Towards a Generalization of Self-Localization

Diedrich Wolter[1] and Christian Freksa[1] and Longin Jan Latecki[2]

[1] SFB/TR 8, Universität Bremen {freksa,dwolter}@sfbtr8.uni-bremen.de
[2] Temple University latecki@temple.edu

Summary. Self-localization is an important task for humans and autonomous robots as it is the basis for orientation and navigation in a spatial environment and for performing mapping tasks. In robotics, self-localization on the basis of monomodal perceptual information has been investigated intensively. The present chapter looks at self-localization in a more general setting where the reference information may be provided by different types of sensors or by descriptions of locations under a variety of conditions. We introduce some of these conditions and discuss general approaches to identifying locations in perceived environments. Taking into account cognitive considerations, we propose an approach to identify locations on a high, abstract level of representation. The approach combines qualitative and quantitative information to recognize locations described as configurations of shape features. We evaluate this approach in comparison to other approaches in a self-localization task and a generalized localization task based on a schematic map.

1 Introduction

Humans and autonomous robots need to know where they are located to successfully orientate themselves, to navigate in a spatial environment, and to perform mapping tasks. The notion of "self-localization" (SL) refers to an agent's procedure of determining where it is located. SL procedures require spatial reference systems, for example a coordinate system or a map.

In autonomous robotics, approaches to SL have been developed that determine a robot's position and orientation (jointly referred to as "pose") based on sensor readings. To accomplish this, the robots relate the sensor information about their environments with their internal knowledge about these environments. Detected correspondences between the two knowledge sources are used to infer the presumed location of the robot.

The approach to SL outlined above even enables wheeled robots to identify locations not visited before. Henceforth, it also enables robots to incrementally build up spatial knowledge about initially unknown environments by determining their location and registering new observations in relation to this location. Coping with a-priori unknown environments is an important ingredient to intelligent autonomous navigation and consequently has been studied intensively.

However, in many situations, agents (humans, robots, software agents) have extensive a priori knowledge about the spatial environment, for example in the form of maps, sketches, natural language descriptions, or (precise or vague) memories of previous observations or descriptions. In such cases it may be desirable to make use of this knowledge to enable robots to localize themselves more efficiently or in ways that are similar to human self-localization.

For certain tasks the utilization of a priori knowledge is not only desirable but indispensable, for example when a robot is expected to visit places which are described by reference to this a priori knowledge; this may frequently be the case in natural instructions by a human instructor. Furthermore, it may be necessary that a robot specifies its position not in terms of its internal reference system but in terms of a reference system that is available to its human instructor and can be understood by him or by her.

From a technical point of view, this is a different task than conventional SL, as the knowledge employed exhibits different structures and characteristics than conventional sensor readings. In particular, this knowledge may not have an immediate geometric interpretation and it may lack details. Different types of reference systems will require different ways of self-localization; this does not imply, however, that localization will be less precise.

From a more abstract point of view, both tasks — sensor-based and knowledge-based self-localization — can be viewed as belonging to the same class of tasks, as both answer the question of the robot's pose with respect to a given spatial reference system. Therefore we will call this class of tasks "generalized self-localization" (GSL).

In the present chapter we explore several variations of the SL problem and investigate how we can extend existing SL approaches in such a way that they can solve the GSL problem. To this end we propose to employ more abstract forms of knowledge in order to integrate the dissimilitude of potential information sources for common treatment. We illustrate this approach using a specific robot task: spatial orientation by means of schematic maps. Schematic maps (e.g. public transportation maps or emergency evacuation maps) are successfully employed by humans due to their fast and efficient use. We will show how GSL can be used for human-robot communication on the basis of schematic maps.

2 The Generalized Localization Task

In robotics, the notion of self-localization has been used in a rather restricted sense: in its most elementary form it is used to denote the task of identifying the robots' locations on the basis of the same type of sensor information that has been retrieved from the location previously. More specifically, self-localization in so-called view-based robot navigation (see for example [12]) is performed with the *same* sensors and the *same* spatial resolution by an agent with more or less the *same* perspective as before. Thus, the robot can use characteristic features to identify a specific place in a finite set of places.

However, we may have situations in which a robot has to localize itself from perspectives it never has encountered before under comparable conditions, possibly not even with the same sensors, or even never encountered before at all. A human, another robot, or a data base may have provided information about the environment; this information is now to be used by the robot for its self-localization task. To cope with such situations, we will adopt a more general notion of self-localization.

2.1 Generalizing the Self-Localization Task

Starting with the aforementioned case of SL, an agent recognizes a location from an observation obtained with the same sensors, with the same spatial resolution, and from the same perspective — a simple task provided the agent receives the same percept as obtained in a reference cognition event. In realistic situations, however, the sameness of all these parameters is never given — let alone guaranteed; therefore it is not a trivial task to solve this self-localization problem. Successful approaches must deal with the unavoidable deviations of parameter values. However, this problem can be solved with little effort purely on the level of sensor data. We refer to this type of SL (not varying any parameters) as the *elementary case* of SL. It is utilized in view-based robot navigation (for an example, see [12]).

Which abilities does an agent need to recognize places under even less favorable conditions: from different locations with different spatial orientation, with different sensors, at different sensor resolution, or under different environmental conditions? In the following, we will consider incremental abilities required with respect to the restricted case of self-localization. We will consider three strands of generalizing the SL problem: (1) different perspective; (2) different spatial resolution; and (3) different kinds of sensors. Fig. 1 presents an overview of these generalization strands and indicates specific classes of SL tasks.

Different perspective

Variations of view poses can be considered a first step of generalizing SL. Robots identify their location using sensor readings taken at different view poses. Perceiving objects from varying locations, their appearance or visibility can change — for example, due to occlusion; such changes are reflected in the generalization axis ‘perspective’. Spatial reasoning allows inferring how a physical phenomenon observed from one perspective appears when observed from another perspective.

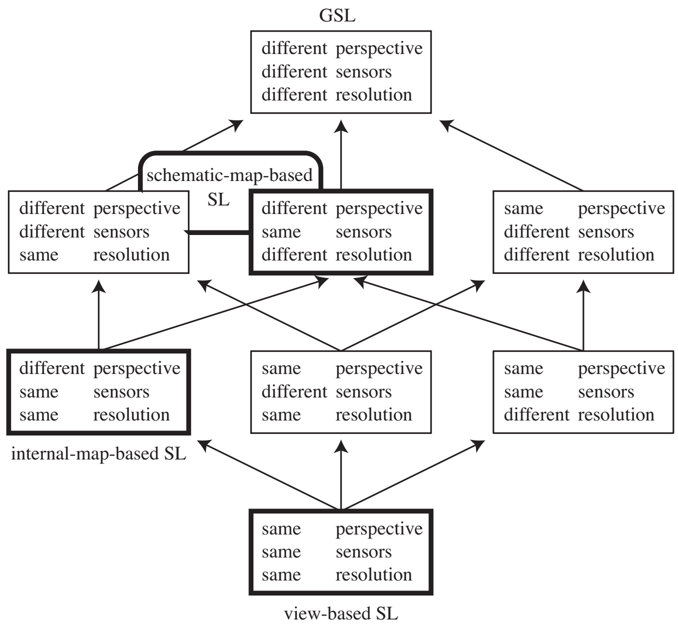


Fig. 1. Generalization in self-localization: The elementary case in SL (sameness in all apsects) is the localization problem faced in view-based robot navigation (bottom). Three strands of generalizing the elementary situation are depicted: perspective (left); sensors (center); and resolution (right). SL using an absolute spatial representation can cope with varying perspectives and handles SL based on a robot's internal map. The approach presented in this article uses a schematic map as reference; it is depicted at the generalization strand from map-based localization to GSL (upper left).

The elementary case of SL permits place recognition in an agent-centered reference system. To recognize locations independently of the agent's perspective we must transform sensory information into a location-independent, absolute reference system, e.g. a geographic map. Transformation from agent-centered observations to an absolute map is an abstraction process that abstracts from individual sensor readings and mediates between differences in multiple observations of the same physical phenomenon. This step is particularly easy for sensor data obtained from range sensors. It is still an unsolved problem if relying on camera images, though.

Using elementary spatial reasoning on an absolute spatial representation allows us to partially infer the expected view caused by a different pose. Most approaches to robot navigation or robot mapping utilize some kind of absolute representation, typically a coordinate-based map (see, e.g. [29, 40]). In this representation, perspective generalization can easily be handled.

Different kinds of sensors and knowledge sources

To describe different kinds of sensors and knowledge sources with a single label, we employ the notion of abstract sensor readings. For example, a map can provide abstract sensor readings by retrieving sensor information available at a given pose. An agent that has to recognize a place through perception with a different kind of sensor than initially will not be able to successfully match the corresponding percepts, in general; rather it will require a representation that relates different perceptions in terms of common traits.

