

---

# Mobile Robot Navigation in Indoor Environments: Geometric, Topological, and Semantic Navigation

---

Ramón Barber, Jonathan Crespo, Clara Gómez,  
Alejandra C. Hernández and Marina Galli

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/intechopen.79842>

---

## Abstract

The objective of the chapter is to show current trends in robot navigation systems related to indoor environments. Navigation systems depend on the level of abstraction of the environment representation. The three main techniques for representing the environment will be described: geometric, topological, and semantic. The geometric representation of the environment is closer to the sensor and actuator world and it is the best one to perform local navigation. Topological representation of the environment uses graphs to model the environment and it is used in large navigation tasks. The semantic representation is the most abstract representation model and adds concepts such as utilities or meanings of the environment elements in the map representation. In addition, regardless of the representation used for navigation, perception plays a significant role in terms of understanding and moving through the environment.

**Keywords:** robot navigation, environment modeling, topological navigation, semantic navigation, geometric navigation

---

## 1. Introduction: navigation problem

Navigation of mobile robots has been traditionally understood as solving the problem proposed by these three questions (Levitt [1]):

- Where am I?
  - Where are other places related to me?
  - How do I get to other places from here?
-

The answer to these questions refers to localization to determine where the robot is, path-planning to know how to reach other places from where the robot is, and navigation to perform the trajectory commanded by the path-planning system.

This approach has ruled robot navigation paradigm and many successful solutions have been proposed. However, the advances in technical developments allow a wider thought of mobile robot navigation giving a solution for a “bigger” problem. This solution would give answer to two additional questions, which are:

- How is this place like?
- How is the structure of the environment I am in?

The answer to these new questions focuses on the importance of perception to determine the place and the elements of the place where the robot is and the importance of modeling the environment to determine the structure of the place and infer the connections between places. When robots are integrated in human environments and live together with humans, perception and modeling gain importance to enable future tasks related to service robots such as manipulation or human-robot interaction. In this chapter, we give a conceptual answer to the traditional robot navigation question and to the questions proposed above.

Robot navigation has been tackled from different perspectives leading to a classification into three main approaches: geometric navigation, topological navigation, and semantic navigation. Although the three of them differ in their definition and methods, all of them have the focus of answering the same questions.

From the beginning, authors have focused on generating metric maps and moving through the map using metric path planners. The most well-known algorithm to build metric maps is simultaneous localization and mapping (SLAM) as proposed by Bailey & Durrant-Whyte [2]. Wang et al. [3] use SLAM and Rapidly-exploring Random Tree (RTT) planning with Monte Carlo localization to drive a wheelchair in indoor environments. Pfrunder et al. [4] use SLAM and occupancy grids to navigate in heterogeneous environments. However, as larger maps are considered, it is computationally expensive to keep metric maps and other authors have focused on topological representations. Fernández-Madrigal et al. [5], design a hierarchical topological model to drive a wheelchair and perform reactive navigation and path-planned navigation. Ko et al. [6] works with topological maps where nodes are bags of visual words and a Bayesian framework is used for localization.

Regarding topological navigation, since the first developments, the global conception of the system has attracted the interest of several authors. In Kuipers & Levitt [7], the authors presented a four-level hierarchy (sensorimotor interaction, procedural behaviors, topological mapping, and metric mapping) to plan trajectories and execute them. Giralt et al. [8] defined a navigation and control system integrating modeling, planning, and motion control and stated that the key to autonomous robot was system integration and multisensory-driven navigation. Mataric [9] defined a distributed navigation model whose main purposes were collision-free path-planning, landmark detection, and environment learning. In Levitt [1], Levitt established a visual-based navigation where landmarks are memorized and paths are sequences of

landmarks. These authors tackled for the first time robot navigation concepts from a topological approach and some of them noticed the importance of perception in the navigational processes.

At the last level of the semantic navigation paradigm, the ability to reason and to infer new knowledge is required. In today's world of robotics, there is a general tendency to implement behavioral mechanisms based on human psychology, taking as example the natural processing of thought. This allows a greater understanding of the environment and the objects it contains. This trend is also valued in the field of mobile robot navigation. Navigators have been increasing their level of abstraction over time. Initially, navigation was solved with geometric navigators that interpreted the environment as a set of accessible areas and areas that were not accessible. Then, the concept of node and the accessibility between nodes was introduced, which allowed increasing the level of abstraction generating graphs and calculating trajectories with algorithms of graphs. However, the level of abstraction has increased a step further, introducing high-level concepts that classify the rooms according to more complex and abstract data such as the utility and the objects they contain.

Another important aspect is the collection of the information of the environment, which has to be available for the navigation systems, among other tasks carried out by mobile robots. This information is provided by a perceptual system, therefore non-trivial problems appear related to object detection and place recognition. One of the most challenging issues in scene recognition is the appearance of a place. Sometimes, the same place may look different or different places may look similar. Also, the position of the robot and the variation of its point of view can affect the identification of the place when it is revisited. Environment models and the way to define the navigation tasks can be remodeled taking into account new technologies and trends. This will provide more autonomy to mobile robots and will help the interaction with humans in their usual environments. In this trend, vision is the main sensor for navigation, localization, and scene recognition. Place recognition is a very well-known challenging problem that not only has to do with how a robot can give the same meaning that a human do to the same image, but also with the variability in the appearance of these images in the real world. Place recognition has a strong relation with many major robotics research fields including simultaneous localization and mapping.

## 2. Geometric navigation

Geometric navigation consists of moving the robot from one point of the environment to another one, given those points by its coordinates in a map. The map of the environment is classically represented by a grid of points, and the trajectory between two points is a sequence of points the robot must reach in the given order. The controller of the robot must reach the next point of the path closing a loop with the encoder information about its position (distance and angle). One of the main objectives in the geometric navigation researches is the path-planning task from an initial point to a final point, creating an algorithm able to find a path ensuring completeness.

Although the path-planning task is widely treated in mobile robots, computational capacity has increased exponentially and more complex algorithms can be developed. Algorithms try to find the shortest or fastest path while maintaining safety constraints. Another challenge is to try to get solutions that provide smoother trajectories, trying to imitate the human trajectories.

In the literature, many different algorithms can be found. One of the most important precedents were the works of LaValle [10], where a classification into two big groups depending on the way information is discretized is proposed: combinatorial planning and sampling-based planning. Combinatorial planning constructs structures containing all the necessary information for route planning; see De Berg et al. [11]. Sampling-based planning is based on an incremental representation of space and uses collision-free algorithms for path search. Here are some of the algorithms most used in the path-finding problem.

