Autonomous Mapping of Kitchen Environments and Applications

Zoltan Marton, Nico Blodow, Mihai Dolha, Moritz Tenorth, Radu Bogdan Rusu, Michael Beetz
Intelligent Autonomous Systems, Technische Universität München
{marton, blodow, dolha, tenorth, rusu, beetz}@cs.tum.edu

***Abstract*— This paper investigates the problem of acquiring 3D semantic maps of indoor household environments, in particular kitchens. The objects modeled in these maps include cabinets, tables, drawers and shelves, and objects in the environment that have particular relevance for a household robotic assistant. Our mapping approach is based on point cloud representations, and a set of sophisticated interpretation methods operating on these representations. We outline the steps of our mapping approach, explain the key techniques that make it work and present their use for different applications.**

I. Introduction

The usage of robots that aid us is becoming more and more widespread, typically in the industry, but increasingly in public and in some cases, home applications as well. Robots are starting to be more flexible, being able to do almost anything from building cars to riding roller-skates, but virtually all of these complex actions have to be preprogrammed, as their ability to recognize complex patterns is fairly limited.

The only way a robot can be truly autonomous is by its capacity to learn ”on the road”, from its own experiences. To accomplish this, it needs to understand what the ”road” is, and what's the best way to explore the information that it provides. Simply put, it needs a *semantic map*. [1]

The purpose of the majority of maps acquired and used by robots is to aid them in localization and navigation [2], thus besides obstacles, they don't represent objects relevant for other tasks. There are a few exceptions, for example in the cognitive mapping area [2], [3], [4]. Performing manipulation requires much more details though: interesting objects need to be differentiated from obstacles and knowledge is needed about how those objects can be manipulated (for example to open a cupboard the handle has to be found and pulled on a trajectory that can be intuited from the placement and shape of the handle).

A very good source of information, for both humans and technical systems alike, is obtained from visual data, which should be understood here in a broader sense, not only image and color information, but also anything that gives hints about the geometry of the environment and the position of the objects in it. So, the tactile senses of plants, animals and humans[1] and their extensions[2], or 2D laser scanners and proximity sensors, etc. give *visual data* just like images from the eye or camera sensors do.

The set of possible actions and prior knowledge on its turn is also acquired naturally and most easily by visual observation. Apart from the genetically encoded reflexes, this is how evolved beings learn, and the machine learning methods developed for computers enable them to try to imitate this process.

This way a simple 3D image of the environment and of the objects in it is transformed into a map filled with semantic information.

The remainder of the paper is organized as follows. The next section gives a brief overview of our system, followed by a discussion on related work in Section III. In Section IV we give an overview of the mapping algorithms. Section V describes our methods for acquiring and interpreting point cloud data. Section VI applies the presented methods to build three dimensional Semantic Maps, and gives a short introduction to their possible usage, and Section VII talks about integrating the results in a knowledge-base. We conclude with a short discussion and give our conclusions together with a sketch of our future work.

II. System Overview

A. Setup of the Problem

Our setup for this paper is the following one. We have a mobile robot equipped with two arms with grippers acting in a kitchen environment (see Figure 1). The task of the robot is to explore the environment and build a comprehensive 3D model of the environment that contains models of objects that are relevant for the robot's task as a household assistant. These objects include cabinets, drawers, and table tops.

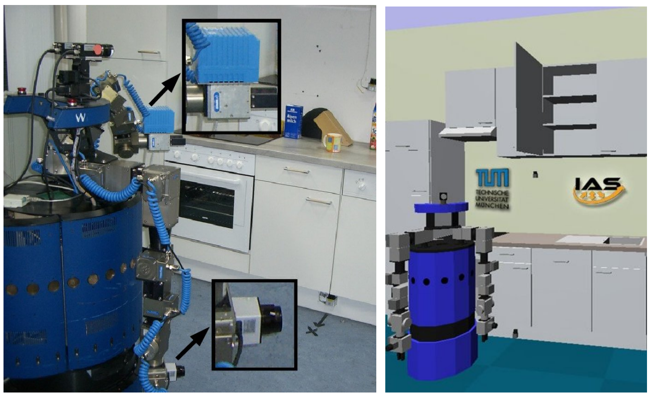

Fig. 1. A mobile B21 robot with two arms and laser scanning capabilities shown in the real (left and midle) and simulated kitchen (right).

The primary sensing device used for map building consists of two laser scanners mounted on the robot's end effectors.