For example, the boundary of a physical object may be perceived visually in terms of a transition between different brightness or hue values, through tactile perception in terms of a transition of physical resistance values, and through distance sensors in terms of an abrupt transition between distance values, while the object surface appearance may exhibit differentiated readings on some sensors and stable readings on others. Therefore, object boundaries are suitable concepts of a spatial scene that support multimodal recognition while object surfaces may be less suitable. Especially object boundaries which are boundaries to passable space are of importance to navigation as they constrain possible movements. We find these boundaries registered in maps, including schematic maps; boundaries are easily accessible to a robot utilizing range sensors.

To enable multimodal recognition on the basis of different abstract sensor readings, we may develop a representation that features the notion of an object boundary while it abstracts from object surfaces, for example. Such a representation also can be used to relate sensory information to conceptual knowledge that has been conveyed through object descriptions in terms of natural language or by graphical means. In other words, to make cross-modal use of a variety of knowledge sources we can abstract from the specifics of individual modalities and identify modality-independent features or concepts. We then must provide mappings between the modality-specific percepts and those concepts.

Different spatial sensor resolution

Even if we stay within the same modality, we will get problems with matching abstract sensor readings from a given place if the sensors provide spatial data at different levels of spatial resolution, as they will identify different sets of sensory features. A suitable abstraction from low-level perceptual features also will be helpful in this case: a resolution-adaptive representation will enable the comparison of sensor data obtained at different levels of spatial resolution.

We point out that a change of resolution (granularity) does not necessarily happen uniformly, as in the case of smoothening filter application. Rather, coarsening can occur selectively like in schematization processes (see [3]). Here information characteristic for a spatial configuration or relevant to a considered task may remain on a high level of detail whereas irrelevant information may be discarded completely. To interrelate different levels of granularity it is advantageous to define a notion of saliency for features; only salient features remain represented when the resolution is reduced. Moreover, it is essential to estimate whether a feature at hand will be represented on a specific level of granularity or not.

2.2 High-Level Knowledge for GSL

In the elementary case of SL, sensory information obtained by independently sensing the same physical phenomenon can be correlated in a rather straightforward manner. Moving along one of the three strands of generalization, adequately abstracted information and abstraction processes are required to enable correlation of sensor readings, i.e. matching, by focusing on essential features. When the perspective of observation changes, sensory information is abstracted to yield view independent images by employing an absolute representation, e.g. a map. To mediate between different abstract sensor readings, information can be abstracted to cross-modal concepts. Features present on different levels of resolution can be related using an abstraction process to reduce spatial resolution or handling information in a granularity-adaptive or granularity-insensitive manner. Qualitative spatial representations provide an anchor to handle varying levels of granularity as only the most relevant relations — which are not subject to change of resolution — are made explicit. We point out that in all strands of generalization abstraction is the key to master generalized localization tasks.

SL is primarily a problem of spatial information processing and we are especially interested in understanding spatial abstraction. Reconsidering the generalization strands in SL from the point of spatial abstraction, it can be organized as the pyramid presented in Fig. 2. On the finest level of granularity, fine-grained metric information is available; sensor and perspective variations cover a wide range. On a coarser level of granularity the multitude of possible variations decreases as the expressiveness in spatial information is reduced.

Coarse spatial information is available through language or through rough or schematic overview maps; it is typically qualitative information that classifies spatial information into distinct categories [22]. Notably, qualitative representations are not restricted to representations of coarse knowledge. Qualitative information can also be retrieved from fine-grained representations and, for example, can be exploited in reasoning process. We conclude that it is advantageous to explicitly address abstract qualitative information when interrelating spatial information on significantly varying levels of granularity or to bridge cross-modal variations.

2.3 Localization Using Schematic Maps

In this paper, we use the term "map" to denote a representation that relates landmarks and features to spatial locations. This subsumes internal spatial

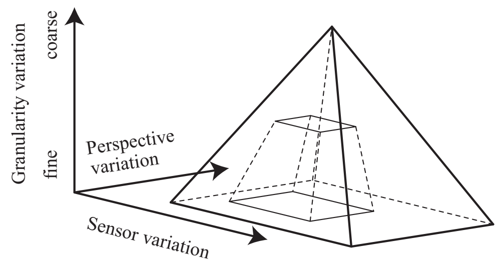


Fig. 2. The pyramid of generalized localization problems from the perspective of spatial information processing. Possible variations due to change of perspective or change of source for (abstract) sensor readings decrease when the level of spatial resolution is decreased. A reduction of granularity abstracts from metric details and gives rise to the importance of characteristic qualitative information.

representations of a robot and external maps, e.g. floor plans. A map typically preserves spatial information metrically (on a certain level of spatial granularity). If it abstracts from metric properties and represents qualitative aspects of spatial information (specifically topological and ordering information), we refer to it as “schematic map”. Schematic maps as characterized by Klippel et al. [22] abstract from information irrelevant to a specific task considered. For example, a schematic floor plan giving directions to visitors typically abstracts from furniture, doors, etc. One may even abstract from the shape of a room. Indeed, a schematic floor plan represents salient boundaries to free space, for example the outline of rooms, corridors, etc. Schematic maps can however be designed for arbitrary environments featuring a great variety of objects. Therefore, approaches to schematic map interpretation by means of recognizing specific objects or specific spatial properties are restricted to specific environments. We are interested in the fundamental principles of recognizing spatial environments and, henceforth, do not aim at recognizing hallways, doors, etc. This allows responding to arbitrary environments and maps.

By considering abstract sensor readings retrieved from a schematic floor plan in relation to a robot capable of scanning the borderline of free space (e.g. by means of a range sensor) we characterize the localization task using a schematic map. First, as with any absolute representation, schematic maps provide information free of a specific perspective. Second, the granularity of schematic maps is coarser than of sensor information. However, not all differences between metric maps and schematic maps are due to reduced resolution. When transforming a metric map into a schematic map, it undergoes a selection process that only retains salient, characteristic information (compare [3]). For a robot not capable of distinguishing different kinds of obstacles the relation between sensor information and schematic map information presents a small modality change, as the set of objects that can be perceived by the robot differs from the set of objects registered in the schematic map. Therefore, we classify the localization using a schematic map on the generalization strand different perspective, different resolution, same sensor towards GSL (see Fig. 1).

3 Localization Strategies

In this section we will analyze approaches to SL that employ some kind of map representation. We will evaluate their applicability to more generalized localization tasks. First, the SL problem is decomposed into distinct subtasks and aspects; this decomposition provides a classification scheme for individual approaches to SL.

3.1 Computing the Pose

The challenge in SL is to find a sensible transformation from an agent-centered perspective to a specific reference system, typically an absolute one. Therefore, SL primarily is a question of spatial reasoning. On a closer look, additional aspects emerge, though.

A robot can localize itself by determining the correspondence between its sensory input and the map. In other words, we compute the pose which — according to its map — explains the sensory input. The problem of determining this correspondence is termed the *correspondence problem* or the task of *data association*; a good solution to the correspondence problem is among the hardest problems in mobile robot navigation [40, 20]. Once a correspondence between perceived features in their local frame of reference and map features in the absolute frame of reference is established, simple trigonometric computation yields the robot's absolute pose. Important criteria of the applicability of specific approaches are the robot's perceptual features. The ability to uniquely identify landmarks, for example, would make the correspondence problem trivial. Industrial applications sometimes use unique artificial tags to simplify recognition in a robot's working environment [18, 15]. In the present chapter, however, we will consider unaltered environments, though.

To approach the correspondence problem if a — possibly vague — pose estimate is available, matching algorithms are employed. These algorithms calculate the most likely correspondence between the sensory input and the expected perception on the basis of the pose estimate and the internal map. On the basis of this correspondence they infer the expected percept. In the context of statistical frameworks for robot localization the role of matching algorithms is providing a solid perceptual model to infer the probability of each individual pose hypothesis (compare [39]). The more robust a correspondence can be determined, even in absence of precise pose estimates, the fewer hypotheses need to be considered; this improves efficiency. Differences between true and estimated robot perspective result in differences between actual and expected percept. Robustness of matching algorithms is important, especially in the context of GSL. Here, variations may also appear due to shifts of modality or granularity.

To sum up, the key challenge in map-based localization is to find a good solution to the correspondence problem. There are four essential factors that shape approaches to localization:

- **Feature representation:** Which features are made explicit in the map? (sensor reflection points, extracted feature points, ...)
- **Representation of configurations:** Which spatial relations are made explicit in the map? (qualitative knowledge, metric data, ...)
- **Spatial reasoning / configuration matching:** Which matching algorithm is used? (Iterative Closest Point, shape matching, ...)
- **Temporal reasoning:** How is history information handled? (stochastic estimators, conceptual neighborhoods, ...)