## 2.1. Deterministic algorithms

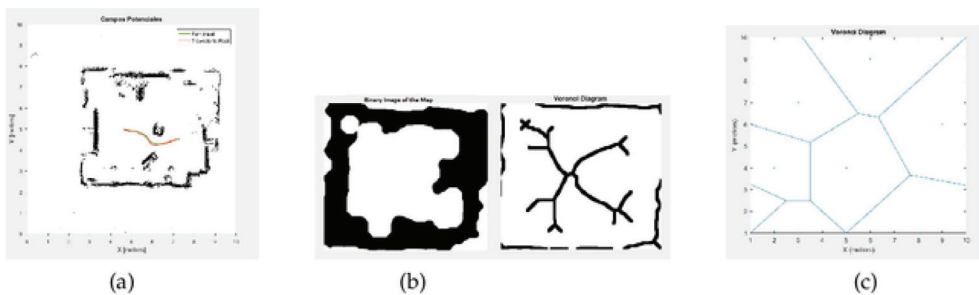
One of the first approaches tries to get all the possible paths between two points and choose the shortest one. Two example algorithms use potential fields and Voronoi diagrams.

### 2.1.1. Potential fields

The potential fields method is based on reactive planning techniques, which can be used to plan locally in unexplored environments. This method assigns to the obstacles similar characteristics that an electrostatic potential might have; in this way, the robot is considered as a particle under the influence of an artificial potential field that pulls it toward a target position, **Figure 1a**. The generation of trajectories is due to an attractive field toward the final position and another repulsive one with respect to the obstacles [12].

In this way, navigation through potential fields is composed of three phases:

- Calculation of the potential acting on the robot using the sensor data.
- Determination of the vector of the artificial force acting on the robot.
- Generation of the movement orders of the robot.



**Figure 1.** Trajectories created using potential fields and Voronoi diagram methods. (a) Trajectory of a real robot created using Potential Fields Method, (b) Trajectories created using Voronoi Diagram by Skeletonization and (c) Trajectories created by Voronoi Diagram Method.

On the one hand, the attractive potential must be a function of the distance to the final destination, decreasing when the robot approaches this point. On the other hand, the repulsive potential should only be influenced when the robot is at a dangerous distance from obstacles. Therefore, the potential fields method allows to be performed in real time, as a local planner, considering obstacles only when the robot is at a minimum distance from them.

The problem that exists in this type of method is the local minimums [13], places where the potential is null but it is not the final position.

### *2.1.2. Voronoi diagram*

The generation of Voronoi diagrams seeks to maximize the distance between the robot and the obstacles, looking for the safest path between two points in the space [14]. In this way, the diagram is defined as the locus of the configurations that are at the same distance from the obstacles.

The algorithm divides the space into sections formed by vertices and segments that fulfill the cost function of maximum distance between obstacles, **Figure 1**. Then, the trajectory is sought from an initial point to the objective. For a more realistic representation, obstacles are considered as polygons, since physically an obstacle is not a point.

One way of building the Voronoi diagram is using image-processing methods (skeletonization) based on the method of Breu [15]. These present a linear (and therefore asymptotically optimal) time algorithm in order to calculate the Euclidean distance of a binary image.

The algorithm is built with the pixels that result after performing the morphological operations on the image, with the nodes being the points where the lines that pass through the pixels of the image intersect.

A trajectory generated by Voronoi diagrams has the disadvantage that it is not optimal from the point of view of length traveled, it can also present a large number of turns [16]. Moreover, this method is not efficient for more than two dimensions.

## **2.2. Probabilistic algorithms**

Over the years, a large number of solutions for the navigation problem have been presented using algorithms with random components, specifically in generic environments and with a large number of dimensions.

### *2.2.1. PRM algorithm*

Probabilistic roadmap (PRM) is a trajectory planner that searches the connectivity of different points in the free space from a starting point to the final goal avoiding collisions with obstacles, using random sampling methods [17].

If the environment in which the robot is located is very complex, this type of algorithm allows finding a solution without a large computational requirement, so it is used in environments with a large number of dimensions.

One of the main problems of PRM is that the solutions it finds do not have to be the optimal trajectory. Also, since the way it generates the nodes is completely random, it produces a nonhomogeneous distribution of the samples in the space. These two disadvantages are seen in **Figure 2a**.

### 2.2.2. Rapidly exploring random tree algorithm

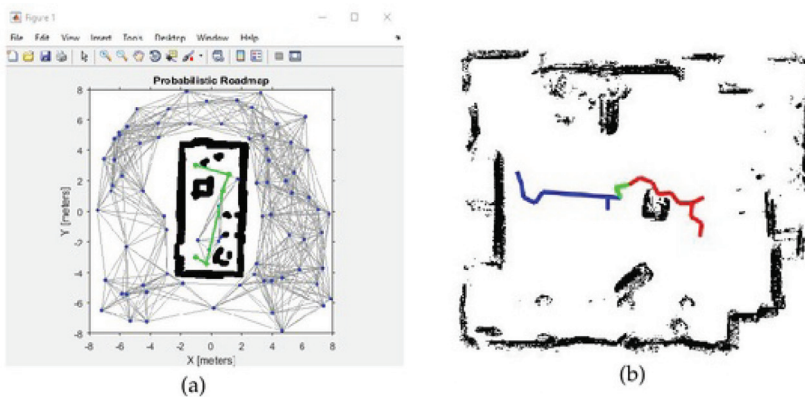
Rapidly exploring random tree (RRT) [18] provides a solution by creating random branches from a starting point. The collision-free branches are stored iteratively and new ones are created until the target point is reached. The algorithm is started with a tree whose source is a single node, the starting point. In each iteration, the tree expands by selecting a random state and expanding the tree to that state. The expansion is performed by extending the nearest node of the tree to the selected random state, which will depend on the size of the selected step. The algorithm creates branches until the tree takes a certain extension approaching the goal.

The size of the step is an important parameter of the algorithm. Small values result in slow expansion, but with finer paths or paths that can take fine turns [19].

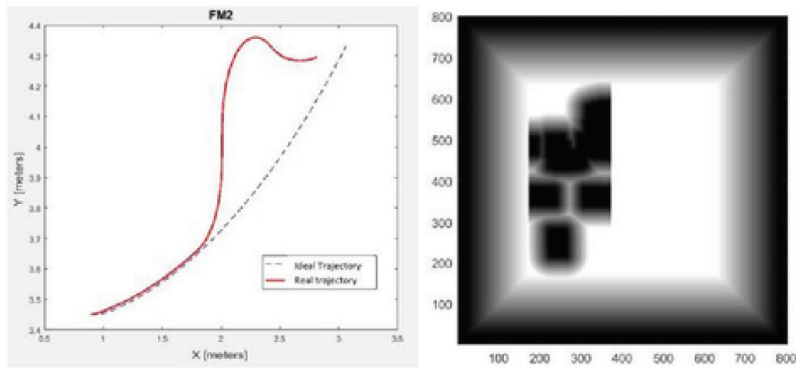
### 2.3. Fast marching square algorithm

Fast marching square ( $FM^2$ ) is a path-planning algorithm, that searches for the optimal path between two points. It uses the fast marching method (FMM) as a basis for calculation. It is a modeling algorithm of a physical wave propagation.

The fast marching method uses a function that behaves similar to the propagation of a wave. The form that this wave propagates, following the Eikonal equation, from an initial point to reach a goal position is the most efficient way in terms of time to reach it. The fast marching algorithm calculates the time ( $T$ ) that the front of the wave, called interface, spends to reach each point of the map from a starting point. The FMM requires as previous step a



**Figure 2.** PRM generated and trajectory created using bidirectional RRT. (a) PRM generated for a real environment and (b) Trajectory created using Bidirectional Rapidly Exploring Random Trees.



