Approaching Qualitative Spatial Reasoning About Distances and Directions in Robotics

Guglielmo Gemignani, Roberto Capobianco, and Daniele Nardi

Department of Computer, Control, and Management Engineering "Antonio Ruberti"
Sapienza University of Rome, Italy
{gemignani,capobianco,nardi}@dis.uniroma1.it

Abstract. One of the long-term goals of our society is to build robots able to live side by side with humans. In order to do so, robots need to be able to reason in a qualitative way. To this end, over the last years, the Artificial Intelligence research community has developed a considerable amount of qualitative reasoners. The majority of such approaches, however, has been developed under the assumption that suitable representations of the world were available. In this paper, we propose a method for performing qualitative spatial reasoning in robotics on abstract representations of environments, automatically extracted from metric maps. Both the representation and the reasoner are used to perform the grounding of commands vocally given by the user. The approach has been verified on a real robot interacting with several non-expert users.

1 Introduction

One of the long-term goals of our society is to build robots able to live side by side with humans, interacting with them in order to understand the surrounding world. However, to reach such a goal, robots need to be able to understand what humans communicate (Fig. 1). The conventional numeric approach used in robotics is in fact deeply different from natural language interaction between people. The former is based on precise metric information about already well known environments, in which each element is uniquely specified only through its coordinates. The latter can deal, instead, with ambiguities and spatial uncertainties, which are solved by referring to purely qualitative properties of objects, or relations among them: correct grounding of spoken information and places of the world can be obtained even with incomplete spatial knowledge. Many difficulties arise when trying to move from a numeric representation of the world to a qualitative one. Especially if the commands to be executed by the robot are given through speech.

Many theories have been developed in the field of Artificial Intelligence, demonstrating that Qualitative Spatial Reasoning [1] can overcome the problems arising from indeterminacy, by allowing inference from incomplete spatial knowledge. Implementations of this kind of reasoners enable disambiguation between objects through spatial relations like directions or distances, thus allowing to improve symbol grounding on robots equipped with a speech recognition system. However, the majority of theories on qualitative spatial reasoning have been developed under the assumption that discretized representations of the world were available. Usually, this is not true for robots, that need an abstraction of the environment strictly depending on its underlying structure, in order to reason about actions which can be executed.

In this paper, we propose a method for performing qualitative spatial reasoning on robots. In detail, we applied the *cone-based* approach, presented by [2], to an abstraction of the environment specifically built for a consistent integration of high-level reasoning and numeric representation. Both the representation and the reasoner are used in order to perform grounding of commands vocally given by the user to the robot. The proposed approach has been subsequently validated. In particular, by analysing the number of grounded commands in different settings, we pointed out multiple relations between such commands and the amount of knowledge available to the robot. Moreover, we performed additional experiments to identify the major issues that occur during the grounding process, by analyzing the user expectations with respect to the system outputs.

Fig. 1: User interacting with a robot through natural language interaction.

The key contributions of our work are the following. First, we introduce an abstract representation of the environment useful for the deployment of many theories of AI on real robotic applications. Second, we present a method for automatically adding, on such a representation, a high-level description of the objects through the interaction with the user: each object is represented as a composition of rectangles, easily enabling for the computation of spatial properties and relationships. Finally, we describe how, by exploiting the easiness of use of both the object and the environment representation, reasoning on areas can be effectively accomplished, according to well founded theories about Qualitative Spatial Reasoning.

The remainder of the paper is organized as follows. Related work is illustrated in Section 2, followed by a description of how the environmental representation is built, given in Section 3. Section 4 will show, instead, how the representation can be used to ground the commands given by a user, by exploiting a specifically implemented qualitative spatial reasoner (QSR). Finally, the experiments undertaken to validate this approach are reported in Section 5, while Section 6 will discuss the work presented and the future developments.