[1]like the Braille script used by blind people (devised by Louis Braille)
[2]like the white cane used to identify obstacles by the visually impaired

The robot has basic problem-specific manipulation skills: it can open and close cabinets and drawers, it can reach into cabinets, and make accurate and smooth sweeping motions with its arms and hands to acquire accurate 3D point clouds.

Besides the sensors on board of the robot, the environment is equipped with a sensor network of distributed and heterogeneous sensing units including fixed laser range sensors on the walls, magnetic sensors that report whether doors are open or closed, and RFID tag readers that report the RFID tags within their sensor range, typically placed in a cabinet or in places such as under the table [5].

In general the mapping problem is to infer the semantic object map that best explains the data acquired during the mapping process. The data is comprised of several parts (i.e. snapshots, scans), which cover the environment that is to be mapped. These partial views of the world have to be assembled together to form a complete model in a process called registration. In our work, we are interested in registration in both time and space domains. We make little assumptions about the overlap between two views, but naturally we expect that a minimal overlap exists.

The resultant *PCD (Point Cloud Data) world model* is considered the input of our functional mapping system. Figure 2 presents the overall system architecture.

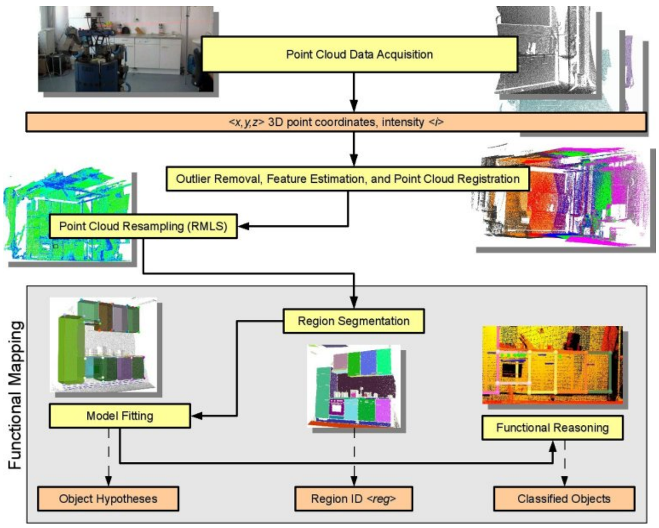

Fig. 2. Brief overview of the overall system architecture.

Since the topic of this paper focuses on mapping and map usage, we will not address the modules outside its scope (namely Outlier Removal, Feature Estimation, Registration, and Resampling), as they have already been covered in our previous work [1], [6], [7], [8].

B. Map Representation

The output of our system consists of a compact obstacle representation of the environment. In addition, our algorithms segment and represent areas of interest, such as cabinets and drawers in their respective states, as well as other objects with their respective furniture category. The system also represents and labels tables and shelves explicitly. Separate actions are taken towards the recognition of objects located on planar surfaces [1], [9].

A map such as the one shown below can be created from a single, partial scan, and updated whenever new scans are registered and made available to the system.

```
<rdf:RDF>
  <map:Cupboard rdf:ID="cupboard1">

    <map:widthOfObject
        rdf:datatype="#float">0.34</map:widthOfObject>
    <map:depthOfObject
        rdf:datatype="#float">0.86</map:depthOfObject>
    <map:heightOfObject
        rdf:datatype="#float">0.60</map:heightOfObject>

    <map:center>
      <map:Point3D rdf:ID="Point3D_2">
        <map:yCoord
            rdf:datatype="#float">2.983215</map:yCoord>
        <map:zCoord
            rdf:datatype="#float">2.15</map:zCoord>
        <map:xCoord
            rdf:datatype="#float">0.17</map:xCoord>
      </map:Point3D>
    </map:center>
    <map:properPhysicalParts>
      <map:UShapedHandle rdf:ID="handleCupboard1"/>
    </map:properPhysicalParts>
    <map:properPhysicalParts>
      <map:Door rdf:ID="doorCupboard1">
        <map:doorHingedTo rdf:resource="#cupboard1"/>
      </map:Door>
    </map:properPhysicalParts>
    <map:properPhysicalParts>
      <map:RfidReader rdf:ID="rfid1">
    </map:properPhysicalParts>
    <map:contains rdf:resource="#plate2"/>
  </map:Cupboard>
</rdf:RDF>
```

