alignment induced by corresponding polylines P and Q by $A(P,Q)$. The difference of the induced alignment $A(P,Q)$ and the estimated alignment is denoted as $\Delta A(P,Q)$. To measure $\Delta A(P,Q)$, SHRIMP utilizes a parametric model that proved reasonable in experiments[6]:

$$D(dt, d\Phi) = \lambda_4 ||\vec{dt}|| + \lambda_5 d\Phi \tag{5.8}$$

Denoting the set of unmatched polylines $\{P_i, P_{i'}, \ldots, Q_j, Q_{j'}, \ldots\}$ by $\overline{PQ}^{\sim}$, determination of the optimal correspondence relation $\sim^\star$ is formulated as follows:

$$\sim^\star = \underset{\substack{\sim \text{ is homomorphic matching} \\ \text{of } P \text{ and } Q \text{ w.r.t. } \prec}}{\arg\min} \underbrace{\sum_{R \in \overline{PQ}^{\sim}} I(R)}_{\text{ignorance penalty}} + \sum_{(P_{\overline{i:j}}, Q_{\overline{i':j'}}) \in \sim} \Bigg(\underbrace{S^\star(P_{\overline{i:j}}, Q_{\overline{i':j'}})}_{\text{shape distance}} + \underbrace{G([P_i, \ldots, P_j]) + G([Q_i, \ldots, Q_j])}_{\text{grouping penalty}} + \underbrace{\Delta A(P_{\overline{i:j}}, Q_{\overline{i':j'}})}_{\text{alignment consistency}} \Bigg) \tag{5.9}$$

The equation can be solved, using the techniques developed in Section 4.3.3. Roughly speaking, the developed DP scheme introduces an update step to the standard DP scheme applicable to incremental computation of $\sim^\star$, if no multiple correspondences would be considered (cp. the proof of Theorem 1). The developed extension of DP can be interpreted in the context of matching configurations of polylines as follows: the extensions enables overcoming a prefix requirement in classical DP by reconsidering correspondences determined in previous steps. Suppose a polyline P shall be matched against two polylines Q_1, Q_2 that are created by splitting P. In classical DP, the result of comparing Q_1 to P cannot be altered in subsequent computation, i.e. the solution to a subproblem (cf. Section 4.3.1) is fixed and it is a proper part of the overall solution. In this example this means that if P and Q_1 are sufficiently dissimilar, Q_1 is irrevocably disregarded. Consequently, Q_2 is likely not to be matched either. In the developed extension to DP, not associating P and Q_1 is again inspected when comparing Q_2 and P; this gives the correct correspondence of P and $Q_{1:2}$, the concatenation of Q_1 and Q_2. In the following, I present the complete algorithm in detail—for comparison, refer also to the proof in Theorem 2.

Let P be a sequence of polylines of length n, i.e. $P = [P_1, \ldots, P_n]$. Analogously, let $Q = [Q_1, \ldots, Q_m]$. For matching two P and Q, a matrix M of size $n \times m$ is used similar to most DP schemes. The cell $M_{i,j}$ corresponds to the matching of sub-sequences $P_{1:i}$ and $Q_{1:j}$. So, $M_{i,j}$ represents the solution of the

[6]See Chapter 6 for concrete choices of the parameters λ_4 and λ_5

subproblem $\Pi_{i,j}$ introduced in Section 4.3.1, and $M_{n,m}$ represents the overall solution. At the beginning, M is empty and is initialized by computing the entry $M_{1,1}$. Two alternatives need to be evaluated:

1. P_1 and Q_1 are matched, yielding the cost $S^\star(P_1, Q_1) + \Delta A(P_1, Q_1)$.
2. P_1 and Q_1 are both ignored, yielding the cost $I(P_1) + I(Q_1)$.

The cost-minimal choice is selected and stored in $\sim_{1,1}$, i.e. if P_1 and Q_1 are matched, $\sim_{1,1}$ is set to $\{\{P_1, Q_1\}\}$, if they are not matched, $\sim_{1,1}$ is set to $\emptyset$. The respective cost is stored in $C_{1,1}$. Both values, $\sim_{1,1}$ and $C_{1,1}$, constitute the cell $M_{1,1}$, i.e. cost and correspondence are stored in the matrix. For ease of description, matrix cells $M_{0,j}$ and $M_{i,0}$ are assumed to be defined and to denote zero cost and no correspondence. Remaining entries $M_{i,j}$ can be computed, when all predecessors $M_{i',j'}$ have already been determined for all $i' < i$ and $j' < j$. To determine $M_{i,j}$, or, equivalently, to solve the subproblem $\Pi_{i,j}$, all cells $M_{i',j'}$ may need to be considered according to the proof of Theorem 2. For clarity and ease of description, three cases of determining $M_{i,j}$ are distinguished:

Skip : $M_{i-1,j} \to M_{i,j}$ Advancing from $M_{i-1,j}$ by introducing P_i into the subproblem, but maintaining the correspondences $\sim_{i-1,j}$, i.e. leaving P_i unmatched.

$$\begin{aligned}\sim_{i,j} &:= \sim_{i-1,j} \\ C_{i,j} &:= C_{i-1,j} + I(P_i)\end{aligned}$$

Skip : $M_{i,j-1} \to M_{i,j}$ Advancing from $M_{i,j-1}$ by introducing Q_j into the subproblem, but maintaining the correspondences $\sim_{i,j-1}$, i.e. leaving Q_j unmatched.

$$\begin{aligned}\sim_{i,j} &:= \sim_{i,j-1} \\ C_{i,j} &:= C_{i,j-1} + I(Q_j)\end{aligned}$$

Match : $M_{i'-1,j'-1} \to M_{i,j}$ Extending the correspondences represented by $M_{i'-1,j'-1}$ by matching $P_{\overline{i':i}}$ and $Q_{\overline{j':j}}$, i.e. the polyline obtained by concatenating $P_{i'}, P_{i'+1}, \ldots, P_i$ and the polyline obtained by concatenating $Q_{j'}, Q_{j'+1}, \ldots, Q_j$. For computing the cost-optimal n-to-m correspondence, the values $i' = 1, \ldots, i$ and $j' = 1, \ldots, j$ need to be selected, such that the summed up cost of the shape distance $S^\star(P_{\overline{i':i}}, Q_{\overline{j':j}})$, the polyline concatenation $G([P_{i'}, P_{i'+1}, \ldots, P_i])$ and $G([Q_{i'}, Q_{i'+1}, \ldots, Q_i])$, the alignment consistency $\Delta A(P_{\overline{i':i}}, Q_{\overline{j':j}})$, and the cost $C_{i'-1,j'-1}$ is minimized.

$$\begin{aligned}\sim_{i,j} &:= \sim_{i'-1,j'-1} \cup \{P_{i'}, \ldots, P_i, Q_{j'}, \ldots, Q_j\} \\ C_{i,j} &:= S^\star(P_{\overline{i':i}}, Q_{\overline{j':j}}) + G([P_{i'}, \ldots, P_i]) + G([Q_{j'}, \ldots, Q_j]) + C_{i'-1,j'-1}\end{aligned}$$

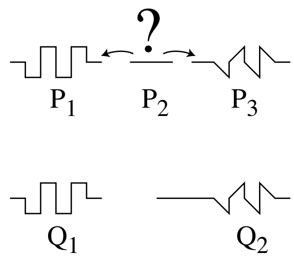


Figure 5.12: Illustration to the extension made to the Dynamic Programming scheme when matching sequences of polylines, $[P_1, P_2, P_3]$ and $[Q_1, Q_2]$. As could be caused by segmentation differences, the polyline Q_2 corresponds to P_2 and P_3, thus, the optimal matching is $P_1 \sim Q_1$, $P_{\overline{2:3}} \sim Q_2$. When only considering the best matching of the sub-sequences $[Q_1]$ and $[P_1, P_2]$ (which is an intermediate step in matching based on Dynamic Programming), it may be optimal to match Q_1 to $P_{\overline{1:2}}$. To enable recovery from such faulty intermediate result, the developed extension to the Dynamic Programming scheme is necessary. Here, the association of P_2 is reconsidered when the matching advances to P_3 and Q_2, giving the correct correspondence $Q_2 \sim P_{\overline{2:3}}$.

Of these three alternatives, the cost-otimal choice is selected. To evaluate the matching step for obtaining $M_{i,j}$, all concatenations $P_{\overline{i':i}}$ for $i' = 1, \ldots, i$ and, simultaneously, all concatenations $Q_{\overline{j':j}}$ for $j' = 1, \ldots, j$ need to be evaluated. Thus, a single step results in a $O(n \cdot m)$ complexity. Roughly speaking, determining the best subsequences allows to recover from faulty intermediate results that are only apparent in a larger context (see Fig. 5.12). As there are $n \cdot m$ cells to evaluate, the overall complexity is $O(n^2 \cdot m^2)$ (see Corollary 3). Notably, the presented complexity is an upper bound—it is not yet clear whether a more efficient equivalent algorithm exists.

As regards practical application, the implementation can exploit the hard constraint in the grouping penalty G: if the grouping penalty increases to infinity, no further grouping needs be considered. In computation of the matching step, this means that the search of i', j' can be restricted. If i' (or j' respectively) leads to an infinite grouping penalty, lower values of i' (or j' respectively) do not need to be evaluated, saving computational costs. In typical situations, where there are some polylines that are not allowed to be joined, i.e. that are farer apart than 0.5 meter, only few alternatives for i', j' need to be evaluated, cutting the computational cost well below the worst case. The developed procedure has proved to be suitable to real world applications, in which typically up to $10 - 20$ features need to be correlated.

An example of the matching is depicted in Fig. 5.13 where information extracted from range finder data is matched against map information. The esti-

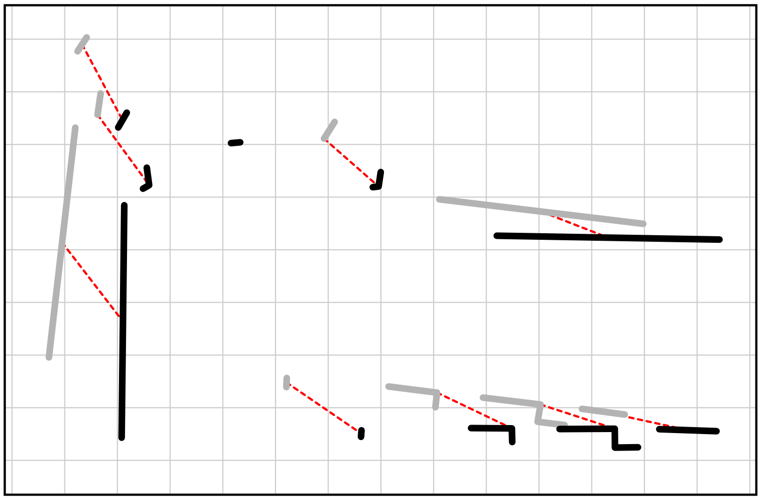


Figure 5.13: Determining the correspondence of two configurations of polylines (grey and black). Here, the estimated alignment is set to zero, i.e. the matching assumed that both views are congruently aligned. Even this very poor estimate of alignment does not disturb the matching (dashed lines). Consideration of shape distance and constraints of circular order yields a robust matching.

mated alignment has been set to $(0, 0)$, i.e. the robot initially believed that both configurations would line up in a direct manner. However, the real difference in view poses (and simultaneously the true alignment) comprises a translation of more than one meter. In other words, the supplied estimate of alignments is of very poor quality. Note that polylines are correctly associated even though potential correspondence partners are positioned much closer according to estimate. Nevertheless, the outlined matching technique masters the task. This demonstrates the utility of a distinctive feature distance measure. Robustness to poor estimates is achieved by strict observation of circular ordering and consideration of shape similarity. In this regard, the developed matching advances on existing techniques that mainly rely on position estimates (cf. Section 3.5). In particular, the shape distance measure that dominates over positional information derived from estimates allows for this improvement.

One question urges: will the algorithm described always determine a globally optimal correspondence? Unfortunately, the answer is no, maybe not. To be more precise, it is unclear under which conditions the utilized edge weighting, i.e. the term minimized in Eq. 5.9, fulfills the requirement of local optimality (cf. Section 4.3.3). In presence of *significant* shape information, it is *likely* that the requirement is satisfied—however, it appears infeasible actually to prove it in terms of a general theorem—maybe it cannot be proved at all. Difficulties of formal treatment arise from the high dimensionality of the problem space: polylines can adopt arbitrary shapes, can be positioned at arbitrary places, and so on. To actually prove optimality of the practical implementation would require to gather all variations in one formal, tractable theory. In this work, an empirical evaluation of the matching is performed in context of a self-localization task in a known environment. The evaluation indicates that the matching determines a plausible solution.

5.6.1 Obtaining an alignment estimate from shape analysis

For small polylines lacking of shape information, consideration of positional information is indispensable to identifying their correspondence. However, for *complex* polylines, matching purely based on shape distance is already capable of plausible data association, as the shape distance measure is more dominant and can exploit distinctive shape properties. This observation lays the foundation to estimating the induced alignment from shape analysis. If one had two corresponding polylines, one could compute the induced alignment and use this as the estimation in the matching. To obtain a corresponding pair, matching is extended to a two-step process. An additional matching phase is introduced, which is performed without considering positional information at all, i.e. the term $\Delta A(\cdots)$ in Eq. 5.9 is ignored. Matching can correctly determine correspondences of complex polylines. Then, the *most reliable* correspondence of *complex* polylines is selected and used to determine the desired estimate. Thereafter, the actual matching can be performed as has been described above.

Thus, the task is to define complexity of a polyline and reliability of a matching. Note that complexity has already been issued in SHRIMP: curve evolution in the shape extraction process relies on a relevance measure of vertices, which measures the contribution of a single vertex to the overall shape information. Put differently, the relevance measure determines the contribution of a single vertex to the complexity of a polyline. It comes natural to also rely on this model here. Vertex relevance can canonically be extended to complete polylines by adding up the individual relevance measures for all vertices. This yields the definition of polyline complexity for a polyline P comprising the vertices $p_1, p_2, \ldots, p_n$[7]:

$$C(P) := \sum_{i=2}^{n-1} K(p_{i-1}, p_i, p_{i+1}) \tag{5.10}$$

Shape distance can be regarded to model reliability of a correspondence

[7]To account for environments presenting mostly straight obstacle boundaries, it is advantageous to assign a complexity to straight lines as well. In the DCE-based complexity measure, straight lines have a zero complexity, since they do not include inner vertices. Assigning a complexity to straight lines (e.g. equal to a symmetrical corner with right angles and the same curvature length), enables exploitation of salient straight lines in determination of the alignment to be anticipated. When including straight lines, two non-parallel lines need to be selected for determining translation and rotation.

computationally. As stated in the above, naive formulation, the goal is to find the correspondence incorporating high shape similarity (i.e. low shape distance) and high complexity of the polylines. So, a straightforward realization of a reliability measure is:

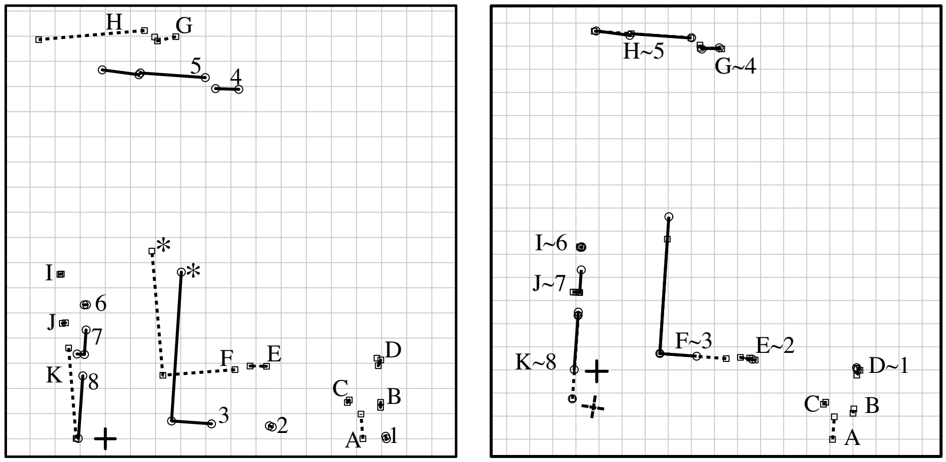
$$Q(P,Q) = C(P) + C(Q) - S^{\star}(P,Q) \tag{5.11}$$

Page et al. (2003) proposed an alternative approach to determining shape complexity which evaluates the uniqueness of turning angles of a polyline using statistical analysis with respect to a shape database. Since SHRIMP aims at determining the reliability of a polyline correspondence, which is a property of the individual polylines involved, the approach proposed in this dissertation appears better suited. In contrast to an analysis based on information theory, SHRIMP determines a shape to be complex, even if multiple shapes are contained in the view that are similar. This is no problem for the matching, as the circular ordering respected in the matching ensures a robust association.

In the experiments, it turned out that this definition worked well and does not require further balancing between shape complexity and shape distance. To illustrate the effectiveness of the two-step matching, refer to Fig. 5.14, where two configurations of polylines are matched against each other without an a-priori estimate of alignments. The polylines marked $\star$ denote the most reliable correspondence. Observe that the feature 'D' is correctly matched to the feature '1', even though features 'A', 'B', and 'C' are closer to '1' at the starting point (image (a)), and these features lack decisive shape information. In this example, the developed matching succeeds correctly associating the configurations, which true origins differ by more than one meter. This is a dramatical improvement compared to the precision required by standard scan matching approaches which typically rely on a hill-climbing strategy (e.g. Hähnel et al., 2002).

5.7 Alignment

The objective of alignment is determination of a two-sided mapping of two reference systems. Computation is based on a known correspondence of objects described with respect to the reference systems. Here, observed polylines described in a local coordinate system need to be aligned with their corresponding map polylines, which are embedded in the absolute coordinate system of the map. By relating the origin of the local coordinate system to the absolute coordinate system, the robot pose can be inferred. Similarly to most other approaches for determination of the alignment(cf. Section 3.6), SHRIMP aims at minimizing the incongruence of observation and map by determining an optimal translation and rotation. In other words, alignment is performed as optimization. In contrast to popular approaches of incremental alignment (cf. Section



(a) (b)

Figure 5.14: (a) Two configurations of shape features obtained from sensor data (numbered 1–8 and A–K) are matched only with respect to the shape distance measure. Observe that at this stage it is impossible to identify the most plausible correspondence of polyline '1'. The most reliable match (marked $\star$) is consulted to infer an estimate for the configurations' alignment. The final matching can then be performed; correspondences found, and the two scans aligned according to the estimation are shown in (b). Observe that the scans' origins (marked by crosshairs) are farther apart than 1m (grid denotes 1m distance) and no odometry has been used to match the views.

3.5.4), a reliable correspondence of features is known before the alignment is computed.

To determine the alignment, I adapt a scan matching technique originally developed by Cox (1990) and improved by Gutmann (2000). The adapted approach is based on a scan point to line matching, i.e. points detected by the range sensor are matched against model lines represented in the map. Distance of scan points and model lines is minimized by aligning the scan. Given a correspondence of points and lines, the optimal alignment minimizing the least square error can be computed in a closed form (cf. Gutmann, 2000). In SHRIMP, mapping of scan points to polylines in the map is restricted by the known correspondence of polylines. The mapping of points to map polylines respects the determined correspondences.

Before starting the alignment process, map polylines participating in the matching are expanded. When retrieving the view from the map, visible parts of the boundary have been determined, while maintaining a link to the generating polyline in the map (see Section 5.3). If occlusion occurs, a single polyline in the map corresponds to multiple polylines in the view. Put differently, polylines in the observation corresponding to a single generating map polyline represent different fragments of the same polyline. Now, these fragments are expanded to the full extent of the map polylines. This helps congruently aligning observation and map, if the map view does not provide full congruency with the observation. Analogously to expanding map polylines, observed polylines are expanded: if multiple polylines in the observation are jointly matched (i.e. concatenated), these polylines are linked to a single polyline. Linking ensures that the order of fragments is maintained in alignment and merging, i.e. if one fragments proceeds another with respect to circular order, and if both fragments are matched to the same polyline, then linking the observed fragments ensures that the fragments are aligned to parts of the map polyline, which also proceed one another accordingly.

In a first step, corresponding polylines are aligned by aligning their endpoints using the technique described by Lingemann et al. (2004). This yields a good approximation of the optimal solution and provides a solid starting point for the incremental alignment according to Gutmann (2000).

For each polyline perceived, sample points are required for employing the alignment technique described by Gutmann (2000). My actual implementation utilizes a sampling distance of 1 cm, however, the concrete choice is not critical[8]. Additionally, it is made sure that at least one sampling point lies on each of the polyline's line segments. Segments introduced by linking perceived polylines are not sampled, as the links are purely artificial. Using these sampling points, corresponding line segments of corresponding polylines are determined based on proximity. For every sampling point, the nearest point contained in the corresponding polyline is computed and the scan is aligned. The procedure is repeated until convergence is detected, i.e. the alignment does not change significantly any more.