2 Related Work

In order to understand commands that use qualitative spatial references for distinguishing objects in the environment, first of all, a robot needs to be able of performing symbol grounding. The problem of symbol grounding, namely the process of matching natural language expression, with entities of the world and their corresponding representation internal to the robot, has been addressed by many authors. For example, in [3] the authors present a system able to follow natural language directions. In this work, the process of grounding the user commands is divided in three steps: Extracting linguistic structures related to spatial references; Grounding symbols to the corresponding physical objects within the environment; Reasoning on the knowledge acquired to extract a feasible path. In [4], instead, the authors describe a robot able to learn and use word meanings in three kind of tasks: indoor navigation, spatial language video retrieval, and mobile manipulation. They propose an approach for robustly interpreting natural language commands, based on the extraction of shallow linguistic structures. In particular they introduce a Generalized Grounding Graph able to handle multiple arguments or sentences nested into the commands. Finally, in [5] a sophisticated robot is described, equipped with a symbolic high-level spoken dialogue system that uses Discourse Representation Structures [6] to represent the meaning of the dialogues that occur with user.

The second key aspect needed to understand commands of the type “go in front of the closet next to the emergency door", is the ability of reasoning about spatial directions in a qualitative manner. In other words, the robot needs to be able of reasoning about an object with respect to another object in a given reference frame. In the literature, spatial relations are studied and used in various research fields. For example, in the “CogX"[1] project [7] the spatial relations “in” and “on” have been used to define object targets for indirect object search. Kunze *et al.* [8] have enhanced this work by using more restrictive spatial models to provide more tightly defined viewing probabilities. In particular, by using information about landmark objects and their spatial relationship to the target object, the authors show how a searching task can be improved by directing the robot towards the most likely object locations. The authors of [9] use, instead, an extension of the *double cross calculus*, introduced by [10], to express robot navigation objectives that include spatial relations in a Mars-like environment. This work, however, lacks an intermediate layer between the metric map and the high-level representation used for reasoning. Finally, Loutfi *et al.* [11] look at this problem in the context of perceptual anchoring to provide qualitative relations inferred from observed metric relations.

As well known in the research community of Qualitative Spatial Reasoning [12] the representation of spatial knowledge is usually divided in “propositional" and “pictorial" representations. The former cannot easily express structural properties, being focused on formal properties of the representation itself. The latter, even preserving structural properties, provide a low level representation which is not suitable for fast computations. Hernández suggests, in his work, the use of a hybrid representation, interfacing these two categories as separate representations. Inspired by [12] and adopting a similar approach, we propose a method for performing qualitative spatial reasoning in robotics, where the interface between the metric information and the symbolic knowledge is represented by a grid-based structure automatically built by our robot (briefly described in

[1] http://cogx.eu/

Section 3). On top of the representation, we adopt a reasoning approach that exploits shapes for distance and orientation qualitative calculus. Specifically, we decompose spaces and objects in rectangles, adopting an intrinsic reference frame. For grounding the command received by the user, instead, we follow the approach recently proposed by [13]. In our system, the output of an automatic speech recognition module (ASR) is matched with one of the frames representing the commands executable by the robots, later grounded using definite clause grammars [14]. A more detailed description of the processing chain from user utterance to task execution will be described in Section 4.

3 Representing the Environment

Starting from a low-level representation of the environment (Metric Map), rich of structural details, we have devised a method for automatically extracting a hybrid representation of the environment able to easily interface the symbolic and the structural layers of knowledge. Such a representation has been obtained by capturing the 2D structure of the environment through a grid, on top of which knowledge can be added. Such a grid, in fact, has the key property of having cells of different size, in order to capture the spatial similarity of close locations, while being uniformly accessible as a matrix. Each element of the matrix, therefore, can be considered independently, being its spatial relations kept by the structure of the matrix itself.

Since this work has been developed for domestic robots, operating in indoor locations, the first step performed in order to create a grid structurally coherent with the environment is the detection of the edges of the map through the Canny Edge Detector (Fig. 2a). Then, we find the segments corresponding to its walls, since they are the basic elements to form the grid itself. This process is performed by extracting the lines in the image through the Hough Transform (Fig. 2b) and then applying a filtering process (Fig. 2c) based on the length and distance of the segments. Starting from those, the grid (called Grid Map) is produced in an adaptive manner: 1) The segments generated through the wall detection are extended to the whole map image; 2) The minimum horizontal and vertical distances $(x_{\min}, y_{\min})$ are computed; 3) each pair of parallel lines is considered, and a new one is inserted between them only if their distance is at least twice $x_{\min}$ or $y_{\min}$, depending on their angle. The resulting grid, therefore, has cells of a size ranging between $x_{\min} \cdot y_{\min}$ and $2x_{\min} \cdot 2y_{\min}$.