In the following sections we will discuss these factors in some detail.

3.2 Feature representation

Sensor data is interpreted in terms of environmental features. Features can range from hardly interpreted sensor patterns to complex objects and their properties. The manifold of features possible can be classified into spatial properties (e.g. position, size, shape) and non-spatial properties (e.g. color, object category). In the following we will focus on spatial features in unprepared environments that can be perceived by robots as well as by humans. Though exploitation of non-spatial properties would support the recognition processes and would complement spatial information, intelligent processing of spatial information is one indispensable ingredient to successful localization.

The choice of features to be used for localization depends on the type of sensors; applicability to GSL adds further requirements. In external representations such as schematic maps a coarse level of granularity entails a complete lack of unimportant features whereas other features may be schematized, i.e. they are coarsely represented. To successfully match information on different levels of granularity, means for determining the saliency of a feature and means for shifting the level of granularity are required. Determination of saliency allows to estimate whether a feature at hand will be represented on a specific level of granularity or not; means of shifting granularity levels are required to identify correspondences. Proceeding from simple to more complex features we examine these properties as well as the contribution of a specific feature to robust localization.

Raw sensor patterns

A prominent approach relying on matching sensor data is the ‘view-based approach’. It matches raw sensor images and does not extract features from sensory input. Typically, sensor snapshots are obtained and stored for different discrete view points. For example, Franz et al. [12] handle linear panoramic camera images taken at specific locations in the environment. Similar to the view-based map representation, the lowest level of Kuipers' spatial semantic hierarchy ([24, 25]) associates the robot's action patterns at decision points with the corresponding locations.

Uninterpreted data does not allow for granularity shifts and cannot be integrated with external information. Furthermore, uninterpreted data provides no information about the local spatial configuration; data can only related to the view point.

Landmarks

Landmarks are objects in space that are easy to identify; for localization purposes, they can be represented by their position. Landmarks are typical environmental features for localization in human navigation (see e.g. [8]). Landmarks are well-researched in the context of human navigation, but the detection of landmarks that are commonly used in human communication (e.g. "the gas station") is not yet possible in computer implementations. Landmarks that can be used in robotics still must be comparatively simple. For example, Forsman [10] developed a tree detection approach on the basis of range data; it was tailored to an outdoor park scenario. Similarly, corners detected in the environment can be used as landmarks [1]. In human-robot communication it is desirable to identify entities in the environment that provides both species a spatial reference for their interaction.

Specific landmark identification approaches restrict applicability to environments that contain those landmarks. It is however possible to derive additional information from landmarks which can be used, for example, to estimate their appearance in a representation at a specific level of granularity. The utilization of landmarks in human-robot interaction is still a challenge; its solution depends on sophisticated object recognition which is still beyond reach.

Free space

The boundary of free space is of special importance to robots and humans since it limits the accessible environment and it constrains possible actions. Consequently, many approaches represent free space, its boundary, or geometric features derived from it. Information about free space also can be obtained from maps that are used by humans. Sensors like laser range finders or sonars measure the boundary of free space directly. We will now review the most important features for representing boundaries of free space.

Cell occupancy

In cell occupancy representations, spatial cells are classified as occupied or free. The spatial domain is partitioned into square-shaped cells of fixed size (e.g., 10cm x 10cm). The typical map representation employed is the so-called occupancy grid [34]. This technique is particularly popular when using range sensors like laser range finders (LRF); sensor output can be used directly without processing (other than noise filtering). A clear advantage is the universality of the approach, as it can be used in arbitrary environments. The simplicity entails severe limitations, though. Occupancy grids are basically bitmap images that, if related to externally provided maps, would require sophisticated image processing techniques for matching. As of today, 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 e.g. [23]).

Free space boundary

Reflection points measured by a range finder represent the boundary of free space. To capture a wider context than single points and to reduce the amount of data, points can be grouped to geometric primitives. For indoor environments grouping into line segments is especially popular (e.g. [32, 36, 7, 9]). In connection with communication tasks it may be desirable to identify salient boundary configurations. A starting-point for defining saliency is given by considering the size of configurations, e.g. the length of a line.

Existing grouping approaches are limited to environments whose boundaries present mostly straight lines. To achieve more universal applicability, Wolter & Latecki [45, 46] propose to use polygonal lines to approximate arbitrarily shaped boundaries. In this way, the universality of point-based representations and the compactness of abstract geometric features can be retained. Feature saliency based on shape complexity and an approach to schematization complex shapes have been proposed by Barkowsky et al. [3].

Routes

A prominent geometric feature derived from free space is the Generalized Voronoi diagram (GVD) [30]. The GVD represents the medial axis of free space ("skeleton"), the set of all points equally and maximally apart from the nearest boundaries. Each point of the GVD is the center of a circle inscribed in the free space that touches at least two points of obstacle boundaries. 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 Voronoi paths emanating from a given point on the GVD. GVGs offer abstract and compact means for representation [38]. Furthermore, routes that follow the GVD are maximally safe as they maintain maximum distance to obstacles. However, the graph structure of GVGs is susceptible to noise in input data; the problem of robust recognition on the basis of GVGs has not yet been solved. It is not yet possible to handle the absence of environmental features in external maps when matching them to perceived information, as the graph structure changes fundamentally when objects disappear. The applicability of GVGs to place recognition depends on improvements in handling multiple levels of granularity and in skeleton-based recognition. These topics are currently under investigation (see [44]).

3.3 Representation of configurations

A configuration describes the spatial arrangement of features that can be perceived in the environment. Frequently coordinate systems are used to represent the position of objects, but qualitative spatial relations describing relative positions (e.g., "A is north of B") or topology information may also be used.

Qualitative representations

Qualitative representations employ a finite, typically small set of relations to model spatial information. Relations usually describe by means of relative information as obtained by comparison; for example, "north of" and "south of" can serve as qualitative relations acquired by comparing the geographic location of two objects.

Some authors confide the set of potential relations to a single connectivity relation, topology (among others, see [6, 25, 48]). Topological information captures connectivity information of distinctive places and can be represented by an (attributed) graph structure. For example, Yeap & Jefferies [48] represent connectivity of local maps. Graph labeling is required to enable agents to identify individual edges that meet in a single node of the graph. Yeap & Jefferies associate edges with exits of the local maps. Kuipers [24] labels directed edges by robot commands. The execution of an action associated with an edge takes the robot from one node to the other. In contrast, Franz et al. [12] use directional information to label edges. Hereby, directions are determined by the relative positions of the two nodes connected. 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 relational representations cannot be made.

Ordering information is another important representative of qualitative information in navigation. Schlieder [37], for example, represents the cyclic order of point-like landmarks and Barkowsky et al. [2] utilize cyclic order of extended landmarks in non-cyclic environments. Cyclic order of perceivable objects has also been used to instruct a mobile robot by means of a schematic map [47]. The self-localization approach proposed in the present chapter utilizes cyclic ordering, as well.

Qualitative representations have been claimed to provide adequate means for communicating spatial information; Moratz & Tenbrink [33] utilize projective relations between objects in a robot instruction setting. A robot is instructed to move to a position described by qualitative relations. This task is strongly connected to the localization problem.

Qualitative calculi

Qualitative calculi extend qualitative relations by introducing means to "calculate with relations", e.g. to infer, if the relations holding between A and B & B and C are known, which relation holds between A and C (relation composition). To relate spatial relations, reasoning — often based on relation composition & constraint propagation — is applied. With respect to correspondence determination, constraint-based reasoning could be exploited to prune the search space. A mapping of objects is only admissible, if it is consistent with qualitative constraints posed on the objects. Thus, qualitative calculi can be employed to introduce hard constraints in correspondence computation (compare [42]). Additionally, conceptual neighborhood structures (see Sec. "Spatio-temporal reasoning" on page 16) have been introduced for qualitative reasoning. Conceptual neighborhoods are in particular valuable to resolve conflicts on the symbolic level by defining an interrelation on the level of relations. However, the application of qualitative reasoning to the correspondence problem, e.g. by means of constraint propagation (see Sec. 3.4) has not yet been thoroughly investigated.

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, e.g. distance information sensed by a range finder, can be mapped directly to a quantitative representation. The most prominent form of quantitative representation is coordinate-based geometry; landmark positions, for example, are represented as points in the Euclidean plane. Most approaches in robotics represent positions as coordinates in the absolute frame of reference given by the global map (see Thrun [40] for an overview).

