ON AN ONTOLOGY OF SPATIAL RELATIONS TO DISCOVER OBJECT CONCEPTS IN IMAGES

Matthias J. Schlemmer, Markus Vincze

Vienna University of Technology
Faculty of Electrical Engineering and Information Technology
Institute of Automation and Control
1040 Vienna, Gusshausstrasse 27-29, Austria

schlemmer@acin.tuwien.ac.at (Matthias Schlemmer)

Abstract

This work tackles the problem of object concept detection in the domain of robot vision. Autonomous robots are supposed to navigate in unknown environments, facing objects that have appearances the robot has never seen before. As a concept of an object we define the collection of necessary properties (philosophically speaking, the thing-in-itself), which is often *not* (only) its appearance but could be any combination of properties (e.g. shape, function, etc.). As a first implementation, we therefore use only a qualitative description of the spatial relations of the parts that make up an object concept. This qualitative information is stored in an ontology that is used for checking relations of object parts found in the robot's camera images. An additional technique used in this approach is abstraction: Abstracting several connected parts to a bigger one leads to a reduction of the overall number of participating objects. Therefore, the specific instance of the object concept found in the image is reduced to its defining high-level shape, which in turn complies to the definition stored in the ontology. One of the tackled questions is therefore how specific the object concept is to be defined in the knowledge repository. We show that with this approach we can discover a column or an arch that is made up of an arbitrary number of parts.

Keywords: Object Concept Detection, Part Ontology, Perceptual Grouping, Robot Vision, Abstraction

Presenting Author's Biography

Matthias Schlemmer. Born in 1979 in Vienna, Matthias Schlemmer studied electrical engineering at the Vienna University of Technology with focus on computer science. Since 2002 he is also studying philosophy at the University of Vienna. After receiving his master's degree in electrical engineering in 2004, he started working on his PhD at the Institute of Automation and Control as a project assistant. His research interests comprise robotics and computer vision as well as related cognitive and philosophical aspects.

ISBN 978-3-901608-32-2

1 Motivation

A burning issue in robot vision is the representation and detection of objects including their structural and task-related properties. Imagine a home robot whose current task for whatever reason is to detect the chairs in a room. Of course, it is wishful that the robot does not only detect those chairs it has previously been trained to detect but every object in the room that fulfils the function of being a chair. Now, the main question that arises is, how do we define or represent something like a chair? A famous example would be the definition by Stark and Boyer [1] who thin out every aspect of a chair to the functional description of having a sit-able surface along with stable support. Further classification, if needed, could comprise the check for having back or arm support, as [1] actually do.

However, the problem is the detection of parts of objects or objects. One classical approach to re-cognise a part or object is learning-from-prototype, which is a shortcut from pixels to objects but suffers from the problem that it is not possible to arrive at generic object recognition or to detect generic properties or parts. Another frequently used possibility is the use of models, i.e. shape-cues, to re-cognise an object. Again, due to the difficulty to detect these shape cues in images directly, often they are broken, superfluous or missing. Yet another possibility is perceptual grouping: finding low-level features and organising them into Gestalts, eventually ending at a proto-object level, e.g. grouped lines forming a closed contour. Here, our approach steps in: out of these grouped proto-objects, concepts of objects are to be detected, easing the bridging to the semantic object level.

Having a robot in mind that tries to gain knowledge about its environment, our approach tries not to redetect – already perceived – instances of objects but rather to group features to a more general object concept. We start with having a rather simple task such as finding something in a room while keeping in mind to extend this approach to a wider variety and complexity of object concepts and application fields.

Questions that are tackled in this paper concern the notion of objects and the role of their parts: what parts are a necessary condition for the object and which are supplemental. For example, a column can be built up or detected in an image of one homogeneous solid or can be made up of several parts stacked on top of each other. Hence we need to find an approach that tackles this issue, correlated with the question of abstraction. On the other hand, a column can be made of stone, steel, wood or any other material – hence, in this case, the appearance of the object (its colour) is probably no defining criterion.

