Challenges in Using Semantic Knowledge for 3D Object Classification

Corina Gurău[1] and Andreas Nüchter[2]

[1] Automation Group, Jacobs University Bremen gGmbH, Germany
[2] Robotics and Telematics, University of Wüzburg, Germany

Abstract. To cope with a wide variety of tasks, robotic systems need to perceive and understand their environments. In particular, they need a representation of individual objects, as well as contextual relations between them. Visual information is the primary data source used to make predictions and inferences about the world. There exists, however, a growing tendency to introduce high-level semantic knowledge to enable robots to reason about objects. We use the Semantic Web framework to represent knowledge and make inferences about sensor data, in order to detect and classify objects in the environment. The contribution of this work is the identification of several challenges that co-occur when combining sensor data processing with such a reasoning method.

1 Introduction

Autonomous recognition of structure in an indoor environment is a challenging task for the robotics community. Relying on depth perception, prior knowledge and logic, humans are particularly adroit at understanding their surroundings. Robotic systems rely on imagery and sensor data to build and encode their knowledge. Yet, we expect some systems to perform tasks such as navigation, manipulation, or interaction, in cluttered environments, structured for humans. To improve the way robots structure their knowledge of the world, we can share a common knowledge management system. Then robots could use our way to represent, make inferences and take decisions. By finding a representation in Description Logic for common-sense statements, and mapping them to ontological concepts and relations between those concepts, information such as *the book is on the shelf* or *the room is empty* is shared between humans and robots. This high level semantic description through ontologies also permits reasoning in a logical way.

In this paper we aim at verifying if the bottom-up, knowledge-based interpretation of indoor scenes is a reliable approach for 3D object detection. This task has been heavily performed using statistical methods and pattern recognition. Detecting and classifying objects by relying on a logical representation has been less considered in recent years, due to the access to large amounts of data and computational resources to learn the structure of our visual world.

Our proposed system is used for knowledge modeling and information retrieval. We divide the task into three main components: 1) geometric analysis and characterization of scanned environment data, 2) semantic description and ontology mapping of geometric shapes, and 3) knowledge query and rule evaluation.

After scanning the environment, we use 3D point cloud segments to identify predefined geometric primitives and formalize spatial relations between object parts (cf. Fig. 1). We store the obtained geometric information and load it in a knowledge management system to populate an ontology with class instances. For answering queries over the computed spatial data, we implement a reasoner in Semantic Web Rule Language (SWRL) and run it under the platform of Protégé, an ontology editor and knowledge-base framework.

Space and spatial organization are the most common sense knowledge for humans. To describe them, we make use of Web Ontology Language (OWL). In our approach, we create an OWL ontology based on Description Logic (DL), which permits defining instances (description logic individuals), creating classes (description logic concepts), properties (binary relation specifying class characteristics), and operations (union, intersection, complement, etc). Our framework relies on reasoning with the 3D geometric information to detect and classify objects in a human environment. We consider properties such as size, orientation, position of point cloud segments, as well as spatial relations between segments, such as intersection, inclusion or parallelism. Our intuition in selecting the features is that it is easier to compute spatial relations for simple planar primitives of a complex object rather than computationally expensive ones for the whole object.

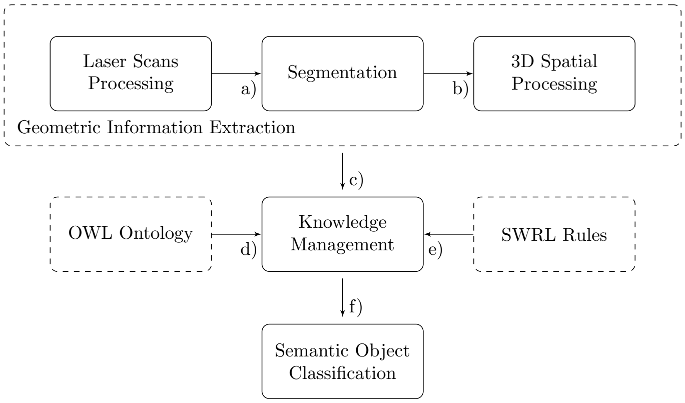

Fig. 1. Semantic Interpretation Pipeline. (a) Acquired laser scans are first segmented into planar regions. (b) Segments and spatial relationships are further analyzed. (c) Extracted geometric information is loaded in Protégé, on which our Knowledge Management framework is built. (d) An OWL Ontology encodes prior knowledge. (e) SWRL rules represent the restrictions we impose on the objects. (f) By running a reasoner on the information from (c), (d) and (e), objects of interest in the environment are detected and classified.