**Figure 3.** Trajectory calculated by fast marching square.

discretization of the environment in a grid of cells, on which the Eikonal equation is solved iteratively (**Figure 3**).

The trajectories obtained using FMM present two drawbacks: great abruptness of turns and trajectories very close to the obstacles. This makes it impossible to use the algorithm as a trajectory planner in real robotics. The main change in  $FM^2$  [20] solves these problems by generating a speed map that modifies the expansion of the wave taking into account the proximity to obstacles.

### 3. Topological navigation

Topological navigation refers to the navigational processes that take place using a topological representation of the environment. A topological representation is characterized by defining reference elements of the environment according to the different relations between them. Reference elements are denominated nodes and the relations between them are characterized as arches. The aim of topological navigation is to develop navigational behaviors that are closer to those of humans in order to enhance the human-robot understanding. Summing up the main implications of topological navigation [21]:

- Topological navigation permits efficient planning. However, as they are based on relations, they do not minimize distance traveled or execution time.
- Topological navigation does not require precise localization.
- It is easy to understand by humans due to its natural conception.
- A complex recognition model is needed.
- Huge diagrams are involved in large environments and diagrams scale better than geometrical representations.



Topological representation classifications, as the one proposed by Vale & Ribeiro [22], differentiate mainly two ways of representing the environment: topological representations based on movements, for example the works developed by Kuipers & Byun [23], and topological representations based on geometrical maps, as proposed by Thrun [24]. These two conceptions differ mainly in the spatial relation between the real world and its representation. Regarding topological maps based on geometry, an exact relation between the environment and the representation is mandatory. Every topological node is metrically associated with a position or place in the environment, whereas, in topological representations based on movements, it is not necessary to have a metrical correspondence with the elements of the environment. An example of the same environment represented as a topological map based on geometry and as a topological map based on movements is shown in **Figure 4**.

In topological representations based on geometry, nodes normally correspond to geometrical positions (characterized as  $(x, y, \theta)$ ) that correspond to geometrical events such as junctions, dead-ends, etc. and arches correspond to the geometrical transition between positions. In topological representations based on movements, the relation between nodes is more abstract as it does not have a geometrical meaning, being a qualitative relation instead of a quantitative one. That is why arches can be associated to different and more complex behaviors. In addition, nodes are associated to important sensorial events which can be determined for each application ranging from important geometrical events to landmarks and objects.

Although there are substantial differences between topological representations based on geometry and topological representations based on movements, both of them share the definition of the required modules so the system works efficiently. These modules will be explained in the following subsections, which are: modeling of the environment, planification and navigation, and perception interface.

### 3.1. Modeling of the environment as a topological graph

A topological graph, as defined by Simhon & Dudek [25], is a graph representation of an environment in which the important elements of an environment are defined along with the transitions among them. The topological graph can be given to the system by the user or can be built through exploration. Exploration strategies differ if the system works with a topological



**Figure 4.** Topological representations based on geometry (left) and on movements (right).



representation based on geometry or a topological representation based on movements. In the case of using representations based on geometry, exploration and map acquisition strategies such as the ones explained previously for geometrical representations can be used adding a processing stage to extract the relevant geometrical positions. In the case of representations based on movements, strategies like Next Best View, as proposed in Amigoni & Gallo [26], or Frontier Exploration, as in the work proposed by Arvanitakis et al. [27] can be used. Generally, these representations are translated into a text file that lists the information of the graph in order to maximize the graph efficiency. In **Figure 5**, an example of a map text file structure and its graphical interpretation is shown. In this example, a topological representation based on movements is used so the position of the nodes in the graph is not necessarily linked to the position of reference elements in the environment.

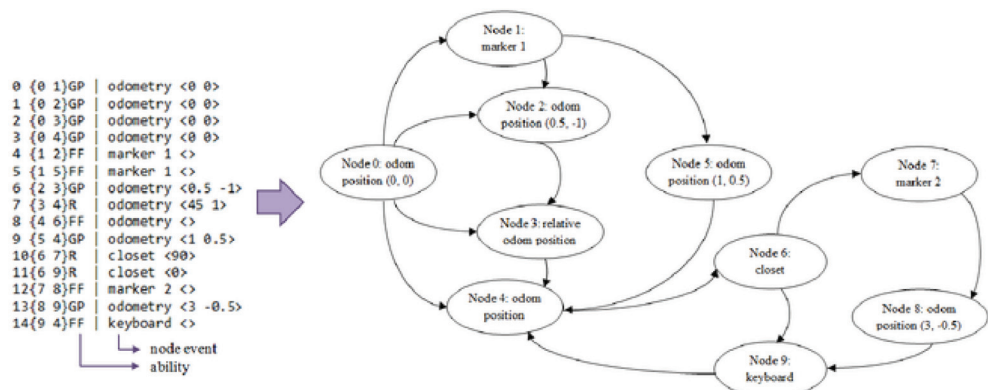
This text file contains all the information required for navigation. Nodes are ordered according to an identifier and they are associated with their corresponding event type (odometry, marker, closet, etc.). Arches are associated with the behavior or translational ability that the robot has to perform (GP, go to point; R, turn; etc.).

A model of the environment can be formed by several topological graphs containing different information or representing different levels of abstraction, in that case the environment is modeled as a hierarchical topological graph.

### 3.2. Topological planification and navigation

Topological navigation behaviors are determined mainly by the path-planning and navigation (meaning strict motor abilities execution) strategies as considered in Mataric [28]. The different nodes conforming the path are associated with the topological place where the robot perceives a sensorial stimulus or where it reaches a required position. In addition, while navigating, the behavior of the robot is subjugated to real-time perception and movement in the environment.

Topological planification is in charge of finding the topological route that the robot has to follow; it has to determine the path that the robot has to perform in order to move between two



**Figure 5.** Example of a topological map and its graphical representation.

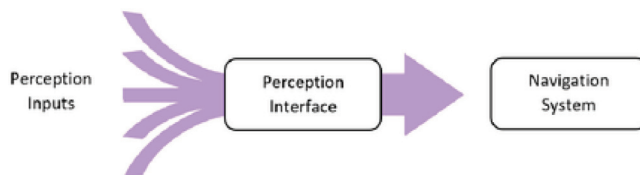
nodes. The path is a subgraph of the original graph of the environment. In order to plan a trajectory in a topological representation of the environment, a graph path-planning algorithm has to be used. There are many algorithms in the literature for this purpose, such as Dijkstra's algorithm, Skiena [29]. The main objective of Dijkstra's algorithm is to obtain the shortest path between nodes in a graph according to some heuristics or cost function. Given an initial node, it evaluates adjacent nodes and chooses the node that minimizes the cost function. This process is iterated until the goal node is reached or every connection between nodes has been explored. Using traditional algorithms such as Dijkstra, many modifications can be implemented to establish heuristics that fit better to real environments. For example, the cost of a translation between nodes can be varied according to previous executions or pursuing a specific global behavior, such as the personality factor proposed in Egido et al. [30].