Experiments indicate that this straightforward adaption of previous work yields good results. The main difference to the original work is that alignment in SHRIMP can reckon on a reliable correspondence determination. An exemplary result of the alignment contrasted to the original work of Gutmann (2000) is depicted in Fig. 5.15—as can be observed, the alignment is correctly determined. In Fig. 5.16 the applicability of this alignment technique to complex contours is demonstrated.

[8]In SHRIMP, I sample the perceived polyline rather than refer to the original data points that let to extraction of the polyline. The technique described by Gutmann (2000) requires knowledge of points and corresponding tangents of object surfaces. From the polyline representation, tangents can easily and robustly be obtained.

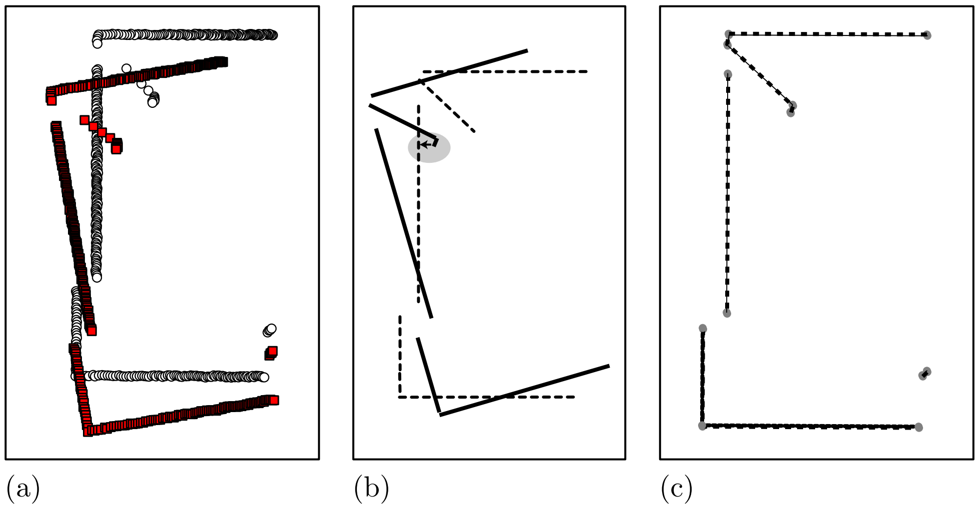


Figure 5.15: (a) Two exemplary observations (marked as boxes and circles respectively) obtained by a simulated LRF in a local frame of reference. (b) The alignment of extracted lines obtained according to Gutmann (2000) is stuck in a local minimum related to an erroneous matching of the marked line segment. (c) The alignment of extracted shape information computed by SHRIMPS relies on the shape-based matching and correctly aligns the scans. Note that the observations depicted in (a) are interpreted slightly differently, since SHRIMPS relies on shape extraction instead of line fitting.

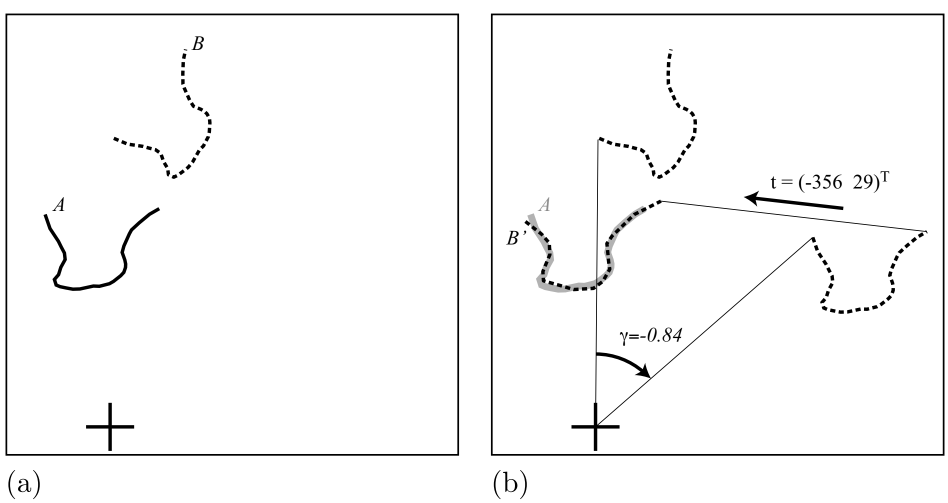


Figure 5.16: (a) Two polygonal contours (A,B) to be aligned; the cross denotes the point of reference (b) Illustration of the alignment computed by SHRIMPS.

5.8 Merging

The map is updated by merging observed polylines to corresponding map polylines, and, by adding new observations emerging in the field of view.

Features in the observation that are not matched to map features are interpreted as newly observed features and are added to the map. Since very small polylines do not display shape information that can be exploited in the matching phase, these polylines may easily remain unmatched. This is particularly the case, when remote fragments of an obstacle are visible. To avoid multiple registrations of features, small polylines outside the immediate surrounding of the robot are not registered in the map. The experimental system utilizes a threshold of 10 cm in polyline diameter as minimum size and a radius of 2 meter for defining the immediate surrounding of the robot. Both values have proven reasonable in the experiments and are uncritical parameters.

The main task of the merging procedure is mediating between differing perceptions of the same polyline in multiple observations and refining the polyline's appearance in the map. My approach to merging is related to the approach on curve morphing suggested by Sebastian et al. (2003) (see also cf. Section 3.7). The task of map merging in robot mapping differs from curve morphing, though. In robot mapping, due to change of viewpoints, only some parts of polylines may actually be visible and correspond. Hence, the morphing procedure can only be applied to the overlapping, i.e. jointly visible part. For example, if a larger fragment of an obstacle boundary is observed than is registered in the map, only the jointly represented fragments can be merged. In SHRIMP corresponding polylines are decomposed into three parts: *head*, *body*, and *tail* (see Fig. 5.17). The body part describes the jointly represented fragments. Head and tail denote the remaining parts. Determination of head, body, and tail exploits that the corresponding polylines have already been aligned. To determine the begin of head and tail parts, each polylines' endpoint are mapped to the nearest points on the other polyline (Fig. 5.17 (b)). If endpoints map to endpoints, the polylines have a common dimension. Otherwise, the mapped point marks the beginning of the head part (or tail part, respectively). For improving robust detection of parts sharing the same dimension, but which are not accurately aligned, a small tolerance is introduced when comparing points.

In accordance to the approach suggested by Sebastian et al. (2003), merging of corresponding body parts would be performed by first determining sample points on the contour, then matching these points using a DP scheme, and finally balancing the sample points between their original position and the position of their corresponding counterpart. As regards an adaption of this approach to robot mapping, I applied some changes.

Sample points on the polyline extracted from observation have already been determined to compute the alignment and are reused here. No sampling points on the map polyline are determined in SHRIMP, but points corresponding to

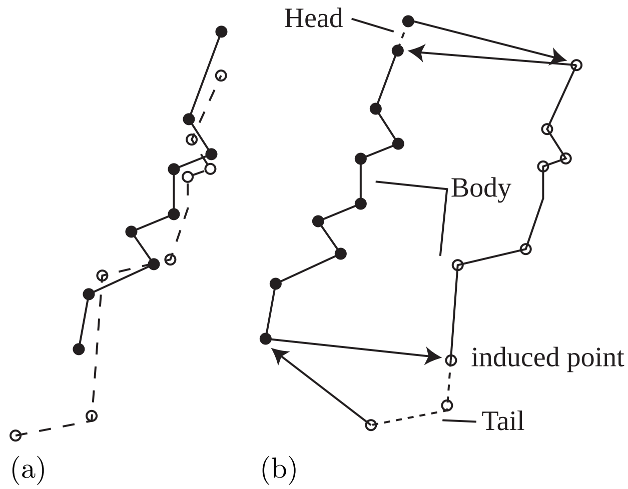


Figure 5.17: (a) Corresponding, aligned polylines. (b) Decomposition into head, body, and tail parts; for illustration purposes, the polylines have been shifted apart in (b).

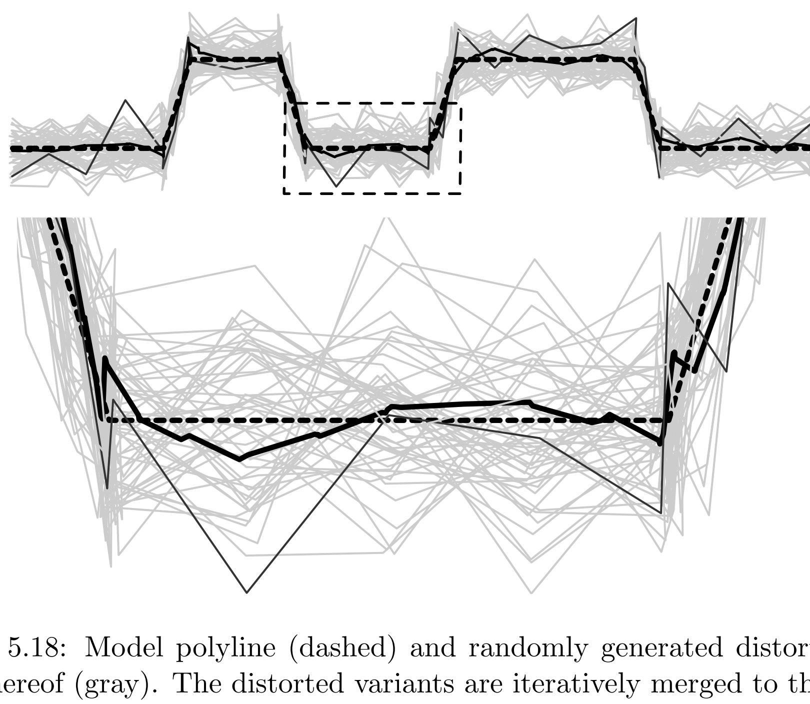


Figure 5.18: Model polyline (dashed) and randomly generated distorted variants thereof (gray). The distorted variants are iteratively merged to the resulting polyline (solid black) which resembles the original model. The lower image depicts the enlarged part that is marked in the upper image.

the sample points of the observed polyline are determined by the matching. The original work addresses shape recognition and, therefore, incorporates a complex cost function that is minimized by adequately matching discrete points. In the context of updating the map, corresponding polylines are both similar and aligned. So, a much simpler weight function may be utilized which does not require any balancing of parameters. In SHRIMP, I employ the same techniques as for aligning observation and map: a proximity-based weight function (Euclidean distance is used) and a point-to-polyline-segment matching. This approach avoids further parameters, while at the same time it is not restricted to association of discrete points, but discrete points from the observation are mapped to arbitrary points of the map polylines. This avoids distortions introduced by a fixed discretization. In other words, merging is performed in direct continuation of alignment. In alignment, all sample points are simultaneously moved to fit the configuration retrieved from the map most congruently. In merging, the points are individually adapted to the map. Associated points are balanced by a weighted average, whereby weighting is defined by the amount of times a polyline has been sensed. The underlying idea is that repeated observation and merging increases the certainty in the appearance of a map polyline. As a result, the shape of a polyline is stabilized the more often it is observed. This technique is suitable to cancel out measurement noise in corresponding polylines. Notably, this is not a stochastic sound modeling of measurement noise. A sound treatment of noise in merging is not within the scope of my work. Indeed, results indicate that this approach already provides good results. In Fig. 5.18 the outcome of iteratively merging strongly distorted polylines is demonstrated. Image (a) depicts a model curve (bold) that has been used to randomly create distorted variants of the model (grayish polylines; one exemplary variant is highlighted). Iteratively merging 30 polylines, the outcome, which is depicted in image (b), closely resembles the model polyline.

5.9 Summary & conclusion

In this Chapter, I have presented my approach to robot mapping, starting by a characterization of plausible data integration. I have argued that plausibility in mapping is shaped by plausibility in correspondence determination in every respect. My requirements for plausible correspondence determination respond to conclusions drawn in earlier Chapters and are realized in the presented computational approach to matching. In summary, plausible data integration must aim at minimizing a suitable feature distance measure, account for grouping differences, respect mutual compatibility of associations, and respect confident knowledge of configuration. These requirements are implemented in the computational model to robot mapping, named SHRIMP. It is an incremental mapping architecture that is centered on a matching algorithm that reflects the theoretical results described in the preceding Chapter.

A second, central component in SHRIMP is its specially designed shape distance measure. This measure provides distinctive information about feature similarity and provides the basis that enables to disregard of estimates of features locations, which are often unreliable, or may not be available when relating observation to an external map. Shape distance computation is based on state-of-the-art techniques stemming from object recognition. Studies on shape similarity in the context of robot mapping[9] have allowed improvement of shape distance measures and return enhancements to object recognition, which is documented in Latecki et al. (2005b).

SHRIMP extends earlier work using polylines in a-priori manual world modeling (e.g. Chatila & Laumond, 1985) by means of autonomous map acquisition and improved self-localization. It improves recent approaches to map acquisition that extract polygonal lines from congruently aligned range finder data (e.g. Veeck & Burgard, 2004), as SHRIMP addresses the full spectrum of map construction, in particular correspondence determination on the basis of a polyline-based map. SHRIMP provides a constructive validation of to the claim of my thesis that shape information provides a solid basis to robust mapping.

To complement key techniques of matching, map representation, shape extraction, alignment, and merging procedures have been described. Computer implementations of all procedures involved in robot mapping are gathered in an prototypical implementation termed SHRIMPS. In the next Chapter, an experimental evaluation of SHRIMPS is presented for providing a proof-of-concept.

[9] (Latecki et al., 2003; Wolter & Latecki, 2004; Wolter et al., 2004)

Chapter 6

Evaluation

Grau, teurer Freund, ist alle Theorie,
Und grün des Lebens goldner Baum.

Goethe, Faust I

In this Chapter, I present an evaluation of my approach to robot mapping. Evaluation is performed in experiments using simulated and real-world sensor data. In a first study, I present results from examining self-localization performance. Self-localization performance is essentially governed by the performance of correspondence determination, thus, providing good insight in results achieved as regards one of my central objectives of developing a robust approach to correspondence determination on the basis of shape analysis. Experiments cover the standard task of relating observations to the robot's internal map. In addition, one setup employs an external, coarse map, to which the observations of the robot need to be related. This setting is included to evaluate achievements as regards my research goal to advance robot navigation towards utilizing externally supplied maps.

In a second study, mapping performance is investigated. This analyzes the robustness to localization in partially unknown environments and the reliable interplay of all functional components developed.

6.1 Implementation notes

SHRIMPS has been implemented according to the presentation in the previous Chapter. The system comprises some parametric models. Referring to a unit size of centimeters, the parameters have been used in the experiments (unless noted otherwise) are as follows:

parameter	value	description
Feature extraction:		
Intermediate DCE threshold:	5.0	Section 5.4.2, pp. 135
Grouping threshold:	10.0 [cm]	Section 5.4.1, pp. 134
Shape distance measure:		
Arc length vs. curvature	$\lambda_1 := 0.025$	Eq. 5.2, p. 142
Removal cost	$\lambda_2 := 2.5 \cdot 10^{-5}$	Eq. 5.3, p. 145
Matching:		
Ignorance penalty:	$\lambda_3 := 5000$	Eq. 5.6, p. 149
Alignment consistency measure:	$\lambda_4 := 1.0$, $\lambda_5 := 6.0$	Eq. 5.8, p. 152
Merging/ map update:		
Min. size of polylines to add:	10.0 [cm]	Section 5.8, p. 161

Suitable parameter values have been determined in experiments. The high distinctiveness of the shape distance measure eases determining suitable parameters, as the matching is not sensitive to variations.

SHRIMPS has been implemented in Macintosh Common Lisp 5.0; reported computing times refer to an Apple G5 dual 2.0Ghz[1] computer with 1GB of memory. Computing times are marked by a superscript ‘A’, e.g. 1:30$^{\mathrm{A}}$ minutes.

6.2 Case study self-localization

In this study, I evaluate self-localization performed by the functional components view extraction, correspondence determination, and alignment. The performance of my approach is compared against other approaches to self-localization documented in the literature. The study considers standard self-localization in a known environment as well as map-based localization using a schematic overview map.

Schematic maps are coarse overview maps that omit unnecessary details and simplify shapes and structures. Schematic maps have been considered to provide a suitable basis for interaction of humans and robots, in particular as regards robot instruction (Freksa et al., 2000b). Thus, schematic maps can be regarded to be an important representative of external maps, which robots should be able to utilize (cf. Section 1.1). This makes it interesting and relevant to study the adequacy of localization techniques to handle schematic maps. Experiments based on the schematic map address evaluation as regards my research goal to devise techniques for correspondence determination that are applicable to both, applications involving the internal map of the robot, and to applications involving an external, maybe coarse map.

[1] Only one processor is used by the Lisp system

Localization is evaluated in a simulated environment in which a virtual robot equipped with a LRF moves. Trajectories are determined and according sensor data is simulated. The data presents a 180° field of view and measurements are distorted using Gaussian noise $\mathcal{N}(0, 1)$, i.e. the true measurement is at a probability of 95.5% within a tolerance of ±2 cm. Experiments are performed in simulation to allow referring to ground truth in evaluation and to enable systematical modification of the experimental setup, thereby gaining a better insight in the capabilities of the methods.

To maintain the focus on spatial aspects, no stochastic models have been employed. I am aware that virtually all practical implementations of robot self-localization employ some stochastic model to address uncertainty and such techniques appear necessary to allow recovering from false decisions made. Here, only the the most plausible pose is determined instead of tracking a complete set of potential poses and updating belief states. This evaluates the performance of fundamental spatial information processing. The more reliable the single, most plausible pose can estimate the robot's true pose, the better an overall localization system including uncertainty models would work (cf. the discussion in Section 1.2). Moreover, incorporating a comprehensive uncertainty handling introduces additional, non-spatial processes and would conceal the ability to judge the performance of spatial representation and reasoning techniques to a certain extent. This makes this study both legitimate and interesting.

6.2.1 Evaluation criteria

Localization is performed for a sequence of observations along a trajectory through a test environment; the true map is provided to the localization methods. Differences between the true view pose and the view pose determined by a localizing method serve as the basis for evaluation. Differences are averaged for individual methods and compared. In addition, a proximity test is defined: whenever the localization is performed with a difference from the true pose of less than a fixed threshold, the proximity test is said to be passed. In localization experiments involving an accurate map of the environment, the threshold is set to 25° in orientation and 25 cm in position. In the experiment involving a schematic map, the threshold is relaxed, since schematic maps are coarse and differ from observation. Thus, localization with respect to a schematic map yields coarser information, too. The proximity test with respect to a schematic map utilizes a maximum difference in orientation of 45°, and of 50 cm in position.

6.2.2 Methods compared

For the experiments a representative selection of state-of-the-art localization methods utilizing different spatial representations and matching techniques has been selected. Their performance is compared against SHRIMPS' localization performance. The following methods have been chosen:

- Feature tracking based on histogram matching developed by Röfer (2002) and also employed by e.g. Jefferies et al. (2004c)

- Map-based localization by line matching described by Gutmann et al. (2001); it relates to approaches by Cox (1990); Gutmann (2000); Lu & Milios (1997)

- Iterative Closest Point (ICP) (Besl & McKay, 1992) used in connection with occupancy grids and employed by e.g. Hähnel et al. (2002); Thrun et al. (2000b)

These methods constitute a reasonable spectrum as regards different map representations and matching techniques. The first two approaches utilize extended features (lines) and the third is based on the very popular representation of occupancy grids (cf. Section 2.1.3).

Röfer's approach to feature tracking (Röfer, 2002) addresses localization by matching consecutive observations against each other. Such an approach to self-localization is referred to as feature tracking—it does not rely on an internal map. However, the technique to determine correlations of observations is identical to map-based approaches. In feature tracking, the previous observation is considered instead of an internal map. Röfer utilizes line features in his approach. Lines are extracted from LRF data using the generalization algorithm described in (Musto et al., 1999) (see also Section 3.4.1). On the level of configurational information, Röfer independently considers the orientation of the lines and their position. Orientation of lines is jointly represented by means of a histogram of directions. Similarly, histograms are employed for representing the x- and y-coordinates of the lines in a local coordinate system. Localization is performed by extracting lines from the LRF, by computing histograms, and by correlating histograms of the current and previous observation. In a first step, orientation histograms are correlated; this determines the robot's turning angle. Then, the robot's local coordinate system is rotated accordingly, aligning the current local coordinate system with the previous one in terms of orientation. In a second step, independent x- and y-histograms are correlated. The correlation yields the robot's translation. Both values are used to update the robot's pose.