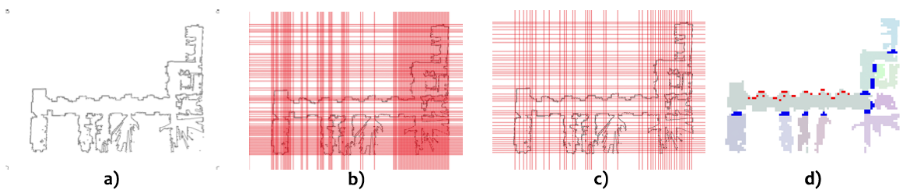

Fig. 2: a) Edges of the Metric Map. b) Segments extracted through the Hough Transform. c) Segments obtained after the filtering process in the wall detection. d) Representation of the Cell Map in which the areas of the environment are represented with different colors, the doors in blue and the objects in red.

On top of such a discretized representation, knowledge is added online by acquiring information through the natural language interaction between the robot and the human, to perform semantic mapping. Specifically, three types of knowledge can be acquired: Areas (e.g., corridors, rooms, etc.); Structural elements (e.g, windows, doors, etc.); Objects in the environment (e.g., tables, closets etc.). Using the Metric Map for the autonomous navigation of the robot in the environment, the positions of the different objects of interest are registered. The user can tag a specific object by naming it and specifying its position through a commercial laser pointer, thanks to which the orientation of the object is obtained by extracting its normal. The robot, in fact, is endowed with a speech recognition system and a Kinect sensor for detecting the laser point and extracting properties and shapes of objects, in addition to the laser range sensor used for the localization process. Once knowledge is acquired, it is first inserted in a relational database and, then, it is processed in order to be reported on the Grid Map, thus obtaining the Semantic Map, composed by the Cell Map and the Topological Graph. The Cell Map (Fig. 2d) contains a high-level description about the regions, structural elements, and objects contained in the environment. The Topological Graph, instead, is created in order to represent the information needed by the robot for navigating and acting in the environment, associating each node of the graph to a cell of the Cell Map. A final representation of the robot's knowledge is shown in Fig. 3, while a more detailed description about the representation, its building algorithms and the knowledge acquisition process can be found in [15].

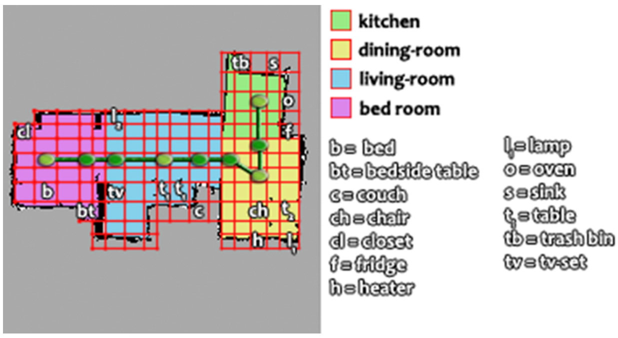

Fig. 3: Semantic map of a domestic environment. The Topological Graph is depicted on top of the Cell Map and the objects in it. The metric map is also depicted in the background.

Given the various kinds of knowledge represented in the Cell Map, various forms of reasoning can be performed. To this end, all the knowledge included in the Cell Map and in the Topological Graph is automatically translated into Prolog assertion predicates. In particular, each element of the Cell Map is represented with a predicate

```
cellIsPartOf(XCoord, YCoord, AreaTag)
```

and each object is represented with the two predicates

```
object(Id, XCoord, YCoord, Properties),
Type(Id, Type).
```

For example, the knowledge of a white plug located in the cell with grid map coordinates 45, 67 belonging to the corridor area will be represented with the three predicates

```
cellIsPartOf(45, 67, corridor),
object(plug1, 45, 67, color-white),
objectType(plug1, plug).
```

The knowledge stored in the Topological Graph is instead represented as an acyclic graph in Prolog with the predicates

```
node(Id, XCoord, YCoord),
arc(Id1, Id2),
```

respectively for the nodes and the arcs of the graph. Having translated in Prolog all the knowledge stored in the representation built with the previously described method, performing certain kind of inference on it becomes straight forward, as it is shown in the next section.