The navigator is in charge of performing the path and reaching the goal node analyzing events and abilities or positions and transitions. An ability is the order the robot has to execute to reach coming nodes and it is intrinsically related to motor control. An event is the sign indicating the robot has reached a node, through sensorial information or odometrically in the case of representations based on geometry. A navigator can be based on a single behavior or on multiple behaviors. Topological representations based on geometry are all based on a single behavior that can be understood as "Going to point" or "Reaching next position." Topological representations based on movements can be based on single behaviors also or define a set of behaviors that would be used for optimizing the transitions between nodes. A multiple-behavior system contains several strategies and abilities in order to perform navigation behaviors and it should enable the inclusion of new strategies. Some of the common abilities implemented are Go to point, Turn, Contour following, and Go to object. Navigation systems are complemented with reactive obstacle avoidance modules to guarantee safe operation.

### 3.3. Managing the information of the environment: perception interface

If the topological navigation system is designed to work with multiple exteroceptive and proprioceptive events and it has to handle them simultaneously, these events have to be managed carefully. In order to manage the information that will be assigned to the nodes, a new module is needed. In this chapter, we will refer to this module as the perception interface.

The perception interface is decoupled from the system translating the specific information of each perception to a general structure which interacts with the other modules of the topological system, as shown in **Figure 6**. When translating the information to a general structure, the power of the system is multiplied exponentially, as adding new perception types becomes very



**Figure 6.** Perception interface and event manager modules distribution.

simple. Perceptions are mainly based on vision, such as object detection and landmark detection, but some perceptions such as magnetic signals, ultrasounds, or proprioceptive perceptions such as odometry can be used. Another advantage of the perception interface is the possibility of establishing priorities and relations between perceptions easily.

#### 4. Semantical navigation

The current tendency in robotics is to move from representation models that are closest to the robot's hardware such as geometric models to those models closer to the way how humans reason, with which the robot will interact. It is intended to bring closer the models the way robots represent the environment and the way they plan to the way the humans do. The current trend is to implement behavior mechanisms based on human psychology. Robots are provided with cognitive architectures in order to model the environment, using semantics concepts that provide more autonomy, and which helps the navigation to be robust and more efficient.

In theory, any characteristic of the space can be represented on a map. But in general, it tends to identify a map with geometric information more or less complemented with additional information. When a mobile robot builds a map, the techniques used generally ignore relevant descriptive information of the environment and quite close to the process of made by humans, such as the navigation of the environment, the nature of the activity that there is, what objects it contains, etc.

The problem of the construction of semantic maps consists of maps that represent not only the occupation and geometric appearance of the environment but the same properties. Semantics is needed to give meaning to the data, mainly to make the simplest interpretation of data. The application of this idea to the construction of maps for mobile robots allows a better understanding of the data used to represent the environment and also allows an exchange of information between robots or between robots and people, if needed, easily. It also allows to build more accurate models and more useful environment models.

Semantic navigation is another step to the logical representation, it implies an additional knowledge about the elements of the world and it allows the robot to infer new information. For instance, the robot can perform this type of associations: "There is a photocopier in this room-> there must be paper nearby".

Semantic navigation allows the robot to relate what it perceives to the places in which it is located. This way, an environment model is managed using the objects and on the concepts that they represent. All this information is used to classify the place where objects are located and it is used to reach a given location or a specific target. Therefore, the robot can find places or rooms Vasudevan & Siegwart [31] semantically related with the target, even if it is in a little-explored or unknown environment. This navigation level allows to complete the information that is necessary from a partial knowledge of the environment.

Semantic navigation is related to classification methods of the environment object and places, which provides more effectiveness, as is described in Vasudevan & Siegwart [32]. An example

of place identification using a Naive Bayes Classifier to infer place identification is shown in Duda et al. [33] and Kollar & Roy [34]. These works show that relations between object-object and object-scenario can be used to predict the location of a variety of objects and scenarios.

#### 4.1. Knowledge representation of the environment

One of the most important issues is how the robot models the environment. In semantic navigation, a large amount of information related to the environment and to the objects of the environment is used in the environment representation. For this representation, an ontology can be used to define the concepts and the links between objects in the environment. **Figure 7a** shows the concepts of the ontology and its relations.

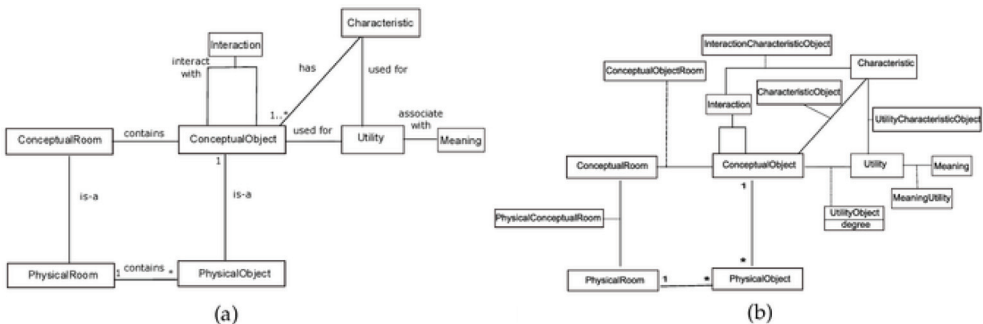
#### 4.2. An ontology design through a database

The ontology design can be implemented using different tools. In this work, a relational model using a database is proposed. **Figure 7b** shows a relational database diagram scheme. This scheme is used to understand the elements of the system and its relations. Tables used to model the ontology are described below.

##### 4.2.1. Basic information tables

Basic information tables are used to store the environment elements. Four elements have been considered: physical rooms (real locations sensorially perceived by the robot), conceptual rooms (the type of room that the robot can recognize), physical objects (objects perceived by sensors), and conceptual objects (each semantic information of the object that the robot must recognize).

- **ConceptualRoom:** This table stores the information of the rooms understood as concepts. The table records are those corresponding to the concepts *bathroom*, *bed room*, *kitchen*, etc.
- **ConceptualObject:** This table contains data related to objects abstractly defined and understood as concepts. For example, *Oven*, *Computer*, *Washbasin*, etc.



**Figure 7.** Ontological model for semantic navigation and its implementation on a database. (a) Ontological model for semantic navigation and (b) Database design for semantic navigation.

- **PhysicalRoom:** It stores specific locations, places in space, regions delimited by walls, having real coordinates, etc., where the robot can move.
- **PhysicalObject:** This table stores each real object that the robot sensor identifies. This table contains specific objects: *Table-1*, *Table-2*, *Computer-1*, etc.

With these tables, objects and places have been modeled. More tables are needed in order to complete the relations and information of the environment that are described below.

#### 4.2.2. Links between primary tables

To complete the model, the following links between the primary tables are needed:

Objects are always in rooms; so, between *PhysicalRoom* and *PhysicalObject* tables, there exists a link. And a physical room may have an indeterminate amount of physical objects. In table *PhysicalObject*, the room associated to each object is also stored.