C. Use Case

An example scenario which depicts the way our system works is described below:

- the robot enters a new kitchen without knowing anything about it
- if the kitchen is equipped with some sensing or computational capabilities, the robot connects to its sensor network and inquires whether the environment has a model of itself [5]
- if the environment can provide a snapshot taken in a previous mapping stage, the robot proceeds in actualizing it by creating a new snapshot and merging the two together; if not, it simply creates a new map
- the map is created by fusing data from various sensors using a few assumptions about the environment, and is represented in a compact format

III. Related Work

Several efforts in the past have been made regarding the creation of environmental object maps out of 3D range data. Since the creation of such maps is a highly complex process and involves the combination of several algorithms, we will try to address the most relevant publications for our work below. Related work on particular dimensions will be addressed in their respective technical sections.

An EM-based algorithm for learning 3D models of indoor environments is presented in [10]. The maps are created using mobile robots equipped with laser range finders, but they do not include any semantic information. The work in [11] uses a stereoscopic camera system and a knowledge base in the form of a semantic net to form 3D models of outdoor environments. Two parallel representations, one spatial and one semantic, are proposed in [12] for an indoor environment, but their approach needs further investigation. An object based approach for cognitive maps is used to recognize objects and classify rooms in different categories in [13]. The work presented in [14] provides a method for classifying different places in the environment into semantic classes like doorways, kitchens, corridors, rooms using simple geometric features extracted from laser data and information extracted from camera data. The semantic interpretation in [15] takes into account generic architectural elements (floor, ceiling, walls, doors) which are identified based on the relationships between the features (parallel, orthogonal, above, under, equal height). Finally, a decomposition of maps into regions and gateways is presented in [16].

With few exceptions, in particular in the area of cognitive mapping [13], [3], but also including [17], [4], maps do not represent objects relevant for other robot tasks, besides navigation. That is why none of the above mentioned representations can be used in an aplication scenario such as ours, where the household robotic assistant needs fine-grained semantic information in order to manipulate objects in the environment, and complete complex tasks.

We consider the development of functional mapping acquisition routines that can reliably recognize objects such as cupboards, drawers, etc, as an essential building block for the realization of autonomous robots capable of building semantic maps of their environments.

IV. Overview of the Mapping Algorithms

At a conceptual level, our mapping algorithms are organized as follows:

- Acquisition and interpretation of 3D scans representing partial views of the environment model. (Section V)
- Based on the augmented point cloud model, generated in the previous processing step, using additional assumptions about the structure of the environment, semantic objects are extracted and their representation is derived. In this step we propose object hypotheses for individual cabinet doors, tables tops, shelves, and recognize if the "cabinets" are appliances such as ovens, dishwashers, etc. (Section VI)
- The spatial knowledge obtained from processing the point cloud data can be combined with ontological knowledge about its properties, uses and functions. The combination of these two kinds of knowledge yields a very rich semantic representation of the environment. (Section VII)

The methods and techniques used in the steps of our mapping algorithm are detailed in the subsequent sections.

V. Point Cloud Data Acqusition and Geometric Interpretation

Our algorithms assume all data as being represented by *point clouds*. We acquire data from various sources, using either real sensors (laser, stereo, time-of-flight cameras), 3D simulators (Gazebo), or 3D CAD models (Google 3D Warehouse) as explained below. We make little or no assumptions about the underlying structure of the data, as our goal is to extract information robustly from it for a larger variety of problems.

The acquisition of point clouds from sensing physical devices is performed through the usage of the Player project [5], of which we are active developers.

No matter what assumptions one can make about a certain type of environment, the solution found for that particular case cannot be generalized properly, unless the same problem has been solved for at least a dozen of scenarios with similar environments. Obviously, this is not an easy thing to do, especially for household environments, as the resources to build apartments or find sites that could be used for research tests are very limited. One of the solutions that we found to tackle this problem is the usage of 3D simulation tools such as Gazebo, which could allow us to get similar data as we would get from a real site. To make the data more realistic we add zero mean Gaussian noise to it with various standard deviations (see Subsection VI-A). We also propose the conversion and use of Google SketchUp models available online as additional training data (see Figure 3), which already proved useful for our application [1].

The resulted point clouds are then all processed in the same manner. This assures that plenty of training examples will be available for learning our environmental and object models.

