On-line Semantic Mapping

E. Bastianelli, D. D. Bloisi, R. Capobianco, F. Cossu, G. Gemignani, L. Iocchi, D. Nardi
Dept. of Computer, Control, and Management Engineering
Sapienza University of Rome, Italy

Abstract—**Human Robot Interaction is a key enabling feature to support the introduction of robots in everyday environments. In fact, robots are currently incapable of building representation of the environments that allow both for the execution of complex tasks and for an easy interaction with the user requesting them. In this paper, we focus on the problem of semantic mapping, which amounts to building a representation of the environment that combines the metric information needed for navigation with symbolic information that conveys meaning to the elements of the environment and the objects therein. Specifically, we extend previous approaches, by enabling on-line semantic mapping, namely the ability to add to the representation elements that are acquired through a long term interaction with the user. The proposed approach has been experimentally validated on different kinds of environments, several users and multiple robotic platforms.**

I. Introduction

A key step in the delivery of robots for consumer applications is the ability to build and maintain a rich representation of the operational environment. Although the perception capabilities of robots are rapidly improving, there are still significant limitations in the categorization of environments and in the recognition of the objects therein. Consequently, the knowledge that robots have about the environment is often restricted to the features that enable for an effective navigation.

Semantic mapping is the term used to denote the process that allows the robot to enrich the map used for navigation with semantics (i.e., knowledge about the environment, according to [1]). Most state of the art semantic techniques are typically focused on the automatic interpretation of perceptions. A few approaches try to leverage the ability to build and maintain a semantic map, by putting the human in the loop. In other words, the robot builds its representation of the environment, by interacting with the user. In this way, the representation can become much richer both in terms of classification of the spaces and in terms of object detection. Current systems are designed to build the semantic map off-line, namely by a preliminary inspection of the environment, possibly with the user help.

The goal of this paper is to push the approach one step further by implementing a system that allows to acquire new objects in the representation through a continuous, on-line interaction with the user. More specifically, the proposed approach builds on the framework presented in [2], where a multi-modal technique has been proposed to develop a semantic map off-line. The idea is that the robot is guided by the user in a tour of the operational environment, that allows for the construction of the representation needed by the robot through a multi-modal interaction combining gesture recognition, speech recognition, and perception (see Fig. 1).

However, the requirement that the semantic map is built through a preliminary inspection of the environment with the user becomes impractical, if the amount of knowledge to be provided increases. Moreover, environments are dynamic, as people and objects change location.

In this paper, we extend the previous approach in two main respects. First, we obtain from the map annotated with the groundings of objects and locations provided by the user, a representation that enables the system to execute commands, such as “go to the printer” or “check whether the window in the corridor is open”. In addition, the representation supports the placement of 3D objects in the semantic map, together with their relevant properties. The resulting representation suitably combines metric and semantic information, supporting topological navigation, as well as the specification of a variety of target locations in the environment.

Second, we aim at extending earlier approaches by enabling the robot to acquire additional knowledge about the environment after the initial set-up of the semantic map. The idea is that the knowledge is acquired on a by-need basis and is incrementally added to the robot’s representation of the environment.

In order to realize the above sketched approach, several functions have been implemented on top of the previously developed framework. A symbolic representation allows to maintain high-level information about the environment (topological graph and object properties); complex robot behaviors enable the acquisition of new knowledge about objects through perception and interaction with the user; the new knowledge acquired about objects permits the update of the semantic map.

The paper is organized as follows. In Section II, we set the context by discussing related approaches to semantic mapping. Then, in Section III and in Section IV we describe the developed system focusing on the representation of the robot’s knowledge and on the main modules that are used for acquiring new knowledge and for interacting with the user. In Section V we present the method used to integrate knowledge acquisition and user interaction in a complex robot behavior and the process of updating the semantic knowledge of the robot. Experiments to validate the performance of the proposed approach are presented in Section VI. In the last section, we draw some conclusions and discuss future work.

II. Related Work

Work aiming at the construction of rich representations of the robot operational environment has become more and more relevant, as the requirements in terms of performance of robots in unstructured and dynamic environments are growing. Specifically, the term “semantic map” refers to a representation that embodies in the spatial structure of the environment additional information concerning the places and objects therein [1]. A key role in semantic mapping is thus played by the association between the symbols and the physical elements of the environment [3].