Related work in the area of combining 3D point cloud processing with knowledge-based reasoning is concerned with architectural reconstructions [2, 3]. A similar 3D object classification approach was taken by [4], however at critical points, the paper does not formulate solutions. In this paper, we focus on identifying the challenges in such an approach.

For the preprocessing phase we use the Felzenszwalb and Huttenlocher segmentation algorithm. Recently, we presented a segmentation method for 3D point clouds acquired with state-of-the-art 3D laser scanners extending the method of Felzenszwalb and Huttenlocher [1]. From the 3D points an unoriented graph is constructed. The graph is then segmented by using a k-nearest neighbor search and a similarity measure based on surface normals, resulting in a point cloud segmentation in planar patches. Fig. 2 shows two examples.

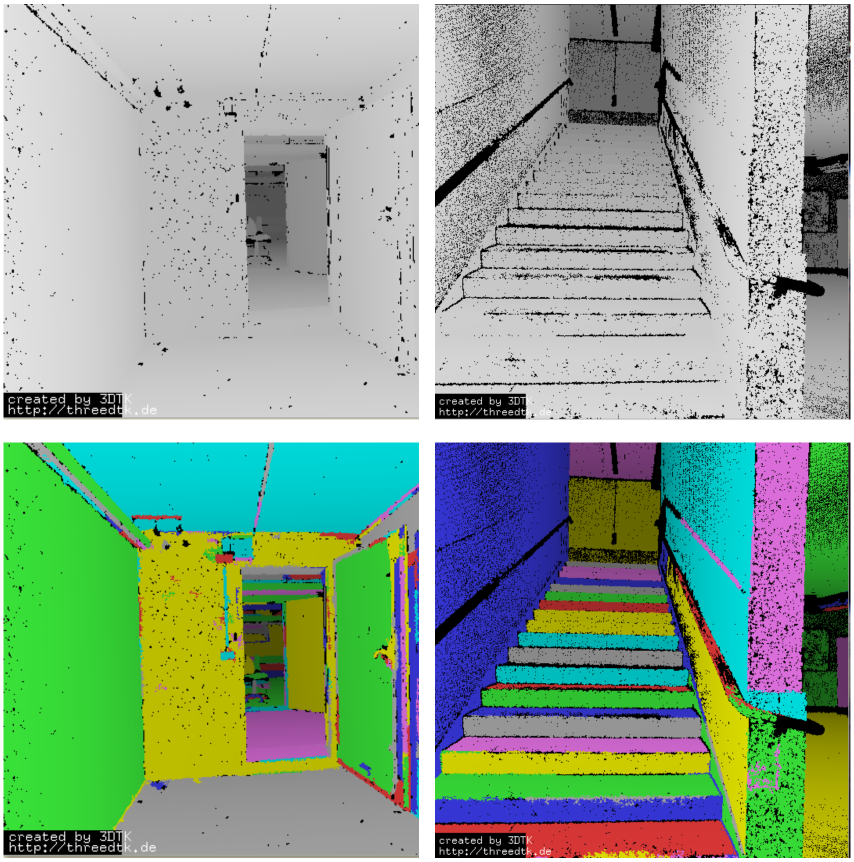

Fig. 2. Felzenszwalb and Huttenlocher Segmentation. Top: 3D Point Cloud of an empty room and of a staircase. The scene has been rendered with black fog to enhance depth perception. Bottom: 3D Point Cloud segmented using the parameters $\mu = 0, \sigma = 1$, $N = 10$, $k = 1000$.

2 The Protégé platform and Ontology Web Language

We model in an ontology our prior knowledge of the environment, making use of the Protégé-OWL editor. Protégé-OWL is an extension of Protégé that permits loading and saving ontologies, define logical class characteristics as OWL expressions, and most importantly, execute reasoners such as description logic classifiers. To complete the modeling process we add semantic rules developed with Semantic Web Rule Language (SWRL) and run Pellet, a Description Logic Reasoner, designed to work with OWL. Pellet is an implementation of a full decision procedure for OWL-DL which provides support for reasoning with individuals (asserted or inferred), user-defined datatypes and debugging and comparing ontologies.