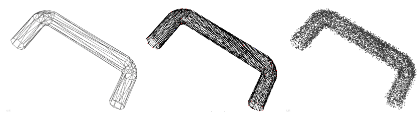

Fig. 3. Using CAD models as additional training data. From left to right: (a) polygonal data set, (c) resampled point cloud model, (d) model with added zero-mean Gaussian noise

The obtained point cloud data (PCD) contains noise and measurement errors, which must be suppressed in order to extract *good* information. Based on assumptions on features and properties of objects and in combination with different sensory data, semantics can be added, which can be used for interpretation and decision making about future actions or in the learning process. The number of points in a PCD consist of 50,000-5,000,000 points for any partial view, thus we need to assure that our algorithms can deal with large amounts of data, and clearly, some optimizations are needed in order to process them.

We apply the methods presented in [1], [6], [7], [8] for estimating point correspondences, registering the scans into a common model and to resample the points on the underlying surfaces in order to reduce sensor noise and unnecesarry point densities while filling small holes.

After segmenting the scene into regions, we look at the regions with the smallest curvature and we approximate them with planar surfaces (see [8]). After projecting the inliers on the model plane, we triangulate them and compute the total area of the region $A = \sum_1^T A_i$, where T is the total number of resulted triangles on the surface, and A_i is the area of triangle i. This information will be used later on together with a set of assumptions to build the semantic map.

For finding out furniture fixtures, we look whether there are any holes in the region, identify them using the boundary point detection and then look in the original point cloud to see whether we have points which are located at a distance $dist_i$ in between some predefined threshold $d_{min} \leq dist_i \leq d_{max}$. The fixtures' position and type are used by the mobile robot to open the cupboards, as presented in Subsection VI-B.

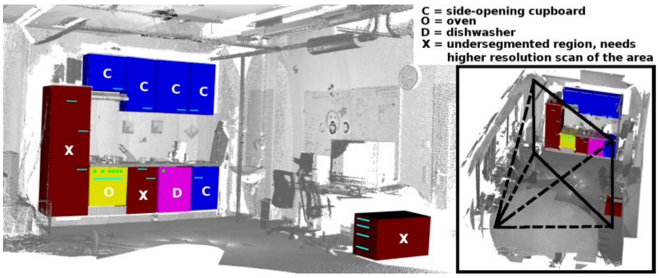

Fig. 4. From raw point cloud data towards semantic maps.

The final result of the mapping procedure is presented in 4, with the identified objects labeled on top of the point cloud.

VI. Semantic Maps

The first step towards semantic maps is comprised of fusing the results presented in the previous section with a given set of assumptions. We have already implemented such a system [1] for detecting several object types, and are extending its definitions towards higher-level features for environment learning. The second step is to add information about the *status* of a certain part of the environment, such as: what type of objects does a cupboard currently contain, as well as add the position and capabilities of the extra installed sensors in the environment (where present) to the map. We start by defining the higher level feature set, and then proceed to an usage example of the map with our robot.

A. Model Learning

In a kitchen environment, the main parts identified in range data are walls, furniture pieces and appliances. While these objects are formed of large and smooth surfaces having characteristics of regular geometric shapes, the features extracted out of their point clouds representation is most of the times inadequate for proper recognition of distinct entities.

The previously extracted geometric features are considered *low-level* in our scenarios, as they lack the ability to generalize a given scene, especially because an indoor environment's appearance varies greatly. Therefore they are not suitable for any kind of semantic interpretations. To overcome this, we propose the use of *higher-level* features, which can be built upon the geometric data by applying a few simple assumptions. By using them in the classification process, the results tend to improve a lot in comparison to the low-level feature approach.

We define regions of interest in a given scene as subsets of a point cloud for which we can extract *some* characteristic information that might indicate something about their shapes or colors. Therefore, we classify the point cloud into dynamic objects (like mugs, silverware, boxes, etc.), regions for we can extract geometric features (cupboards, handles, knobs, etc) and we assume that everything else is either an obstacle or noise.

Our goal is to extract those parts that add information that could be useful in the classification and recognition stage. With this in mind, we first segment the environment into two major categories of areas: a) horizontal surfaces and b) vertical surfaces. Horizontal surfaces will define places such as tables, chairs, etc, while vertical surfaces might hint the existence of cupboards, or other kitchen furniture. The assumption is that both types must be comprised within some geometrical limits, therefore we take only those parts with an area between A_{min} and A_{max} based on common sense knowledge, and start particularizing from that.