Initially, many works attempted a fully automated approach to the construction of a semantic map, focusing on extracting attributes of rooms [4] or on labeling a topological map using anchoring [5]. Another set of works addressed also automatic segmentation of spaces [6], typically building topological maps [7], [8], [9]. Recently, techniques for object recognition and place categorization based on visual features [10], or a combination of visual and range information [11] have been proposed.

However, relying on fully automated interpretation of sensor data is error prone and has limitations in the knowledge that can be acquired, henceforth several researchers suggested to include the human in the learning loop. The user can thus support the semantic map construction by helping the system in grounding the symbols through speech. In [12] a contextual topological map is built through the interaction with the user. In [13] a multivariate probabilistic model is used to associate a spatial region to a semantic label, while a user guide supports the robot in this process, by instructing it in selecting the labels.

A more general approach to human-robot collaboration for semantic mapping is taken in [14] where clarification dialogues between human and robot, using natural language, support the mapping process. Subsequently, in [15] the approach is extended to create conceptual representations of human-made indoor environments. Not only the user supports the robot in place labeling, but the representation is also used in human-robot dialogue. A fully user-centered approach to semantic mapping has also the advantage that the system does not need to be specialized beforehand with the knowledge about a specific type of environment. Randelli *et al.* [2] propose a rich multi-modal interaction, including speech, gesture, and vision enabling for a semantic labeling of environment landmarks that makes the knowledge about the environment actually usable, without many pre-requisites on the features of the environment itself. A suitable representation of the acquired knowledge into an expressive semantic map is however missing. Pronobis and Jensfelt [16] also use heterogeneous modalities for a comprehensive multi-layered semantic mapping algorithm, aiming at place categorization and topological map construction. The system builds a probabilistic representation to estimate room labels that includes information about the existence of objects and properties of space, such as room size, shape and appearance. The support of the user is considered, but it does not play a central role in the process; indeed, the approach is mainly focused on automated perception and inference of the labeling.

The key distinguishing feature of our work as compared with previous approaches to semantic mapping is the incremental construction of the semantic map. As a matter of fact, we address the construction of the semantic map not as a start-up procedure that allows the robot to build the representation before starting operation. Our aim is to enable the robot with the ability to add semantic knowledge to the map, by a continuous process where the robot requires the help of the user according to the approach named “Symbiotic Autonomy” [17], whenever it does not know how to ground a command in the map. In this respect, our approach is similar to the one proposed in [18], where a knowledge base with the grounding of expressions referring to places of the environment is learnt incrementally through the interaction with the user. Here, however, we are not restricting to the the grounding of expressions into places, but we address also objects and we build a more full-fledged representation of the objects in the semantic map.

III. System Description

In this section we provide a description of the main system components that support the incremental construction of the map. We specifically focus on the representation adopted for the semantic map, then we address the main perception processes supporting knowledge acquisition and finally we describe the component that handles the voice interaction with the robot.

A. Symbolic Representation

The robot’s knowledge is divided in two layers: the *world knowledge*, that represents the specific knowledge of a certain environment that the system acquires, and the *domain knowledge* which resembles the general knowledge about a domain. It is important to point out that, while the two components may recall the extensional and intentional components of a classical knowledge base, here they are independent of each other. The world knowledge, in fact, may be inconsistent with the domain knowledge, which is used to support the action of the robot only when specific world knowledge is not available. For example, if the user asks the robot to go to the printer, the system does not blindly infer the location of the printer from the representation, rather it asks the user to confirm its location. In this way, the exceptions that are typical of each environment can be explicitly stored in the world knowledge (e.g., a printer could be in the rest rooms, as in offices that lack space, while this exception does not need to be explicitly considered in the domain knowledge).