Objects of interest in the scene are modeled under the class BuildingObject, while the rest map to geometries: either point cloud segments or pairs of point cloud segments. We therefore restrict our definition of an object to anything composed of them.

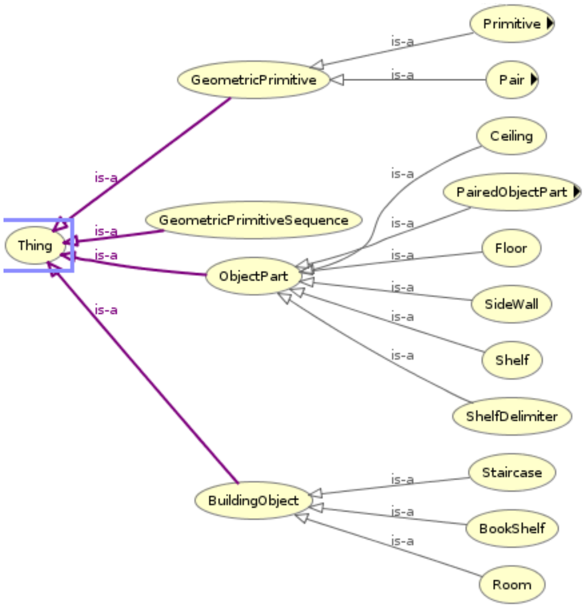

Fig. 3. Classes used for semantic interpretation.

Within the OWL ontology, not only we create appropriate object classes, but also class properties, through which we encode object geometry and spatial relations between segments in the scene (cf. Fig. 3). To integrate 3D data processing with Semantic Web technologies, we considered attributes such as:

size (Since we are only considering planar surfaces, we refer to size as the area of the segment. It is the most distinguishable segment property.), position (We consider minX, maxX, minY, maxY, minZ, maxZ as some objects are expected at a certain relative position inside a scene.), orientation (Individuals of vertical or horizontal segments are directly instantiated under the appropriate class.). Equally important as segment attributes, are the spatial relations between segments: connected, parallel, perpendicular, the pairs being instantiated under the classes Pair or PairedObjectPart.

3 SWRL Rules

The purpose of our semantic interpretation approach is to enable querying the spatial knowledge base. After populating our Protégé classes with individuals, we see their properties and their relationships as logical predicates (*asserted knowledge*), and we use logical rules to derive new facts and instances (*inferred knowledge*). The SWRL rules incorporate the restrictions that we impose on the environment: our knowledge about the scene configuration and about the shape of the objects. A rule takes the form of an implication between an antecedent and a consequent, and supports either a final decision or an intermediate decision in interpretation process. For instance, we know that a bookshelf essentially consists of a series of parallel segments at certain intervals. We make a similar judgement that if we have two stairs in the same sequence of primitives, the object is a staircase. Two example rules are as follows:

$$\begin{aligned}&\mathrm{LowShelf}(?\mathrm{x}) \rightarrow \mathrm{HorizontalSegment}(?\mathrm{x}) \wedge \mathrm{hasSize}(?\mathrm{x}, ?\mathrm{size})\\ &\quad \wedge \mathrm{swrlb}:\mathrm{greaterThan}(?\mathrm{size}, 0.02) \wedge \mathrm{swrlb}:\mathrm{lessThan}(?\mathrm{size}, 1.0)\\ &\quad \wedge \mathrm{hasMaxY}(?\mathrm{x}, ?\mathrm{maxY}) \wedge \mathrm{swrlb}:\mathrm{greaterThan}(?\mathrm{maxY}, 0.6)\\ &\quad \wedge \mathrm{swrlb}:\mathrm{greaterLess}(?\mathrm{maxY}, 1.5)\end{aligned}$$

$$\begin{aligned}&\mathrm{Staircase}(?\mathrm{x}) \rightarrow \mathrm{hasHVConnectedPair}(?\mathrm{x}, ?\mathrm{pair1}) \wedge \mathrm{Stair}(?\mathrm{pair1})\\ &\quad \wedge \mathrm{hasHVConnectedPair}(?\mathrm{x}, ?\mathrm{pair2}) \wedge \mathrm{Stair}(?\mathrm{pair2})\\ &\quad \wedge \mathrm{GeometricPrimitiveSequence}(?\mathrm{x})\end{aligned}$$

4 Results

To show the potential of our approach we exhibit three different simulations in which we query the knowledge system for different building objects. Our approach is also viable for different geometries, in particular after extending the method to curved spaces by adding properties and rules accordingly.