4 Reasoning on the Representation

In this section, we show how the commands uttered by a user are understood by our robot, by qualitatively reasoning about orientations and distances of objects in the environment. In particular, we describe the processing chain applied to the input command (Fig. 4), first describing the natural language processing operations performed on it, while later showing the qualitative reasoning method adopted.

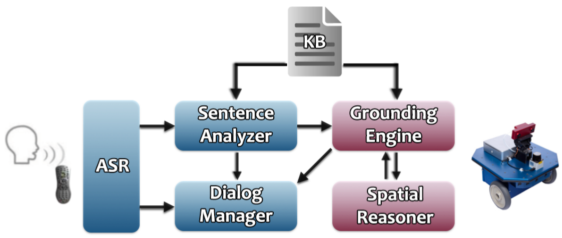

Fig. 4: Processing chain applied to the commands uttered by a user, starting from the ASR and ending with the command execution.

Starting from the sentence uttered by the user, the processing chain for spoken commands follows a well-established pattern: based on a language model built using *SRGS*[2], the system tries to match the output of the ASR within a knowledge base of frames, which represent the commands that can be executed by the robot. According to the results of such a matching, either the processed command is sent to the grounding engine, or an interaction with the user is started, in order to ask for clarifications or for a new command. When a frame is correctly instantiated and an output can be passed to the grounder, the Sentence Analyzer serializes it into a command keyword and a list of

[2] http://www.w3.org/TR/speech-grammar/

tokens representing the specification of the action to be performed. For example, if the command uttered by the user is “go in front of the socket on the right of the closet", the output received by the grounder will be the keyword “*GO_FRONT*", in addition to the list of tokens “[to, the, socket, on, the, right of, the, closet]".

The output of the Sentence Analyzer is then passed to the Grounding Engine. By using definite clause grammars implemented in *Prolog*, we parse the input tokens in order to extract the located object, the reference object and the spatial relation that relates them. Having extracted these three elements from the user command, the system tries to ground them by querying the knowledge base and the spatial reasoner: two objects that fall in the categories of the located object and the reference object are searched, filtering the results by requiring their positions to agree with the relation specified by the user. To better explain this process, if the list “[to, the, socket, on, the, right of, the, closet]" is received as an input, the tokens “socket", “right of" and “closet" are identified as the located object, the spatial relation and the reference object respectively. The knowledge base is then queried for all the sockets and closets known in the environment with their positions. Finally, the known objects are filtered by the spatial reasoner that discards all the sockets that are not on the right of the closet.

In order to perform this latter operation, by exploiting the representation previously described, we built a spatial reasoner with an intrinsic reference frame. Three vicinity relations (near, next to and nearest), their three opposite relations (far, not next to and furthest) and four orientations (behind, in front, on the right and on the left) have been implemented. In particular, by defining C_{Loc} and C_{Ref} the set of cells belonging to the cell map that include a portion of the objects Loc and Ref respectively, we say that Loc has a vicinity relation with Ref if and only if:

$$d(centroid(C_{Loc}), centroid(C_{Ref})) < t$$

where d is the euclidean distance, t is a threshold constant and $centroid(x)$ is a function that takes a set of cells x in input and returns the coordinates of its centroid in the metric map coordinate system. By specifying a threshold constant for both the relations “near" and “next to", respectively t_{near} and t_{next}, we therefore define the six distance relations (the nearest attribute is computed finding the object that minimizes the above defined distance). In order to define the orientation relations, instead, analogously to [12], we exploited the “intrinsic front side" of the objects (identified with the normal n̂ of the surface tagged by the user during the knowledge acquisition phase previously described). Specifically, we have used it to define a forward orientation, later deriving, by rotating clock-wise, respectively the concept of left, backwards, and right regions, as shown in Fig. 5. By defining the general concept of directions, we adopted the *cone-based* approach [2] to explicate the four directional relations, starting from the centroid of the reference object, as shown in Fig. 5. As for the definition of vicinity, in order for two object Loc and Ref to be related from a relation R, by defining A_R^{Ref} the area corresponding to a region in the direction R with respect to the reference object Ref (e.g. A_{right}^{closet} is the area on the right of the closet), we require that:

$$centroid(C_{Loc}) \in A_R^{Ref}$$

where, again, the $centroid(x)$ is a function that takes a set of cells x in input and returns the coordinates of its centroid in the metric map coordinate system. Note that, if the centroid of the located object corresponds to the centroid of the reference object, we consider them connected with all the four directional relations, since our representation is an abstraction of the physical world. After applying the cone-based approach to our representation, by collapsing the represented objects in their centroid, all the properties derived for this approach from the theory are automatically inherited by our system. Finally, by exploiting the representation of the environment automatically built two advantages can be identified: The objects can be automatically inserted in the environment representation through the interaction with the user; The qualitative spatial reasoning can be performed on points (the centroid of the cells representing the object in the Cell Map), allowing for an easy and straightforward approach.

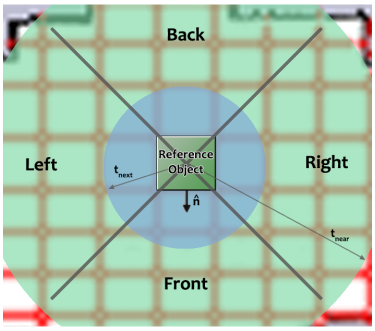

Fig. 5: Reference frame adopted for the implemented spatial reasoner. The image shows how the concept of “near" and “next to" have been implemented as well as how regions in the different direction have been identified through the cone-based approach.

5 Experimental Evaluation

Several tests have been conducted in order to demonstrate the improvements that qualitative spatial reasoning can determine in grounding the commands given by the users to a robot, as well as the efficacy of implementing such an approach on a real robot. Our validation work has been therefore focused on two different kinds of experiments.

The purpose of the first experiment was evaluating the impact of a qualitative spatial reasoner on an agent whose amount of knowledge continuously grows, as well as the influence of the already available knowledge on such a reasoning. The goal of this work was enabling a real robot to disambiguate the instructions given by a human-being on the basis of the relations between objects in the environment. Such an evaluation has been carried out considering the number of unambiguous and ambiguous commands (i.e., commands referring to more than one object with a specific spatial property, see Fig. 6) grounded by the agent. Indeed, when full knowledge about the environment is available, grounding ambiguous commands would mostly lead to the execution of the wrong action with respect to the user expectation, while all the unambiguous commands are supposed to be correctly grounded.

We therefore analyzed first the impact of the presence or absence of the qualitative spatial reasoner (QSR) and then the amount of knowledge available to the agent.

Fig. 6: Setting in which the command *"go to the socket near the closet"* has an ambiguous meaning. The command, in fact, could be grounded with either the socket on the left or the one on the right of the image, since they are both near the closet.

In detail, we first asked each member of a group of 26 students from the First Örebro Winter School in Artificial Intelligence and Robotics[3] to provide a set of 3 commands containing spatial relations between objects, by looking at pictures of the test environment. Then, from the 78 acquired commands, we extracted two types of tasks: 28 ambiguous and 50 unambiguous. By gradually adding knowledge about the objects inside the knowledge base of the agent, we therefore measured how many commands were grounded. We repeated the experiment for both categories of commands, with or without the qualitative spatial reasoner. Since the curves depend on the order of the objects inserted in the knowledge base, the experiment has been performed five times in order to obtain its average trend (Fig. 7). In case the QSR was not present (first curve), only the objects in the environment, whose category has a unique member, were correctly identified. For example, since we had two closets in the test environment, there was no way of distinguish them without exploiting spatial relations. By comparing the first and the second curve in the image, it can be noticed that the presence of the QSR does not greatly affect their trend, when a little amount of knowledge is available, due to the absence of exploitable spatial relations between objects. On the contrary this is not true when substantial environmental information is accessible. In this case, Curve 2, 3 and 4 show that the QSR is essential for grounding all the unambiguous commands, lowering and eventually zeroing the errors that derive from the grounding of ambiguous ones (which should not have been grounded). In order to better understand this point, suppose you have a test environment where two trash bins are in front of two different windows: by not knowing the existence of one of the two trash bins, if the ambiguous command *"go to the trash near the window"* is given, the robot will erroneously ground the command with the only trash known. Differently, if both trashes are known the robot, it will correctly ground both objects, warning the user of such an ambiguity.