The representation formalism of the world knowledge contains the following elements, that are typical also of other approaches of semantic maps [15]; however, since the user validates the knowledge acquired by the system, we do not introduce a probabilistic representation as in [16].

a) The **Metric Map**, that is represented as an occupancy grid generated by a SLAM method.
b) The **Instance Signatures**, that is represented as a data base of structured data, where each instance has a unique label, an associated concept, and a set of properties expressed as attribute-value pairs.
c) The **Cell Map**, that is represented as a discretization of the environment in cells of variable size. Each cell represents a portion of a physical area and is an abstraction of locations that are not distinguishable from the point of view of robot high-level behaviors.
d) The **Topological Graph**, that is a graph where nodes are locations associated to cells in the Cell Map and edges are connections between these locations. Locations are distinguished in two types: static, which have fixed positions, and dynamic, that correspond to a variable position within a given area. Since the Topological Graph is used for navigation purposes, the edges also contain the specific navigation behavior that is required for the robot to move from one location to another. In this way the topological map is also used to generate appropriate sequences of behaviors to achieve the robot’s navigation goals.

The domain knowledge base contains, instead, a taxonomy of the concepts involved in the environment tied by an *is-a* relation, as well as their properties and relations (see [5], [7]). However, our aim is not the automatic classification of places; rather, we use this knowledge when a new object is acquired, to associate both spatial properties and functional properties to its representation in the world knowledge. Such properties are used in the grounding process, and they are also associated with the knowledge that is later used to support robot operation. Therefore, in our case, the main reasoning amounts to inheritance of properties from the taxonomy.

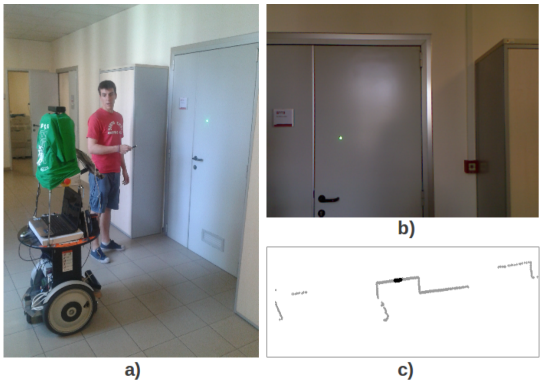

Fig. 1. Object tagging in an office environment. a) A commercial laser pointer is used to tag the object. b) The dot is detected in the HSV color space. c) The object is located with respect to the laser scan of the robot (represented as black pixels).

B. Robot Perception

The robot can acquire two types of perception data: RGB-D data coming from a Kinect sensor and 2D data coming from a laser range finder. The two sources are used to recognize the object that the user wants to insert into the semantic map. To label an object of interest in the scene, the user can point it through a commercial laser pointer (see Fig. 1) and provide, through the vocal interface, semantic information about it.

The laser dot is detected by using the RGB-D data. The RGB image is converted into the HSV color space to search for the color of the light emitted (green in our case). In our OpenCV implementation, we set the value to locate the dot as $40 < H < 70$, $S < 120$, and $V > 230$. To filter out false positives (e.g., due to illumination sources coming from windows or ceiling lighting) the background model of the scene is computed before detecting the dot, thus obtaining a binary foreground mask. Then, the depth image is used to discard all the points that are above a certain height (2.00 meters) or that are too far from the camera (3.00 meters).

Once the dot is found, it can be re-projected onto the 3D point cloud of the scene. The segmentation of the shape of the tagged object is performed by using the laser dot as the seed point of the expansion and the size information contained in the knowledge base, which provides a stopping criteria.

The pose estimation module outputs the 2D position x, y and the bearing θ of the tagged object. The pose (x, y, θ) is calculated by taking into account the normal corresponding to the surface of the segmented object in the reference frame of the Kinect and applying a transformation to the reference frame of the robot.

Person Detection

As already mentioned, the approach to the semantic map update described here is based on human-robot interaction and thus a basic functionality of the robot is to detect and follow the user who is guiding the robot to acquire new knowledge.

The detection of the person to be followed is realized by integrating information coming from the laser range finder and the RGB-D camera. The poses of these two sensors are calibrated so that we obtain a merged set of range data to process.