After obtaining several regions from the point cloud, we analyze the relationships between them and the surfaces they were collected from. These relationship indicate the existence of knobs, handles, their number, dimension and position, the area of the surfaces they lie on, etc. With this new information, we create a higher level semantic feature map, which will be used for training a classifier.

We are currently investigating ways of automatically creating and extracting such higher level feature maps out of sensory data, but preliminary tests showed that classifiers trained on synthetic data have an accuracy of about 95%. These results are very preliminary and are not obtained from real data yet, but they show that the techniques might scale towards realistic varieties of kitchen environments. To make sure that we obtain sufficiently diverse sets of training data, we use kitchen models from furniture department stores and specialized design companies[3].

In order to use this data we have implemented software tools for translating the CAD models into Gazebo compact simulation models, so that we can simulate the aquisition by laser sensor of point cloud data that is similar to the ones coming from real sensors (see Figure 5).

B. Interacting with the Environment

We demonstrate the usage and importance of having a detailed semantic map of the environment in an application scenario using our mobile robot. Given a robot equipped with manipulation capabilities in a household kitchen environment, our goal is to learn what a cupboard is, where it is

[3]We acknowledge the support of Plantek and Segmüller department stores for providing the three dimensional models.

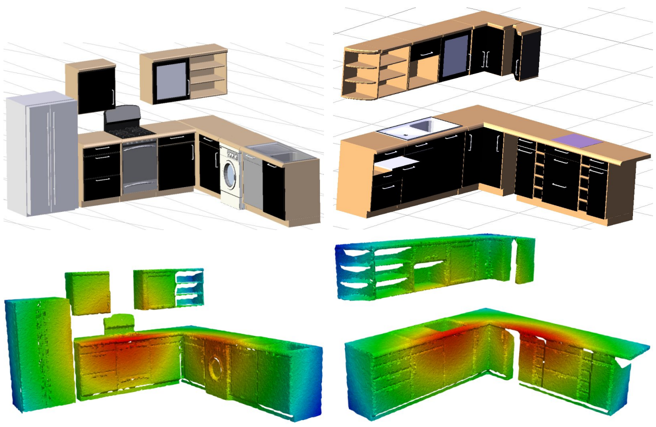

Fig. 5. 3D kitchen models (top) and the triangulated point cloud data obtained by a simulated scan (bottom) – the colors represent the distance to the laser sensor (red is closest, blue is farthest).

located, and how to open it. Obviously, in order to interact with the real world and perform such complex manipulation tasks, the robot needs a detailed 3D description of the environment, such as what type of cupboard and handle it has to deal with. It can use the physical simulation of the obtained map in Gazebo to verify it's assumptions and to test different manipulation strategies (see Figure 6).

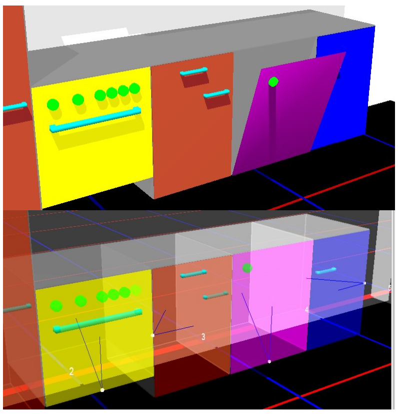

Fig. 6. Functional object map automatically imported in the Gazebo 3D simulator and the identified opening positions (top) and hinges (bottom).

To define the trajectory that would open a specific container we make use of the located handle's position to identify the opening direction [8]. In our current hardware setup, the mobile robot's manipulators are not equipped with force/torque sensors. Since there is no feedback from them during an action, any closed-loop control technique is rendered useless. Therefore, we are relying on a very precise trajectory generator which controls the end-effector. To help building this trajectory planner, we make use of the direct and inverse kinematics equations of the arm (see Figure 7.

To achieve its goals, the robot can also employ the additional sensors present in the environment as well.

The kitchen is instrumented with various sensing devices, such as laser sensors, magnetic sensors in the cupboard

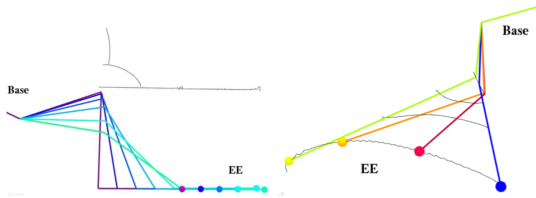

Fig. 7. Arm trajectories while opening a drawer (left) or cupboard (midle).