The goal of the second experiment was, instead, to understand the limitations of the proposed approach rather than to perform a usability study. In detail, we do not want to analyze in a quantitative manner the obtained results but, our intention is to identify the kind of errors perceived by non-expert users during the interaction and

[3] http://aass.oru.se/Agora/Lucia2013/

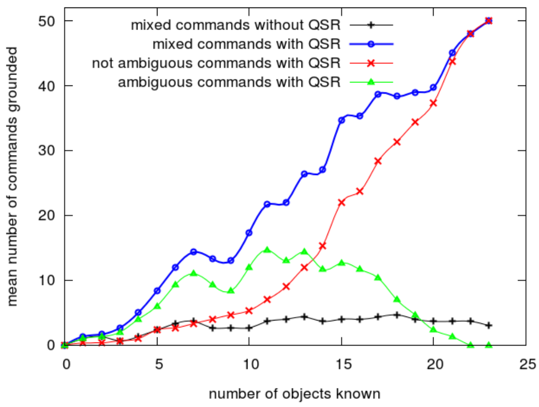

Fig. 7: Mean number of grounded commands with respect to the number of objects known in the environment, added in a random order. Three different curves (“mixed commands without QSR”, “mixed commands with QSR”, “not ambiguous commands with QSR” and “ambiguous commands with QSR”), respectively, report the results obtained by giving to the robot the complete (mixed) set of commands, only unambiguous commands or only ambiguous commands. As expected, with a qualitative spatial reasoner and a complete knowledge about the relevant elements of the environment, the robot correctly grounds only the not ambiguous commands.

the grounding process. To this end, we implemented our method on a robot, able to interact with a user through natural language: in this setting we measured the agreement between the user expectations and the grounding performed by the robot. In particular, we first produced a Cell Map by carrying the robot on a tour of the environment and tagging 23 objects within the environment, as well as the doors and the functional areas in it, through an online augmentation of the Map. Then, we asked 10 different non-expert users to assign 10 distinct tasks to the robot, asking them to evaluate if the robot correctly grounded their commands, meeting their expectations. The commands have been directly acquired through a Graphic User Interface, in order to avoid possible errors due to misunderstandings from the speech recognition system. In detail, the users had the possibility to choose the action to be executed by specifying the located object, the reference object and one of the 10 spatial relations implemented in our reasoner. Table 1 shows that approximately 80% of the uttered commands have been correctly grounded. The remaining 20% of the wrongly grounded commands where due to two different phenomena:

- The command given was ambiguous, requiring other proprieties, in addition to direction and distance, to identify the object;
- The users did not behave coherently during the interaction with the robot, by varying their concept of vicinity or by adopting different reference frames.

While the first issue is intrinsic to the nature of the command and it can be solved by exploiting other proprieties (e.g., the color), the second one could be addressed by using adaptive parameters, learnt over time through the interaction with the user. For example we noticed that the concept of vicinity for a reference object varies with the number of objects around it. By keeping track of the feedbacks given by the user when the system wrongly grounds a command, the change of the vicinity concept (represented in our system with the two thresholds t_{next} and t_{near}) over different settings could be modelled. Moreover, the well established concept that the reference frame adopted for spatially relating objects changes with respect of where the user is standing ([16] and [17]), could be addressed by dynamically changing the robot's reference frame based on the position of the human. Such solutions, however, go beyond the scope of this paper.

Table 1: Results of the second experiment. Ten different users have been asked to give ten different tasks to the robot, using spatial relations about distances and directions. The table shows the number of correctly and wrongly grounded commands with respect to the expectations of the users.