The method is based on filtering the current range data with information coming from the localization module, and in particular the distance map. In this way, it is possible to filter out the range data that are associated to objects that are in the map (low values in the distance map). The remaining data points are assumed to belong to objects not in the map and in particular to the user of the robot. After a clustering and the application of an additional filter on the metric size of the cluster, the closest cluster which is compatible with the typical size of a person is used as target detection for the person following behavior.

The procedure implemented here is not completely robust to the presence of multiple people in the scene and it may generate false positives when objects that have a similar size of a human (e.g., a plant) appear as new objects in the scene. However, since the focus of this paper is not on people perception and tracking, we use this simple implementation that is robust enough for the described semantic mapping task.

C. Voice Interaction

A key component of the multi-modal interface of the proposed system is the speech understanding subsystem. While the combined use of perception and pointing with the laser allows to acquire the position and the visual description of an object, the utterances of the user provide a mechanism to acquire the symbolic representation of the pointed object that populates the knowledge base. In this way, the natural language reference to an object is grounded, and the grounded representation can be used to refer to the object in question during the interaction with the user.

The sentences that are recognized by the system currently belong to two basic categories: commands and object descriptions. In fact, a dialogue with the user allows the system to acquire a user command, and if an argument of the command denoting a location can not be grounded, an acquisition process is started. During this process, the user can guide the system with voice commands such as *"turn right"*, *"follow me"* or *"go to the Phd room"*. For the references to objects in the environment, when the robot is in front of the new object/location to be grounded, the user can point to the object and tell the robot the reference for the object, e.g., *"this is the emergency door"*.

The dialogue is embedded within Petri Net Plans, that are used to specify the robot's behaviors; more details are given in Section V. In order to have better performance of the ASR, in the implementation of the dialogue, the set of grammars are contextual with the robot's behavior and are loaded dynamically. Thus, two basic sets of grammars have been implemented, one for the motion commands and one for the categorization commands. We defined also a third set of grammars used when the robot asks for confirmations about some performed action.

In the rest of this section, we first describe our basic approach to speech processing, then we focus on the process for grounding the user expressions into the locations in the semantic map and finally we address the references to new objects/locations to be added to the knowledge base.

1) Speech Processing: In order to find a correspondence between speech commands and robot actions, a bridge between the linguistic level and the ground world of the robot actions is needed. The first step is the creation of a semantic representation of the linguistic expressions contained in the commands. To this end, we adopt the FrameNet representation paradigm, as in [19]. This representation is based on the concept of *frame*, a conceptual structure representing a situation in the world, typically an action. The general meaning expressed by each frame can be enriched by semantic arguments, called *frame* *elements*, that are part of the sentence and provide additional specification of the action.

In order to obtain the semantic interpretation of a spoken command we rely on a grammar-based speech recognition approach, where the language model is specified by defining grammars that drive the recognition process (as in the Speaky for Robots project[1]). By attaching a proper semantic output to each grammar rule, it is possible to get a representation of the linguistic semantics of a recognized utterance. The final semantic interpretation is obtained by composing the outputs of the rules applied in the recognition process. The output is a parse tree containing syntactic and semantic information, that is used to instantiate the associated frame. For example, the command *"go to the Phd room"*, will be mapped to the MOTION frame, while the sub-phrase *"to the Phd room"* will fill the specific frame element GOAL representing the destination of the MOTION action.

A limitation of the grammar-based approach in the specification for the ASR is that the recognized language is restricted by the grammar. Here, we assume that the names of additional objects and locations not present in the map are known to the speech recognition module, so that they can be understood when the user refers to them. In fact, different speech processing chains, that rely on general purpose ASRs do not need this *a priori* knowledge.

2) Grounding: The information contained in the semantic map, in terms of places as rooms or positions of other objects (even the ones acquired in the on-line mode) is used to ground the natural language references to positions in the space. The overall grounding process is then realized by first selecting the action according to the parsed semantic frame. Then, specific arguments of the frame are grounded by querying the semantic map with the corresponding predicate. For example, the destination of a MOTION command as *"to the Phd room"* is translated into the appropriate query predicate, i.e., `position(phdRoom)`. This will be executed on the Cell Map in order to retrieve the pose of the referring expression.