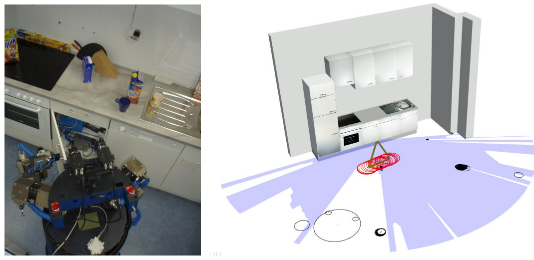

Fig. 8. The robot's movement in the kitchen as seen by a laser sensor installed in the environment; by assuming that the robot geometry is known or that the robot is the only entity moving in the environment we can detect the laser's position on the wall, thus making it a *valid* sensor for the robot's usage in future.

doors, RFID readers in the cupboards, etc [5], [1]. A simple laser sensor is enough for determining the relative position of the robot and the sensor. By determining the position of the laser in the environment and assigning an entry in the semantic map, the robot will be able to use that sensor in the future as part of its distributed sensing capabilities. In our example we assume that the mobile robot is localized with respect to the environment, and we try to detect the absolute position in the room of the laser sensor. Since our mobile robot is cylindrical in nature, we assume that looking for arcs in the set of laser range measurements suffices [18]. During the time the robot moves in the room the laser is queried repeatedly and arcs are determined. By assuming either that the robot is the only object moving in the scans, or that the robot knows it's own geometric structure, we can determine that the feature which constantly changes it's position is in fact the robot (see Figure 8).

VII. Integrating Ontological and Common-sense Knowledge

Classifying and localizing objects is just part of the way towards rich semantic environment maps. The sole fact that a set of points is classified as a "cupboard" does not give it a semantic meaning as long as the system does not know what a cupboard actually is, which properties it has and what the robot can do with it. Examples of knowledge that is required in addition to the pure spatial information are that cupboards can contain food, tableware or silverware, have a hinged door with a handle and make their interiors invisible when closed.

Our approach is to link the semantic map to a knowledge base in description logics that contains much of the desired facts: knowledge about the taxonomy of objects (cupboards are containers and can therefore contain things), their physical parts (doors, hinges, handles etc.) as well as functional knowledge about how and in which context are objects used.

We use the Cyc ontology and facts imported from the Open Mind Indoor Common Sense project as the basis for our robot's knowledge base and extend it manually as required, e.g. to adapt it more to the needs of a mobile robot.

Objects that are detected in the environment are represented as instances of the classes in our knowledge base which allows for querying all relevant knowledge and performing reasoning about objects which are grounded in actual sensor data. This makes it possible to query for objects that serve for a purpose, e.g. for storing cups (see Figure 9), and get their spatial properties as the result. In the figure, all objects identified as cupboards are returned since the query asks for objects where cups are stored.

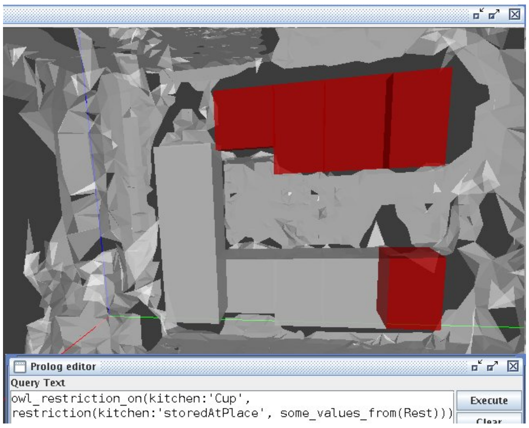

Fig. 9. The results obtained by querying the OWL-DL representation to locate containers where cups can be stored.

VIII. Conclusions and Future work

We have presented a system for building semantic maps of indoor household environments (i.e. kitchens). Given partial views of the environment as point clouds, we have developed and extended techniques for segmentation, that look promising for applications such as ours. The significance of this work lies in the development, integration and improvements of several techniques from different research fields such as computer graphics, robotics, machine learning and scientific computing, as well as in the results presented.

We are currently investing ways of adding more semantic information to our maps, by making use of sensor networks [5]. In this paper, we've already taken the first steps by attempting to detect laser sensors potentially installed in the environment, based on simple, yet realistic assumptions.