#### 4.2.3. Tables that allow searches by semantic proximity

As an example of the implementation of the tables, **Figure 7b** shows the design of the database. This database contains several tables that manage important information to help the robot to find objects and relations. With the information of this table, if the robot does not locate a specific object, it can deduce with what other objects it is linked to, helping the robot locate the object. To get additional information about objects, some tables are defined:

- **Interaction:** Objects interact with other objects. This table can handle any type of interaction, although tests have been performed with a limited number of them. The interactions taken into account are: *BE\_A*, when an object is a subtype of another object; *IS\_USED\_WITH*, when an object is used with an object; and *IS\_INSIDE\_OF*, when usually an object is inside another object.
- **Utility:** All objects have one or more utilities. The tendency is to group objects that have the same utility (or something related) in the same places. Also, concepts are created regarding kinds of room depending on what they are used for.
- **Meaning:** The actions or utilities that an object has often are associated with a specific meaning. The goal of the navigation may also be oriented to emotions or places that give a feeling, such as calm or fun. For example, *read* is an action *quiet*.
- **Characteristic:** Objects may have features to best define the concept or even differentiate them from others. For example, on the one hand, water may be cold, warm, or hot. And that implies differences in location. Cold water may be found in the refrigerator or in a fountain, but hot water would come out from the faucet.

These tables include the semantic information of the objects in the environment that can help the robot to locate on it. For example, considering the utility, it can be said that a *computer* is used to *work*. The robot can go to an office if it needs to locate a computer.

This relational database model provides a simple way to define and manage semantic knowledge using simple queries without the need to define rules as it has been described in previous works (Galindo et al. [35], Galindo et al. [36]).

### 4.3. Semantic information management

Once the environment and its components are modeled, the next step is to manage the information in order to accomplish navigation tasks. Robots need to localize in the environment and calculate the path to the targets defined with semantical information. In the real robot, semantic targets must be transferred to the topological and geometrical levels in order to complete the movement of the robot.

#### 4.3.1. Semantic target obtention

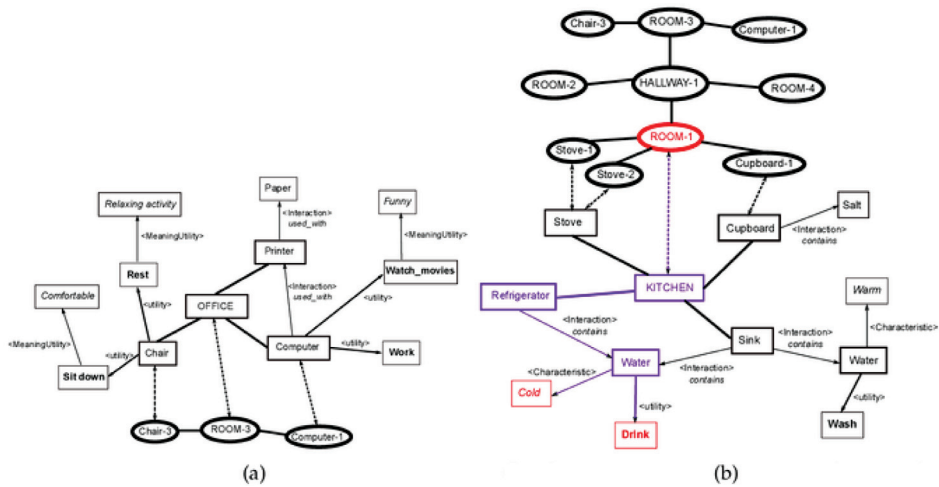
Similar to the previous navigation system, a way to establish how to reach a target is needed. In the proposed semantic system is defined the concept of *Semantic Position* as a contextual information unit related to the robot's position. *Semantic Position* contains attributes that correspond to a room code (the target room in this case) and to an object code (the target object in this case). It is therefore understood that a position is obtained in relation to a semantic reference point, which can be either objects or rooms. The target of the robot is defined as a *Semantic Position*.

Depending on the information available in the database about the requested destination and the type of destination (object or room), several options can be found in order to establish the semantic path:

- If the destination is a known object in a known room, the robot has all the information to reach the *Semantic Target*.
- If the destination is an unknown room, a semantic exploration routine must be used to manage the semantic information to locate a room with similar semantic information of the destination.
- If the destination is an unknown object in a known room, the robot would try to explore it to get an object with similar semantic information in the room to define it as a *Semantic Target*.
- If the destination is an unknown object in an unknown room, all information of the object and room in the database must be used to get an object in a room which matches with the semantical information of the destination.

To get the *Semantic Target*, several consultations to the database must be done. One query is used to obtain the specific object instances that have already been stored, other query to know the physical room where a specific object is found, and a last query to match a room with a specific type of object.

In addition, this system allows the search of destinations by semantic proximity. For example, using the knowledge of **Figure 8a** if the requested objective is to do something *fun*, the answer would be to go to the *computer-1*. This is because the *computer-1* is an object associated with the



**Figure 8.** Ontology concepts and semantic relations. (a) Ontology concepts and semantic relations of a office and (b) Ontology concepts and semantic relations of a kitchen.

computer concept. The *computer* concept is associated with the *watching movies* utility and the *watching movies* utility is associated with the *fun* concept. Using the knowledge of **Figure 8b**, searches for objects with characteristics and actions can be done. If the destination is to *drink something cold*, the system recognizes *cold water* as a valid objective. The *cold water* is related to the *refrigerator*, the *refrigerator* is related to the *kitchen*, and the *kitchen* is related to Room-1. Then, the system goes to Room-1.

#### 4.3.2. Semantic identification of a place

A place, in this case a room, can be identified as a vector of detected objects. The object observed from the environment must be analyzed and with a query we can get a list of the types of room where the object can be found. If an object is found, it defines the room where it is located. For instance, the bed is an object that can only be in a bedroom. If the result of the previous query returns several records, it is because the object is not discriminatory enough to be able to conclude any identification, and more additional semantic information and more queries are needed to get the destination room.

#### 4.4. Example

In this example, the robot is asked for several targets in a home. In the experiment, the robot has the information provided by the previous exploration of the environment. This information is stored in several tables: 1a is the table with physical objects information, 1a is the table with physical objects information and 1b is the table with deduced knowledge.

The objective is to use semantic information to get the semantic target, that is attached to a topological or to a geometrical target to which the robot must move (**Table 1**).



First, the robot is asked for the kitchen. Checking the table *PhysicalConceptualRoom* with a query, the result is *Room-1*. Repeating the query, in the tables, there are no more kitchens in the database, so the process finishes.

In a second experiment, the robot is asked for a chair. In the initial environment, there are three real objects of the CHAIR type. In the query, the robot identifies three chairs: *Chair-1*, *Chair-2*, and *Chair-3*. The robot gets the information of the rooms in which the chairs are and moves to the first option. If it indicates that this is not the correct room, the robot moves to the rest of the options. This process is described in **Figure 9**.

In case the object has no match with the observed physical object, it is necessary to ask again for a chair, after having discarded the chairs from the previous test. The operation sequence is shown in **Figure 10a**. In **Figure 9**, another query on chair is shown, and the robot starts to explore searching in new types of rooms where chairs could be found.