3) Object Reference: The system has been also provided with the possibility to acquire new knowlege about the objects in the environment. A specific CATEGORIZATION frame is associated with a description command as *"this is the emergency door"*. The CATEGORY argument, that represents the description or the specification of the subject, is instantiated with the referenced object (e.g., *"the emergency door"*). The lexical representation expressed in the CATEGORY frame element is then linked to the pointed position in the space by using the information obtained with the dot detection system. Finally, given the position of the robot, the position of the detected dot and the categorization command, a new instance signature is created. Every time a new structure of this sort is instantiated, a processing step is performed in order to update the Cell Map with the symbolic representation of the new object paired with information about its position, as described in the following section.

IV. INITIAL SEMANTIC MAP BUILDING

The module that builds the semantic map takes as inputs the metric map and the tags added to the map through the interaction with the user and produces, as output, the representation described in Section III-A, specifically the cell map and the topological graph.

The tags associated with the instance signature can be of two types: area tags, each consisting of the name of the area and the position of the robot when the operator provides the information, or object tags, each consisting of the name of the object, its position (x, y, θ), its size and its properties. The procedure for building the semantic map relies on two basic sub-modules, which directly work on the metric map and on the tagged objects. The first one, denoted as the *Grid Extractor*, creates the grid that defines the cell map, by applying the Canny Edge Detector and using the Hough Transform with a multi-resolution approach to find the lines corresponding to the edges of the map. A grid, which is consistent with the structure of the building scanned by the laser, is produced adaptively from such lines. In detail, the smallest distances between each pair of horizontal and vertical lines (respectively y_{min} and x_{min}) are computed: the cells produced have therefore a size between $y_{min} \cdot x_{min}$ and $2y_{min} \cdot 2x_{min}$.

The second sub-module, the *Room Map Builder*, determines the areas of the environment that can be aggregated, based on the tags provided by the user, i.e., it associates each tag with a certain area of the metric map. In order to achieve this and fully reflect the structure of the building, some lines corresponding to the tagged doors (which simply are objects containing the substring "door" in its name) are added to the map. The Watershed algorithm is then applied, using the positions corresponding to the area tags as markers. From the output of the algorithm, each area can be simply extracted through a color based segmentation procedure, considering the color of the resulting image in the position corresponding to the tag of that specific area. Furthermore, a module logically parallel to the Room Map Builder, called *Object Handler*, computes the vertices in the map for each tagged object, based on its position and size.

The resulting grid determines the Cell Map, where each element, corresponding to a cell, contains the following data:
1) A string representing the tag of the area: if the string is empty (no tag), the whole cell is considered to be empty;
2) A vector (which can be empty) containing the strings representing the tags of each object, that is located in the cell;
3) A vector of boolean values representing the presence of a door between that cell and the surrounding ones;
4) A vector containing an item for each tagged object, each of them listing the corresponding properties (e.g., the color).

Based on the Cell Map, a topological graph is also created automatically. The constituting nodes of such a topological graph represent physical locations on the map, while the edges are the connections between them. For each room, two kinds of nodes are produced: static and dynamic ones. Static nodes represent locations that have a fixed position on the map and are used to deal with critical pathways in the map (i.e., doors), where a specific behavior is required. Dynamic nodes represent, instead, variable positions within a given area of the environment. Thus, for each room mapped in the semantic map, a dynamic node is created while, for each doorway that connects such a room to another, a static node is generated with its fixed position set in front of the considered door. Each static node is connected to the dynamic node of the room it belongs to and to the static node associated with the the location across the door. Finally, the facts embodied in the matrix are stored into a Prolog knowledge base. For each object tagged by the user, two predicates *object(Name, Position, Properties)* and *ObjectType(ObjectName)* are created in order to perform inference on it.