Map-based localization by means of line models as originally developed by Cox (1990) utilizes a uniform map representation that represents line features in an absolute coordinate system. Gutmann (2000) extended Cox' algorithm by introducing a line detection for extracting lines from LRF data—the recursive split line fitting is employed (see Section 3.4.1). Gutmann also introduced a distance measure for line features, which simultaneously considers difference

setup	Fig.	average movement between scans	average rotation between scans	total distance traveled
simple environment (A)	6.1 (a)	11 mm	5.0°	24.32 m
simple environment (B)	6.1 (b)	974 mm	49.2°	22.40 m
complex environment	6.4	11 mm	4.0°	43.03 m
complex environment & schematized map	6.4	104 mm	30.6°	42.07 m

Table 6.1: Experimental setups used for evaluation of self-localization performance

in line orientation and difference in line position. The matching is performed indirectly by means of iterative alignment (see Section 3.5.4).

The third approach examined in this study is based on an occupancy grid representation, which is the most commonly used representation (cf. Thrun et al., 2005). To align observation and map, the Iterative Closest Point (ICP) algorithm is employed (cf. Section 3.5.4). Similarly to the line-based localization, matching and alignment in ICP is performed indirectly by iterative alignment (see Section 3.5.4). The fundamental difference between ICP and line-based localization according to Gutmann et al. (2001) lies in the underlying spatial representation.

6.2.3 Experiments & discussion

The experiments were conducted in three settings: a simple environment displaying only obstacles delimited by straight walls, a more complex environment derived from a floor map containing arbitrary-shaped obstacles, and a setting where localization in the complex environment has to be performed using a coarse schematic map that causes the actual robot perception and map to differ significantly.

The methods listed in Section 6.2.2 have been implemented according to the literature. Bins of width 2° are used for determination of rotation and 50 mm for position in the histogram-based localization. The resolution of 50 mm has been used as grid size for occupancy grids in ICP. Results are presented in the following Figures and discussed in the below.

In the results obtained for the simple test environment (A) (see Tab. 6.1, Fig. 6.1, Tab. 6.2, and Fig. 6.2), it can be observed that most methods correctly determined the robot's trajectory; the depicted trajectories are printed on top of the true trajectory. Only the histogram-based localization fails to resemble

simple environment (A):

Method	average deviation from true position [cm]	heading [°]	proximity test [%]
histogram	155.1	1.2	0
ICP	10.0	1.2	97
line-based	1.2	0.5	100
SHRIMPS	2.0	3.0	94

simple environment (B):

method	average deviation from true position [cm]	heading [°]	proximity test [%]
histogram	215.1	54	30
ICP	18.5	1.68	87
line-based	128.2	11.5	48
SHRIMPS	1.9	1.27	100

complex environment:

Method	average deviation from true position [cm]	heading [°]	proximity test [%]
histogram	240.0	26.4	8
ICP	53.4	1.5	68
line-based	516.7	65	4
SHRIMPS	14.4	1.21	87

schematic map:

method	average deviation from true position [cm]	heading [°]	relaxed proximity test [%]
histogram	212.8	26.5	14
ICP	223.4	25.9	50
line-based	183.6	16.6	30
SHRIMPS	55.3	3.3	86

Table 6.2: Tabular overview of localization results obtained

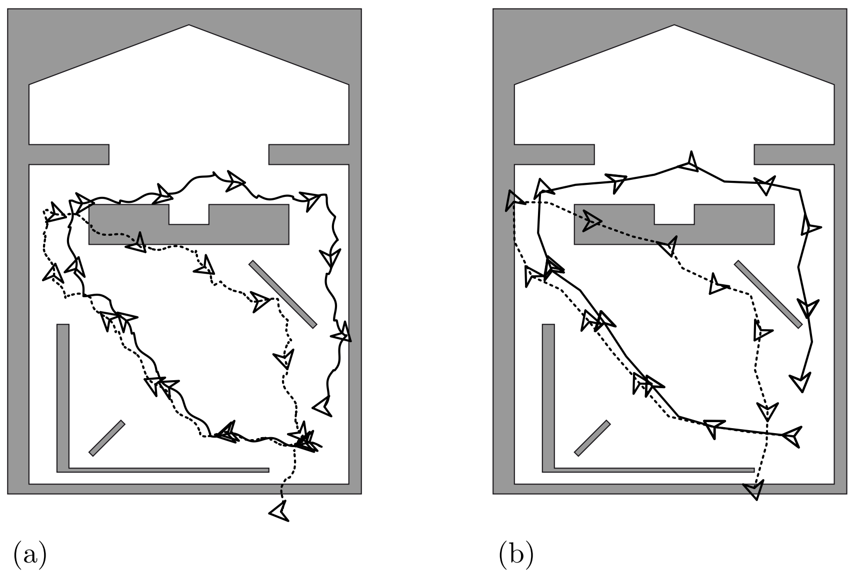


Figure 6.1: Experimental settings A (left) and B (right) for the simple environment of size 8 × 11 m. The robot's true path is depicted in black (triangles denote robot orientation), trajectory as reconstructed from simulated odometry readings is shown as dashed lines.

the robot's path. Investigating deeper into this failure, it can be observed that the method fails at places showing a high rotational symmetry like the two top corners of the robot's path.

The setup in the simple test environment (B) increases the distance traveled between successive observations. Larger errors in odometry accumulate, so this setting investigates the capability of methods to cope with unreliable pose estimates. The results (see Tab. 6.2 and Fig. 6.3) indicate that in contrast to setting (A) the line-based localization fails too. The estimated trajectory is only correct until the robot reaches the top left corner. At this point, a wrong correspondence of lines is determined, since lines are matched based on a distance function relying on a pose estimate, which is not reliable here. ICP can estimate the robot's trajectory correctly, except from cutting the top right corner. SHRIMPS handles the increased uncertainty without loss in localization quality.

In the settings using the complex environment depicted in Fig. 6.4, differences of localization results become more apparent (see Fig. 6.6). Both localization methods relying on detection of lines in the LRF data, namely the histogram-based tracking and the line-based matching, soon loose track of the correct path, estimating no more than 8% of the trajectory close enough to the true trajectory as regards the proximity test. At a first glance at Fig. 6.6

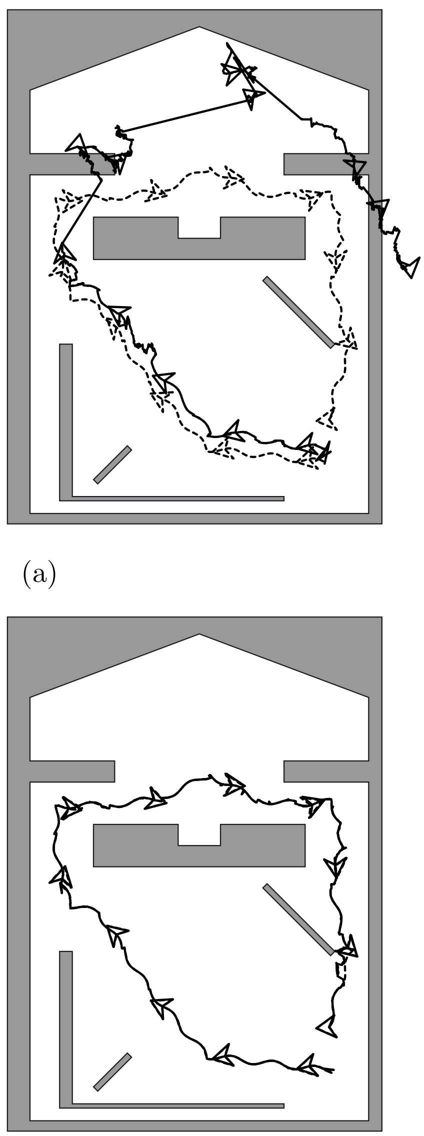

(a) (b) (c) (d)

Figure 6.2: Resulting localization for the simple test environment A (see Tab. 6.1) plotted against true path (dashed line). (a) Histogram-based localization, (b) ICP, (c) line-based, and (d) SHRIMPS

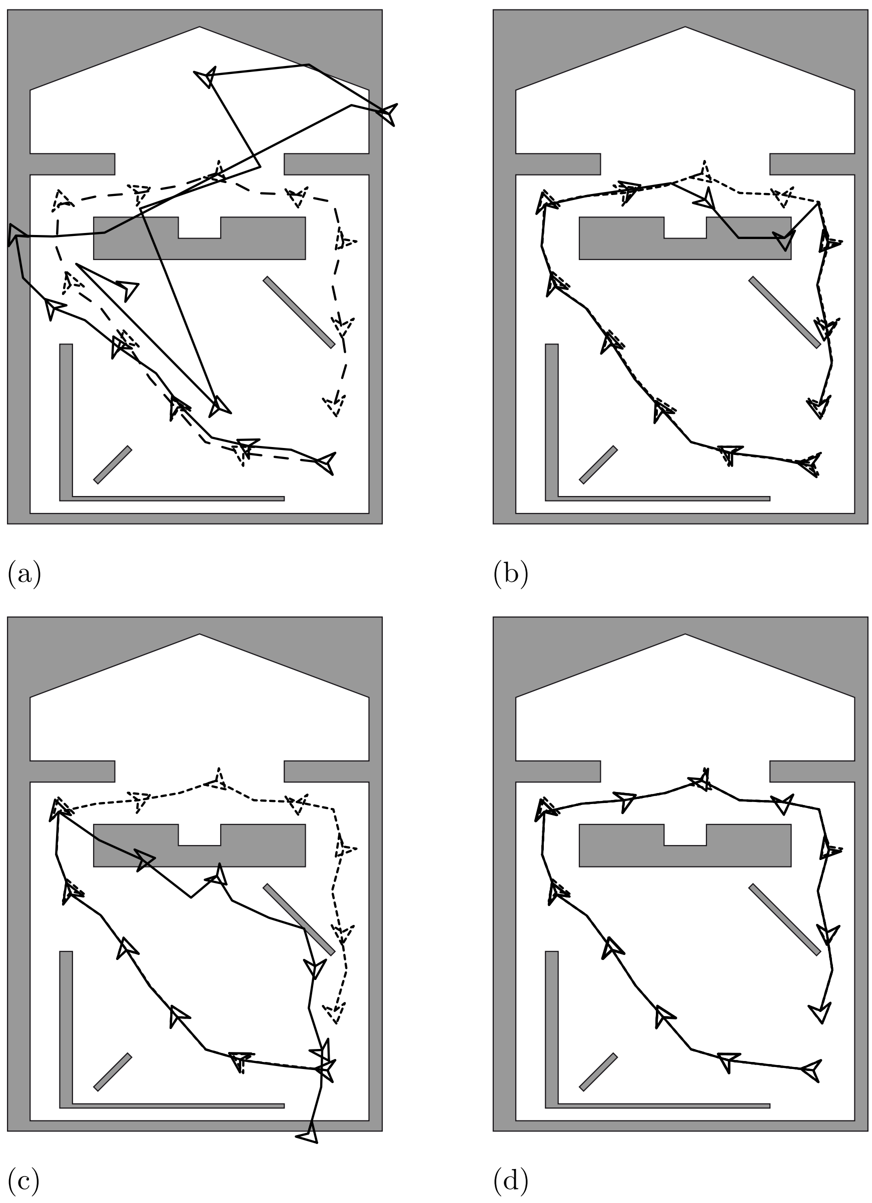
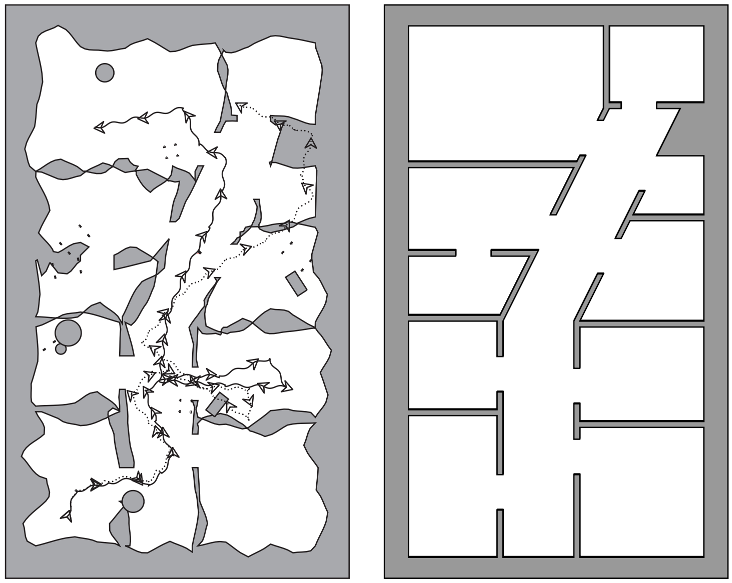


Figure 6.3: Resulting localization for the simple test environment B (see Tab. 6.1) plotted against true path (dashed lines). (a) Histogram-based localization, (b) ICP, (c) line-based, and (d) SHRIMPS



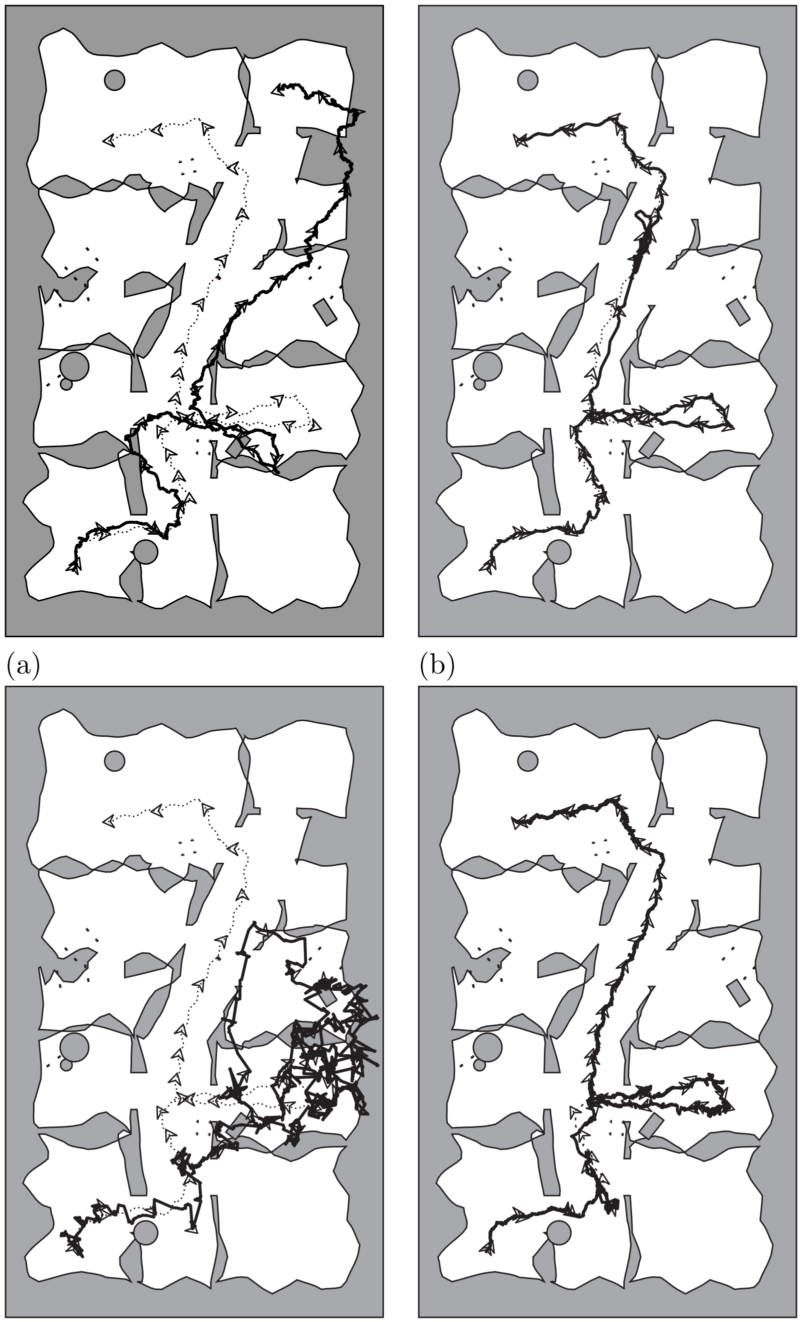
(a) (b)

Figure 6.4: Results of the experimental settings for the complex environment (see Tab. 6.2) of size 14 × 24 m. The environment is shown in (a), whereas (b) depicts the schematized map used in one experiment.

(b), ICP seems to resemble the robot's trajectory accurately. However, due to susceptibility for local minima in the iterative alignment of the ICP process, pose estimates often get stuck at locally optimal solutions. This results in an average deviation of about 220cm from the true trajectory. However, ICP recovers when the robot moves on further. After all, 68% of the estimated poses satisfy the proximity test. SHRIMPS achieves an average deviation from the true trajectory of about 55cm, and 87% of estimated poses satisfy the proximity condition.

Using LRF data corresponding to the complex environment, but providing a simplified, schematized map for localization instead of the true map, simulates wayfinding using an external overview map[2]. The schematic map has been derived from the environment similar to the process described by Barkowsky et al. (2000). Recall, in this setting the relaxed proximity test is applied, since LRF data and schematic map differ significantly. Due to the large differences between map and perception, most localization techniques fail. ICP meets the proximity constraint in 50% of the estimated poses, whereas SHRIMPS passes it in 86% of the cases. It estimates the path precisely until the robot enters the last room in the top-left corner. The freestanding column in the top-left room

[2]Using an truly external map, the scale may be unknown. Here, no scaling of the map is performed.



(a) (b) (c) (d)

Figure 6.5: Results obtained in the map-based localization experiment. Determined poses and true poses are plotted. (a) Histogram, (b) ICP, (c) line-based, and (d) SHRIMPS

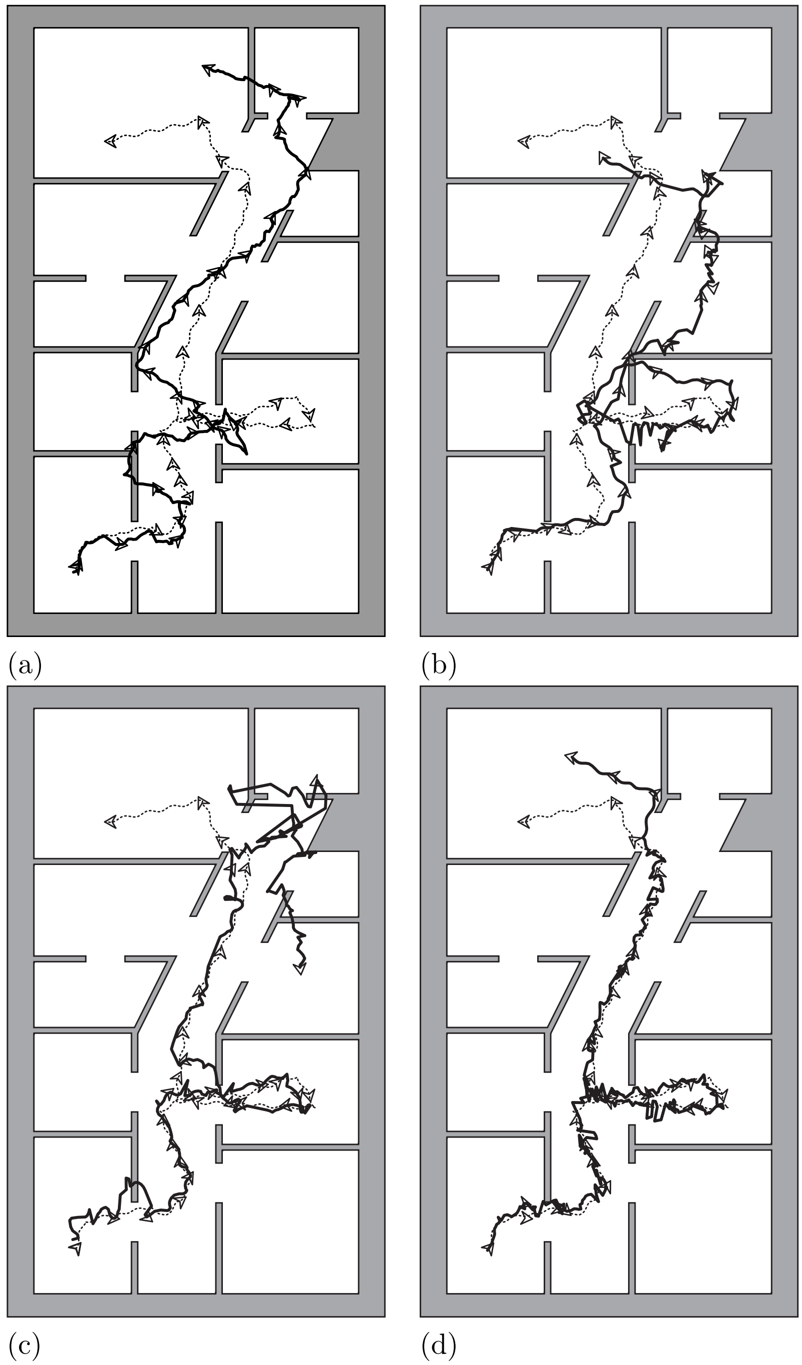


Figure 6.6: Results obtained in the localization experiment involving a schematic map. Determined poses and true poses are plotted. (a) histogram, (b) ICP, (c) line-based, and (d) SHRIMPS

is erroneously linked to the top wall, which yields an overall shape of the room that does not align with the schematic map and causes the deviation.