Arguing from a philosophical point of view, we are trying to model a sort of thing-in-itself. The discussion about what constitutes an object is practically as old as philosophy itself. Starting from Plato's *ideas* that – according to him – we have seen in heaven and that we can recall when encountered with a specific (unideal) instance, to Aristotle who introduced his well-known categories, culminating in Immanuel Kant's distinction between the thing-in-itself, which is inherently not ascertainable, and the thing-for-us. Our (subjective) reality is made up of appearances that come from but are not the same as the things-in-theirselves. The latter only affect our senses. Nicolai Hartmanns *critical realism* purges the Kantian position even more from metaphysical touches. In his opinion, we are actually able to perceive important traits of the thing-in-itself and not only the thing-for-us. For him, this is the reason for us being able to talk about things, that we did not actually perceive. The object concept that we would like to suggest in this paper as foundational object representation for a robot is the thing-in-itself in the Hartmann sense.

1.1 Related Work

There is a lot of research in appearance-based recognition of objects, e.g. [2] which is a recent paper dealing with learning-from-prototype. Classical works on model-based recognition are [3] and [4]. A Gestalt-based perceptual grouping framework which aims at proto-object detection can be found in [5].

[6] introduces a hierarchy of primitives that is ordered into structure-type, solid-type, face-type and pair-type. For 3D, Biederman [7] segments images into regions of deep concavities in order to arrange simple geometric components to objects. In [8], for example, vehicles of a special kind need to be classified (SUVs, Taxis, etc.). In this paper, it is underlined that the „[...] extraction of features for recognition cannot be separated from the act of recognition itself.“.

Ontologies start being used in computer vision (beside the typical application field of the semantic web) in the last few years. [9] uses the large semantic ontology *WordNet* for detecting boundaries between objects leading to a segmentation of the input image into different classes of objects. [10] uses an ontology for object categorization, combining machine learning and knowledge representation techniques. The introduction of a visual concept ontology bridges between domain knowledge and the image processing.

2 Contribution

The contribution of this paper is to separate properties that are found to be characteristic for objects (the object concept as we call it – the abstract or generic idea generalised from particular instances) from the specific instance of the object in the image. The expectation is that then we can also detect the object more easily in the image, if it appears with changes to its shape or appearance but still constitutes the same class of object.

The difficulty is to find the ridge between getting too specific or too general when defining an object through characteristic properties. As a first approach, we start with a basal description of the spatial relations between their different parts. It leads to the usage of a naive physics approach in which, for example, the support of an object is defined as lying stable on top of another object. Certainly, further work needs to implement more kinds of relations.

The usage of an ontology as repository of object characteristics for such a system appears to be an elegant means. This is due to the fact that „An ontology is an explicit specification of a conceptualization“ as the popular definition of [11] states. The emphasis lies on *explicit*, stating that in an ontology we model the knowledge of our domain (i.e. the world our robot acts in) clearly in natural language, making all assumptions visible. Additionally, the well-known problem of symbol-grounding can be shortcut elegantly, too. Pure feature extraction alone might deliver us a rectangle without any information to what it may belong. Grouping these features guided by an ontology allows for usage of semantic knowledge stored along with the description of the object concept in the database. This way, the concept of, say, a chair immediately also helps us in retrieving its function.

An important advantage is that through the decoupling of the current hypotheses found in images and the fixed knowledge, the expansion of the ontology is straightforward. New concepts can be included fast by users, without having to pre-plan all eventualities in the vision algorithms. This is backed by the strength of an ontology language that allows describing facts in natural language.

Last but not least, this approach enables a step-wise abstraction of the objects and parts in the world around us. Starting with the identification of some part, we might come up with a bigger object to which this part belongs and that finally might again be part of an even larger object. This should also accelerate the whole computation – dealing with grouped objects instead of all subparts highly reduces the search space. Especially in the domain of robots acting in cluttered environment, pixel data is far too complex to be processed in reasonable time. Therefore, reducing this data to higher-level objects should accelerate computation and decision making.

3 Implementation

The implementation of ontology and image processing is shown in the following with the help of a fundamental example: The detection of an arch in an image. Fig. 1 shows a simple case: In this blockworld scenario, the arch consists of two pillars and one cross-bar. The pillars as well as the cross-bar are made up of one solid piece. In Fig. 2 the result of the detection is shown.