[1] http://labrococo.dis.uniroma1.it/?q=s4r

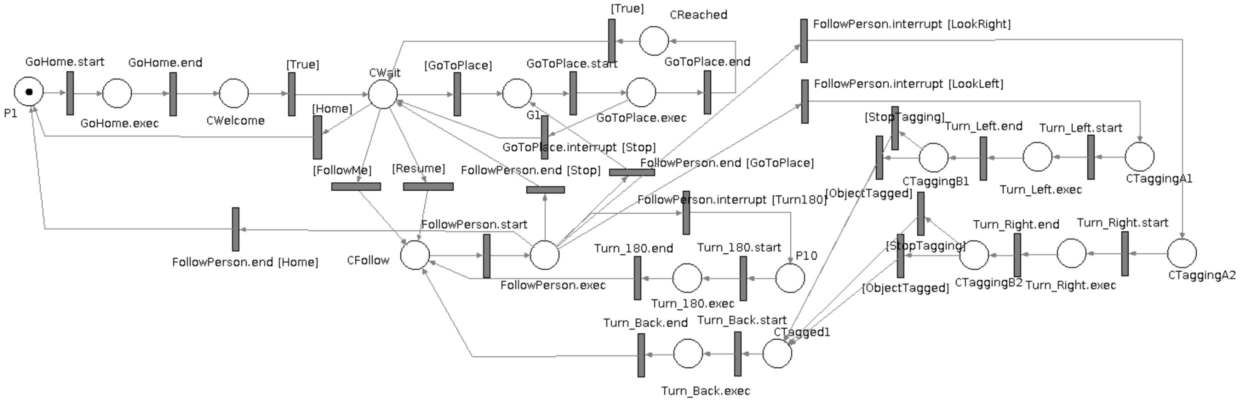

Fig. 2. The Petri Net Plan (PNP) used in the experiments for acquiring knowledge through user interaction.

V. On-line Knowledge Acquisition

The second contribution of our work is on providing the robot with the ability of on-line knowledge acquisition in the previously outlined framework, through a multi-modal human-robot interaction. To this end, we have developed and integrated in the overall system a high-level plan that represents the complex behavior of the robot interacting with the user. When the plan is successfully executed, the semantic map is augmented with the newly acquired knowledge.

The overall behavior of the robot during the test has been represented using the Petri Net Plans (PNP) formalism [20]. PNPs are used here to properly combine all the actions and the perceptions of the robots, including mobile base navigation, speech synthesis and recognition, and perceptions, as described in Section III. More precisely, PNPs implement the general behavior of the robot for gathering information about the environment through human-robot interaction and for providing these information to the module that updates the semantic map.

An example of PNP used in the experiments is shown in Fig. 2. It contains a combination of *actions* and *conditions*: actions are behaviors implemented in the robot, while conditions are used to enable the transitions to different behaviors, that are evaluated either through sensor processing routines or in the semantic map. The actions considered in this plan are the following: 1) 'GoHome': go to a predefined starting position; 2) 'FollowPerson': follow the person in front of the robot within the environment; 3) 'Turn_Left'/'Turn_Right'/'Turn_180': turn to look for objects as suggested by the user; 4) 'GoToPlace': go to the place named by the user.

The implementation of the navigation actions is based on standard localization, path planning and obstacle avoidance modules provided by the ROS framework, while the person following behavior is implemented with a simple PD controller that maintains the robot at a given relative position with respect to the person. The speech synthesis actions are implemented by exploiting the Microsoft TTS.

The conditions used in this plan are: 1) 'FollowMe': a voice command to follow the robot is received; 2) 'LookLeft', 'LookRight': voice commands to let the robot turn left/right; 3) 'ObjectTagged': the user specifies the category of the object and the laser dot has been detected, 4) 'GoToPlace': the user specifies a location or an object to be reached by the robot; 5) 'Home': the user commands the robot to go in the starting position and (re-)start the demonstration; 6) 'Stop'/'Resume': voice commands to interrupt/resume the demonstration.

Roughly speaking, the overall behavior of the robot is as follows: the robot goes to a predefined starting position and waits for the user commands; if the command can be grounded in the semantic map, it is executed, otherwise the user is asked for help; when a command to follow the user is received, the robot follows the user looking around as suggested; when the dot projected by the laser pointer on an object is recognized, the robot estimates the position of the dot and the orientation of the surface around the dot, then when it receives the object description from the user, it associates the object pose and its description and builds the representation of the object to be added to the semantic map. Upon the final confirmation by the user, the update of the semantic map is achieved by creating the corresponding cell map as explained in the previous section and then replacing it in the map.