Further work has to be done in the area of object segmentation and recognition. Using the features presented in [1], [9] for object recognition to train a classifier gave good preliminary results. We plan on extending these to deal with partial occluded objects in natural scenes, as well as generating CAD-like models from the data and then using them against a priori learned models.

Acknowledgments

This work is supported by the CoTeSys (Cognition for Technical Systems) cluster of excellence.

References

[1] R. B. Rusu, N. Blodow, Z. Marton, A. Soos, and M. Beetz, "Towards 3D Object Maps for Autonomous Household Robots," in *Proceedings of the 20th IEEE International Conference on Intelligent Robots and Systems (IROS), San Diego, CA, USA, Oct 29 - 2 Nov.*, 2007.

[2] S. Vasudevan, S. Gachter, V. Nguyen, and R. Siegwart, "Cognitive Maps for Mobile Robots - An object based approach," 2007.

[3] J. Modayil and B. Kuipers, "Bootstrap learning for object discovery," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-04)*, 2004, pp. 742–747.

[4] R. Triebel, Óscar Martínez Mozos, and W. Burgard, "Relational Learning in Mobile Robotics: An Application to Semantic Labeling of Objects in 2D and 3D Environment Maps," in *Annual Conference of the German Classification Society on Data Analysis, Machine Learning, and Applications (GfKl)*, Freiburg, Germany, 2007.

[5] R. B. Rusu, B. Gerkey, and M. Beetz, "Abstractions, interfaces and reusable code: Robots meet Sensor Networks," 2007.

[6] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Aligning Point Cloud Views using Persistent Feature Histograms," in *Proceedings of the 21st IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Nice, France, September 22-26*, 2008.

[7] R. B. Rusu, Z. C. Marton, N. Blodow, M. E. Dolha, and M. Beetz, "Functional Object Mapping of Kitchen Environments," in *Proceedings of the 21st IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Nice, France, September 22-26*, 2008.

[8] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D Point Cloud Based Object Maps for Household Environments," *Robotics and Autonomous Systems Journal (Special Issue on Semantic Knowledge)*, 2008.

[9] R. B. Rusu, Z. C. Marton, N. Blodow, and M. Beetz, "Learning Informative Point Classes for the Acquisition of Object Model Maps," in *Proceedings of the 10th International Conference on Control, Automation, Robotics and Vision (ICARCV), Hanoi, Vietnam, December 17-20*, 2008.

[10] Y. Liu, R. Emery, D. Chakrabarti, W. Burgard, and S. Thrun, "Using EM to Learn 3D Models of Indoor Environments with Mobile Robots," in *ICML '01: Proceedings of the Eighteenth International Conference on Machine Learning*, 2001, pp. 329–336.

[11] O. Grau, "A scene analysis system for the generation of 3-D models," in *NRC '97: Proceedings of the International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, 1997, p. 221.

[12] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, and et al., "Multi-Hierarchical Semantic Maps for Mobile Robotics."

[13] S. Vasudevan, S. Gächter, V. Nguyen, and R. Siegwart, "Cognitive maps for mobile robots-an object based approach," *Robotics and Autonomous Systems*, vol. 55, no. 5, pp. 359–371, 2007.

[14] O. M. Mozos, A. Rottmann, R. Triebel, P. Jensfelt, and W. Burgard, "Semantic Labeling of Places using Information Extracted from Laser and Vision Sensor Data," in *In Proceedings of the IEEE/RSJ IROS Workshop: From sensors to human spatial concepts*, Beijing, China, 2006.

[15] A. Nuechter, H. Surmann, and J. Hertzberg, "Automatic Model Refinement for 3D Reconstruction with Mobile Robots," 2003.

[16] D. Schröter, T. Weber, M. Beetz, and B. Radig, "Detection and Classification of Gateways for the Acquisition of Structured Robot Maps," in *Proc. of 26th Pattern Recognition Symposium (DAGM), Tübingen/Germany*, 2004.

[17] D. Schröter and M. Beetz, "Acquiring Modells of Rectangular Objects for Robot Maps," in *Proc. of IEEE International Conference on Robotics and Automation (ICRA), New Orleans/USA*, 2004.

[18] J. Xavier, M. Pacheco, D. Castro, A. Ruano, and U. Nunes, "Fast Line, Arc/Circle and Leg Detection from Laser Scan Data in a Player Driver," in *International Conference on Robotics and Automation (ICRA)*, 2005, pp. 3930–3935.