Generally speaking, in quantitative representations all available information is maintained while in qualitative approaches some details may be intentionally discarded. In quantitative approaches all values are treated equally and no aspects are made explicit. This can hamper recognition, as a small example on coordinate-based geometry shows. Consider an agent that observes two landmarks that are located close to one another. By measuring their position the agent determines two similar coordinates that are both subject to measurement errors. By evaluating the measurements and taking into account the error margins, we may not be able to decide which of the landmarks is located on the left and which is located on the right. The agent 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 we 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.

3.4 Matching

Matching establishes the correspondence between observed features and features represented in the robot's internal map.[3]. A transformation from an agent-centered to the absolute frame of reference can then be computed on the basis of correspondences between observed features and map features. In other words, by establishing the correspondence the agent is localized.

The correspondence problem is challenging in three regards: obtaining a feasible solution, handling uncertainty, and integrating spatio-temporal knowledge. In the following we will review strategies addressing these problems and we will analyze how these strategies meet the requirements of GSL.

Achieving feasibility in data association

Considering a map containing n features and an observation comprising m features, there are

$$\sum_{i=0}^{n} \binom{n}{i} \cdot \binom{m}{i} \cdot i! \tag{1}$$

potential correspondences if observed features are not necessarily represented in the map 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. Confident knowledge, for example, can be exploited in terms of hard constraints restricting the search space. If a pose estimate is available, the *projection filter* [32] can be employed to disregard map features that are estimated 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; this would affect the matching result. In many robot applications pose estimates are provided by odometry.

Computational complexity can be further reduced, if distinguishable features are exploited. For example, using extremes in range scans, Lingemann

[3] In the case of SL by means of feature tracking (e.g. [36, 31]), the agent's previous observation assumes the role of the internal map in map-based SL.

and Hertzberg [31] restrict consideration of correspondences to features of the same type (minimum or maximum). If uncertainty in feature classification is an issue, a feature similarity measure is used as heuristic, i.e. similarity provides a soft constraint, and matching is transformed into a discrete optimization task that assigns the most similar features to one another. In the case of using occupancy as feature, feature similarity considers difference of cell occupancy; this difference is typically represented as probability value [19, 40]. Utilization of complex features allows for fine distinctions in the similarity measure and yields both efficient matching and robustness. In our approach we will argue for shape features that represent the boundary of free space to exploit distinctive shape similarity in the matching procedure.

An alternative approach to increase the efficiency of the matching procedure is to respect the spatial configuration of observed features in relation to the configuration of map features. Admissible mappings from perception to the map preserve the configuration of the features. This can, in principle, be achieved similarly as in constraint propagation (compare [42]), treating relative position 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 model confident knowledge. In our approach, we utilize circular order of visibility as a source of certain information (compare Sec. 5.2). Notably, the application of the Mahalanobis distance for pruning potential candidates can be interpreted as an application to constraint propagation. Here, correlations of distances are exploited for gating in a statistical framework (compare [35]). To our knowledge, constraint propagation has not been further utilized in this context and remains an open research issue. Instead, correspondences are sometimes pruned in a successive step; correspondences which entail a transformation from an agent-centered to an absolute frame of reference that deviate significantly from the transformation obtained by averaging the individually obtained transformations can be removed [16].

To avoid costly computation of robust matching, some SL approaches handle the correspondence problem indirectly. They seek to directly determine the robot pose which explains the percepts (e.g. [40, 17, 32, 5, 7]). In this family of approaches, the robot pose is no longer derived from the discrete correspondence problem; instead, it is obtained by a continuous optimization search for an optimal pose. A pose estimate is required as a start value. Within each step, a simple but fast matching procedure relates perceived features already transformed to the absolute frame of reference to map features. Typically, nearest neighbor algorithms are applied to perform the matching [17, 32, 5, 7]. Embedded in an iterative optimization framework, erroneous results of the matching algorithm can be recovered in successive steps. Notably, all optimization algorithms are susceptible to local minima and erroneous matching can further affect the overall performance. Therefore, this family of approaches relies on a high quality pose estimate as start value.

Handling uncertain information

Inescapable uncertainty in real-world data inhibits perfectly congruent correspondences. Therefore, the goal must be to find those correspondences which explain the agent's observations *best*. This requires integrating differences of assigned features on the level of feature appearance and configuration. The most successful approaches today use statistical methods to "explain" and correct for these differences (see Thrun [39, 40] for an extensive overview). The role of matching algorithms in a statistical framework is to determine the degree of belief in a specific hypothesis of observation, robot pose, and map appearance [19].

Statistical models also are helpful to handle uncertainty beyond sensor noise, e.g. sporadical errors in feature detection — given that a stochastic distribution can be found to model this phenomenon. Hähnel et al. [19] regard a uniform distribution as sufficient to handle erroneous measurement of individual laser beams by a laser range finder. However, in cross-modality, granularity, or perspective shifts of GSL it is unclear if and how differing appearances for a specific source of abstract sensor readings can be adequately modeled by means of a probability distribution. For example, it appears impractical to model which perceived objects are registered in a schematic map. Therefore, we argue for an additional utilization of qualitative knowledge in GSL which, by advancing to a more abstract representation, allows disregarding deviations on a fine level of granularity.

Spatio-temporal reasoning

Spatio-temporal reasoning ties spatial and temporal information together. The possible sequences of physical robot locations and orientations constrain hypotheses about its actual and future pose; therefore spatio-temporal reasoning is an important ingredient to determining the pose of a robot.

In robotics, spatio-temporal reasoning often is tightly coupled with stochastic models to represent uncertainty. Therefore, robot movements are modeled stochastically. SL can then, for example, be approached by means of Markov processes [21] or Monte Carlo methods [41, 11]. This is advantageous in a stochastic framework of SL, but likewise to the aforementioned considerations it is questionable how to express spatio-temporal constraints when information on different levels of granularity needs to be interrelated. In qualitative representations, changes on the level of qualitative information can be represented by discrete *conceptual neighborhood* [14, 13] structures of qualitative spatial relations. Conceptual neighborhoods denote transitions between qualitative relations. Two relations are neighbored, if and only if they can be directly transformed into each other by steady motion. For example, when distinguishing four cardinal directions, "north" and "west" are conceptual neighbors, but "east" and "west" are not. If, for example, a landmark is expected in direction "north" but cannot be observed, this conflict may be resolved most easily by searching in the conceptually neighboring directions "west" or "east". Conceptual neighborhoods allow expressing spatio-temporal constraints in terms of admissible transitions on the qualitative level.

3.5 Conclusions

In reviewing the variety of map features, we identify three main categories of information sources used in localization: features represented in absolute maps, sensor patterns in egocentric observations, and approaches completely abstaining from map representations (see Fig. 3 for an overview). Completely abstaining from maps either requires employing position sensors like GPS — this requires external information to establish a frame of reference — or to incrementally determine the robot's movement by tracking static features and updating the pose estimate. In principle, any approach to feature tracking can be related to a map-based approach, if considering a map that exclusively represents the last observation. However, we are interested in approaches that allow expressing the robot's pose in an externally supplied reference system, i.e. a schematic map. Thus, approaches handling allocentric map information are most adequate.

There are two principle alternatives in map features to choose from, namely features that represent landmark positions and features that describe free space (either directly, e.g. occupancy grids, it's boundary, e.g. line-based maps, or derived geometric information, e.g. Voronoi diagrams). Considering maps commonly used by humans we conclude that landmarks and representation of free space are both suitable choices, boundary of free space being the more fundamental feature, though. Moreover, landmarks typically used by humans are difficult to identify for a robot. Therefore we suggest anchoring map representations on a representation of free space boundaries.

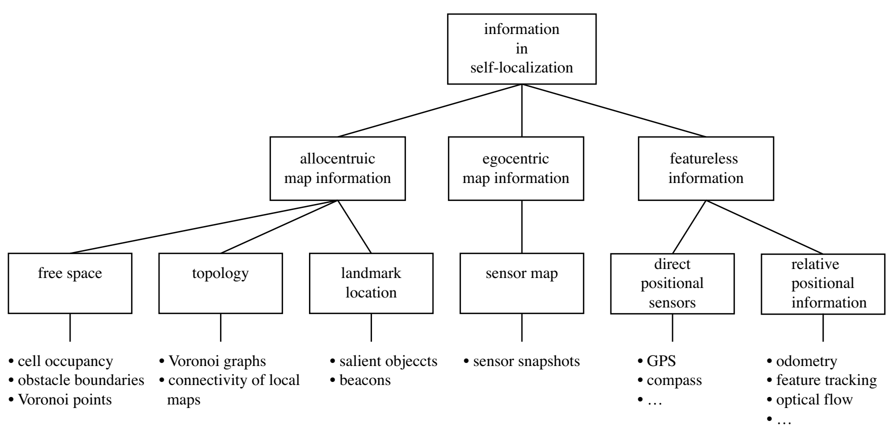


Fig. 3. Categories of information employed in localization

With regards to representing configurations, we reviewed relational approaches which link features by means of a graph, qualitative approaches describing the relative position of objects, and quantitative approaches employing coordinate systems. An overview of this classification is presented in Fig. 4. Quantitative approaches support expressive and precise pose representation. Relational and qualitative approaches, on the other hand, are valuable for handling spatial information on a coarser level of granularity. They abstain from metrics and, by doing so, avoid inescapable differences on the metrical level, e.g., if two configurations on a different level of granularity are related. In localization tasks employing coarse external maps we therefore propose to explicitly integrate qualitative or relational information.