In addition to actions and events, robot states are also associated with the places of execution of the Petri Net. Some of these states are used for contextual execution of some behaviors and for the human-robot dialogue. For example, in the state '*CWelcome*', the robot greets the person and tells it is ready to start; in the states '*CTagging**', the robot is in tagging mode, where a special grammar for receiving object descriptions is loaded and a corresponding dialogue is activated.

VI. Experiments

The main goal of our experiments is to show the feasibility, the effectiveness, and the robustness of the proposed approach for on-line construction of the semantic map. Consequently, we have measured the performance at system level and not on the single components.

The experiments have been conducted by executing robot behaviors similar to the one described in the previous section, across different settings, including multiple robots, several users, and two very different kinds of environments: our Department office (Fig. 1) and a home (Fig. 3). Videos of some of the experiments and several data acquired during them can be found at *www.dis.uniroma1.it/~iocchi/RobotExperiments/SemanticMaps/ICAR13.html*.

More in detail, we used two different wheeled robots, suitable for operation in indoor environments and equipped with laser range finders and Kinect cameras. On the two robots similar functionalities for autonomous navigation and other behaviors are implemented by relying on the ROS architecture.

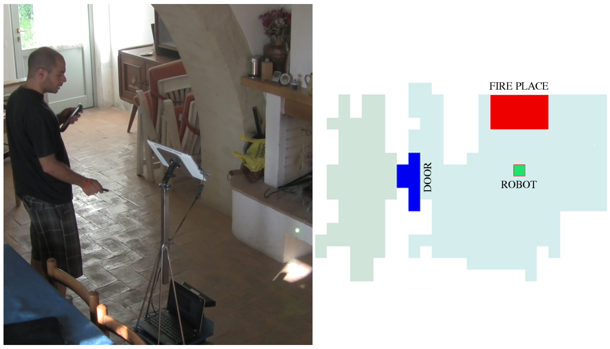

Fig. 3. Experiment in the home environment. A user is tagging the fire place (left), which is automatically added to the cell map (right).

It is worth emphasizing that, with the exception of hardware dependent modules, the two robots run the same code. Another difference in the hardware set-up is in the use of an on-board microphone vs. a hand-held microphone. The comparison of the two settings is beyond the scope of the work reported here. Our experience confirms that the hand-held microphone ensures better performance in the speech recognition module, but the device needs to be carried and operated by the user.

The experiments have been run during three weeks in our Department and in a weekend in the home shown in the above picture. Given the interactive nature of the task, we aimed mainly at verifying the robustness of the proposed approach.

The system allowed to consistently build and update semantic maps of the two environments, over a large set of tests, that included 6 rooms and 20 objects in the office environment, and 2 rooms and 41 objects in the home environment. The robot behavior has been designed so that the user can smoothly compensate for robot perception and positioning inaccuracies, thus eventually achieving an accurate and rich semantic map.

Overall, the system recorded more than 2,000 sentences uttered by at least 10 different persons (including all the authors of this paper) and more than 1,000 images with the dot detected during tagging. As a side result, we have also obtained an interesting data set including spoken commands and RGB-D images acquired in real operation (see above URL), that can be used to test the performance of speech understanding and object recognition in similar robotics applications.

VII. Conclusion

In this paper we have presented an incremental method for on-line semantic map acquisition, based on multi-modal human-robot interaction. While previous approaches to semantic mapping typically are based on a systematic exploration of the environment, here we start from an initial state-of-the-art semantic map, which includes a metric map produced by a SLAM algorithm, a labeling of the different locations and areas of the environment, and, possibly, a description of a first set of objects. In the paper, we have shown how to extract an effective symbolic representation from such semantic map and how to incrementally add new elements to it through an open-ended process.

The proposed method has been tested with two different robots, in different environments and with many users. In our view, the system has been designed for the acquisition of new knowledge for extended periods of time. To this end, a number of additional features need to be addressed. Among them, we are looking at a seamless extension of the language, so that the knowledge about unknown objects can be gathered through external sources, and to the issues that arise in the update and maintenance of the knowledge base, when considering dynamic objects that change location.