6.2.4 Discussion

Localization experiments have been carried out under conditions ranging from a simple, idealized environment to a realistic one, and in connection to a schematic map. In the simple environment (setup simple environment (A)), all techniques yield reasonable results. Merely, histogram-based matching exposed difficulties in handling views that present a high rotational symmetry. In real-world conditions, such constellation are unlikely, though. When used in environments presenting a dominant main direction and with good odometry, histogram-based pose tracking is likely to give good results. Increasing the travel distance between successive observations (setup simple environment (B)) increases the effect of accumulating odometry errors. Some approaches are particular sensitive to odometry errors and fail to handle these conditions. Histogram-based matching appears to be dependent on reliable odometry and a significant overlap of scans; line-based matching appears to be sensitive to odometry errors too. As there are only few lines, a single wrong correspondence has a strong adverse effect to the iterative alignment procedure, which appears to be getting stuck at sub-optimal solutions. In contrast, ICP, which considers correspondences on the level of individual points, can recover from such errors. In the realistic environment (setup complex environment) presenting hardly linear obstacle outlines, line-based techniques fail; ICP and SHRIMPS handle the setup reasonably. Evaluating the localization with respect to a schematic map, the difference in performance between ICP and SHRIMPS becomes more apparent. Considering the average differences between true trajectory and estimated one, it can be concluded that only SHRIMPS is able to master this setting: poses determined by shape-matching differ about 55 cm in average, whereas the pose estimated by ICP differs by more than 2m in average (see Table 6.2).

To sum up, the experimental evaluation demonstrated the applicability of SHRIMPS to standard map-based localization and to self-localization using a schematic map. The experiments highlight that SHRIMPS performs comparably well as often-used ICP-based localization relying on an internal map. In the case of self-localization using a schematic map, SHRIMPS is still able to robustly perform localization in most cases. The experiments support my conclusions on spatial representations and reasoning techniques: a universal representation of spatial information in combination with a sensible matching strategy provides the best means to master correspondence determination robustly. This substantiates the claim of this thesis to advance robot mapping performance by improving spatial information processing and to improve self-localization towards capabilities required for exploiting external maps.

Sensor data:	Simulated LRF data
Travel distance:	approx. 42 meter
# of scans:	400 (3840)
Odometry:	unavailable (only available to GMAPPING)
Average robot movement between scanning:	0.1 meter translation, 8.5° rotation (0.01 meter, 1.7°)

Table 6.3: Experimental setup of the simulated environment

6.3 Mapping experiments

In this Section, mapping experiments for evaluating SHRIMPS are described. Experiments comprise setups using simulated data and real-world data. The output of SHRIMPS is compared to the freely available GMAPPING software (Grisetti et al., 2005), which provides an advanced extension to the popular CARMEN software[3]. GMAPPING can be regarded to demarcate the state-of-the-art in stochastic-based mapping. It comprises an occupancy grid representation, ICP-based scan matching, and particle filters to simultaneously track different hypotheses of the robot's pose. GMAPPING has been installed on a Linux PC comprising a Pentium IV 2.66 Ghz processor and 512MB Ram. Computing times of this program are indicated by a superscript 'B' to indicate the referring computer clearly. GMAPPING is initialized using the parameter file "fr079.5cm.ini" distributed with the software.

6.3.1 Mapping with simulated sensor data

In this experiment, a synthetic indoor environment of approximately 6 × 7 meter is used (see Fig. 6.7). A simulated robot moves along the path marked in the image, collecting 400 laser scans similar to data as could be obtained by a SICK laser range finder. The experimental setup is summarized in Tab. 6.3. The data presents a 180° field of view and measurements are distorted using Gaussian noise $\mathcal{N}(0, 1)$, i.e. the true measurement is at a probability of 95.5% within a tolerance of ±2 cm. Simulated sensor readings have been processed by SHRIMPS and a map has been constructed (see Fig. 6.8); in the experiment no odometry data has been used. The processing time took approximately 2:15$^{\mathrm{A}}$ minutes. The same scans have also been processed by GMAPPING in 1:43$^{\mathrm{B}}$ minutes; the result is depicted in Fig. 6.9 (a). However, since GMAPPING requires odometry information, pose estimates for view poses

[3] Carmen—Carnegie Mellon Robot Navigation Toolkit is freely available from `http://carmen.sourceforge.net/`

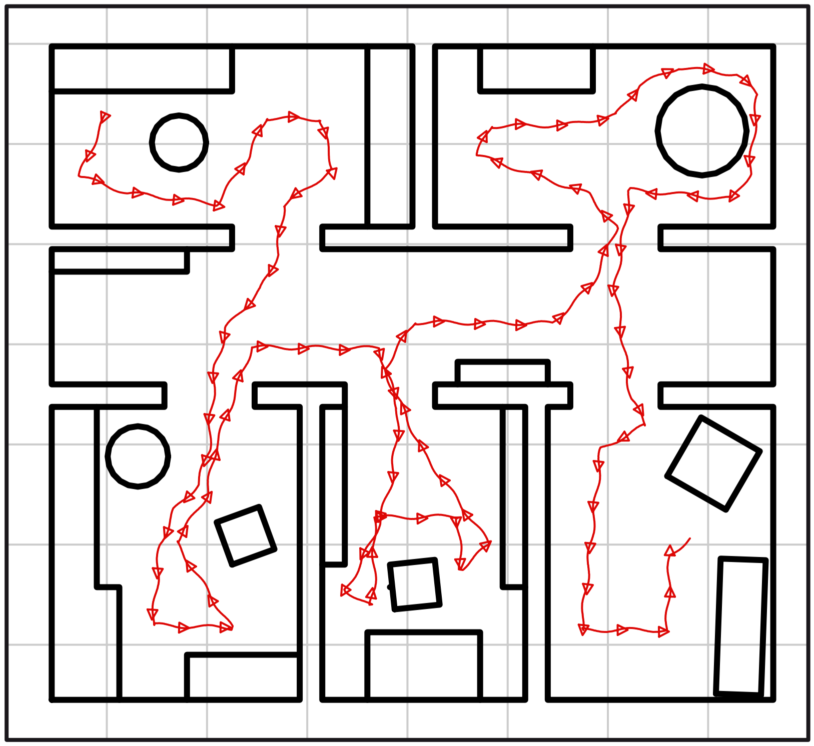


Figure 6.7: Setup of the mapping experiment using simulated sensor data, the environment measures approximately 6×7 meter. The grid denotes a distance of 1 meter. The simulated robot traversed the path and collected 400 scans in total.

have been supplied. Here, the true view poses are used. This means, by simply superimposing the scans according to their "estimated" view poses, a congruent map could be constructed. Of course, GMAPPING is not aware that true poses are available.

Another data set has been created comprising 3840 scans in total. Here, additional scans are determined for poses at equally spaced poses in between the view poses in the first set and odometry information has been added. The summarized setup is denoted in parentheses in Tab. 6.3. The data set has been processed by GMAPPING in approximately 6^{B} minutes. (see Fig. 6.9 (b)).

Discussion

In Fig. 6.8, the output of SHRIMPS is overlaid with the ground truth map. As can be observed, the determined map accurately resembles the environment. Unfortunately, the two round columns in the top left and top right corner show some artifacts of polylines put on top of each other; this has been caused by

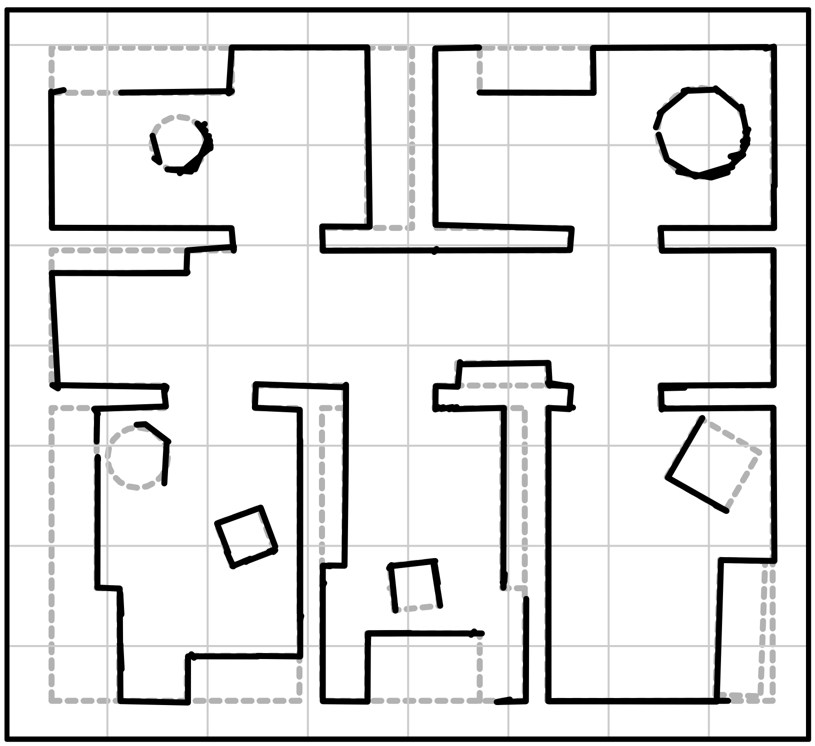
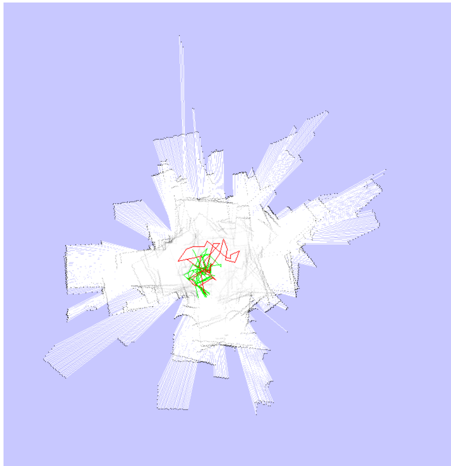
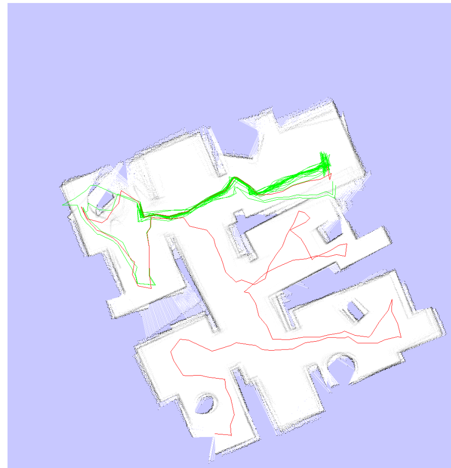


Figure 6.8: Results of the mapping obtained with simulated data. Map computed by SHRIMPS is superimposed on the ground truth map (dashed gray lines).



(a)



(b)

Figure 6.9: Results of GMAPPING for mapping the simulated data. (a) Processing the same 400 scans as processed by SHRIMPS. (b) Processing additional scans corresponding to intermediate scan poses to provide a strong overlap between consecutive scans; in total, 3840 scans are processed.

occasionally not matching a perceived feature to the map, thereby establishing a new map object. Further observations of these columns should allow the matching and merging components to recover from this situation by detecting the correspondence and merging the polylines together. In contrast to the upper columns, the round column in the bottom left room is already appropriately approximated by a single polyline spanning the part visible from the robot's path.

The same LRF data has been input to GMAPPING; the resulting map is depicted in Fig. 6.9 (a). It clearly indicates a mapping failure. The reason for this can be narrowed down to difficulties of processing consecutive scans, which do not present a nearly complete overlap. To obtain a comparable output, additional scans at intermediate view poses have been simulated. Despite the perfect odometry supplied, the resulting map depicted in Fig. 6.9 (b) is of poorer quality as the one obtained by SHRIMPS—several artifacts are apparent. To conclude, SHRIMPS can outperform state-of-the-art mapping with occupancy grids in a stochastic framework.

6.3.2 Mapping with real sensor data in a home environment

This experiment is based on real LRF data obtained in a home environment; home environments present one important representative of working environments for future service robots. The environment regarded in this experiment is

Sensor data:	Sick-LMS LRF mounted on a Pioneer-2 robot
Travel distance:	approx. 31 meter
# of scans:	41
Odometry:	available
Average robot movement between scanning:	0.75 meter translation, 56.6° rotation

Table 6.4: Experimental setup of the home environment

most likely an ordinary representative of its class and it is challenging in many regards. First, it is a crowded environment and most view poses only offer a strongly restricted view as obstacles block sight. Matching needs to handle the limited configuration information available and must reliably detect correspondences of the few objects in sight. Second, various small objects are spread all over the place and may only be visible to the sensor, if the sensor is located close to the obstacle. This results in unpredicted occlusion. Similarly, complex outlines of the obstacles (e.g. curtains, piled up clothing, ripped radiators) appear differently depending on the view pose and their appearance is particularly sensitive to any misalignment of the sensor. To correlate observations under these conditions correctly, careful determination of feature distance is essential. A third difficulty of this setup is a large movement of the robot between two consecutive scans, resulting in a small overlap of consecutive scans[4]. Many new objects can emerge from one scan to the other. Matching needs to reliably detect the few correspondences available and, at the same time, reliably disregard newly emerged features.

The experimental setup is summarized in Tab. 6.4; two consecutive scans are illustrated in Fig. 6.10. In this depiction, the scans are combined to one coordinate system using odometry. The Figure illustrates the small amount of overlap between consecutive scans. The scan depicted using black circles contains some false readings, possibly caused by disturbance of the sensor related to specific object surfaces.

In this experiment, two parameters of SHRIMPS have been changed:

Skip penalty:	$\lambda_3 := 10000$ (see Eq. 5.6 on page 149)
Grouping threshold:	5.0 [cm]

The penalty for not matching an object has been increased to respond to sig-

[4]Unfortunately, some sensor data has been corrupted during transmission from the robot to the computer recording the data. The unrecoverable loss of some sensor data caused differences in view pose between remaining scans of up to three meters; odometry information of reasonable quality is available.

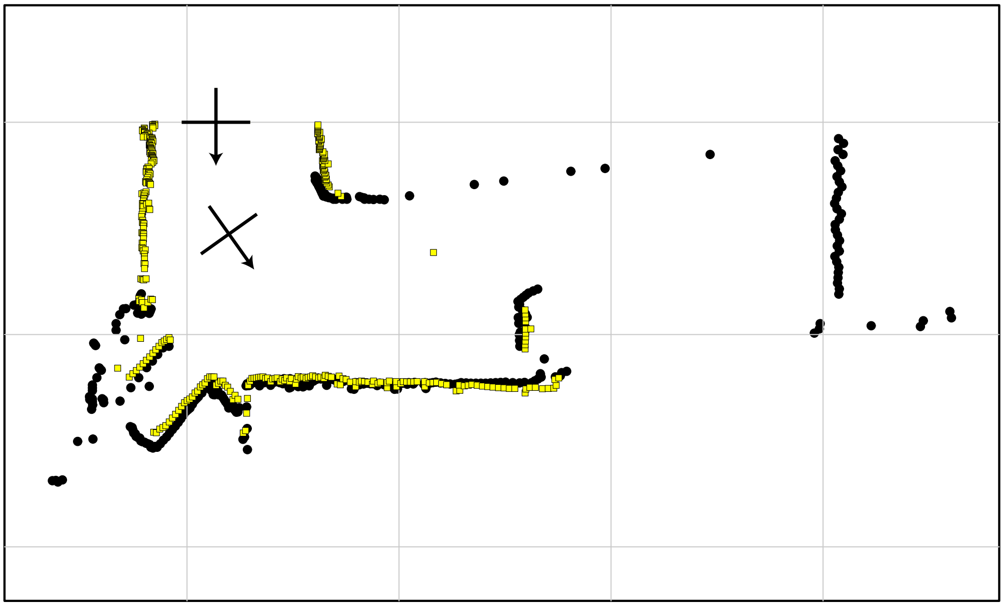


Figure 6.10: An overlay of two consecutive scans from the home environment according to odometry; scan points from one scan are depicted as boxes, scan points from the other scan as circles. View poses of the robot are indicated by cross hairs; the grid denotes a distance of 1 meter. The scan depicted by solid circles contains some measurement errors that do not correspond to any obstacle in the environment, e.g. the solitary scan points in the upper and lower left part of the image.

nificant differences in feature appearance. Using this parameter value, features appearing less similar can also be matched. The grouping threshold has been decreased, as there are many objects close to one another in this environment.

The map produced by SHRIMPS is depicted in Fig. 6.11; its computation took about 11^{A} seconds and it consists of 59 polylines with 238 points in total. The same sensor data is also used as input to GMAPPING, the corresponding output is depicted in Fig. 6.12; the computing time amounts to 41^{B} seconds.

Discussion

By visual impression, the map computed by SHRIMPS resembles the true environment; ground truth data to compare with is not available. This demonstrates that correspondences between consecutive scans have been robustly determined. Small artifacts resulting from undetected correspondences remain, though. For example, a corner of a polyline is overlaid with another polyline displaying the same shape (see fifth grid cell from the left, second from top in Fig. 6.11).

As regards the results obtained by GMAPPING, it can be observed in the

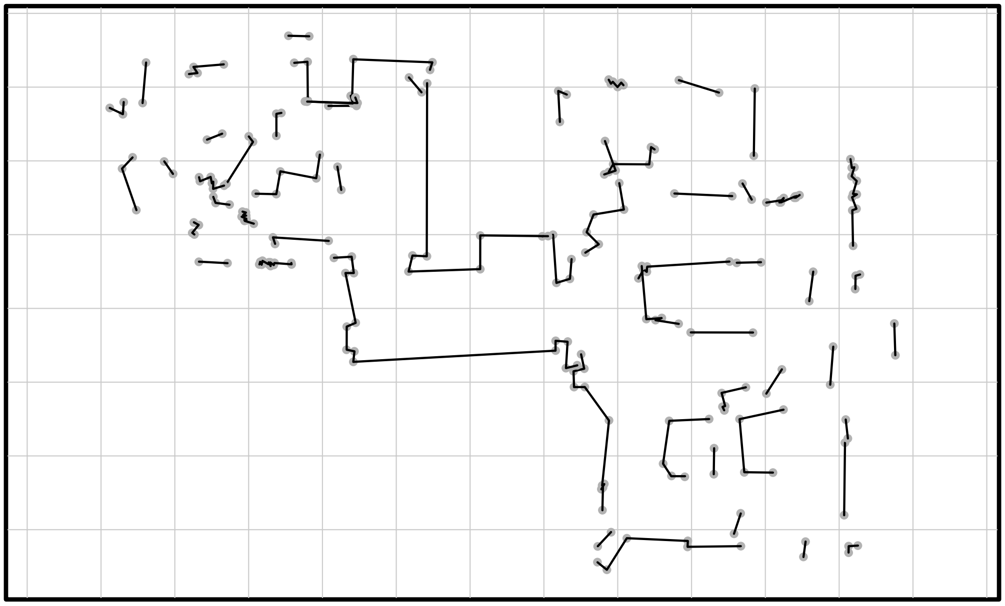


Figure 6.11: Map of the home environment computed by SHRIMPS.

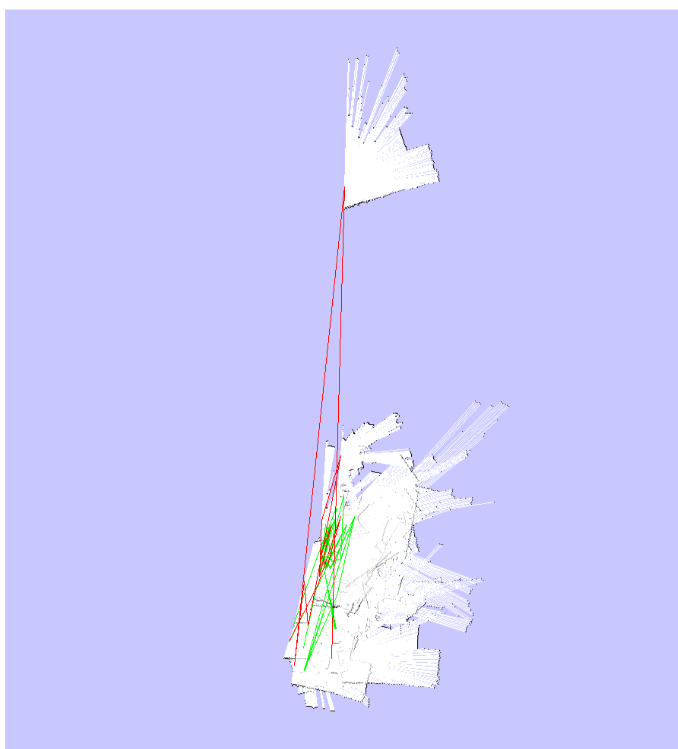


Figure 6.12: Map of the home environment computed by GMAPPING.