3.1 Ontology

As mentioned above, the first preliminary implementation accounts for spatial arrangements of simple objects, namely polygons in 2D. A vital consideration when designing an ontology for automated interaction with image feature extraction is to somehow cover the span between human readability (which is a big advantage of ontologies) and good automated handling possibilities. A number of different ontology languages can be found, each with specific advantages and disadvantages. Our choice is to use OWL (Web Ontology Language) [12] as recommended by the W3C. It extends RDF (Resource Description Framework) [13] and is therefore based on triples, consisting of a subject, a predicate and an object. Relations between different classes is done via *Restrictions* that assert specific necessary (and partly sufficient) conditions to classes, in our case: object concepts that are to be detected. As an editor to model this information, we use Protégé, developed by Stanford University [14]. For the handling of the ontology, *Jena* [15], a semantic web framework, is used that is capable of instantiating and querying a given OWL ontology.

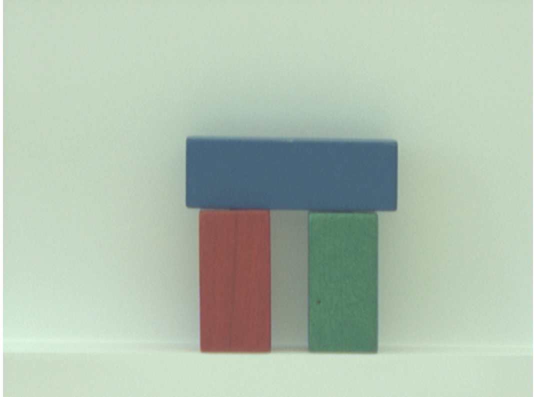

Fig. 1 A simple blockworld example: an arch

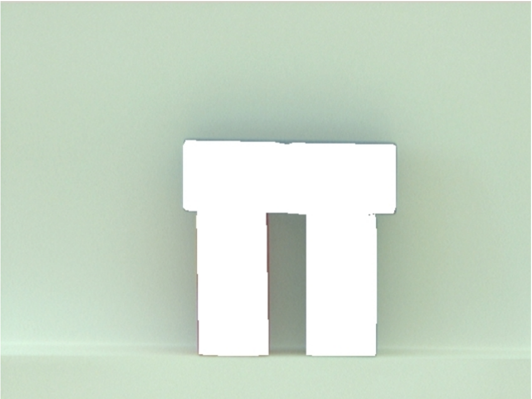

Fig. 2 The arch of Fig. 1 found by the system.

For a start, our ontology holds the classes *Column* and *Arch* as known object concepts as well as the definition of the following spatial relations: left_of, right_of, on_top_of and below_of. The latter two additionally have subproperties defining whether the object on top is stably supported by the subjacent object (in a naive physics way). Of course, this is very basal and needs to be refined in future work. A column, for example, is currently defined as being made up of two polygons one resting on the other. Due to the stepwise abstraction technique, we are, of course, also able to find a column made up of more parts as each step connects two polygons to one, new larger object – which will be shown below.

The class *Column* uses a property *hasParticipatingObjects*: „hasParticipatingObjects exactly 2“. This accounts for the fact that with the definition of a column mentioned above, we need to detect two objects. As we want them to be specifically arranged, we need to assert further restrictions. In this case, this would be that one object needs to rest stably on the other. This way, the concepts *Column* and *Arch* were defined which are used in the next step to find the arch in the actual image.

It needs to be mentioned that it is difficult to describe relations between different classes (object concepts) in OWL. This would imply a type of restriction applied to a class describing the relation between two or more other classes. This is not defined in OWL, which is due to the fact that only triples are defined. However, the constraint of only allowing triples has an important advantage: it allows the usage of a description logic reasoner in order to infer knowledge.

3.2 Combining Vision and Ontology

The vision part of the framework is liable for extracting features from images, which is currently done using the perceptual grouping framework *vs2*, developed by [5]. This results in polygons found in the image.