Our simulations concern half of an empty room, a staircase and a bookshelf. For each scenario, a set of SWRL rules was designed that allows for labeling of intermediate object parts such as a ceiling, shelf planes or stairs, as well as labeling of the entire object of interest. Labels correspond to object categories. We map the segmentation output to the ontology via a mapping language, and obtain asserted instances. By running the reasoner, we further label the segments, and create inferred instances. For the three examples, the results are shown in Table 1. Not all mapped segment get a labeling, which is due to the challenges described next.

Table 1. Asserted and inferred segment labels

point cloud	#mapped segments	#labeled segments
empty room	9	5
bookshelf	18	18
staircase	17	5

5 Challenges

Several challenges were encountered during the implementation of the presented object classification method. More precisely, we coped with:

Thresholds. As the description logic reasoner uses crisp logic, we had to set hard thresholds for property predicates, e.g. $\mathrm{swrlb} : \mathrm{greaterLess}(?\mathrm{maxY}, 1.5)$. Finding these constants was done manually and it was time-consuming.

Coordinate frames. In our experiments the constants refer to the scanner own coordinate system and we used single scans. It is an open question how this extends to an arbitrary (project or robot) coordinate system or even to global coordinates, i.e., to include georeferencing.

Missing data. We experienced that mapped segments are not labeled due to missing data. The laser scanner gages only objects visible. However, multiple 3D scans and scan registration are necessary to completely digitalize scenes.

Efficiency for multi-values predicates. For extracting relations between individuals they have to be compared. Currently, we perform this comparison while processing the point cloud in C++, exploiting spatial data structures such as k-d trees.

Memory efficiency. Due to the presence of many segments in realistically sized real-world scenes, Pellet reasoner tends to run out of memory due to the complexity of the used description logic.

Designing the data processing tool chain. It is not clear, which parts of the interpretation process should be implemented at the point cloud processing level, i.e., in the C/C++ part that acquires the sensor data, calculates the normals and performs the segmentation, and which parts should be performed by description logic reasoning in the knowledge-based system. The question is, when and where to call Pellet and the used ontology.

6 Conclusion

We presented a framework for semantic interpretation of point clouds which takes advantage of Semantic Web technologies. Built on the platform of Protégé-OWL, our alternative method of linking top level semantic qualification with low level geometric calculations uses a connectivity-preserving segmentation algorithm, an ontology structure and a reasoner. We believe that the logical structure of an ontology is suitable for semantic knowledge representation and that under the Semantic Web framework, Web Ontology Language is appropriate for defining spatial knowledge. Such an approach provides a better understanding of a 3D scene, by facilitating detection and recognition in 3D point clouds.

Needless to say, a lot of work remains to be done. To avoid the use of crisp thresholds, we plan to add fuzziness to the system and/or use probabilistic reasoning. A promising approach is given by Pu and Vosselmann in [5]. They use semantic building knowledge to reconstruct a polyhedron model of outdoor terrestrial 3D scans. They also describe the uncertainty and make expected decisions [6]. Further future work will aim at interpreting multiple registered 3D scans. As our system relies on plane segmentation, this extension seams straightforward. However, a combination with next-best-view planning is highly desirable.

References

1. Sima, M.C., Nüchter, A.: An extension of the Felzenszwalb-Huttenlocher segmentation to 3D point clouds. In: International Conference on Machine Vision (2012)
2. Duan, Y., Cruz, C., Nicolle, C.: Architectural reconstruction of 3D building objects through semantic knowledge management. In: 11th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (2010)
3. Hmida, H., Cristophe, C., Frank, B., Christophe, N.: Knowledge Base Approach for 3D Objects Detection in Point Clouds Using 3D Processing and Specialists Knowledge. International Journal On Advances in Intelligent Systems, vol. 5, pp. 114 (2012)
4. Günther, M., Wiemann, T., Albrecht, S., Hertzberg, J.: Model-based object recognition from 3D laser data. KI 2011: Advances in Artificial Intelligence, Springer (LNAI 7006), pp. 99-110 (2011)
5. Pu, S., Vosselman, G.: Knowledge based reconstruction of building models from terrestrial laser scanning data. {ISPRS} Journal of Photogrammetry and Remote Sensing **64**(6), 575 – 584 (2009)
6. Pu, S.: Knowledge based building facade reconstrcution from laser point clouds and images. PhD thesis, University of Twente (2010)