Fig. 6.12 that the data is not congruently integrated. A precise analysis of the failure of GMAPPING in this setup is not in the scope of my work, but the output of the system repeatedly acclaimed a failure of scan matching. GMAPPING appears unable to cope with sequences of laser scans that do not present very high degree of congruence, even though there are significant overlaps.

6.3.3 Data integration purely considering shape

In a third study, I return to the motivating example presented in the introduction of this dissertation. I discussed in the motivation that exploitation of spatial information is essential to robot mapping. In an example I demonstrated that assembling scans can be easy to humans, if a salient object can be identified on the scans. So far, no techniques in robot mapping can exclusively rely on spatial information, but are dependent on reliable odometry information, which provides suitable start estimates for data integration by iterative alignment. In the following, I detail the computation performed by SHRIMPS to process the four scans depicted in Fig. 1.2.

At the beginning, the internal map is empty and the first observation is stored in the map, identifying the origin of the local coordinate system with the origin of the map's coordinate system. When the second scan is processed by SHRIMPS, the map view is retrieved according to the previous robot pose, i.e. the origin—this yields the same view as provided by the first observation. Since no odometry is available, shape analysis is regarded to provide an estimate for the alignment of scans (see Section 5.6.1). The two views are matched disregarding the alignment consistency (cf. Eq. 5.9 on page 152), and the most reliable correspondence is determined (cf. Section 5.6.1). Based on this correspondence, the estimate of the alignment can be computed. Fig. 6.13 depicts these steps, image (a) shows the matching obtained (the most reliable correspondence is highlighted), and image (b) illustrates the estimated alignment by arranging the views accordingly. In the following steps, the actual matching is carried out, the observation is aligned to the map, corresponding observations are merged, and unmatched polylines in the observation are registered in the map. The actual matching is depicted in Fig. 6.13 (c), and the updated map is depicted in Fig. 6.13 (d). This two-phase matching procedure is repeated for the remaining observations, yielding the result presented in Fig. 6.14. As can be seen, the output is similar to the manually determined solution (see Fig. 1.3 on page 26).

6.3.4 Discussion

Experimental evaluation of mapping tasks indicates that SHRIMPS indeed provides adequate means to tackle the mapping problem. In simulated and real environments SHRIMPS has been demonstrated to provide good results. Fur-

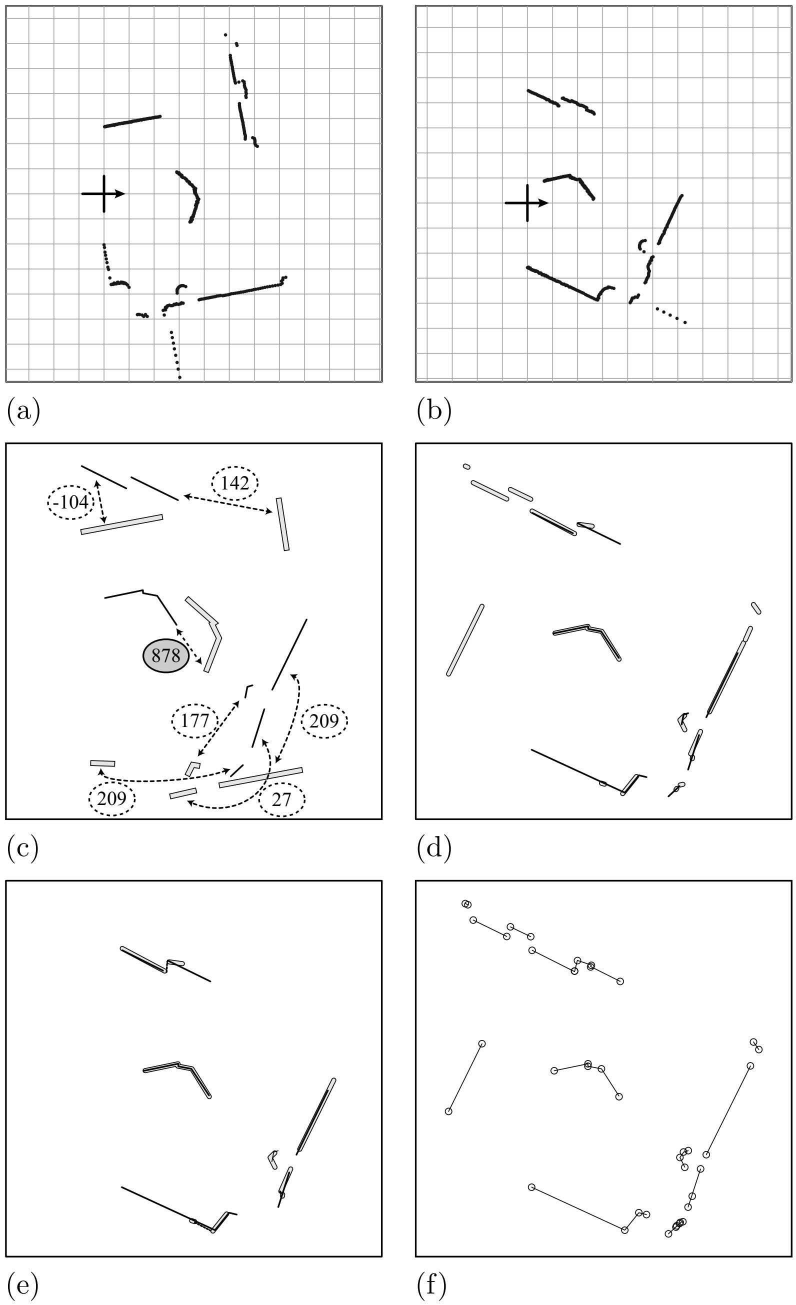


Figure 6.13: (a) Scan from Fig. 1.2-(a). (b) Scan from Fig. 1.2-(b). (c) First matching and correspondence reliability (only matched polylines are depicted). (d) Polylines aligned according the alignment derived from the most reliable correspondence. (e) Actual matching (only matched polylines are depicted), and (f) map after integrating the first two scans.

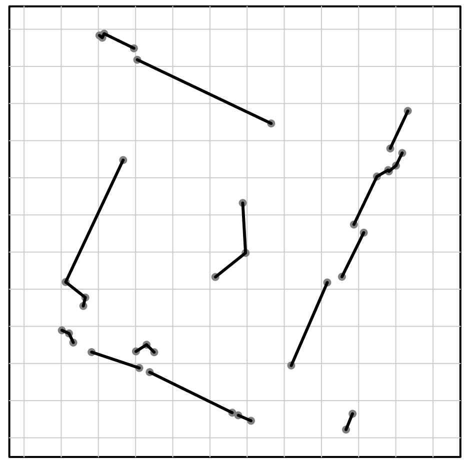


Figure 6.14: The four scans depicted in Fig. 1.2 (page 26) assembled by SHRIMPS, purely considering shape information.

thermore, it has been shown that SHRIMPS is potentially able to outperform state-of-the-art approaches to mapping that comprise a full-fledged stochastic model handling uncertainty, but less sophisticated methods for representing and reasoning about spatial information. Thus, spatial information processing is essential to the performance of mapping. It is thrilling future work (see Section 7.3.1) to integrate state-of-the-art uncertainty handling to spatial information processing in SHRIMPS and to evaluate the performance.

In SHRIMPS, there are several components that need to be supplemented for real-world application to address the full spectrum of difficulties, e.g. handling of uncertainty, implementation of loop closing techniques, etc. In its present form, the SHRIMP architecture presents adequate means to tackle the core problem, which is correspondence determination, even succeeding in unfavorable situations like when missing on odometry to provide position estimates. Dynamics in populated environment are not addressed in this thesis, but remain an important ingredient to real-world application. Object maps such as utilized in the proposed SHRIMP architecture are regarded to provide good means to approach this topic (cf. Thrun (2002) or see Section 2.1.4).

6.4 Summary & conclusion

In this Chapter, I have presented experiments for evaluating the performance of SHRIMPS. Experiments are performed for self-localization using internal and one external, coarse map and for mapping in unknown environments.

In the experiments on self-localization poses computed by SHRIMPS have been compared to results obtained by other techniques for localization. Stochastic frameworks for handling uncertainty have been masked out in the experiments. Tests have been performed in simulated environments to provide a ground truth. The evaluation shows that SHRIMPS performs equally well as other approaches in standard localization tasks. If pose estimates (odometry) are of poor quality, SHRIMPS maintains its self-localization performance. In a second study the performance of localization with respect to an external (coarse) schematic map has been examined. It turns out that SHRIMPS is able to master this task in contrast to all other methods analyzed.

In the experiments on mapping, SHRIMPS has been confronted with simulated and real-world environments. In both cases, the mapping procedure has performed well. However, to yield performance suitable for real applications, some techniques need to be introduced. Primarily, techniques for propagating accumulating pose uncertainty and techniques for detecting and addressing ambiguity in the matching are required. In principal, such techniques are known, but, nevertheless, their integration is a non-trivial task that exceeds the scope of my work.

To conclude, the experimental evaluation indicates that shape-based robot mapping is a suitable approach which in its present form already allows improving on state-of-the-art techniques. Adapting techniques to handle different hypotheses and to correct for accumulating localization and registration errors in cyclic paths presents itself as a promising step to achieve a comprehensive mapping system applicable to general real-world environments.

Chapter 7

Conclusion & outlook

In liehter varwe stat der walt,
der vogele schal nu donet,
div wunne ist worden manichvalt;
des meien tugende chronet
senide liebe; wer were alt,
da sih div çit so schonet?
her meie, iv ist der bris geçalt!
der winder si gehonet!

Carmina Burana, CB 138a

At this point, I summarize my approach, summarize the results achieved, and evaluate the contribution of this dissertation on shape-based robot mapping to current research. Additionally, I elaborate on further research directions that respond to questions raised in this work, and I discuss research tasks that benefit from the results of this dissertation.

7.1 Summary of the dissertation and its contribution

Outlining the robot mapping task, I have exposed its key challenges. Of those, this thesis addresses spatial representation and reasoning techniques underlying robot mapping. A sensible approach to robot mapping demands an adequate spatial representation and sophisticated reasoning techniques to tackle the central correspondence problem. One characteristic of robot mapping is the requirement of handling uncertain information and of mediating between conflicting knowledge, for instance between multiple, but differing observations of the same physical entity. Resolving conflicts involves ambiguities and complicates a definition of robot mapping in terms of a precise computational goal.

In this dissertation, the term plausibility in data integration is introduced to seize the computational goal of mapping. I have argued that plausible integration of observations must focus on spatial knowledge, attending more to abstract, confident knowledge than to uninterpreted estimates derived from uncertain sensor data. Matching techniques addressing the correspondence problem must be empowered to handle relaxed prerequisites on input data, e.g. inaccuracy of sensor data. This can be achieved by improving the spatial representation underlying robot mapping, and by advancing matching techniques. An exemplary task of integrating observations illustrated that it is essential to utilize spatial information inherent in the observations. By exploiting spatial knowledge, the influence of pose estimates derived from odometry can be reduced. This allows for a relaxation of requirements posed on the reliability of pose estimates in terms of availability as well as accuracy, ultimately leading to independency from odometry. As odometry information is often unreliable, gaining independence of it is a desirable goal. Robustness against poor pose estimates and the capability of mastering the correspondence problem by purely regarding spatial information inherent in the observations also supports utilization of external maps, e.g. floor plans.

A generalized interpretation of robot mapping is proposed in this dissertation, which covers a wider range of tasks than the typical interpretation of a single robot integrating information obtained from its own sensors. Robot mapping can naturally be extended to include a wider range of knowledge sources, such as external overview maps or communication with fellow robots. To advance robot mapping towards such a generalized interpretation, I have taken a first step by aiming at the design of spatial representation and reasoning techniques that allow a robot to relate its spatial knowledge to external maps.

The scenario covered in this dissertation is the acquisition of spatial knowledge in indoor environments using a high-quality range sensor (a laser range finder) as primary sensor. This setup is relevant to practical applications like service robots for home or office usage, while at the same time it puts the focus on intelligent information processing, as one can easily abstract from technical issues. The central claims of my thesis are that (a) improved spatial representation and reasoning techniques can be designed using shape analysis originating from the field of visual object recognition, and that (b) a robust and efficient matching algorithm can be designed based on shape analysis and spatial reasoning. To address my research questions, I have analyzed relevant approaches in two regards: by the representation of spatial information, and by the realization of functional components.

To analyze spatial representations, I have distinguished three layers: feature representation, configuration representation, and map organization. On the level of feature representation, I have linked approaches originating from the field of robotics to shape representation techniques related to visual object recognition. From my review, I have concluded that shape can constitute a well-founded map representation and I committed myself to forming such representation and developing reasoning processes that empower robot mapping on such basis. Polygonal lines that capture the boundary of navigable space can be employed as map features. They offer means to represent arbitrary environments compactly, and to enable mediating between sensor data and an object-centered representation, which is most adequate to higher level reasoning processes. Shape distance measures developed for retrieval applications in computer vision context can be transferred to the robot mapping domain and support identification of features in the matching. Generally, a universal representation of the boundary of navigable space supports linking the internal representation of a robot to external maps like floor plans.

Examining functional components, I have identified and analyzed the tasks view acquisition (feature extraction), matching, aligning (localization), and merging (map update). I have compared my view on robot mapping focusing on these functional components to a perspective centering on uncertainty handling and subordinating functional components to means of reasoning about uncertainty. Statistical techniques are widely employed to handle uncertain information. From evaluating properties of computationally tractable statistical frameworks, I have derived inherent complications of approaches subordinating spatial reasoning to a statistical framework. In response, I have argued for making the functional components dealing with spatial information the key point in robot mapping.

The central functional component is matching, which correlates observation and internal map, establishing correspondences on the level of features. Placing state-of-the-art techniques to matching in the context of graph theory and combinatorial optimization, I have advanced the problem formulation by introducing homomorphic matchings, which are mappings between two sets of features that adhere to constraints over features. By introducing constraints into the matching, confident, qualitative information can be modeled.

I have posed matching as the task of finding a homomorphic matching with respect to confident information about feature configurations that is optimal to feature-intrinsic information, such as feature similarity that models plausibility of feature correspondence. In terms of the developed graph matching technique, features are modeled as vertices and feature-intrinsic information is modeled as an edge weighting. Matching is posed as the task of identifying the correspondence relation of vertices that is optimal with respect to the edge weighting. By basing this new formulation of the correspondence problem on hypergraphs, n-to-m correspondences are expressible. This is an important step towards utilization of extended geometric primitives as map features, since multiple associations of a single feature appear necessary to address inescapable grouping differences in feature extraction. I have proposed exploitation of robust qualitative ordering of extended features in the matching by introducing it as side condition into the matching. I have showed that matchings which are homomorphic with respect to ordering information constitute a tractable problem class, and I have derived an algorithm to solve such problems. The algorithm relates graph matching to Dynamic Programming techniques originating from operations research.

Based on the conceived matching technique, I have developed a new approach to the correspondence problem that is based on a shape distance measure modeling shape similarity. The developed shape distance measure is especially tailored to robust recognition of simple shapes that lack of informative shape information. Polylines extracted from range finder data comprise a comparatively high level of noise as compared to shape information available. By encapsulating an existing, solid shape distance measure in a process that searches for supporting shape information in a noisy polyline, a new shape distance measure is obtained that allows for robust recognition under the challenging conditions faced in robot mapping. The shape distance measure provides decisive information to the matching, facilitating an efficient and robust solution to the correspondence problem, which is neither depended on the availability of pose estimates nor on a specific quality thereof. Moreover, the matching enables the feature extraction to be independent of parameters, as is can be adapted by the hypergraph matching and the shape distance measure.

Centering an incremental mapping architecture on the developed matching component, I have described a comprehensive computational model of shape-based mapping, termed SHRIMP. Besides matching, it comprises functional components for feature extraction, alignment (localization), and map update (merging). Each of these components is realized by bringing together approaches developed in the research areas of computer vision and robotics.

For obtaining a proof-of-concept, my approach is evaluated in experiments. The results achieved in a case study on self-localization demonstrate that the developed techniques can outperform state-of-the-art techniques. The capability of relating the robot's perception to an external, coarse floor plan is evaluated, showing that SHRIMP is empowered to relate spatial information across different levels of granularity. Mapping experiments demonstrate that SHRIMP allows for a robust integration of observations to a compact survey map based on shape information. To sum up, the experiments indicate that the shape-based map representation and the reasoning processes developed in this dissertation provide a sound basis for realizing an advanced robot mapping system.

Currently, the field of robot mapping is dominated by engineering approaches that make intense use of statistical filters. My approach differs from these approaches in that it is more abstract and cognitively motivated—it focuses on spatial representation and reasoning techniques. I acknowledge principles of visual similarity and of grouping rather than pursuing to determine optimal data integration with respect to a statistical model. Thus, my shape analysis techniques constitute an alternative to approaches developed. In the domain of robotics, explicit matching is often avoided by integrating matching with alignment to a combined, iterative algorithm that identifies physical entities mainly by their estimated position. In my approach, the correspondence problem is addressed explicitly. An analytical solution is proposed that combines cognitively motivated similarity of shape features with qualitative spatial information. The approach establishes a close connection between the research areas spatial cognition, robotics, and visual object recognition—this connection has not been thoroughly investigated yet and encourages future research. Challenges in robot mapping are substantially characterized by uncertainty in information available—uninterpreted use of sensor information is not possible, since for instance measurement errors accumulate and distort the mapping process increasingly. Uncertainty may be tackled in two ways: by aiming at engineering away uncertainty or by aiming at devising techniques robust to uncertainty. Robotics commonly adopts the first alternative and develops stochastic reasoning techniques that can shift uncertain data to a level of sufficient detail, given that real-world processes can be captured in a computationally tractable stochastic model. In contrast, my approach facilitates a robust handling of uncertain information by employing an adequate level of abstraction. In this dissertation I argued that robust spatial reasoning directly addresses uncertainty, whereas stochastic techniques provide means to address the effects of uninterpreted use of uncertain information. Since an informative interpretation of sensor data that is independent of uncertainty may not be achievable, but alternative interpretations may need to be evaluated, combining these complementary approaches is an important future research task.

Results achieved in this dissertation also has an impact outside the domain of robot mapping: the developed shape distance measure can be generalized (Latecki et al., 2005b) and is well-suited to tasks in visual object recognition. The matching algorithm is a new contribution to the formulation of the correspondence problem relevant to a wide range of navigational tasks, but can also be applied to more general matching tasks.

7.2 Evaluation of the achievements

To evaluate the achievements of this dissertation with respect to my research question, I reconsider my thesis and the initial claims raised in the introduction.

> *1. A connection exists between mapping using range information and visual object recognition on the level of shape information.*

Reviewing spatial representations underlying robot maps, I compared feature representation employed in robotics and shape representation techniques originating from visual object recognition. It turned out that spatial representations capturing the navigability of space have a counterpart in shape representation techniques. Occupancy grid representations are basically bitmap images and both scan matching and image retrieval techniques can be based on the Hausdorff distance for comparing representations. Roadmap representations represent routes extracted from range data similar to skeleton-based shape representations. Similar feature extraction processes are used too. Representations of the boundary of navigable space correspond to boundary-based shape representations, polylines can serve as a adequate representation in both domains—however: techniques operating on poylines like e.g. recognition processes have been researched in visual object recognition exclusively. My dissertation transfers state-of-the-art shape analysis on the basis of polylines to robot mapping and advances these techniques.

Shape information plays a central role in my approach to robot mapping. The spatial representation underlying SHRIMP is based on polylines which represent shape information. Shape analysis in terms of a shape distance measure is applied to approach shape similarity computationally, providing decisive information to the central matching component in SHRIMP.

Besides tackling the correspondence problem, shape analysis is also involved in aligning perception and map. In alignment, a connection between robotics and visual object recognition has been shown too, as similar alignment tasks are studied in the context of visual object recognition. To approach the merging problem of combining polylines that correspond to the same physical entity, shape transformation techniques have been adapted which originate from an object recognition application. These links which are realized in SHRIMP demonstrate the close connection of the research fields visual object recognition/ computer vision and robot mapping.

> *2. A spatial representation based on shape information is well-suited to robot mapping and navigation and it allows utilizing external maps.*

An object-centered representation can be regarded as the most adequate representation to robot mapping, as compact representations provide a solid foundation to efficient algorithms. Object maps provide good means to tackle robot mapping in its full generality, given that the employed primitives allow an adequate representation of any potential working environment (cf. Section 2.1.4). I respond to this discussion by employing universal geometric primitives based on shape information which are extractable from widely employed range finders. Polylines can be employed to represent the boundary of navigable space adequately and, hence, provide valuable information to navigation in a compact manner. The evaluation demonstrates that this approach is indeed adequate to robot mapping. It even provides solid means to extend today's robot capabilities towards desired communicational skills, as has been investigated in the self-localization experiments relating realistic simulation of perception to a schematic map that significantly differs from the true environment. The matching is challenged by relating fine-grained perception to a coarse, schematic map. In the SHRIMP architecture, matching relies on a robust feature distance measure and on robust qualitative ordering information about extended objects. This empowers SHRIMP to successful and robust self-localization.