User	Correctly Grounded Commands	Wrongly Grounded Commands
1st	7	3
2nd	8	2
3rd	10	0
4th	6	4
5th	8	2
6th	8	2
7th	10	0
8th	7	3
9th	9	1
10th	8	2
Total	81	19

6 Conclusion

In this paper we have presented a method for applying qualitative reasoning about directions and distances on real robots. In particular, we have shown how a suitable representation of the environment can be automatically extracted from the Metric Map, by creating a grid-based abstraction of the world with the aid of the user. By embedding in such a representation a high-level description of the objects, qualitative spatial reasoning can be performed by the robot to accomplish tasks in real scenarios. Indeed, this is an important task, for performing a further step in the direction of implementing effective human-robot interaction. The proposed approach has been validated by considering the number of grounded commands with respect to different amounts of knowledge available to the robot, as well as with the presence and the absence of a qualitative spatial reasoner. From such an analysis the essential role of a qualitative spatial reasoner for grounding spoken commands has been pointed out. Finally, we have performed a second experiment to identify the major issues that occur during the grounding process. Specifically, several non-expert users have been required to give specific commands to the robots, comparing their expectations with the output of the system. Two issues intrinsically embedded in the human-robot interaction have been identified: the ambiguity of certain commands and the incoherence of reference frames adopted by the users. Solving these issues, as well as using the proposed representation for a further improvement of the reasoner (e.g., considering different kind of commands that exploit other properties to identify objects), will be the focus of our future work. Moreover, we are planning to extend our approach in a 3D representation.

References

1. M. Knauff, The cognitive adequacy of allen's interval calculus for qualitative spatial representation and reasoning, Spatial Cognition and Computation.
2. A. U. Frank, Qualitative spatial reasoning with cardinal directions, in: 7. Österreichische Artificial-Intelligence-Tagung/Seventh Austrian Conference on Artificial Intelligence, 1991, pp. 157–167.
3. S. Tellex, T. Kollar, S. Dickerson, M. R. Walter, A. G. Banerjee, S. J. Teller, N. Roy, Understanding natural language commands for robotic navigation and mobile manipulation., in: AAAI, 2011.
4. S. Tellex, T. Kollar, S. Dickerson, M. R. Walter, A. G. Banerjee, S. Teller, N. Roy, Approaching the symbol grounding problem with probabilistic graphical models, AI Magazine.
5. C. Theobalt, J. Bos, T. Chapman, A. Espinosa-Romero, M. Fraser, G. Hayes, E. Klein, T. Oka, R. Reeve, Talking to godot: Dialogue with a mobile robot, in: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2002), 2002.
6. H. Kamp, U. Reyle, From discourse to logic: Introduction to modeltheoretic semantics of natural language, formal logic and discourse representation theory, 1993.
7. K. Sjöö, A. Aydemir, P. Jensfelt, Topological spatial relations for active visual search, Robotics and Autonomous Systems.
8. L. Kunze, K. K. Doreswamy, N. Hawes, Using qualitative spatial relations for indirect object search, in: IEEE International Conference on Robotics and Automation (ICRA), 2014.
9. M. McClelland, M. Campbell, T. Estlin, Qualitative relational mapping for planetary rovers, in: Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence, 2013.
10. K. Zimmermann, C. Freksa, Qualitative spatial reasoning using orientation, distance, and path knowledge, Applied Intelligence.
11. A. Loutfi, S. Coradeschi, M. Daoutis, J. Melchert, Using knowledge representation for perceptual anchoring in a robotic system, Int. Journal on Artificial Intelligence Tools 17 (2008) 925–944.
12. D. Hernández, Diagrammtical aspects of qualitative representations of space, Proc. AAAI Spring Symposium on Reasoning with Diagrammatic Representations.
13. E. Bastianelli, D. Croce, R. Basili, D. Nardi, Unitor-hmm-tk: Structured kernel-based learning for spatial role labeling, in: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). Association for Computational Linguistics, 2013.
14. F. C. Pereira, D. H. Warren, Definite clause grammars for language analysisâĂŤa survey of the formalism and a comparison with augmented transition networks, Artificial intelligence.
15. E. Bastianelli, D. D. Bloisi, R. Capobianco, F. Cossu, G. Gemignani, L. Iocchi, D. Nardi, Online semantic mapping, in: Advanced Robotics (ICAR), 2013 16th International Conference on, 2013, pp. 1–6. doi:10.1109/ICAR.2013.6766501.
16. R. Baruah, S. M. Hazarika, Qualitative directions in egocentric and allocentric spatial reference frames, International Journal of Computer Information Systems and Industrial Management Applications.
17. B. Tversky, Cognitive maps, cognitive collages, and spatial mental models, in: Spatial Information Theory A Theoretical Basis for GIS, Springer, 1993, pp. 14–24.