(a) Content of the table physical object			(b) Content of the table conceptual physical room	
Physical object			Conceptual physical room	
Name	Conceptual name	Room name	Conceptual name	Physical name
Chair-1	CHAIR	Room-1	Room-1	KITCHEN
Chair-2	CHAIR	Room-2	Room-2	LIVING ROOM
Chair-3	CHAIR	Room-3	Room-3	OFFICE
Refrigerator-1	REFRIGERATOR	Room-1		
Sofa-1	SOFA	Room-2		
Desk-1	DESK	Room-3		

Table 1. Tables with semantic information.

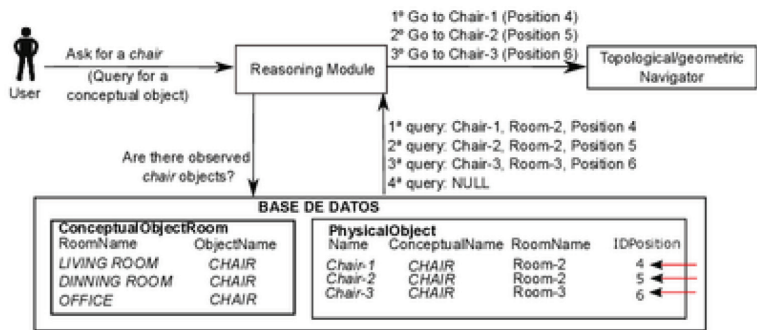
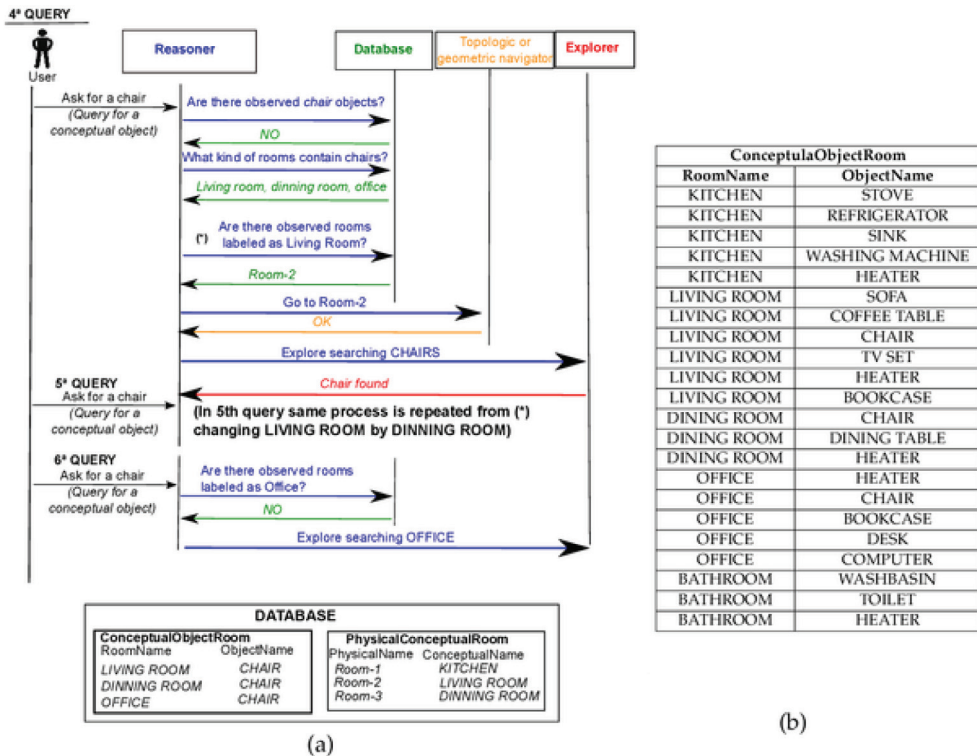


Figure 9. Test sequence with the CHAIR object.



**Figure 10.** Ontology concepts and semantic relations. (a) Test sequence with the CHAIR object when there is no match with no observed physical object and (b) Content of the table ConceptualObjectRoom.

## 5. Perception model for navigation

Talking about autonomous robots implies carrying out movements safely and having a complete knowledge of the environment. These elements define the capabilities of action and interaction between the environment, humans, and robots. Tasks performed by mobile robots such as navigation, localization, planning, among others, can be improved if the perceptual information is considered. So, the general idea is detecting and identifying meaningful elements (objects and scenes) of the environment. There are different ways to obtain the information about the environment. One of them consists of detecting recognizable features of the environment (natural landmarks) using several types of sensors. The detection of artificial landmarks, based on the work of Fischler & Elschlager [37], can be used to acquire a representation of the environment. However, due to the technological advances in 3D sensors (e.g., RGB-D sensors) and according to Huang [38], vision has become the main sensor used for navigation, localization, and scene recognition. Vision provides significant information about

the objects present in a place; at the same time, it is capable of providing semantic information of the scene where it is. Scene recognition is a very well-known challenging issue that deals with how robots understand scenes just like a human does and the appearance variability of real environments. Therefore, regardless of the type of navigation used, whether geometrical, topological, or semantic, place recognition and object identification play a significant role in terms of representation of the environment.

### 5.1. Object recognition

To perform several tasks in common indoor environments, mobile robots need to quickly and accurately verify and recognize objects, obstacles, etc. One of these crucial tasks is to move safely in an unknown environment. Autonomous robots should be able to acquire and hold visual representations of their environments.

The most important stages in an object recognition model are: feature extraction and prediction. As for feature extraction, the identification of significant aspects of different objects that belong to the same class, independently of the appearance variabilities, such as scaling, rotation, translation, illumination changing, among others, is crucial to obtain a suitable representation of the objects present in a scene. Some techniques are based on local and global descriptors such as the works presented by Bay et al. [39] and Hernández et al. [40], or a combination of both of them (Hernandez-Lopez et al. [41]). Also, in other approaches such as the work presented by Csurka et al. [42], visual vocabularies (e. g., bag of words) are commonly used to create a proper representation of each object.

Regarding prediction, through the vectors created from the extracted features, it is possible to learn these characteristics in order to identify objects that correspond with each class. In the literature, different classification techniques based on machine learning such as nearest neighbor classifier, neural networks, AdaBoost, etc., have been proposed depending on the kind of extracted features (Pontil & Verri [43]). Machine learning is a field of computer science that includes algorithms that improve their performance at a given task considering the experience. In this way, support vector machine (SVM) is one of the most helpful classification algorithms. The aim of SVM is to generate a training model that is capable of predicting the target classes of the test dataset, considering only its attributes.

Generally, an object recognition system that works in real time is divided into two stages: offline and online. Offline stage includes all the processes to reduce the execution time and guarantee the efficiency of the system, which are image preprocessing, segmentation, feature extraction, and training process. Online stage refers to the processes carried out in real time with the goal to ensure the interaction between the robot and the environment. Mainly, the processes included in the online stage are image retrieval and classification.