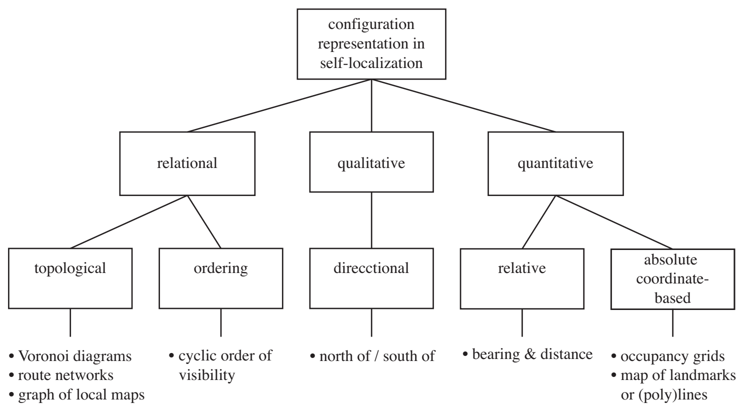


Fig. 4. Categorization of representing configurations

Many matching algorithms employed in robotics align perception and map by means of continuous optimization which searches for the pose value which best aligns perception and map. The correspondence problem is eclipsed. This approach has two major drawbacks. First, they require a pose estimate to start the search. In the case of using an external map no good start 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, i.e. when features identified in the sensor information are not registered in the map, or vice versa. This may be the case, for example, if we use a schematic map.

We suggest focusing on the correspondence problem in order to find an optimal correspondence between perception and map. The problem may then be formulated as a discrete optimization problem that can be solved analytically, i.e. without the risk of getting stuck in a local minimum. Certain information can be explicitly introduced by means of qualitative information, its exploitation allows for an efficient approach. In summary, by incorporating qualitative constraints on spatio-temporal processes, on one hand, and by relaxing requirements on insignificant distinctions, on the other hand, we can considerably reduce the number of alternatives that must be taken into consideration. This approach resembles knowledge-based hypothesis matching in natural cognitive systems more closely and considerably cuts down the computational complexity.

4 Spatial representation based on shape information

In our approach, the spatial representation utilizes shape features that describe the boundary of free space as basic map entities. Shapes are represented by configurations of polygonal lines. In these configurations, scene features are simultaneously related by qualitative ordering information and by quantitative position information. In the following we refer to this approach as shape-based localization or, shortly, SBL. This section presents details on the construction of its underlying representation.

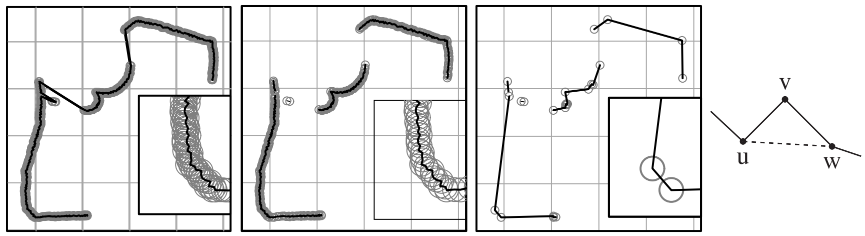
From the sensor readings of a range sensor we extract shape information as polygonal lines, termed *polylines*. Polylines resemble the discrete structure of sensor data; they allow us to approximate arbitrary contours with arbitrary precision. SBL differs from other approaches to extracting complex features in that it is parameter-free and does not require a noise model of the sensor[4]. All control-values are determined adaptively, but preset values reduce computational cost. In the following, we will present a brief description of the OA algorithm; for an extensive description refer to [45, 46]; intermediate stages of the shape extraction process are shown in Fig. 5.

4.1 Extracting shape features from range information

Shape extraction starts by grouping sensor reading points. The maximum distance between neighboring points within a single polyline is controlled by a threshold. Ideally, each polyline represents a single object in view and each object is represented by a single polyline. As different view points and noise can cause different groupings, we need to account for differences in later processing stages. When we match perceived shapes against the map, we allow for re-joining and splitting polylines. The threshold that controls the grouping is chosen to resemble an assumed minimum object distance of 10 cm.

To obtain a compact representation of shapes without loss of important shape information and to cancel the effects of sensor noise, we apply Latecki's & Lakämper's Discrete Curve Evolution method (DCE) [26]. DCE describes a context-sensitive process of evolving polylines by iterative vertex removal.

[4] Veeck & Burgard [43] also suggest to use polygonal lines. Their approach requires an accurately aligned set of scans as starting point of their computation. In contrast, we pursue incremental map construction [46, 45]



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

Fig. 5. States in extracting shape features; grid size is 1m x 1m. (a) input data obtained from a range sensor in an indoor environment, (b) grouping, (c) application of DCE, (d) vertex labels in the relevance computation. Framed boxes in (a), (b), and (c) show enlargements.

A vertex relevance measure is defined to determine individual vertices' contribution to the shape information; the measure can be computed locally. It is defined for neighboring vertices u, v, w (see Figure 5 (d)) as

$$r(u, v, w) = d(u, v) + d(v, w) - d(u, we) \tag{2}$$

where d denotes Euclidean distance.

After vertex removal, the relevance measures of neighboring vertices get updated. Hence, DCE is a fast process (complexity $O(n \log n)$ for polylines with n vertices). In practical use, DCE can process laser range scans consisting of 361 measurements in just a few milliseconds. Besides for noise cancellation, DCE can be used in schematization processes to coarsen the granularity level [3], it simplifies the contour but maintains the overall appearance.

DCE selects vertices to be removed in the context of a single polyline. The identification of relevant vertices can be improved by extending the context to sets of corresponding polylines. To this end, we do not stop the evolution of polylines on the basis of a fixed threshold; rather, we terminate the evolution process on the basis of shape similarity (see Sec. 5.1). Efficiency is improved by first performing DCE without consideration of shape similarity until an intermediate, fixed stop threshold is reached. Then, DCE is continued under consideration of shape similarity. The evolution process for exemplary polylines obtained from a simulated laser range finder is depicted in Fig. 5 (a) – (c).

4.2 Representing configurations

Configurations describe the relative positions of shape features, i.e. polygonal lines. Most importantly, the cyclic order of visibility is represented. As laser range data is ordered cyclically to begin with — i.e. ordered by angle of perception — we simply need to retain the sequence of shape features. We can consider ordering information as reliable information, i.e., there is no uncertainty about the ordering of perceived features. Respecting ordering as a hard constraint in the matching process greatly improves efficiency and robustness. However, if we were to restrict the representation of configurations to cyclic ordering, we would face some limitations. For example, if the map were to contain two objects of identical shape, but only one similar object was found in the sensor data, it would not be possible to determine which of the two objects in the map represents the sensor data. To overcome this limitation, we include metric positional information along with the ordering information.

5 Matching based on ordered shape information

Matching integrates the recognition of individual shape features and the recognition of configurations. We first describe the recognition of polylines which is based on shape similarity. Thereafter, recognition of configurations is described.

5.1 Shape similarity

SBL examines shape similarity to determine potentially corresponding shape features. Shape similarity is modeled by a shape distance measure — the minimum distance of 0 refers to identical shapes.

Shape distance measures play an important role in computer vision, particularly in object recognition. They measure the difference between two shapes and aim at mirroring human intuition. There is a strong connection between object recognition in vision and recognition processes in localization, although the connection between computer vision and robot mapping has not yet been sufficiently exploited according to Thrun [40]. We derived a shape distance measure from state-of-the-art shape matching used in computer vision [28, 26]. To tailor the approach to the domain of range data, some adaptations have been made (for details see [28, 27]).

The idea of measuring the distance between a polyline p and a model q is to disregard irrelevant features that make polylines dissimilar from one and another; in other words, we focus on the subset of vertices that exhibit maximal similarity. Therefore, the measure has been termed *partial optimal similarity* [28]. Here, p corresponds to a polyline extracted from LRF data (which may still contain some noise), whereas q will be a matching candidate extracted from the map. The map is typically derived from multiple observations; we consider map data as absolute reference. The algorithm proceeds as follows: Evolution by means of DCE is continued for polyline p while a simplification of p improves the similarity to q.