> *3. Sophisticated shape analysis originating from the field of computer vision can be transferred to the robot mapping domain.*

Shape provides rich, distinctive information. It is of high importance to object recognition in the field of computer vision and can be regarded as the most informative single attribute of an object with respect to recognizing it. Sophisticated shape analysis techniques originating in this area laid the foundation to develop the sensitive shape distance measure that is particularly robust to noise and allows even vague shape information to be exploited. Evaluating shape complexity and shape similarity allows the matching to adhere from unreliable odometry information.

> *4. An analytical, efficient, and robust approach to the correspondence problem can be designed on the basis of spatial reasoning and shape analysis.*

The matching technique utilized in the developed SHRIMP approach relies on theoretical foundations of balanced hypergraph matching that can be expressed as an extended Dynamic Programming task. The distinctive shape distance measure can be assumed to meet the requirements of applying the theoretical results for determining an optimal correspondence with respect to the computationally modeled plausibility of data integration. Distinctiveness of the shape distance measure essentially influences the matching, though in first matter not in terms of computational complexity but to disregard not plausible feature associations. However, upon a closer look application of my theoretical results requires means to reliably detect correspondences based on feature-intrinsic properties. In principle, shape distance measures can fulfill this premise. As an ultimate consequence, cost-optimal n-to-m matching can be performed by efficient analytical means which can be applied to online mapping tasks. Thus, shape distance significantly contributes to efficiency of the matching as well. The compactness of a shape-based representation, i.e. just few polylines are sufficient to represent a typical perception, is another ingredient to efficient matching. Decreasing the overall amount of features to associate cuts down the overall computational cost as well.

> *5. Sophisticated matching strategies substantially attack the correspondence problem and allow for robust self-localization in context of relaxed requirements on input data. In particular, the absence of odometry information can be mastered.*

The experimental evaluation, in particular the case study on self-localization using an external schematic map indicates that SHRIMP achieves robust localization. Experimental settings not providing odometry information are mastered by SHRIMP reliably, which is a significant improvement over existing approaches. Robot mapping is the task of determining the most *plausible* map given observations made.

Computational modeling of plausibility means describing common knowledge about space in terms of computational models, which is a central objective of spatial cognition research (Freksa, 2004). Learning and incorporating models that reflect a human's understanding of plausible maps can help to improve robot mapping towards human capabilities while at the same time making the robot's performance more transparent to human users. Besides computationally modeling the characteristics of plausible data integration, spatial cognition is involved with qualitative reasoning techniques that I have introduced in the robot mapping task, namely by describing homomorphic matching with respect to the circular order of extended objects. The qualitative constraints empower an efficient analytical approach to the correspondence problem. It remains an challenging research question, which additional spatial relations provide solid means to enhance matching algorithms in terms of tractable problem classes.

7.3 Looking ahead

This dissertation presents an alternative approach to robot mapping, focusing on the design of a shape-based spatial representation and on the development of reasoning techniques, in particular with respect to an analytical and robust approach to the correspondence problem. Interesting research questions have been raised that address potential enhancements and complements. Additionally, some current research tasks receive new input from the results achieved in this work.

In the field of robot mapping several questions remain to be answered for reaching the ultimate goal of a general, fully autonomous mapping by mobile robots that is suitable to a wide range of applications and that also empowers a robot to communicate with fellow robots or humans about its surroundings. The evaluation of SHRIMP indicates that the developed techniques provide solid means for further approaching this goal. In the following, I discuss some particular relevant and promising future steps.

7.3.1 Addressing ambiguities introduced by uncertainty

Challenges of robot mapping are characterized by inescapable uncertainty in sensor data. This dissertation addresses uncertainty by devising spatial reasoning techniques that are robust to uncertain information. As has been discussed in Section 1.2, this is an alternative to approaches primarily aiming at reducing uncertainty by means of stochastic reasoning frameworks. These alternatives are not competing, but complement one another. Uninterpreted utilization of uncertain information introduces conflicts. Resolving these conflicts introduces ambiguities that can be handled by stochastic reasoning frameworks. Sensible interpretation of spatial information already avoids some conflicts and thereby reduces ambiguity, as fewer alternatives appear plausible in the context of a more comprehensive data analysis. Combining the advanced spatial reasoning techniques developed in this dissertation with state-of-the-art techniques for handling ambiguities introduced by interpreting uncertain information, a significant improvement of robustness in robot mapping appears to be possible. This would demarcate one important step towards suitability for reliable real-world application. In the following, I briefly discuss a few starting points for combining these alternative approaches.

As I have argued, a fully stochastic approach to mapping that subordinates spatial information processing to a computationally tractable stochastic model appears questionable to provide a suitable basis. Stochastically modeling all dimensions of uncertainty is unlikely to be possible. It appears more appropriate to me, introducing stochastic processes for handling ambiguities into the distinct functional components of robot mapping.

Stochastic reasoning provides a solid basis to correct for measurement noise. Given that a system can be described by linear process models, the extended Kalman filter provides sound means to propagate and to reason about Gaussian noise. The main deficit of approaches relying on Kalman filters is that the correspondence of observed features needs to be known with certainty. A straight-forward matching algorithm, e.g. using nearest neighbor techniques, does often not provide a suitable basis for application of Kalman filters (cf. Neira & Tardós, 2001). The matching techniques developed in this dissertation empower a robust correspondence determination that can enable the application of Kalman filter techniques. A second complicating aspect of Kalman filters is their efficiency that can exceed time constraints for online mapping. Kalman filters bear a quadratic computational complexity of the filter update with respect to the number of features in the map. Decreasing the amount of features registered in a map (e.g. by advancing the map representation to extended geometric primitives like polylines) significantly decreases the overall computational cost. Only few distinct polylines are required to represent comparatively complex environments, as individual polylines are very expressive.

As a consequence of interpreting uncertain sensor data, any interpreted information must also be considered uncertain information. In its ultimate consequence regarding the proposed SHRIMP architecture, this means that the result of the matching is uncertain too. For handling residual uncertainty in the matching, multi-hypothesis tracking appears applicable, which is closely related to particle filters. Regarding shape similarity of related polylines and regarding the degree of compatibility of related configurations can serve as a basis for evaluating the likelihood individual of hypotheses about alternative matching outcomes. This information can be derived from the central matching equation underlying SHRIMP (Eq. 5.9 on p. 152), considering the value the right-hand side expression yields for a specific correspondence relation. The outlined approach decomposes the overall uncertainty in sensor data into uncertainty in the matching and uncertainty of measured positions, which is referred to as the technique of Rao-Blackwellization (cf. Section 3.1).

Unfortunately, deriving a sound computational modeling is far from being trivial. It appears challenging to derive a model of residual uncertainty in extracted polylines on the basis of sensor data, even if a noise model of the sensor is known. Techniques need to be devised for propagating uncertainty in polylines along matching, alignment, and merging. To track multiple hypotheses stemming from uncertain matching, the matching algorithm would need to be extended from computing only the most plausible correspondence to computing all correspondences that appear sufficiently plausible. These correspondences need to be processed individually and their consequences need to be tracked. As hypotheses branch, a naive approach would lead to combinatorial explosion. So, sophisticated techniques to accomplish suitable hypotheses tracking need to be devised.

In summary, directly addressing uncertainty in perception and map is advantageous to master real-world problems robustly which is required for realizing practical applications. Devising suitable techniques for addressing uncertainty in SHRIMP can rely on extensive research on stochastic modeling for robot mapping, but this step appears not to be a straight-forward application of existent techniques. Instead, introducing methods for handling uncertainty into SHRIMP is more likely a challenging endeavor. Though, I am convinced that this step appears necessary to advance robot mapping to real-world applications, while at the same time allowing for integration of abstract reasoning to robots that rely on an object-centered representation as is offered by SHRIMP and which appear necessary for synthesizing versatile service robots.

7.3.2 Map-based communication

Often, robots should not act purely autonomously, but interact with humans as well. Most importantly, robots need to be enabled to receive orders from humans. Robot instruction involves communication of spatial information. In principal, there are many ways to establish such a communication. Though it seems natural that the methods that govern human-human interaction should also be applied to interaction with robots. In first matter, natural language serves this purpose in human-human interaction. Even though it is a desirable long-term goal to set up communication of humans with robots in natural language, this is a research topic in its own right, going well beyond natural language understanding (see e.g. Fischer & Moratz, 2001; Ligozat, 2000; Moratz & Tenbrink, 2003). Apart from general difficulties in natural language processing, the design of a natural language interface still poses unanswered research questions, as human users are unsure about how to address a specific robot, about which reference system to use, and what kind of commands (e.g. step-by-step instructions vs. goal-based instructions) to choose (Fischer & Moratz, 2001; Moratz & Tenbrink, 2003).

Maps also provide suitable and established means of communication but appear easier to handle; Freksa et al. (2000a) argued for utilization of schematic maps for robot instruction. One can communicate spatial information by indicating locations on a map. For example, to instruct a service robot to move to a specific place, the target location can be selected on an input device displaying a map.

Some foundations have already been provided by this work. The proposed matching provides means to handle schematic maps with respect to localization. The developed shape distance measure does not rely on a fixed level of granularity but allows to bridge between fine-grained perception and coarse map information. We have already outlined some techniques that allow a schematic map to be interpreted in terms of executable robot commands (Wolter & Richter, 2004). However, these are just two aspects in the overall task. Schematic maps commonly employed in human-human interaction comprise a rich repertoire of symbols or spatial relations which need to be adequately interpreted by the robot. Beyond instructing a robot to move to a specific location, the robot should also be able to use the external map as basis for its internal map, supplementing and refining it where more detailed information is available through observation.

7.3.3 Multi-robot mapping

Multi-robot mapping can be regarded to be a special case of map-based communication that is of practical importance. Robots communicate their individual maps to gather a joint, comprehensive view. In principle, multiple robots can more efficiently explore environments and faster provide map information to human users. For example, in robot rescue scenarios robots are employed to gather information in order to guide human rescuers. So far, multi-robot mapping has been extensively studied in situations where the relative start pose of the individual robots are known or can be derived by the robots observing one another. Knowledge about relative start positions eases the problem and allows to extend existing single-robot approaches (see Thrun, 2001). However, requiring this knowledge can be a severe limitation in some situation. For example, all robots would need to start from either known places or the robots need to be simultaneously present to observe one another.

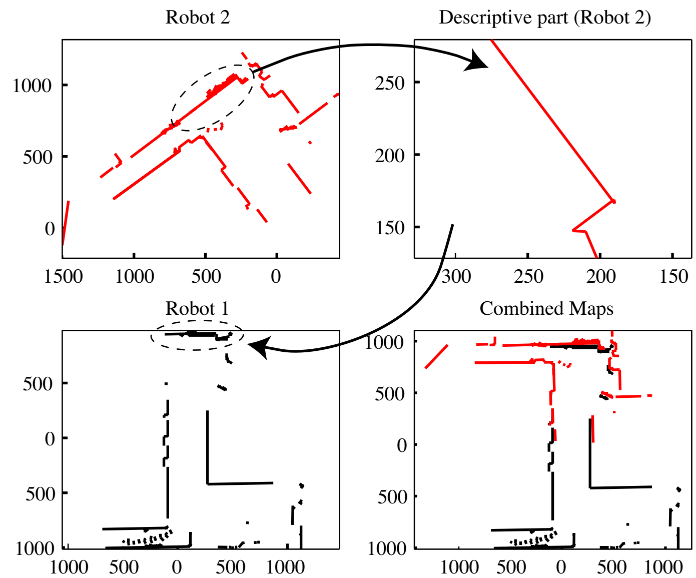


Figure 7.1: Exemplary result of shape-based map merging. To the left (top and bottom image), two partial maps constructed independently by two robots are depicted. From one map, a descriptive shape is extracted (top right image). A similar shape is searched for in the other robot's map. Based on the correspondence found, both partial maps can be merged (bottom right image).

To address multi-robot mapping generally, one requires means to correlate partially constructed maps. In other words, matching needs to be extended from determination of observation-to-map correspondences to map-to-map correspondences. Determination of map-to-map correspondences and integration of corresponding parts is commonly referred to as *map merging* or *map fusion*. Extending matching techniques results in a dramatic increase of the computational burden, as there many more potential correspondences to consider and no estimate of map alignment is available like it is in incremental mapping. Therefore, map merging is a hard and yet unsolved problem. Importance of a sophisticated feature handling in merging maps has been stressed by Konolige et al. (2003).

In this spirit, shape-based object maps can be regarded to provide solid means to tackle the problem. In a proof-of-concept study we evaluated the utility of shape information (Latecki et al., 2005a). The central recognition algorithm is based on shape analysis, in particular on shape similarity consideration by means of a shape distance measure and on shape complexity. The idea behind shape-based map merging is to define a cognitively motivated search strategy that overcomes the immense search space. The set of potential correspondences is traversed by starting with the most salient shapes. There are typically only few potential correspondence partners for a salient object that need to be inspected in detail. This allows to limit the overall computational cost.

First experimental results obtained by the outlined idea are illustrated in Fig. 7.1. The experiments indicate that the approach is promising, but, many open research questions remain. Most importantly, given that a—in some sense—plausible correspondence of polylines has been determined, sensible means to align the remaining map need to be formulated that take into account residual uncertainty in the map. For example, banana-shaped maps of straight corridors are a typical phenomenon of incremental mapping. Even though the most plausible map may not resemble the true environment well, estimates about the residual uncertainty may exist. Knowledge about uncertainty allows the robot to agree to a map presenting a different appearance of the same corridor, given that sensible techniques to relate the two representations exists. However, aligning two maps containing differently bend corridors in any naive manner would not allow detection this correspondence. To develop suitable techniques that allow determination of correspondences in an efficient manner pose great challenges. Relating uncertain information in terms of coarser qualitative spatial relations and advancing on exploitation of qualitative spatial information in matching processes could be a promising starting point.

7.3.4 Advancing on matching techniques

Introducing qualitative information into the matching facilitated a robust and efficient approach to the correspondence problem. Ordering information as utilized in SHRIMP is unlikely to be the only source of confident knowledge valuable to exploit though. Ordering information is handy due to the simplicity of handling binary constraints which can also be utilized in the matching frameworks using association graphs. However, many spatial information is expressed in terms of ternary relations, relating one objects in relation to the relative position of other objects (Freksa & Röhrig, 1993). Ternary qualitative calculi can also present valuable spatial information that could and should be exploited in terms of interpretation as confident knowledge to improve modeling of plausible data integration, while at the same time facilitating an efficient matching. It is yet unclear how such ternary calculi can most appropriately be integrated into the matching.

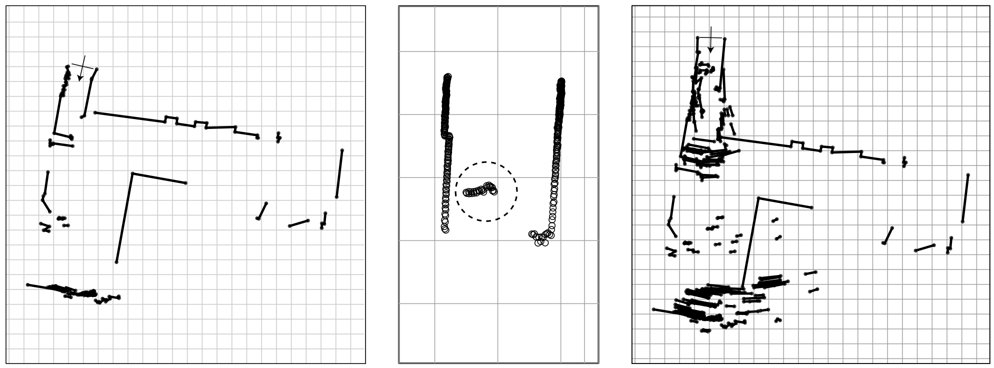
Matching has been advanced directly to address different outcomes of the grouping process by introducing n-to-m matches whereby sequences of features are re-grouped and matched. Joining of features currently cannot be performed if two features relate to a single physical entity but are separated in the view due to occlusion, i.e. an additional feature between the two other ones is detected. To address the full bandwidth of occlusion, a more general approach to mapping needs to developed. Such approach would allow unanticipated occlusion to be handled. However, it could turn out that extending the matching increases computational complexity beyond limits of feasibility. In my proposed approach, occlusion is primarily addressed by anticipating the occlusion. When retrieving a view from the map, polylines are trimmed according to the occlusion expected. Expectation is derived from an pose estimate by retrieving the anticipated view from the internal map. If the map view resembles the observation, the occlusion is handled by the robust matching and the map update. However, if a robot that does not provide odometry measurements moves a large distance between sensing, visibility can change dramatically. A similar situation occurs, if the environment is altered by placing a new object in front of another, partially occluding the rear one. In such situations, the developed matching technique is not capable of deriving the interpretation that an unexpectedly emerged object partially occludes another.

Besides these practical considerations it appears necessary further to explore the theoretical properties of the matching, relating the developed techniques to current research in graph theory and combinatorial optimization. Bipartite graph matching is known to be closely connected to network flow problems (Loncaric, 1998): can this relation be generalized to accommodate matching in generalized hypergraphs? If so, what are the implications?

To conclude, there are still important research questions to answer with respect to a deep understanding of matching, its computational modeling, and its complexity.

7.3.5 Handling dynamics

Real-world environments often display dynamics, as people or other robots are moving around. Not addressing dynamics in robot mapping renders any application to general real world environments impossible. Unfortunately, handling dynamics is a very hard problem and has not been solved yet. Common techniques for handling dynamics aim at filtering out moving objects (e.g. Fox et al., 1999) from the sensor data. A severe complication of dynamic environments is their effect on stochastic techniques for handling uncertain information: the recursive Bayes filter requires maps to be constant in time in order to derive a computational approach (cf. Section 3.1). However, this is no longer the case in dynamic environments. Approaches relying on object maps are claimed to provide a suitable basis for addressing dynamics, if features in the map correspond to moving real-world objects—see Thrun (2002) or refer to the discussion in Section 2.1.4. The present dissertation provides a map representation that meets this condition. A clue to handling dynamics is that, if map objects and physical entities in the environment correspond, one could assign motion models to map features to estimate their position in future observations. At a first glance, this appears to be an uncomplicated extension to mapping procedures



(a) (b) (c)

Figure 7.2: (a) Map computed by SHRIMPS, the current robot pose is denoted by the crosshairs. The grid denotes a distance of 1 meter. (b) A person passes by the robot (marked area). (c) The maps becomes corrupt, if dynamics are not handled.

relying on object maps, but determination and handling of motion models does not constitute the true problem. Rather, moving people appear different in sensor readings while they are moving. The challenge is to track an object, whose appearance is changing. This requires to balance the in matching between congruence of expected position and congruence of shape. By clearly differentiating these factors in the matching (see Eq. 5.9), the developed techniques can provide a solid foundation.

Most importantly, the classification of an object not to move must be reliable. If, for example, a robot would relate its position to an object erroneously regarded as a static one, then the robot would infer to be moving by itself with the same speed as the observed object truly moves. As a result, the mapping procedure is significantly disturbed. This is exemplified in Fig. 7.2. Image (a) depicts a map autonomously constructed by SHRIMPS from 845 scans recorded at the Universität Bremen by the autonomous wheelchair ROLLAND[1] (Lankenau & Röfer, 2001). At this point, a human passes by the robot (image (b)) and, by integrating few successive scans such that they are aligned to the moving object, the map is distorted (image (c)).

Interpretation of dynamics in the presence of uncertainty is particularly difficult. Observe that state-of-the-art techniques for handling uncertainty rely on a static world assumption (see Section 3.1.2) and, hence, cannot be applied to dynamic environments.

[1] The robot is equipped with a LRF facing backwards. Sensor data has been kindly supplied by Thomas Röfer.

7.3.6 Retrieving shape information from 3D sensor data

Three-dimensional, fine-grained spatial knowledge provides rich, distinctive information that can be exploited to disambiguate places. For example, if a robot equipped with a single laser range finder moves around a cafeteria, it perceives mainly legs of chairs and tables—similar to a human moving in the woods. Perceptions crowded with similar objects complicate matching and can even hinder a reliable operation. However, if the robot would be granted a 3D view of its surrounding it could exploit more salient, discriminative features of the environment, e.g. memorize the outline and position of table tops rather than the position of individual legs. Such 3D views can be acquired using rotating laser range finders (Surmann et al., 2003). The discussed example suggests that information derived from 3D data can be valuable, and it also demonstrates that the information required is often still of 2D nature. Dependable feature extraction can therefore aim at detecting suitable 2D shapes in 3D data. Such approach has two advantages: first, sound techniques for 2D mapping can be extended to benefit from richer 3D information to reduce ambiguity. Second, the resulting representation remains compact and does not require to represent and interrelate complex 3D configurations.