Object detection can be very useful for a navigation system since it allows the robot to relate what it perceives to the scenes in which it is. For this reason, it is necessary to consider that the designed systems and the sensors are not perfect. Therefore, the object detection model has to incorporate uncertainty information. The uncertainty management is a relevant aspect in an object recognition system because it allows to represent the environment and its elements in a

more realistic way. The uncertainty calculation can be determined considering the dependency relations between different factors. Some of the factors are: the accuracy of the model that can be determined empirically and a factor based on the outcomes of the classification algorithm. Also, other factors can include the influence of other elements of the environment, for example the distance during the detection process. Finally, considering these factors makes it possible to obtain a robust object recognition model to serve as an input to a mobile robot.

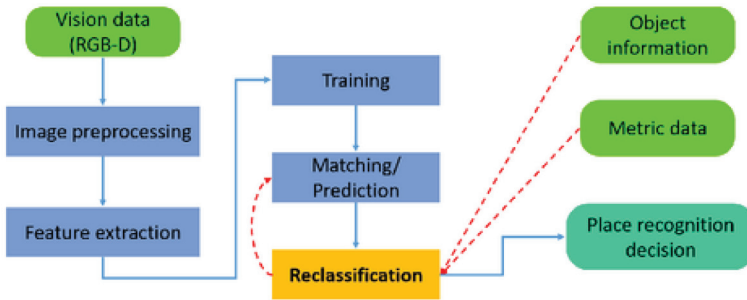
## 5.2. Place recognition

The growth of service robotics in recent years has created the needed for developing models that contribute to robots being able to adequately handle information from human environments. A semantic model can improve the high-level tasks of a robot, such as, semantic navigation and human-robot and robot-environment interaction. According to Wang et al. [44], semantic navigation is considered as a system that takes into account semantic information to represent the environment and then to carry out the localization and navigation of the robot.

In view of the above, place recognition deals with the process of recognizing an area of the environment in which there are elements (objects), actions are developed, and robot-environment and human-robot interaction is possible. The scene-understanding issue can be defined as a combination between scene recognition and object detection. There are approaches that try to solve the scene recognition problem through computer vision algorithms, including the creation of complex feature descriptors (Xie et al. [45]), and a combination of feature extraction techniques (Nicosevici & Garcia [46] and Khan et al. [47]), among others. Moreover, if it is desired to get a robust model as close as possible to reality, the incorporation of environmental data and errors of the sensors is needed.

Some approaches are based on machine learning and select support vector machine as classification technique. Commonly, this type of place recognition model is composed by the following processes (**Figure 11**): image preprocessing, that includes the selection of the datasets and the initial preprocessing of the images to obtain the training model; feature extraction, that can implement some techniques such as bag of words, local and global descriptors, among others; training process, where the parameters of the classifier are defined and the model of the scene is generated; prediction process that generates the final results of the possible scene where the robot is located; and, finally, a last process called reclassification has to be considered in which it is possible to generate a relationship between the place recognition model and the elements (objects) of the environment.

In this way, the influence of the objects in the scene can improve, worsen, and correct the final results on where the robot is located. The process implies the adjustment of the probabilities of being in a place and therefore the management of uncertainties. To do that, it is necessary that the place recognition model and the object recognition system work simultaneously in real time. The uncertainty can be modeled from a set of rules based on learning to determine the probability of co-occurrence of the objects. Also, it is important to incorporate the prior information of the place and object recognition models. Finally, it is possible to apply different



**Figure 11.** Schematic of a general scene recognition model. Vision data are the initial input of the recognition model. Then, the metric data and object information can be included into the model to update the final outcome of the possible place where the robot can be.

theorems such as Bayes to determine the final relation between objects and scenes, and the last result about where the robot is. The adjusted prediction has to be available as input for relevant tasks such as localization, navigation, scene understanding, and human-robot and robot-environment interaction. All this contributes to the main goal that is to have an autonomous and independent robot, with a wide knowledge of the environment.

## 6. Conclusion

The aim of this chapter was to describe different approaches for global navigation systems for mobile robots applied to indoor environments. Many researches and current developments are focused on solving specific needs for navigation. Our objective is to merge all these developments in order to classify them and establish a global frame for navigation.

Robot navigation has been tackled from different perspectives leading to a classification into three main approaches: geometric navigation, topological navigation, and semantic navigation. Although the three of them differ in their definition and methods, all of them have the focus on driving a robot autonomously and safely.

In this chapter, different trends and techniques have been presented, all of them inspired by biological models and pursuing human abilities and abstraction models. The geometric representation, closer to the sensor and actuator world, is the best one to perform local navigation and precise path-planning. Topological representation of the environment, which is based on graphs, enables large navigation tasks and uses similar models as humans do. The semantic representation, which is the closest to cognitive human models, adds concepts such as utilities or meanings of the environment elements and establishes complex relations between them. All of these representations are based on the available information of the environment. For this reason, perception plays a significant role in terms of understanding and moving through the environment.

Despite the differences between the environment representations, we consider that the integration of all of them and the proper management of the information is the key to achieve a global navigation system.

## Author details

Ramón Barber\*, Jonathan Crespo, Clara Gómez, Alejandra C. Hernández and Marina Galli

\*Address all correspondence to: [rbarber@ing.uc3m.es](mailto:rbarber@ing.uc3m.es)

University Carlos III of Madrid, Spain

## References

- [1] Levitt TS. Qualitative navigation for mobile robots. *International Journal of Artificial Intelligence*. 1990;**44**(3):305-360
- [2] Bailey T, Durrant-Whyte H. Simultaneous localization and mapping (slam): Part II. *IEEE Robotics and Automation Magazine*. 2006;**13**(3):108-117
- [3] Wang C, Meng L, She S, Mitchell IM, Li T, Tung F, et al. Autonomous mobile robot navigation in uneven and unstructured indoor environments. In: *Intelligent Robots and Systems (IROS)*, 2017 IEEE/RSJ International Conference on. IEEE; 2017. pp. 109-116
- [4] Pfrunder A, Borges PV, Romero AR, Catt G, Elfes A. Real-time autonomous ground vehicle navigation in heterogeneous environments using a 3d lidar. In: *Intelligent Robots and Systems (IROS)*, 2017 IEEE/RSJ International Conference on. IEEE; 2017. pp. 2601-2608
- [5] Fernández-Madrigal J-A, Galindo C, González J. Assistive navigation of a robotic wheelchair using a multihierarchical model of the environment. *Integrated Computer-Aided Engineering*. 2004;**11**(4):309-322
- [6] Ko DW, Kim YN, Lee JH, Suh IH. A scene-based dependable indoor navigation system. In: *Intelligent Robots and Systems (IROS)*, 2016 IEEE/RSJ International Conference on. IEEE; 2016. pp. 1530-1537
- [7] Kuipers BJ, Levitt TS. Navigation and mapping in large scale space. *AI Magazine*. 1988; **9**(2):25
- [8] Giralt G, Chatila R, Vaisset M. *An Integrated Navigation and Motion Control System for Autonomous Multisensory Mobile Robots, Autonomous Robot Vehicles*. New York: Springer; 1990. pp. 420-443
- [9] Mataric MJ. *A distributed model for mobile robot environment-learning and navigation*. Massachusetts Institute of Technology Cambridge. MA, USA; 1990
- [10] LaValle SM. *Planning Algorithms*. Cambridge University Press; 2006
- [11] De Berg M, Van Kreveld M, Overmars M, Schwarzkopf OC. *Computational Geometry*. Berlin Heidelberg: Springer; 2000. pp. 1-17