The basic similarity measure for comparing simplified ps and qs as detailed in [45, 27] establishes an optimal correspondence of maximal arcs and accumulates differences in relative angular directions. Optimal correspondence of arcs is computed by means of Dynamic Programming — see [26] for details.

5.2 Matching configurations

Provided we have two configurations of features; the task of the matching algorithm is to determine a sensible correspondence relation on the level of polylines. In SBL the currently observed configuration is related to the configuration extracted from the map using the robot's last location as view point. In other words, we do not make use of odometry information to achieve a pose estimate. Due to the distinctiveness of the shape information and the sensibility of the shape distance measure we do not require such pose estimate [45]. Differences of perceived configurations are small on the qualitative level of ordered shape features if the robot has not traveled too far (e.g. less than 1m). These differences can easily be handled by configuration matching.

Matching is formulated here as a discrete optimization problem. We seek to determine the optimal correspondence of shape features. When matching two configurations, changes in the environment, variations of perspective, or noise can cause differences. Constraints and observations that must be considered are as follows:

- Only polylines showing similar shape may correspond.
- The cyclic order of shape features must not be violated. For example, when finding corresponding counterparts for polylines p and q, where p proceeds q, p's counterpart must also proceed q's counterpart.
- An object's visibility can change. Therefore, some polylines may need to be disregarded.
- Correspondences are not necessarily of the type 1-to-1 due to different outcomes of the segmentation process. Instead, 1-to-n, n-to-1, and n-to-m types of correspondence must also be considered.
- Each potential correspondence of two polylines induces an alignment that would adjust the complete shapes involved. We demand that all alignments induced by corresponding polylines are consistent.

We now formulate the discrete minimization problem. Let $S^\star$: polyline $\times$ polyline $\rightarrow \mathbb{R}^+ \cup \{0\}$ denote the shape distance measure described earlier. We will denote configurations, i.e. cyclic ordered vectors of polylines by $\mathbf{P} = (p_1, p_2, \ldots, p_n)$ and $\mathbf{Q} = (q_1, q_2, \ldots, q_m)$ respectively; a sub-vector $(p_i, p_{i+1}, \ldots, p_{j)}$ will be denoted $P_{i,j}$. $P_{i,i}$ will be abbreviated P_i. Sub-vectors represent a single polyline composed by concatenating a sequence of polylines; they are introduced to correct segmentation differences. Furthermore, let $\sim$ denote the relation of correspondence which pins polylines from two configurations together. Our aim is to compute the *optimal* correspondence relation $\sim$.

The quality of a match $\sim$ is determined as the sum of corresponding polylines' shape distances. To compute the optimal match as an optimization process, a penalty for not finding a polyline's counterpart is introduced; otherwise, the empty correspondence relation would yield 0, the lowest possible value, i.e. the optimal choice. The counterweight used is a penalty function

R : polyline $\rightarrow \mathbb{R}^+ \cup \{0\}$ that grows linearly with the polyline's angular size in the field of view. A linearly growing penalty reflects the observation that the shape distance of two polylines that differ only by independent noise grows linearly, too ([45, 46]). This penalty function also addresses feature saliency by consideration of size. Preferring larger features over smaller ones is advantageous in matching a perceived configuration with many details against a schematic map which only presents salient shape features.

The observation that an object is to the left (or right, respectively) of another object is not affected by noise in sensor data. Cyclic order of visibility can therefore be considered certain knowledge. This allows to exploit order as hard constraint and reduce the search space. Observe that the task of determining the optimal correspondence relation of polylines restricted to only correspondences of type 1-to-1 which respect the cyclic order, i.e. $(p_i \sim q_{i'} \wedge p_j \sim q_{j'} \wedge i < j) \rightarrow i' < j'$, is a standard application of Dynamic Programming [4]. Therefore, the unconstrained search space declared in Eq. 1 is reduced to

$$n \cdot m \tag{3}$$

We now formulate the matching which respects the constraints and observations listed above as a minimization problem and we show how it can be solved by an extended Dynamic Programming scheme.

We require that an estimate for the alignment induced by any pair of corresponding polylines exists. This estimate can either be derived from odometry or it can be computed purely based on shape information (see Sec. 5.3). Let us now assume that such an estimate, i.e. a translation vector $\mathbf{t}$ and a rotation by Φ exists. We denote the 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 $\Delta A(P,Q)$ our experimental system utilizes

$$D(\mathbf{dt}, d\Phi) = ||dt|| + 10 d\Phi \tag{4}$$

Denoting the set of polylines $\{p_i, p_{i'}, \ldots, q_j, q_{j'}, \ldots\}$ not belonging to any correspondence by $\overline{PQ}$, determination of the optimal correspondence relation $\sim^\star$ is formulated as follows:

$$\sim^\star = \operatorname{argmin} \sum_{(\mathbf{P}_{i,j}, \mathbf{Q}_{i',j'}) \in \sim} \left(\overbrace{S(\mathbf{P}_{i,j}, \mathbf{Q}_{i',j'})}^{\text{shape distance}} + \overbrace{\Delta A(\mathbf{P}_{i,j}, \mathbf{Q}_{i',j'})}^{\text{robot pose consistency}} \right) + \overbrace{\sum_{r \in \overline{PQ}} R(r)}^{\text{penalty}} \tag{5}$$

To solve the equation, the Dynamic Programming scheme is extended. To enable detection of correspondences of types 1-to-n, n-to-1, or n-to-m, we introduce an updating step that reconsiders the correspondence determined in the previous step. This overcomes the prefix requirement of Dynamic Programming: Suppose a polyline p shall be matched against two polylines q_1, q_2 which are created by splitting p. In classical DP, the result of comparing (the prefix) q_1 to p cannot be altered in subsequent computation. Thus, if p and q_1 are significantly dissimilar, q_1 is disregarded once and for all. Consequently, q_2 would not be matched either. In our extension to DP, we reconsider q_1 when comparing q_2 and p; this gives us the correct correspondence of p and the concatenation of q_1 and q_2.

5.3 Shape complexity & correspondence quality

Matching correlates two sets of shape features which are expected to have a correspondence relation. In the case of relating significantly different configurations (e.g. relating robot perception with schematic map information or perceptions from significantly different view points) metric information about position of objects is of little help; yet if considered, different metric information can even hinder correspondence association. To overcome this limitation we introduced a shape complexity measure that allows us to perform the matching restricted to salient shape features (compare [45] for details). Matching the subset of the most salient shape features in a configuration is more robust than matching nearly featureless, small shapes. Hence we perform matching as a two-step process. In a first step we only consider the most similar and most complex pairs of the corresponding shape features; we can estimate the metric displacement required for the robot pose consistency measure ΔA. In a second matching step, this knowledge can be taken into account; it allows to robustly associate simple shape features even in significantly different configurations [45].

6 Experimental Comparison

To evaluate different approaches to localization, we set up a simulated environment using a virtual robot equipped with a laser range finder. Simulation allows us to easily measure the performance of individual simulation methods, as the ground truth is known. Additionally, we can systematically alter the environment and other parameters like sampling rate or sensor quality to gain a better understanding of the capabilities of individual methods.

To maintain the focus on spatial aspects we do not incorporate stochastic models; we only determine the most plausible pose. In the context of a hypothetical stochastic framework this would mean that we focus on the development of individual hypotheses. The more reliably a single hypothesis can estimate the robot's true pose, the better a complete system including stochastics performs. Furthermore, incorporating comprehensive uncertainty handling would conceal the ability of judging the performance of spatial representation and reasoning techniques, to a certain extent.

6.1 Experiments & discussion

We examined two experimental setups. The first setup is a typical map-based robot SL task. A simulated robot traveled a total distance of 43.03m in the environment depicted in Fig. 6 (a). The average travel distance of the robot between sensing the environment amounts to 11 mm and the average rotation between sensing amounts to 4.0°. The true map was accessible to the localization methods. Hence, the main challenge of this setup is to robustly extract features from noisy input data and to robustly handle the correspondence problem.

In the second setup we investigated into generalized SL using a schematic map as reference system. The robot traveled along the same route as in the first experiment, localizing once every 104 mm on the average; this entails an average rotation between sensing and SL of 30.6°. In this experiment, the schematic map presented in Fig. 6(b) was supplied to the localization methods. This added an extra challenge to mediating between information present in different levels of resolution and to robustly handle objects that were missing in the map (compare Sec. 2.3).

In the experiments we compare our approach with the following localization methods from different categories discussed in Sec. 3.