7.4 Closing remarks

At a first sight it is remarkable, yet even surprising, that a problem that specific and that closely connected to practical applications as robot mapping is keeping a research community occupied for more than two decades. However, investigating deeper into robot mapping, it becomes apparent that robot mapping is by far not a concrete technical challenge, but it involves several fundamental research problems: designing adequate representations of real-world environments, developing means for communication, and devising techniques for reasoning about spatial, temporal, and uncertain information, to name but a few. Truly mastering robot mapping requires solutions to these fundamental problems,which are addressed in various disciplines related to Artificial Intelligence research. In particular with respect to tasks involving interaction with humans, further disciplines contribute too.

Robot mapping is one central problem of building truly autonomous robots. It is no self-contained endeavor, but it is thoroughly animated by the tasks a robot needs to carry out and which rely on the map. As a complicating fact, many potential robot applications that require autonomous acquisition of maps are not yet explored, let alone understood. As of today, successful mobile robot applications remain restricted to fairly constrained scenarios. Helpful, versatile service robots need to master arbitrary environments, though. Maintaining a usable map is difficult in many regards and requires consideration of objects frequently changing their position, of appearance of new objects, and of other objects vanishing. On top of that, handling dynamics is unavoidable. Versatile service robots that meet these requirements are desired and such robots have fired the imagination of science fiction authors, but their realization still remains largely unsolved and involves techniques yet unknown.

Thus, there is more to robot mapping than the concrete technical challenge. Understanding how a suitable representation of one's surroundings can be learned is one fundamental research problem. Robot mapping is closely connected with potential applications and needs to be considered in the context of these applications. Fundamental research questions of representation, reasoning, and interaction need to be addressed. To truly solve the mapping task requires an understanding of all disciplines contributing to the solution of these questions. This makes robot mapping an interdisciplinary endeavor and its objective grows with new robot applications desired. So, it might very well take another while before (if at all?) this research chapter can be closed eventually.

Appendix A

Publications resulting from this work

Parts of this research have already let to the following peer-reviewed publications:

- Longin Jan Latecki, Rolf Lakämper, and Diedrich Wolter (2003). Shape similarity and visual parts, In: Proceedings of the 11th International Conference on Disrecte Geometry for Computer Imagery (DGCI), Naples, Italy

  To this article I have contributed a discussion about the close connection between processing shape information in visual applications and robot mapping; my first experiments on mapping are presented.

- Longin Jan Latecki, Rolf Lakämper, Xinqu Sun, and Diedrich Wolter (2004). Building polygonal maps from laser range data, In: *Proceedings of the ECAI-workshop cognitive robotics (CogRob)*, Patrick Doherty (eds.), pp. 56–62, Valencia, Spain

  This article presents mapping experiments that have been conducted using earlier versions of the shape analysis techniques presented in this work; I contributed description and evaluation of shape extraction and matching.

- Diedrich Wolter and Kai-Florian Richter (2004). Schematized maps for robot guidance, In: *Proceedings of the ECAI-workshop cognitive robotics (CogRob)*, Patrick Doherty (eds.), pp. 71–76, Valencia, Spain

  In this article, I described spatial representation and reasoning techniques that appear suitable to instruct an autonomous mobile robot using a schematic map as medium.

- Diedrich Wolter and Longin Jan Latecki (2004). Shape matching for robot mapping, In: *Proceedings of 8th Pacific Rim International Conference on Artificial Intelligence (PRICAI)*, Chengqi Zhang, Hans W. Guesgen, and

Wai K. Yeap (eds.), pp. 693–702, LNAI Vol. 3157, Auckland, New Zealand

In this article I describe the two-phase correspondence determination that allows disregarding odometry information purely by shape analysis.

- Diedrich Wolter, Longin Jan Latecki, Rolf Lakämper, and Xinqu Sun (2004). Shape-based robot mapping, In: *KI 2004: Advances in Artificial Intelligence, Proceedings of the 27th German Conference in AI (KI-2004)*, Susanne Biundo, Thom Frühwirth, and Günther Palm (Eds.), pp. 439–452, LNAI Vol. 3238,Ulm, Germany

  In this article, the incremental mapping architecture and its components are described.

- Longin Jan Latecki, Rolf Lakämper, and Diedrich Wolter (2005). Incremental multi-robot mapping. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*

  The article presents results of a proof-of-concept study showing that shape analysis provides promising means to tackle map merging in multi-robot mapping; I contributed to the algorithm of correspondence determination.

- Longin Jan Latecki, Rolf Lakämper, and Diedrich Wolter (2005). Partial optimal shape similarity, *Image Vision Computing Journal*, 23:2, pp. 227 – 236

  This article describes an approach to shape similarity generalizing the shape similarity measure developed in this work; I contributed the specific formulation of partial optimal similarity for matching range information and demonstrated that the same technique can also be applied in the context of visual object recognition to recognizing shapes in coarse pixel images.

- Longin Jan Latecki, Rolf Lakämper, Xinqu Sun, and Diedrich Wolter (2005). Geometric robot mapping, In: *Proceedings of the 12th International Conference on Discrete Geometry for Computer Imagery (DGCI)*, Poitiers, France, 2005

  This article presents merging techniques for map construction; I contributed by providing matched shape information autonomously derived from sensor data.

- Diedrich Wolter, Christian Freksa, and Longin Jan Latecki. Towards a Generalization of Self-Localization, In M. E. Jefferies and W. K. Yeap, editors, Robot and Cognitive Approaches to Spatial Mapping, Springer-Verlag, *accepted contribution*

  This article analyses spatial representations and reasoning for self-localization in a generalized context; I present some of the results of the case-study on self-localization detailed in this dissertation.

Bibliography

Altermatt, M., Martinelli, A., Tomatis, N., & Siegwart, R. (2004). SLAM with corner features based on a relative map. In *Proceedings of the IROS-2004*.

Ambler, A. P., Barrow, H. G., Brown, C. M., Burstall, R. M., & Popplestone, R. J. (1973). A versatile computer-controlled assembly system. In *Proceedings of the Internatoinal Joint Conference on Artificial Intelligence*, (pp. 298–307).

Arkin, M., Chew, L., Huttenlocher, D., Kedem, K., & Mitchell, J. S. B. (1991). An efficiently computable metric for comparing polygonal shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13.

Arulampalam, S., Maskell, S., Gordon, N., & Clapp, T. (2001). A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*.

Austin, D. J. & McCarragher, B. J. (2001). Geometric constraint identification and mapping for mobile robots. *Robotics and Autonomous Systems*, 35:59–76.

Bachelder, I. A. & Waxman, A. M. (1994). Mobile robot visual mapping and localization: a view-based neurocomputational architecture that emulates hippocampal place learning. *Neural networks; Special issue: models of neurodynamics and behavior*, 7(6–7):1083–1099.

Bailey, T. (2002). *Mobile Robot Localization and Mapping in Extensive Outdoor Environments*. Ph.D. thesis, University of Sidney, Department of Aerospace, Mechanical, and Mechatronic Engineering.

Bailey, T., Nieto, J., & Nebot, E. (2006). Consistency of the FastSLAM algorithm. In *Proceedings of the 2006 IEEE International Conference of Robotics and Automation*, (pp. 424–429). Orlando, Florida, USA.

Baker, C., Morris, A., Ferguson, D., Thayer, S., Whittaker, C., Omohundro, Z., Reverte, C., Whittaker, W., Hähnel, D., & Thrun, S. (2004). A campaign in autonomous mine mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

Barkowsky, T., Berendt, B., Egner, S., Freksa, C., Krink, T., Röhrig, R., & Wulf, A. (1994). The REALATOR: How to construct reality. In *Proceedings of the ECAI'94 Workshop of Spatial and Temporal Reasoning*.

Barkowsky, T., Latecki, L. J., & Richter, K.-F. (2000). Schematizing maps: Simplification of geographic shape by discrete curve evolution. In C. Freksa, W. Brauer, C. Habel, & K. Wender (eds.), *Spatial Cognition II - Integrating Abstract Theories, Empirical Studies, Formal Methods, and Practical Applications*, (pp. 41–53). Springer; Berlin.

Bartoli, M., Pelillo, M., Siddiqi, K., & Zucker, S. W. (2000). Attributed tree homomorphism using association graphs. In *Proceedings of the 15th International Conference on Pattern Recognition (ICPR'00)*, vol. 2.

Basri, R., Costa, L., Geiger, D., & Jacobs, D. (1998). Determining the similarity of deformable shapes. *Vision Research*, 38.

Bauer, H. (1991). *Wahrscheinlichkeitstheorie*. de Gruyter, 4. ed.

Bellman, R. (1957). *Dynamic Programming*. Princeton University Press.

Bennewitz, M., Burgard, W., & Thrun, S. (2002). Using EM to learn motion behaviors of persons with mobile robots. In *Proceedings of International Conference on Intelligent Robots and Systems (IROS)*.

Berendt, B., Barkowsky, T., Freksa, C., & Kelter, S. (1998). Spatial representation with aspect maps. In C. Freksa, C. Habel, & K. F. Wender (eds.), *Spatial Cognition — An interdisciplinary approach to representing and processing spatial knowledge*, (pp. 313–336). Springer; Berlin.

de Berg, M., van Kreveld, M., Overmars, M., & Schwarzkopf, O. (2000). *Computational Geometry. Algorithms and Applications*. Springer-Verlag.

Berge, C. (1970). Sur certains hypergraphes generalisant les graphes bipartites. In P. Erdős, A. Rény, & V. Sós (eds.), *Combinatorial Theory and its Applications I*. Colloq. Math. Soc., North Holland, Amsterdam.

Besl, P. & McKay, N. (1992). A method for registration of 3D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256.

Biedermann, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94(2):115–117.

Blum, H. (1967). A transformation for extracting new descriptors of shape. In W. Walthen-Dunn (ed.), *Models for the Perception of Speech and Visual Form*. MIT press.

Blum, H. & Nagel, R. N. (1978). Shape description using weighted symmetric axis features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10:167–180.

Bober, M. (2001). Mpeg-7 visual shape descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):716–719.

Bober, M., Kim, J. D., Kim, H. K., Kim, Y. S., Kim, W.-Y., & Muller, K. (1999). Summary of the results in shape descriptor core experiment. *Tech. rep.* MPEG-7, ISO/IEC JTC1/SC29/WG11/ MPEG99/M4869.

Bomze, I. M., Budinich, M., Pardalos, P. M., & Pelillo, M. (1999). The maximum clique problem. *Technical Report Series in Computer Science CS-99-1*, Dipartimento di Informatica, Università Ca' Foscari di Venezia.

Bosse, M., Newman, P., Leonard, J., Soika, M., Feiten, W., & Teller, S. (2003). An atlas framework for scalable mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA' 03)*, vol. 2, (pp. 1899–1906).

Burgard, W., Cremers, A., Fox, D., Hähnel, D., Lakemeyer, G., Schulz, D., Steiner, W., & Thrun, S. (1999a). Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(1-2):3–55.

Burgard, W., Fox, D., Jans, H., Matenar, C., & Thrun, S. (1999b). Sonar-based mapping with mobile robots using EM. In *Proceedings of the International Conference on Machine Learning*.

Buschka, P. (2006). *An Investigation of Hybrid Maps for Mobile Robots*. Ph.D. thesis, Örebro University, Institutionen för teknik.

Castellanos, J. & Tardós, J. (2000). *Mobile Robot Localization and Map Building: A Multisensor Fusion Approach*. Boston, MA: Kluwer Academic Publishers.

Chatila, R. & Laumond, J.-P. (1985). Position referencing and consistent world modeling for mobile robots. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'85)*, (pp. 138–145).

Cheng, K. (1986). A purely geometric module in the rat's spatial representation. *Cognition*, 23:149–178.

Choset, H., Konukseven, I., & Burdick, J. (1996). Mobile robot navigation: issues in implementing the generalized voronoi graph in the plane. In *1996 IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems, New York, NY, USA*, (pp. 241–248). IEEE.

Choset, H., Walker, S., Eiamsa-Ard, K., & Burdick, J. (2000). Sensor-based exploration: Incremental construction of the hierarchical generalized voronoi graph. *International Journal of Robotics Research*, 19(2):126–148.

Choset, H., Lynch, K., Hutchinson, S., Kantor, G., Burgard, W., Kavrakij, L., & Thrun, S. (2005). *Principles of Robot Motion - Theory, Algorithms, and Implementations*. MIT-Press.

Cortelazzo, G., Mian, G. A., Vezzi, G., & Zamperoni, P. (1994). Trademark shape description by string-matching techniques. *Pattern Recognition*, 27(8):1005–1018.

Cox, I. & Leonard, J. (1994). Modeling a dynamic environment using a bayesian multiple hypothesis approach. *Artificial Intelligence*, 66:311—344.

Cox, I. J. (1990). Blanche: Position estimation for an autonomous robot vehicle. In I. J. Cox & G. Wilfong (eds.), *Autonomous Robot Vehicles*, (pp. 221–228). Springer-Verlag.

Denis, M. (1997). The description of routes: A cognitive approach to the production of spatial discourse. *Cahiers Psychologie Cognitive*, 16(4):409–458.

Dimitrov, P., Phillips, C., & Siddiqi, K. (2000). Robust and efficient skeletal graphs. In *Proc. of Conference on Computer Vision and Pattern Recognition*. Hilton Head, South Carolina.

Dissanayake, G., Newman, P., Clark, S., Durrant-Whyte, H., & Csorba, M. (2001). A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions of Robotics and Automation*.

Djugash, J., Singh, S., & Corke, P. I. (2005). Further results with localization and mapping using range from radio. In *International Conference on Field & Service Robotics (FSR '05)*.

Dong, T. (2005). *Recognizing variable spatial environments — the theory of cognitive prism*. Ph.D. thesis, Universität Bremen.

Doucet, A., de Freitas, J., Murphy, K., & Russell, S. (2000). Rao-Blackwellised particle filtering for dynamic Bayesian networks. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*.

Duda, R. & Hart, P. (1972). Use of hough transform to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15.

Duda, R. & Hart, P. (1973). *Pattern classification and scene analysis*. Wiley, New York.

Elfes, A. (1989). *Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation*. Ph.D. thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University.

Fischer, K. & Moratz, R. (2001). From communicative strategies to cognitive modelling. In *Proceedings of the First International Workshop on 'Epigenetic Robotics'*.

Forsberg, J., Larsson, U., & Wernersson, Å. (1995). Mobile robot navigation using the range-weighted hough transform. *IEEE Robotics & Automation Magazine*, 21:18–26.

Forsman, P. (2001a). Feature based registration of 3D perception data for indoor and outdoor map building. In *Int. Conference on Field and Service Robotics*. Helsinki, Finland.

Forsman, P. (2001b). *Three-dimensional localization and mapping of static environments by means of mobile perception*. Ph.D. thesis, Helsinki university of technology.

Forsman, P. & Halme, A. (2004). Feature based registration of range images for mapping of natural outdoor feature based registration of range images for mapping of natural outdoor. In *Second International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT'04)*, (pp. 542–549).

Fox, D., Burgard, W., & Thrun, S. (1999). Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11:391–427.

Frank, A. (1992). Qualitative spatial reasoning about distances and directions in geographic space. *Journal of Visual Languages and Computing*, 3:343–371.

Franz, M. O., Schölkopf, B., Mallot, H. A., & Bülthoff, H. H. (1998). Learning view graphs for robot navigation. *Autonomous Robots*, 5:111 – 125.

Freksa, C. (1991). Conceptual neighborhood and its role in temporal and spatial reasoning. In M. Singh & L. Travé-Massuyès (eds.), *Decision Support Systems and Qualitative Reasoning*, (pp. 181 – 187). North-Holland, Amsterdam.

Freksa, C. (1992). Temporal reasoning based on semi-intervals. *Artificial Intelligence*, 54(1):199–227.

Freksa, C. (2004). Spatial cognition — an AI perspective. In de Mántaras & Saitta (2004).

Freksa, C. & Röhrig, R. (1993). Dimensions of qualitative spatial reasoning. In *Proceedings of the III IMACS International Workshop on Qualitative Reasoning and Decision Technologies - QUARDER'93*, (pp. 483–492).

Freksa, C., Moratz, R., & Barkowsky, T. (2000a). Robot navigation with schematic maps. In E. P. et al. (ed.), *Intelligent Autonomous Systems 6*. IOS Press, Amsterdam.

Freksa, C., Moratz, R., & Barkowsky, T. (2000b). Schematic maps for robot navigation. In C. Freksa, W. Brauer, C. Habel, & K. Wender (eds.), *Spatial Cognition II: Integrating Abstract Theories, Empirical Studies, Formal Methods, and Practical Applications*, vol. 1849 / 2000, (pp. 100–114). Springer.

Frese, U. (2005). Treemap: An $o(logn)$ algorithm for simultaneous localization and mapping. In C. Freksa (ed.), *Spatial Cognition IV*, (pp. 455 — 476). Springer Verlag.

Galton, A. & Meathrel, R. (1999). Qualitative outline theory. In *Proceedings of the Internatoinal Joint Conference on Artificial Intelligence*.

González-Baños, H. & Latombe, J.-C. (2001). Robot navigation for automatic model construction using safe regions. In D. Rus & S. Singh (eds.), *Experimental Robotics VII*, vol. 271 of *Lecture Notes in Control and Information Sciences*, (pp. 405 – 416). Springer-Verlag.

González-Baños, H., Mao, E., Latombe, J., Murali, T., & Efrat, A. (1999). Planning robot motion strategies for efficient model construction. In *Robotics Research — The Eight International Symposium*. Salt Lake City (UT), USA.

Gottfried, B. (2005). *Shape from Positional-Contrast — Characterising Sketches with Qualitative Line Arrangements*. Ph.D. thesis, Universität Bremen.

Grimson, W. E. L. (1990). *Object Recognition by Computer: The Role of Geometric Constraints*. Cambridge (MA), USA: MIT Press.

Grisetti, G., Stachniss, C., & Burgard, W. (2005). Improving grid-based SLAM with Rao-Blackwellized particle filters by adaptive proposals and selective resampling. In *In Proceeding of the IEEE International Conference on Robotics and Automation (ICRA)*.

Gutmann, J.-S. (2000). *Robuste Navigation autonomer mobiler Systeme*. Ph.D. thesis, University of Freiburg. (in German).

Gutmann, J.-S. & Konolige, K. (1999). Incremental mapping of large cyclic environments. In *International Symposium on Computational Intelligence in Robotics and Automation (CIRA '99)*.

Gutmann, J.-S. & Schlegel, C. (1996). Amos: Comparison of scan matching approaches for self-localization in indoor environments. In *Proceedings of the 1st Euromicro Workshop on Advanced Mobile Robots (Eurobot'96)*.

Gutmann, J.-S., Weigel, T., & Nebel, B. (2001). A fast, accurate and robust method for self-localization in polygonal environments using laser range finders. *Advanced Robotics*, 14(8):651–667.

Hähnel, D. (2004). *mapping with mobile robots*. Ph.D. thesis, University of Freiburg.

Hähnel, D., Schulz, D., & Burgard, W. (2002). Map building with mobile robots in populated environments. In *Proceedings of International Conference on Intelligent Robots and Systems (IROS'02)*.

Hähnel, D., Burgard, W., Fox, D., & Thrun, S. (2003). An efficient FastSLAM algorithm for generating maps of large-scal cyclic environments from raw laser range measurements. In *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*.

Hähnel, D., Burgard, W., Fox, D., Fishkin, K., & Philipose, M. (2004). Mapping and localization with RFID technology. In *n Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

Hermer, L. (1997). Internally coherent spatial memories in a mammal. *Neuroreport*, 8:1743–1747.

van Hoeve, W. (2001). The alldifferent constraint: A survey. In *Sixth Annual Workshop of the ERCIM Working Group on Constraints*. Prague.

Hu, M. K. (1962). Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8:179–187.

Huang, W. H. & Beevers, K. R. (2005). Topological map merging. *International Journal of Robotics Research*, 24(8):601–613.

Huttenlocher, D., Klandermann, G., & Rucklidge, W. (1993). Comparing distances using the hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15:850–863.

Jähne, B. (1997). *Digitale Bildverarbeitung*. Springer-Verlag, 4 ed.

Jefferies, M., Yeap, W., Smith, L., & Ferguson, D. (2001). Building a map for robot navigation using a theory of cognitive maps. In *Proc. IASTED International Conference on Artificial Intelligence and Applications*. Marbella, Spain.

Jefferies, M. E. & Yeap, W. K. (2001). The utility of global representations in a cognitive map. In *Proceedings of the Conference on Spatial Information Theory (COSIT)*.

Jefferies, M. E., Baker, J., & Wang, W. (2003). Robot cognitive mapping — a role for a global metric map in a cognitive mapping process. In *Workshop on Robot and Cognitive Approaches to Spatial Mapping*.