- [12] Hwang YK, Ahuja N. A potential field approach to path planning. *IEEE Transactions on Robotics and Automation*. 1992;**8**(1):23-32
- [13] Poty A, Melchior P, Oustaloup A. Dynamic path planning for mobile robots using fractional potential field. In: *Control, Communications and Signal Processing*, 2004. First International Symposium on. IEEE; 2004. pp. 557-561
- [14] Lee D-T, Drysdale RL III. Generalization of voronoi diagrams in the plane. *SIAM Journal on Computing*. 1981;**10**(1):73-87
- [15] Breu H, Gil J, Kirkpatrick D, Werman M. Linear time euclidean distance transform algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1995;**17**(5):529-533
- [16] Alt H, Cheong O, Vigneron A. The voronoi diagram of curved objects. *Discrete & Computational Geometry*. 2005;**34**(3):439-453
- [17] Gang L, Wang J. Prm path planning optimization algorithm research. (n.d.). *Wseas Transactions on Systems and Control*. 2016;**11**. E-ISSN: 2224-2856
- [18] Kuffner JJ, LaValle SM. Rrt-connect: An efficient approach to single-query path planning. In: *Robotics and Automation*, 2000. Proceedings of ICRA'00. IEEE International Conference on, Vol. 2. IEEE; 2000. pp. 995-1001
- [19] LaValle SM. Rapidly-exploring random trees: A new tool for path planning. 1998
- [20] Valero-Gomez A, Gomez JV, Garrido S, Moreno L. The path to efficiency: Fast marching method for safer, more efficient mobile robot trajectories. *IEEE Robotics and Automation Magazine*. 2013;**20**(4):111-120
- [21] Thrun S, Bücken A. Integrating grid-based and topological maps for mobile robot navigation. In: *Proceedings of the National Conference on Artificial Intelligence*. 1996. pp. 944-951
- [22] Vale A, Ribeiro M. Environment mapping as a topological representation. In: *Proceedings of the 11th International Conference on Advanced Robotics-ICAR*. 2003. pp. 1-3
- [23] Kuipers BJ, Byun YT. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Robotics and Autonomous Systems*. 1991;**8**(1-2):47-63
- [24] Thrun S. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*. 1998;**99**(1):21-71
- [25] Simhon S, Dudek G. A global topological map formed by local metric maps. In: *IEEE/RSJ International Conference on Intelligent Robotic Systems*. 1998. pp. 1708-1714
- [26] Amigoni F, Gallo A. A multi-objective exploration strategy for mobile robots. In: *Robotics and Automation*, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on. IEEE; 2005. pp. 3850-3855
- [27] Arvanitakis I, Giannousakis K, Tzes A. Mobile robot navigation in unknown environment based on exploration principles. In: *Control Applications (CCA)*, 2016 IEEE Conference on. IEEE; 2016. pp. 493-498



- [28] Mataric MJ. Behaviour-based control: Examples from navigation, learning, and group behaviour. *Journal of Experimental & Theoretical Artificial Intelligence*. 1997;**11**:323-336
- [29] Skiena S. *Dijkstra's Algorithm. Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*. MA: Addison-Wesley, Reading; 1990. pp. 225-227
- [30] Egido V, Barber R, Boada MJL, Salichs MA. A planner for topological navigation based on previous experiences. *IFAC Proceedings Volumes*; 2004;**37**(8):615-620. ISSN 1474-6670. [https://doi.org/10.1016/S1474-6670\(17\)32046-3](https://doi.org/10.1016/S1474-6670(17)32046-3)
- [31] Vasudevan S, Siegwart R. Bayesian space conceptualization and place classification for semantic maps in mobile robotics. *Robotics and Autonomous Systems (RAS)*. 2008;**56**: 522-537
- [32] Vasudevan S, Siegwart R. A bayesian conceptualization of space for mobile robots. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2007
- [33] Duda RO, Hart PE, Stork DG. *Pattern Classification*. New York, NY, USA: Wiley Interscience; 2000
- [34] Kollar T, Roy N. Utilizing object-object and object-scene context when planning to find things. In: *IEEE International Conference on Robotics and Automation (ICRA)*. 2009
- [35] Galindo C, Fernáñdez-Madrigal J-A, González J, Saffiotti A. Robot task planning using semantic maps, *Robotics and Autonomous Systems*. 2008;**56**(11):955-966
- [36] Galindo C, Saffiotti A, Coradeschi S, Buschka P, Fernáñdez-Madrigal JA, González J. Multi-hierarchical semantic maps for mobile robotics. In: *Proceedings of IROS*. 2005
- [37] Fischler MA, Elschlager RA. The representation and matching of pictorial structures. *IEEE Transactions on Computers*. 1973;**100**(1):67-92
- [38] Huang T. Computer vision: Evolution and promise. In: *International conference; 5th, High technology: Imaging science and technology*. Chiba, Japan; 1996. pp.13-20
- [39] Bay H, Tuytelaars T, Van Gool L. Surf: Speeded up robust features. In: *European Conference on Computer Vision*. Springer; 2006. pp. 404-417
- [40] Hernández AC, Gómez C, Crespo J, Barber R. Object detection applied to indoor environments for mobile robot navigation. *Sensors*. 2016;**16**(8):1180
- [41] Hernandez-Lopez J-J, Quintanilla-Olvera A-L, López-Ramírez J-L, Rangel-Butanda F-J, Ibarra-Manzano M-A, Almanza-Ojeda D-L. Detecting objects using color and depth segmentation with kinect sensor. *Procedia Technology*. 2012;**3**:196-204
- [42] Csurka G, Dance C, Fan L, Willamowski J, Bray C. Visual categorization with bags of keypoints. In: *Workshop on statistical learning in computer vision, ECCV, Vol. 1; Prague*. 2004. pp. 1-2
- [43] Pontil M, Verri A. Support vector machines for 3d object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1998;**20**(6):637-646

- [44] Wang L, Zhao L, Huo G, Li R, Hou Z, Luo P, et al. Visual semantic navigation based on deep learning for indoor mobile robots. *Complexity*. 2018;**2018**. Article ID: 1627185
- [45] Xie L, Tian Q, Wang M, Zhang B. Spatial pooling of heterogeneous features for image classification. *IEEE Transactions on Image Processing*. 2014;**23**(5):1994-2008
- [46] Nicosevici T, Garcia R. Automatic visual bag-of-words for online robot navigation and mapping. *IEEE Transactions on Robotics*. 2012;**28**(4):886-898
- [47] Khan SH, Bennamoun M, Sohel F, Togneri R. Geometry driven semantic labeling of indoor scenes. In: *European Conference on Computer Vision*. Springer; 2014. pp. 679-694