- Map-based localization by line matching [7, 32, 16, 17]
- Iterative Closest Point (ICP) used in connection with occupancy grids [5]
- SBL based on shape matching and ordering information [45, 46]

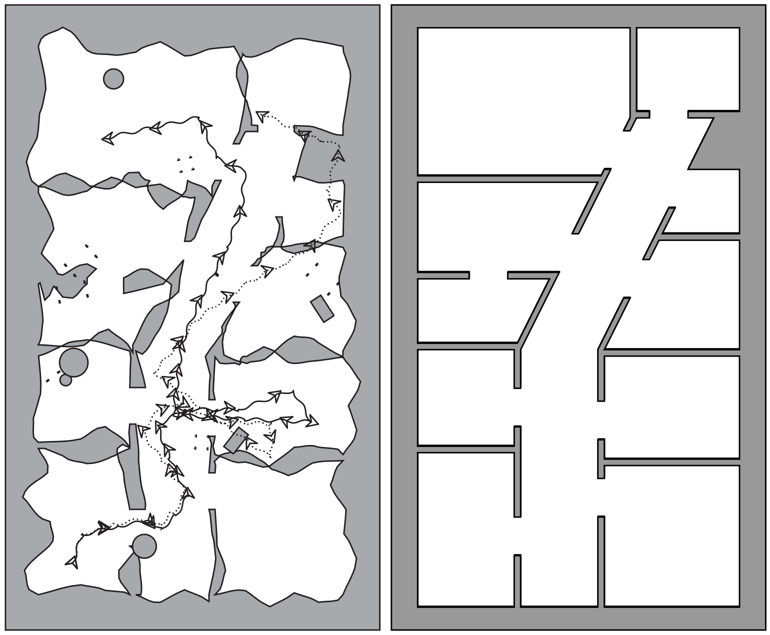
map-based SL:

method	average difference to true position [mm]	heading [°]	proximity test [%]
ICP	534	1.5	21
line-based	5167	65	0.4
shape-based	144	1.21	78

SL using schematized map:

method	average difference to true position [mm]	heading [°]	proximity test [%]
ICP	2234	25.9	50
line-based	1836	16.6	30
shape-based	553	3.3	86

Table 1. Tabular overview of localization results obtained. The proximity test evaluates, if a determined pose is close enough to ground truth. In the map-based localization, the test is passed if the difference is less than 100mm in position an less than 45° in heading. The proximity test for SL using a schematic map allows for a difference in position less than 500 mm and 45° in heading.



(a) (b)

Fig. 6. Experimental setups to test robot self-localization performance. (a) depicts the test environment of aprox. 14×23 meters containing furniture, complex obstacles, etc. The path of the robot (dark line), and the path as reconstructed from the simulated odometry readings (dashed line); (b) shows a schematic map of the test environment.

The methods listed above have been implemented according to the specifications given in the literature. Grid size for occupancy grids in ICP was 50mm x 50mm. We determined the quality of the localization by comparing the differences between true pose (ground truth) and localized pose. A proximity test was applied to compare the deviation between computed pose and ground truth against a threshold. In map-based localization, the test is passed if the position deviation is less than 100mm and the heading deviation is less than 45°. The proximity test for SL using a schematic map allows for a difference in position less than 50cm and 45° in heading. Results presented in Table 1 and in the Fig. 7 and 8 will be discussed in the following.

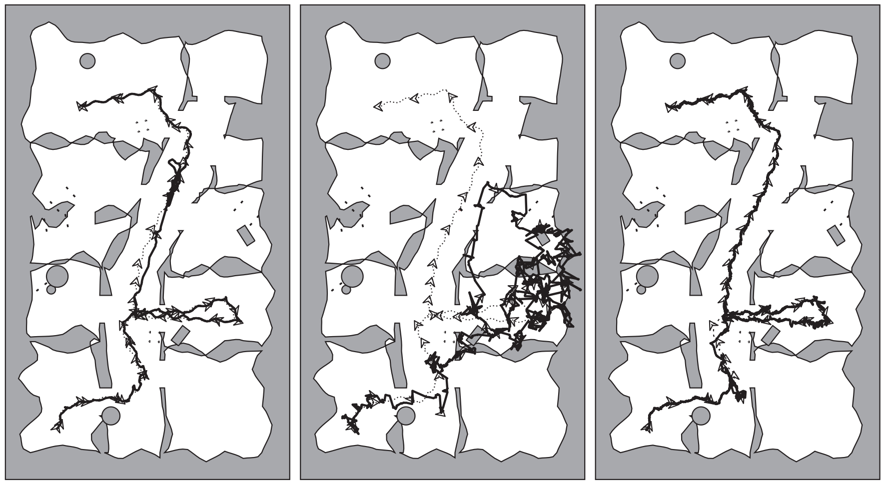
Considering the map-based localization experiment, we observe that line-based localization relying on line detection in the LRF data quickly looses track of the correct path; it passed the proximity test in less than 1% of the cases. At first glance, ICP seems to resemble the robot's true trajectory (see Fig. 7 (a)). However, due to susceptibility for local minima in ICP's optimization process, pose estimates often get stuck in their local surroundings; however, ICP recovers when the robot moves on further. 21% of the estimated poses satisfied the proximity test. Shape-based localization passed the proximity test in 78% of the cases. This demonstrates that our approach is able to robustly perform standard SL tasks.

Using LRF data corresponding to the same test environment as before, but providing a simplified schematic map for localization instead of the true map simulates wayfinding using an overview map. In this setting we relaxed the proximity limit to a distance of 50cm, since sensed LRF data and schematic map differ significantly. We observe a decrease in localization performance which is caused by the large differences between map and perception. ICP met the proximity constraint in 50% of the computed poses, line-based localization succeeded in 30%, and shape-based localization in 86% of the cases. Of these methods, ICP first looses track of the robot's path, but coarsely resembles the true path (see Fig. 8 (a)). An interesting observation is that line-based localization performs better in the localization using the schematic map than in the classical localization task; a reason for this can be seen in the eased line extraction from the schematic map as compared to the true environment map containing mostly non-linear obstacles. However, as regards the average localization error, line-based localization is outperformed by ICP with an average error in line-based localization of about 2.2 m as compared to about 1.8 m in ICP. In contrast, SBL estimates poses with an average error of 0.55m, just about the proximity test threshold of 0.5m. SBL estimates the path closely until the robot enters the last room in the top-left corner. The failure when entering the room was caused by erroneously matching the perceived circular obstacle against the wall registered in the schematic map. Considering the average differences between true and estimated trajectory (see Table 1), it can be concluded that only SBL is able to master the generalized localization setting.

7 Conclusion

We proposed a generalization of the self-localization problem for robots to integrate a variety of localization tasks. These tasks include localization with respect to an externally supplied coarse or schematic map and localization based on route descriptions. We identified three strands of generalization: change of perspective, change of sensor, and change of resolution.

SL approaches are classified with respect to their choice of map features, their representation of configuration information, their approach to the correspondence problem, and their integration of spatio-temporal reasoning. For utilizing external floor plans in a generalized localization task, map features describing the boundary of free space are particularly valuable. The involvement of coarse maps requires an abstraction of configuration information from fine-grained metric details which are meaningless in schematic maps to a qualitative level. We argue in favor of improving matching algorithms to approach the correspondence problem analytically rather than by means of optimiza-



(a) (b) (c)

Fig. 7. Results obtained in the map-based localization experiment. Determined poses and true poses are plotted. (a) ICP, (b) line-based, and (c) shape-based

(a) (b) (c)

Fig. 8. Results obtained in the localization experiment involving a schematic map. Computed and true poses are plotted. (a) ICP, (b) line-based, and (c) shape-based

tion. Analytical solutions do not get stuck in local minima; getting stuck in local minima inevitably occurs when differing views are correlated, as in cross-modal setups or due to a granularity change.

We describe our approach to SL which makes use of expressive shape features. Configuration information makes qualitative knowledge explicit alongside metric information. Qualitative knowledge about cyclic ordering enables design of an efficient analytical approach to the correspondence problem.

In an experimental evaluation we demonstrated the applicability of our approach to standard map-based localization and SL using a schematic map. The experiments highlighted that our approach performs comparably well as often-used ICP-based localization in map-based localization. In the case of SL using a schematic map only the shape-based approach is able to robustly perform localization.

To sum up, several tasks exist that have a close relation to SL and can all be integrated into a more general task definition. For all dimensions of generalization, a sensible abstraction is the key to finding a solution. Sensible abstraction of spatial information can be achieved by including abstract qualitative knowledge and advancing to more expressive map features.