Depending on the given situation, we have the following possibilities of using the ontology along with the current image. Either we want to know whether any known object structures are in the image or we have a specific task, e.g. „Find an arch!“. In the first case, we compute the relations of the objects in the image and then look for corresponding configurations in the ontology. The latter case comprises the following steps:

Given the task of finding an *Arch*, a query on the ontology first tells us that we need three objects. The specific arrangement (two being the support of one object on top) is the result of the next query and this is checked for compliance in the vision unit. Finally, if all restrictions are met, we know that there is an arch, see Fig. 2, where the objects of Fig. 1 belonging to the concept are marked in white.

Let’s get a bit more complex! One of the main ideas of our work is the gradual abstraction of the features in the image in order to actually find objects that form a concept without having to know how much parts contribute. As an example, Fig. 3 shows an arch where the right pillar is itself composed of two objects.

A direct query for location of an arch without previous simpler abstraction steps, leads to a finding like the one in Fig. 4. This is due to the definition of an arch consisting of three objects, two supporting one.

If we, however, first locate the column (see Fig. 5), we are able to use this – abstracted – knowledge as new object for later computations. For example, this information can be used for a query on *Arch* now. As shown in Fig. 6, now again three objects are used, but one having been abstracted earlier and providing therefore a better (abstracted) solution.

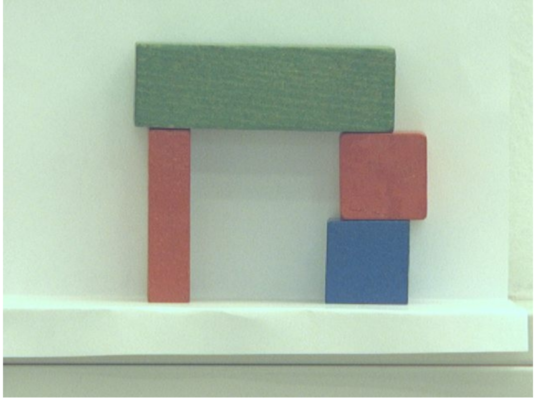

Fig. 3 A more exciting blockworld: again, an arch consisting of two pillars, but with one of them now composed of two objects.

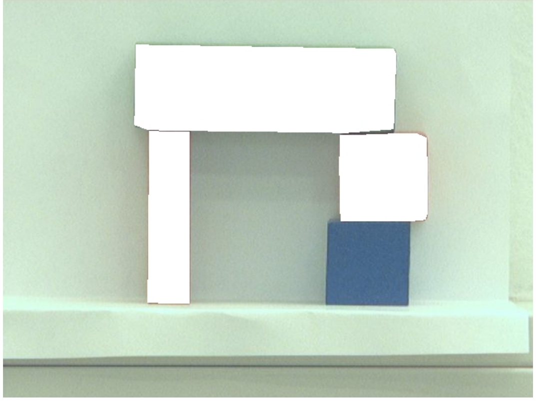

Fig. 4 Result of a direct query for finding an arch on the input image in Fig. 3. The restrictions are fulfilled with the set of objects marked in white. The lower block on the right side is not considered.

By using this kind of interaction between explicitly defined object concepts and hard-coded image analysis, an arbitrary object may be defined in the ontology and as long as the properties (which is for the case of spatial relations and naive physics a finite number) are implemented in the vision algorithms, we can find its occurrence in the camera image. With other words, we define *qualitative relations* between objects in the ontology, but we retrieve a *quantitative description* of their existence in the current view without having a representation of the object concept (column, arch) in the vision part. With this combination we discover object classes such as column or arch independent of how many pieces contribute.

4 Conclusion

In this paper we presented an approach for combining world knowledge in an ontology with image analysis. Object concepts are defined using their necessary conditions only and not their exemplary appearances. Images are first pre-processed by a perceptual grouping tool the output of which is used for checking relations defined in the ontology. Currently spa-

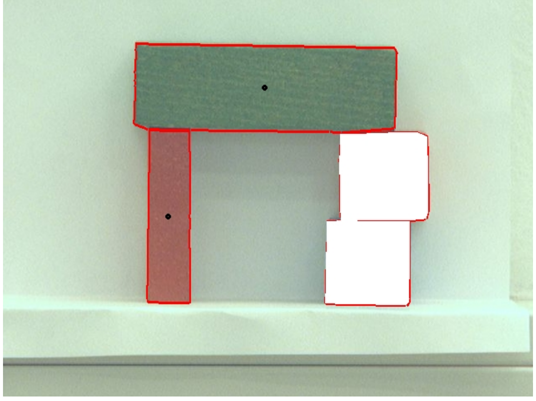

Fig. 5 Result of query for finding a column in the image of Fig. 3.