Jefferies, M. E., Cosgrove, M., Baker, J. T., & Yeap, W. (2004a). The correspondence problem in topological metric mapping — using absolute metric maps to close cycles. In *Proceedings of the Eighth International Conference On Knowledge-based Intelligent Information and Engineering Systems*, (pp. 232 – 239).

Jefferies, M. E., Cree, M., Mayo, M., & Baker, J. T. (2004b). Using 2D an 3D landmarks to solve the correspondence problem in cognitive robot mapping. In C. Freksa, M. Knauff, B. Krieg-Brückner, B. Nebel, & T. Barkowsky (eds.), *Spatial Cognition IV. Reasoning, Action, Interaction: International Conference Spatial Cognition*, vol. LNAI 3343, (pp. 434–454).

Jefferies, M. E., Weng, W., Baker, J. T., & Mayo, M. (2004c). Using context to solve the correspondence problem in simultaneous localisation and mappin. In C. Zhang & H. W. G. W. K. Yeap (eds.), *PRICAI 2004: Trends in Artificial Intelligence: 8th Pacific Rim International Conference on Artificial Intelligence*, vol. LNCS 3157 / 2004, (pp. 664–672). Auckland, New Zealand: Springer-Verlag.

Jensfelt, P. & Kristensen, S. (1999). Active global localisation for a mobile robot using multiple hypothesis tracking. In *Proceedings of the IJCAI Workshop on Reasoning with Uncertainty in Robot Navigation*, (pp. 13–22). Stockholm, Sweden.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME—Journal of Basic Engineering*, (pp. 35–45).

Kim, Y.-S. & Kim, W.-Y. (1998). Content-based trademark retrieval system using visually salient features. *Journal of Image and Vision Computing*, 16(12):931–940.

Kimia, B. B., Tannenbaum, A., & Zucker, S. W. (1990). Toward a computational theory of shape: An overview. In *Proceedings of the First European Conference on Computer Vision*. Antibes, France.

Koenderink, J. & van Doorn, A. (1979). The internal representation of solid shape with respect to vision. *Biological Cybernetics*, 32:211 — 216.

Koenig, S. & Simmons, R. G. (1996). Passive distance learning for robot navigation. In *In Proceedings of the International Conference on Machine Learning*, (pp. 266–274).

Kolesnikov, A. & Fränti, P. (2005). Data reduction of large vector graphics. *Pattern Recognition*, 38(3):381–394.

Konolige, K., Fox, D., Limketkai, B., Ko, J., , & Stewart, B. (2003). Map merging for distributed robot navigation. In *In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (pp. 212–217).

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, 2:pp. 83–97.

Kuipers, B. (2000). The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233.

Kuipers, B. & Byun, Y.-T. (1991). A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Journal of Robotics and Autonomous Systems*, 8:47–63.

Lankenau, A. & Röfer, T. (2001). A safe and versatile mobility assistant. *IEEE Robotics and Automation Magazine*, 8(1):29 – 37.

Larsen, R. & Eiriksson, H. (2001). Robust and resistant 2D shape alignment. *Tech. Rep. 17/2001*, Technical University of Denmark, Informatics and Mathematical Modelling, Lyngby, Denmark.

Latecki, L. J. & Lakämper, R. (1999). Convexity rule for shape decomposition based on discrete contour evolution. *Computer Vision and Image Understanding*, 73.

Latecki, L. J. & Lakämper, R. (2000). Shape similarity measure based on correspondence of visual parts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(10).

Latecki, L. J. & Lakämper, R. (2006a).
URL `http://knight.temple.edu/~shape/issUsage.html` ISS database — the electronic PIcture RObot 'epiro'.

Latecki, L. J. & Lakämper, R. (2006b). Polygonal approximation of laser range data based on perceptual grouping and EM. In *IEEE International Conference on Robotics and Automation (ICRA)*. Orlando, Florida, USA.

Latecki, L. J. & Rosenfeld, A. (2002). Recovering a polygon from noisy data. *Computer Vision and Image Understanding (CVIU)*, 86(3):1–20.

Latecki, L. J., Ghadially, R.-R., Lakämper, R., & Eckhardt, U. (2000a). Continuity of the discrete curve evolution. *Journal of Electronic Imaging*, 9(3).

Latecki, L. J., Lakämper, R., & Eckhardt, U. (2000b). Shape descriptors for non-rigid shapes with a single closed contour. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. Hilton Head Island, South Carolina.

Latecki, L. J., Lakämper, R., & Wolter, D. (2003). Shape similarity and visual parts. In *Proceedings of the 11th International Conference on Discrete Geometry for Computer Imagery (DGCI), Naples, Italy*.

Latecki, L. J., Lakämper, R., & Wolter, D. (2005a). Incremental multi-robot mapping. In *Proceedings of IROS-2005*.

Latecki, L. J., Lakämper, R., & Wolter, D. (2005b). Partial optimal shape similarity. *Image and Vision Computing Journal*, 23(2):227 – 236.

Latombe, J.-C. (1991). *Robot Motion Planning*. Norwell (MA), USA: Kluwer Academic Publishers.

Laumond, J.-P. (1983). Model structuring and concept recognition: Two aspects of learning for a mobile robot. In *Proceedings of the International joint conference on Artificial Intelligence (IJCAI)*, (pp. 108–120).

Lee, D. T. & Drysdale, R. L. (1981). Generalization of Voronoi diagrams in the plane. *SIAM Journal on Computing*, 10(1):73 – 87.

Leonard, J., Durrant-Whyte, H., & Cox, I. (1992). Dynamic map building for an autonomous mobile robot. *International Journal of Robotics Research*, 11(4):89—96.

Leonard, J. J., Newman, P. M., Rikoski, R. J., Neira, J., & D.Tardós, J. (2001). Towards robust data association and feature modeling for concurrent mapping and localization. In *Proceedings of the 10th International Symposium of Robotics Research (ISRR'2001)*. Lorne, Victoria, Australia.

Ligozat, G. (2000). From language to motion, and back: Generating and using route descriptions. In N. Christodoulakis (ed.), *Natural Language Processing. Proceedings of the 2nd International Conference*, (pp. 328–345). Berlin: Springer.

Lingemann, K., Surmann, H., Nüchter, A., & Hertzberg, J. (2004). Indoor and outdoor localizations for fast mobile robots. In *Proceedings of International Conference on Intelligent Robots and Systems (IROS)*.

Loncaric, S. (1998). A survey of shape analysis techniques. *Pattern Recognition*, 31(8):983–1001.

Lovasz, L. (1986). *Matching Theory*. Elsevier Science Ltd.

Lu, F. & Milios, E. (1997). Robot pose estimation in unknown environments by matching 2D range scans. *Journal of Intelligent and Robotic Systems*.

Mallot, H. A., Bülthoff, H. H., Georg, P., Schölkopf, B., & Yasuhara, K. (1995). View-based cognitive map learning by an autonomous robot. In *Proceedings of the International conference on Artificial Neural Networks (ICANN 2)*, (pp. 381–386).

de Mántaras, R. L. & Saitta, L. (eds.) (2004). *Proceedings of the 16th Eureopean Conference on Artificial Intelligence, ECAI'2004, including Prestigious Applicants of Intelligent Systems, PAIS 2004, Valencia, Spain, August 22-27, 2004*. IOS Press.

Margules, J. & Gallistel, C. R. (1988). Heading in the rat: determination by environmental shape. *Animal learning and behaviour*, 16(4):404–410.

Marques, J. S. & Abrantes, A. J. (1997). Shape alignment — optimal initial point and pose estimation. *Pattern Recognition Letters*.

Matsumoto, Y., Ikeda, K., Inaba, M., & Inoue, H. (1999). Exploration and map acquisition for view-based navigation in corridor environment. In *Proceedings of the International conference on Field and Service Robotics*, (pp. 29–31). Pittsburgh (PA), USA.

Meathrel, R. & Galton, A. (2000). Qualitative representation of planar outlines richard meathrel and antony galton. In *Proceesings of the European Conference on Artifical Intelligence (ECAI)*.

van der Merwe, R., Doucet, A., & de Freitas E. Wan, N. (2000). The unscented particle filter. *Advances in Neural Information Processing Systems*, 8(351–357).

Mokhtarian, F. & Mackworth, A. K. (1992). A theory of multi-scale, curvature-based shape representation for planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(8):789–805.

Montemerlo, M., Thrun, S., Koller, D., & Wegbreit, B. (2002). FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Proceedings of the AAAI National Conference on Artificial Intelligence*. Edmonton, Canada: AAAI.

Montemerlo, M., Thrun, S., Koller, D., & Wegbreit, B. (2003). FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of IJCAI*.

Moratz, R. & Freksa, C. (1998). Spatial reasoning with uncertain data using stochastic relaxation. In W. Brauer (ed.), *Proceedings of Fuzzy-Neuro-Systems '98*, (pp. 106–112). St. Augustin: Infix.

Moratz, R. & Tenbrink, T. (2003). Instruction modes for joint spatial reference between naive users and a mobile robot. In *Proceedings of the IEEE International Conference on Robotics, Intelligent Systems and Signal Processing (RISSP 2003)*. Changsha, Hunan, China. Special Session on New Methods in Human Robot Interaction.

Moratz, R. & Tenbrink, T. (2006). Spatial reference in linguistic human-robot interaction: Iterative, empirically supported development of a model of projective relations. *Spatial Cognition and Computation*. In press.

Moratz, R., Renz, J., & Wolter, D. (2000). Qualitative spatial reasoning about line segments. In W. Horn (ed.), *ECAI 2000 Proceedings of the 14th European Conference on Artificial Intelligence*. IOS Press, Amsterdam.

Moravec, H. P. & Elfes, A. E. (1985). High resolution maps from wide angle sonar. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

Museros, L. & Escrig, M. T. (2004). A qualitative theory for shape representation and matching for design. In de Mántaras & Saitta (2004), (pp. 858–862).

Musto, A., Stein, K., Eisenkolb, A., & Röfer, T. (1999). Qualitative and quantitative representations of locomotion and their application in robot navigation. In *Proceedings of International Joint Conference on AI (IJCAI)*, (pp. 1067–1073).

Neira, J. & Tardós, J. D. (2001). Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on robotics and automation*, 17(6):890–897.

Nieto, J. I., Guivant, J. E., & Nebot, M. (2004). The HYbrid metric maps (HYMMSs): A novel map representation for DenseSLAM. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

Page, D. L., Koschan, A. F., Sukumar, S. R., Roui-Abidi, B., & Abidi, M. A. (2003). Shape analysis algorithm based on information theory. In *Proceedings of the IEEE International Conference on Image Processing (ICIP03)*, vol. 1, (pp. 229–232).

Palmer, S. E. (1999). *Vision science—photons to phenomenology*. MIT press.

Pavlidis, T. (1978). A review of algorithms for shape analysis. *Computer Graphics and Image Processing*, 7:243–258.

Pavlidis, T. (1995). A review of algorithms for shape analysis. In *Document image analysis*, (pp. 145–160). Los Alamitos, CA, USA: IEEE Computer Society Press.

Pelillo, M. (1998). A unifying framework for relational structure matching. In *14th International Conference on Pattern Recognition (ICPR'98)*, vol. 2.

Pfister, S. T., Roumeliotis, S. I., & Burdick, J. W. (2003). Weighted line fitting algorithms for mobile robot map building and efficient data representation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

Rafflin, C. & Fournier, A. (1996). Learning with a friendly interactive robot for service tasks in hopsital environments. *Autonomous Robots*, 3(4):399–414.

Randell, D. A., Cui, Z., & Cohn, A. G. (1992). A spatial logic based on regions and "Connection". In *Proceedings of KR92*.

Reiss, T. H. (1993). *Recoginizing planar objects using invariant image features*. Springer-Verlag.

Remolina, E. & Kuipers, B. (2004). Towards a general theory of topological maps. *Artificial Intelligence*, 152:47–104.

Röfer, T. (2002). Using histogram correlation to create consistent laser scan maps. In *Proceedings of the IEEE International Conference on Robotics Systems (IROS-2002)*.

Roumeliotis, S. & Bekey, G. (2000). Bayesian estimation and kalman filtering: A unified framework for mobile robot localization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, (pp. 2985–2992). San Francisco (CA), USA.

Rucklidge, W. J. (1996). Locating objects using the Hausdorff distance. *Tech. rep.*, Xerox Palo Alto Research Center.

Rucklidge, W. J. (1997). Efficiently locating objects using the hausdorff distance. *International Journal of Computer Vision*, 24(3):251 – 270.

Sack, D. & Burgard, W. (2003). A comparison of methods for line extraction from range data. In *Proceedings of the 5th IFAC Symposium on Intelligent Autonomous Vehicles (IAV)*.

Schlieder, C. (1993). Representing visible locations for qualitative navigation. In N. Piera-Carrete & M. Singh (eds.), *Qualitative reasoning and decision technologies*, (pp. 523–532).

Schlieder, C. (1994). Qualitative shape representation. In *Spatial conceptual models for geographic objects with undetermined boundaries*. Taylor & Francis.

Schlieder, C. (1995). Reasoning about ordering. In *Proceedings of the 3rd International Conference on Spatial Information Theory (COSIT)*.

Schölkopf, B. & Mallot, H. A. (1995). View-based cognitive mapping and path planning. *Adaptive Behavior*, 3(311–348).

Se, S., Lowe, D., & Little, J. (2002). mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *International Journal of Robotics Research*, 21(8):735–758.

Sebastian, T. B., Klein, P. N., & Kimia, B. B. (2003). On aligning curves. *Pattern Analysis and Machine Intelligence*, 25(1):116–125.

Shokoufandeh, A., Marsic, I., & Dickinson, S. J. (1999). View-based object recognition using saliency maps. *Image and Vision Computing*, 17:445 — 460.

Siddiqi, K., Bouix, S., Tannenbaum, A., & Zucker, S. W. (1999a). The hamilton-jacobi skeleton. In *Proc. of International Conference on Computer Vision*. Corfu, Greece.

Siddiqi, K., Shokoufandeh, A., Dickinson, S. J., & Zucker, S. W. (1999b). Shock graphs and shape matching. *International Journal of Computer Vision*, 35(1):13–32.

Skiadopoulos, S. & Koubarakis, M. (2005). On the consistency of cardinal directions constraints. *Artificial Intelligence*, 163(1):91–135.

Smith, R., Self, M., & Cheeseman, P. (1990). Estimating uncertain spatial relationships in robotics. In I. Cox & G. Wilfong (eds.), *Autonomous Robot Vehicles*, (pp. 167–193). Springer-Verlag.

Smith, R. C. & Cheeseman, P. (1986). On the representation and estimation of spatial uncertainty. *International Journal of Robotics Research*, 5(4):56—68.

Stachniss, C. (2006). *Exploration and Mapping with Mobile Robots*. Ph.D. thesis, Universität Freiburg.

Stachniss, C. & Burgard, W. (2003a). Exploring unknown environments with mobile robots using coverage maps. In *In Proceedings of the International Conference on Artificial Intelligence (IJCAI)*, (pp. 1127–1132). Acapulco, Mexico.

Stachniss, C. & Burgard, W. (2003b). Mapping and exploration with mobile robots using coverage maps. In *In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (pp. 476–481). Las Vegas, NV, USA.

Stachniss, C. & Burgard, W. (2003c). Using coverage maps to represent the environment of mobile robots. In *In Proceedings of the European Conference on Mobile Robots (ECMR)*, (pp. 59–64). Radziejowice, Poland.

Stachniss, C., Hähnel, D., & Burgard, W. (2004). Exploration with active loop-closing for FastSLAM. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Stein, K. (2003). *Qualitative Repräsentation und Generalisierung von Bewegungsverläufen*. Ph.D. thesis, Institut für Informatik der Technischen Universität München. In German.

Surmann, H., Nüchter, A., & Hertzberg, J. (2003). An autonomous mobile robot with a 3d laser range finder for 3d exploration and digitalization of indoor environments. *Journal of Robotics and Autonomous Systems*, 45(3–4):181–198.

Teague, M. R. (1980). Image analysis via the general theory of moments. *Journal of the Optical Society of America*, 70:920–930.

Thrun, S. (1998). Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):21–71.

Thrun, S. (2000). Probabilistic algorithms in robotics. *AI Magazine*, 21(4):93–109.

Thrun, S. (2001). A probabilistic online mapping algorithm for teams of mobile robots. *International Journal of Robotics Research*, 20(5):335–363.

Thrun, S. (2002). Robotic mapping: A survey. In G. Lakemeyer & B. Nebel (eds.), *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann.

Thrun, S., Bücken, A., Burgard, W., Fox, D., Fröhlinghaus, T., Henning, D., Hofmann, T., Krell, M., & Schmidt, T. (1998a). Map learning and high-speed navigation in RHINO. In D. Kortenkamp, R. Bonasso, & R. Murphy (eds.), *AI-based Mobile Robots: Case Studies of Successful Robot Systems*. MIT Press.

Thrun, S., Fox, D., & Burgard, W. (1998b). A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning*.

Thrun, S., Beetz, M., Bennewitz, M., Burgard, W., Cremers, A., Dellaert, F., Fox, D., Hähnel, D., Rosenberg, C., Roy, N., Schulte, J., & Schulz, D. (2000a). Probabilistic algorithms and the interactive museum tour-guide robot minerva. *International Journal of Robotics Research*, 19(11):972–999.

Thrun, S., Burgard, W., & Fox, D. (2000b). A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. San Francisco, CA: IEEE.

Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic Robotics*. MIT-Press.

Tsang, E. (1993). *Foundations of Constraint Satisfaction*. London: Academic Press.

Tversky, A. (1977). Features of similarity. *Psychological Review*, 84(4):327–352.

Ullman, J. D. (1976). An algorithm for subgraph isomorphism. *Journal of the ACM*, 23(1):31—42.

Veeck, M. & Burgard, W. (2004). Learning polyline maps from range scan data acquired with mobile robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Veltkamp, R. C. (2001). Shape matching: Similarity measure and algorithms. In *Proceedings of Shape Modelling International*, (pp. 188–197).

Wagner, T., Visser, U., & Herzog, O. (2004). Egocentric qualitative spatial knowledge representation for physical robots. *Robotics and Autonomous Systems*, 49:25–42.

Wallgrün, J. O. (2002). *Exploration und Pfadplanung für mobile Roboter basierend auf Generalisierten Voronoi-Graphen*. Master's thesis, University of Hamburg.

Wallgrün, J. O. (2005). Autonomous construction of hierarchical Voronoi-based route graph representations. In C. Freksa, M. Knauff, B. Krieg-Brückner, B. Nebel, & T. Barkowsky (eds.), *Spatial Cognition IV. Reasoning, Action, Interaction: International Conference Spatial Cognition 2004*, vol. 3343 of *Lecture Notes in Artificial Intelligence*, (pp. 413–433). Berlin, Heidelberg, New York: Springer.

Wang, Y., Makedon, F., Ford, J., & Huang, H. (2004). A bipartite graph matching framework for finding correspondences between structural elements in two proteins. In *26th Annual International Conference IEEE Engineering in Medicine and Biology Society*.

Werner, S., Krieg-Brückner, B., & Herrmann, T. (2000). Modelling navigational knowledge by route graphs. In C. Freksa, W. Brauer, C. Habel, & K. F. Wender (eds.), *Spatial Cognition II — Integrating Abstract Theories, Empirical Studies, Formal Methods, and Practical Applications*, vol. LNAI 1849, (pp. 295 – 317). Springer-Verlag.

Wertheimer, M. (1925). Über Gestalttheorie. *Philosophische Zeitschrift für Forschung und Aussprache*, 1:39–60. Transcription of a presentation at the Kant-Gesellschaft, Berlin 17th December, 1924.

Wolter, D. & Latecki, L. J. (2004). Shape matching for robot mapping. In C. Zhang, H. W. Guesgen, & W. K. Yeap (eds.), *Proceedings of 8th Pacific Rim International Conference on Artificial Intelligence (PRICAI-04)*. Auckland, New Zealand.

Wolter, D. & Richter, K.-F. (2004). Schematized aspect maps for robot guidance. In P. Doherty (ed.), *Proceedings of the workshop cognitive robotics (CogRob)*.

Wolter, D., Latecki, L. J., Lakämper, R., & Sun, X. (2004). Shape-based robot mapping. In *Proceedings of the 27th German conference on Artificial Intelligence (KI-2004)*.

Yeap, W. & Jefferies, M. (1999). Computing a representation for the local environment. *Artificial Intelligence*, 107.

Yeap, W. K. & Jefferies, M. E. (2000). On early cognitive mapping. *Spatial Cognition and Computation*, 2(2):85–116.

Zahn, C. T. & Roskies, R. Z. (1972). Fourier descriptors for planar closed curves. *IEEE Transactions on Computers*, 21:269–281.

Zhang, D. & Lu, G. (2002). Generic fourier descriptor for shape-based image retrieval. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '02)*, vol. 1, (pp. 425–428).

Zhang, K. & Shasha, D. (1989). Simple fast algorithms for the editng distance between trees and related problems. *SIAM Journal on Computing*, 18(6):1245 – 1262.