Acknowledgment

This work was carried out in the framework of the SFB/TR 8 Spatial Cognition, project R3 [Q-Shape]. Financial support by the Deutsche Forschungsgemeinschaft is gratefully acknowledged. This work was supported in part by the National Science Foundation under grants NSF INT-0331786 and NSF IIS-0534929. We thank the editors of this volume for their comprehensive review and for their valuable comments that helped us to improve the paper.

References

1. M. Altermatt, A. Martinelli, N. Tomatis, and R. Siegwart. SLAM with corner features based on a relative map. In *Proceedings of the IROS-2004*, 2004.
2. T. Barkowsky, B. Berendt, S. Egner, C. Freksa, T. Krink, R. Röhrig, and A. Wulf. The REALATOR: How to construct reality. In *Proceedings of the ECAI'94 Workshop of Spatial and Temporal Reasoning*, 1994.
3. T. Barkowsky, L. J. Latecki, and K.-F. Richter. Schematizing maps: Simplification of geographic shape by discrete curve evolution. In C. Freksa, W. Brauer, C. Habel, and K. Wender, editors, *Spatial Cognition II - Integrating Abstract Theories, Empirical Studies, Formal Methods, and Practical Applications*, pages 41–53. Springer; Berlin, 2000.
4. R. Bellman. *Dynamic Programming*. Princeton University Press, 1957.
5. P. Besl and N. McKay. A method for registration of 3D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

6. H. Choset, S. Walker, K. Eiamsa-Ard, and J. Burdick. Sensor-based exploration: Incremental construction of the hierarchical generalized voronoi graph. *International Journal of Robotics Research*, 19(2):126–148, 2000.
7. I. J. Cox. Blanche: Position estimation for an autonomous robot vehicle. In I. J. Cox and G. Wilfong, editors, *Autonomous Robot Vehicles*, pages 221–228. Springer-Verlag, 1990.
8. M. Denis. The description of routes: A cognitive approach to the production of spatial discourse. *Cahiers Psychologie Cognitive*, 16(4):409–458, 1997.
9. J. Forsberg, U. Larsson, and Å. Wernersson. Mobile robot navigation using the range-weighted Hough transform. *IEEE Robotics & Automation Magazine*, 21:18–26, 1995.
10. P. Forsman. Feature based registration of 3D perception data for indoor and outdoor map building. In *Int. Conference on Field and Service Robotics*, Helsinki, Finland, 2001.
11. D. Fox, S. Thrun, F. Dellaert, and W. Burgard. Particle filters for mobile robot localization. In A. Doucet, N. de Freitas, and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*. Springer Verlag, New York, 2000.
12. M. O. Franz, B. Schölkopf, H. A. Mallot, and H. H. Bülthoff. Learning view graphs for robot navigation. *Autonomous Robots*, 5:111 – 125, 1998.
13. C. Freksa. Conceptual neighborhood and its role in temporal and spatial reasoning. In M. Singh and L. Travé-Massuyès, editors, *Decision Support Systems and Qualitative Reasoning*, pages 181 – 187. North-Holland, Amsterdam, 1991.
14. C. Freksa. Temporal reasoning based on semi-intervals. *Artificial Intelligence*, 54(1):199–227, 1992.
15. U. Frese. *An O(log n) Algorithm for Simulateneous Localization and Mapping of Mobile Robots in Indoor Environments*. PhD thesis, University of Erlangen-Nürnberg, 2004.
16. J.-S. Gutmann. *Robuste Navigation autonomer mobiler Systeme*. PhD thesis, University of Freiburg, 2000. (in German).
17. J.-S. Gutmann, T. Weigel, and B. Nebel. A fast, accurate and robust method for self-localization in polygonal environments using laser range finders. *Advanced Robotics*, 14(8):651–667, 2001.
18. D. Hähnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose. Mapping and localization with RFID technology. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2004.
19. D. Hähnel, D. Schulz, and W. Burgard. Map building with mobile robots in populated environments. In *Proceedings of International Conference on Intelligent Robots and Systems (IROS'02)*, 2002.
20. M. E. Jefferies, M. Cosgrove, J. T. Baker, and W. Yeap. 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*, pages 232 – 239, 2004.
21. F. Kirchner and J. Hertzberg. A prototype study of an autonomous robot platform for sewerage system maintenance. *Autonomous Robots*, (4):319–333, 1997.
22. A. Klippel, K.-F. Richter, T. Barkowsky, and C. Freksa. The cognitive reality of schematic maps. In L. Meng, A. Zipf, and T. Reichenbacher, editors, *Map-based Mobile Services — Theories, Methods and Implementations*, pages 57—74. Springer, Berlin, 2005.

23. J. Ko, B. Stewart, D. Fox, K. Konolige, and B. Limketkai. A practical, decision-theoretic approach to multi-robot mapping and exploration. In *Proceedings. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-03)*, 2003.
24. B. Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233, 2000.
25. B. Kuipers and Y.-T. Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Journal of Robotics and Autonomous Systems*, 8:47–63, 1991.
26. L. J. Latecki and R. Lakämper. Shape similarity measure based on correspondence of visual parts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(10), 2000.
27. L. J. Latecki, R. Lakämper, and D. Wolter. Shape similarity and visual parts. In I. Nyström, G. S. di Baja, and S. Svensson, editors, *Proceedings of the 11th International Conference on Discrete Geometry for Computer Imagery (DGCI), Naples, Italy*, volume LNCS 2886 / 2003, pages 34–51. Springer, November 2003.
28. L. J. Latecki, R. Lakämper, and D. Wolter. Partial optimal shape similarity. *Image and Vision Computing Journal*, 23(2):227 – 236, 2005.
29. J.-C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Norwell (MA), USA, 1991.
30. D. T. Lee and R. L. Drysdale. Generalization of Voronoi diagrams in the plane. *SIAM Journal on Computing*, 10(1):73 – 87, 1981.
31. K. Lingemann, H. Surmann, A. Nüchter, and J. Hertzberg. Indoor and outdoor localizations for fast mobile robots. In *Proceedings of International Conference on Intelligent Robots and Systems (IROS)*, 2004.
32. F. Lu and E. Milios. Robot pose estimation in unknown environments by matching 2D range scans. *Journal of Intelligent and Robotic Systems*, 1997.
33. R. Moratz and T. Tenbrink. Spatial reference in linguistic human-robot interaction: Iterative, empirically supported development of a model of projective relations. *Spatial Cognition and Computation*, 2006. In press.
34. H. P. Moravec and A. E. Elfes. High resolution maps from wide angle sonar. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 1985.
35. J. Neira and J. D. Tardós. Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on robotics and automation*, 17(6):890–897, 2001.
36. T. Röfer. Using histogram correlation to create consistent laser scan maps. In *Proceedings of the IEEE International Conference on Robotics Systems (IROS-2002)*, 2002.
37. C. Schlieder. Reasoning about ordering. In *Proceedings of the 3rd International Conference on Spatial Information Theory (COSIT)*, 1995.
38. S. Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):21–71, 1998.
39. S. Thrun. Probabilistic algorithms in robotics. *AI Magazine*, 21(4):93–109, 2000.
40. S. Thrun. Robotic mapping: A survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002.
41. S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, 128(1-2):99–141, 2001.

42. E. Tsang. *Foundations of Constraint Satisfaction*. Academic Press, London, 1993.
43. M. Veeck and W. Burgard. 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)*, 2004.
44. J. O. Wallgrün. Autonomous construction of hierarchical Voronoi-based route graph representations. In C. Freksa, M. Knauff, B. Krieg-Brückner, B. Nebel, and T. Barkowsky, editors, *Spatial Cognition IV. Reasoning, Action, Interaction: International Conference Spatial Cognition 2004*, volume 3343 of *Lecture Notes in Artificial Intelligence*, pages 413–433, Berlin, Heidelberg, New York, 2005. Springer.
45. D. Wolter and L. J. Latecki. Shape matching for robot mapping. In C. Zhang, H. W. Guesgen, and W. K. Yeap, editors, *Proceedings of 8th Pacific Rim International Conference on Artificial Intelligence (PRICAI-04)*, volume LNAI 3157 / 2004, pages 693–702, Auckland, New Zealand, August 2004.
46. D. Wolter, L. J. Latecki, R. Lakämper, and X. Sun. Shape-based robot mapping. In S. Biundo, T. Früwirth, and G. Palm, editors, *Proceedings of the 27th German conference on Artificial Intelligence (KI-2004)*, volume LNCS 3238 / 2004, pages 439–452. Springer, 2004.
47. D. Wolter and K.-F. Richter. Schematized aspect maps for robot guidance. In P. Doherty, editor, *Proceedings of the workshop cognitive robotics (CogRob)*, pages 71–76, 2004.
48. W. K. Yeap and M. E. Jefferies. On early cognitive mapping. *Spatial Cognition and Computation*, 2(2):85–116, 2000.