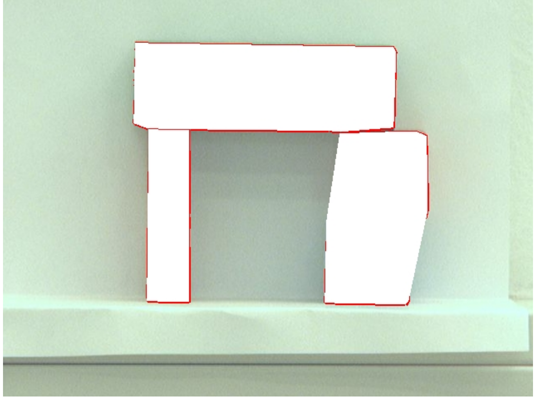

Fig. 6 Result of a query for finding an arch of Fig. 3. The information from Fig. 5 is used, the arch is therefore of higher abstraction than in Fig. 4.

tial relations make up the restrictions that define those necessary conditions, the expansion of the ontology is relatively easy. Important is the possibility to perform abstraction: Using already found object concepts helps finding higher-level concepts.

4.1 Further work

A hard issue is to extract the necessary conditions for building up the ontology. As in some cases (e.g. a traffic light) spatial relations might not suffice, the ontology has to be extended to account for other types of restrictions as well (e.g. colour, texture,...). Furthermore, a lot more object concepts need to be implemented to widen possible test cases.

Another problem that will have to be tackled is to decide which higher-level concepts found should be stored and which underlying polygons shall be kept or rejected. Fig. 7 shows a last example that should clarify this point. As can be seen, again, an arch is shown, this time built up of a lot more subparts. Fig. 8 shows the optimal detection after intermediate abstraction steps for retrieving the whole columns first. Fig. 9, however, also depicts a possible retrieval of an arch, which is one of the intermediate steps. In this case, it would be easy to argue that the columns should be further expanded until the result of Fig. 8 is reached and that the result shown in Fig. 9 is not optimal. However, there are cases where either the lower parts are not found by the segmentation tool or where this lower part might itself take part in some other object concept (say, the basement

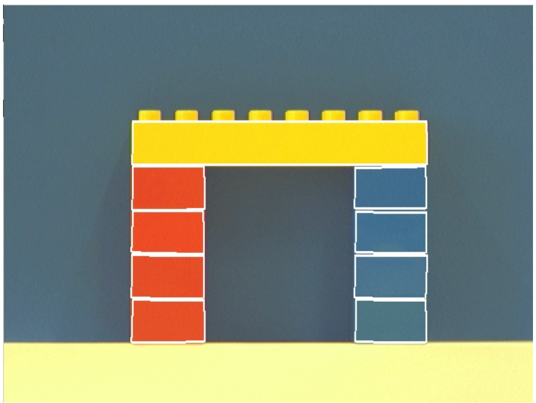

Fig. 7 Another arch example. In this case, there are a lot more parts per column (The single blocks are framed in white in order to provide better visualisation).

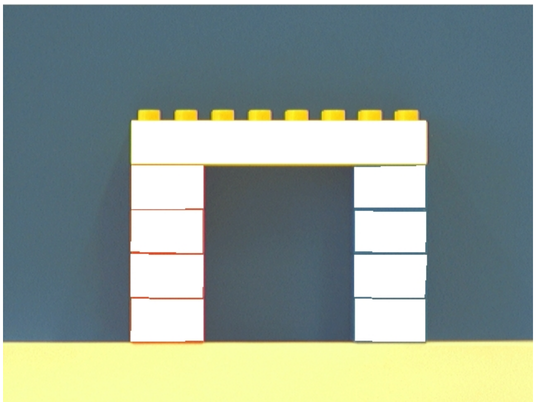

Fig. 8 Optimal (highly abstracted) detection of the arch of Fig. 7.