References

[1] A. Nüchter and J. Hertzberg, “Towards semantic maps for mobile robots,” *Robot. Auton. Syst.*, vol. 56, no. 11, pp. 915–926, 2008.

[2] G. Randelli, T. M. Bonanni, L. Iocchi, and D. Nardi, “Knowledge acquisition through humanrobot multimodal interaction,” *Intelligent Service Robotics*, vol. 6, pp. 19–31, 2013.

[3] J. Hertzberg and A. Saffiotti, “Using semantic knowledge in robotics,” *Robotics and Autonomous Systems*, vol. 56, no. 11, pp. 875–877, 2008, semantic Knowledge in Robotics.

[4] P. Buschka and A. Saffiotti, “A virtual sensor for room detection,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2002, pp. 637–642.

[5] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J. Fernández-Madrigal, and J. González, “Multi-hierarchical semantic maps for mobile robotics,” in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2005, pp. 3492–3497.

[6] A. Nüchter, O. Wulf, K. Lingemann, J. Hertzberg, B. Wagner, and H. Surmann, “3D Mapping with Semantic Knowledge,” in *RoboCup 2005: Robot Soccer World Cup IX*, 2005.

[7] O. Martínez Mozos and W. Burgard, “Supervised learning of topological maps using semantic information extracted from range data,” in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2006, pp. 2772–2777.

[8] N. Goerke and S. Braun, “Building semantic annotated maps by mobile robots,” in *Proceedings of the Conference Towards Autonomous Robotic Systems, Londonderry, UK*, 2009.

[9] E. Brunskill, T. Kollar, and N. Roy, “Topological mapping using spectral clustering and classification,” in *Proc. of IEEE/RSJ Conference on Robots and Systems (IROS)*, 2007.

[10] J. Wu, H. I. Christenseny, and J. M. Rehg, “Visual place categorization: Problem, dataset, and algorithm,” in *Proc. of IEEE/RSJ Conference on Robots and Systems (IROS)*, 2009.

[11] O. M. Mozos, H. Mizutani, R. Kurazume, and T. Hasegawa, “Categorization of indoor places using the kinect sensor,” *Sensors*, vol. 12, no. 5, pp. 6695–6711, 2012.

[12] A. Diosi, G. Taylor, and L. Kleeman, “Interactive slam using laser and advanced sonar,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 1103–1108.

[13] C. Nieto-Granda, J. G. R. III, A. J. B. Trevor, and H. I. Christensen, “Semantic map partitioning in indoor environments using regional analysis,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010, pp. 1451–1456.

[14] G. Kruijff, H. Zender, P. Jensfelt, and H. Christensen, “Clarification dialogues in human-augmented mapping,” in *Proc. of the 1st Annual Conf. on Human-Robot Interaction (HRI’06)*, 2006.

[15] H. Zender, O. Martínez Mozos, P. Jensfelt, G. Kruijff, and W. Burgard, “Conceptual spatial representations for indoor mobile robots,” *Robotics and Autonomous Systems*, vol. 56, no. 6, pp. 493–502, 2008.

[16] A. Pronobis and P. Jensfelt, “Large-scale semantic mapping and reasoning with heterogeneous modalities,” in *Proc. of the 2012 IEEE Int. Conf. on Robotics and Automation (ICRA’12)*, 2012.

[17] S. Rosenthal, J. Biswas, and M. Veloso, “An effective personal mobile robot agent through symbiotic human-robot interaction,” in *Proc. of 9th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2010.

[18] T. Kollar, V. Perera, D. Nardi, and M. Veloso, “Learning environmental knowledge from task-based -robot dialog,” in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA-13)*, vol. 21, 2012.

[19] L. Aiello, E. Bastianelli, L. Iocchi, D. Nardi, V. Perera, and G. Randelli, “Knowledgeable talking robots,” in *Artificial General Intelligence*, ser. Lecture Notes in Computer Science, 2013, vol. 7999, pp. 182–191.

[20] V. Ziparo, L. Iocchi, P. Lima, D. Nardi, and P. Palamara, “Petri Net Plans - A framework for collaboration and coordination in multi-robot systems,” *Autonomous Agents and Multi-Agent Systems*, vol. 23, no. 3, pp. 344–383, 2011.