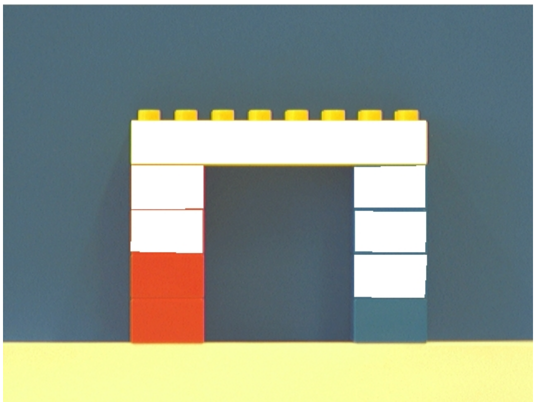

Fig. 9 One found arch of Fig. 7. As can be seen, the result is not optimal. This could occur if the participating columns are wrongly chosen or detected.

of the arch which shall for some reason be detected as well). If we just keep any information, computation time gets high due to combinatorial explosion. Further work will have to intelligently prune the search space.

Additionally, this shows another possible extension. If some data is missing in order to fulfil a hypothesis about an object concept, say, one of the blocks in Fig. 7, the system should be able to provide feedback to the segmentation algorithm. As it is likely that at the gap a block is present, this feedback might lead to a detection by the segmentation algorithm with different parameters, tuned by this high-level feedback.

5 Acknowledgements

The work described in this article has been funded by the European Commission's Sixth Framework Programme under contract no. 029427 as part of the Specific Targeted Research Project XPERO (Robotic Learning by Experimentation) as well as by the Austrian Science Foundation under the grant # S9101 (Cognitive Vision).

6 References

[1] L. Stark and K. Bowyer. Achieving Generalized Object Recognition through Reasoning about Association of Function to Structure. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 13, pages 1097–1104, Oct. 1991.

[2] S. Agarwal, A. Awan, and D. Roth. Learning to Detect Objects in Images via a Sparse, Part-Based Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1475–1490, Nov. 2004.

[3] D.G. Lowe. Three-Dimensional Object Recognition from Single Two-Dimensional Images. *Artificial Intelligence*, 31(3):355–395, 1987.

[4] F. Solina and R. Bajcsy. Recovery of Parametric Models from Range Images: The Case for Superquadrics with Global Deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(2):131–147, 1990.

[5] Michael Zillich. *Making Sense of Images: Parameter-Free Perceptual Grouping*. PhD thesis, Vienna University of Technology, 2007.

[6] F. Tomita and M. Koizumi. A Step Toward Generic Object Recognition. In *Proceedings of the 11th IAPR International Conference on Pattern Recognition*, volume Vol. 1: Computer Vision and Applications, pages 632–636, 1992.

[7] I. Biederman. Recognition-by-Components: A Theory of Human Image Understanding. *Psychological Review*, 94(2):115–147, 1987.

[8] J.L. Mundy. Generic Object Recognizer Design. In *Proceedings of the Computer Vision for Interactive and Intelligent Environment (CVIIE 05)*, pages 135–144, 2005.

[9] Anthony Hoogs and Roderic Collins. Object Boundary Detection in Images using a Semantic Ontology. In *Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW 06)*, 2006.

[10] Nicolas Maillot, Monique Thonnat, and Céline Hudelot. Ontology Based Object Learning and Recognition: Application to Image Retrieval. In *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2004)*, pages 620–625, 2004.

[11] Tom Gruber. What is an Ontology? online at *http://www-ksl.stanford.edu/kst/what-is-an-ontology.html*, May 2007.

[12] World Wide Web Consortium (W3C). OWL Web Ontology Language Overview. online at *http://www.w3.org/TR/owl-features/*, June 2007.

[13] World Wide Web Consortium (W3C). Resource Description Framework (RDF). online at *http://www.w3.org/RDF/*, June 2007.

[14] Stanford Medical Informatics at the Stanford University School of Medicine. Protégé – open source ontology editor. online available at *http://protege.stanford.edu/*, June 2007.

[15] Semantic Web Framework *Jena*. online available at *http://jena.sourceforge.net/*, June 2007.