OBOC: Ontology Based Object Categorisation for Robots

Rogan Mendoza[1], Benjamin Johnston[1*], Fangkai Yang[2],
Zhisheng Huang[3], Xiaoping Chen[2], Mary-Anne Williams[1]

[1]University of Technology, Sydney, Australia
[2]University of Science and Technology of China, Hefei, China
[3]Vrije University of Amsterdam, The Netherlands

[*]Contact: johnston@it.uts.edu.au

Abstract

Meaningfully managing the relationship between representations and the entities they represent remains a challenge in robotics known as grounding. Useful insights can be found by approaching robotic systems development specifically with the grounding and symbol grounding problem in mind. In particular, Semantic Web technologies turn out to be not merely applicable to web-based software agents, but can also provide a powerful extension to existing proposals for grounded robotic systems development. Given the interoperability and openness of the Semantic Web, such technologies can increase the ability for a robot to introspect, communicate and be inspected - benefits that ultimately lead to more grounded systems with open-ended intelligent behaviour.

Keywords: grounding, symbol grounding, ontologies, robotics

1 Introduction

The symbol grounding problem [4,10, 28,30] is a prevalent topic in Artificial Intelligence relevant to problems in knowledge representation, psychology and robotics. The problem is that of making the interpretation of a formal symbolic system intrinsic to the system itself, rather than depending on the interpretation in the minds of the human designers of the system [9].

The symbol grounding problem is related to the more recent and general concept of groundedness. Groundedness is the degree to which the entities of a system's representations correspond *meaningfully* to the entities that they represent [28]. Grounding and groundedness is a multidimensional and graded property of systems: different representations and systems can afford many different degrees of expressiveness, relevance, faithfulness, correctness, accuracy/precision, robustness, adaptability, timeliness, efficiency, self-awareness, awareness of others, functionality, transparency, testability and uncertainty management. Given these many dimensions, there is often no clear answer as to whether the entities in one system are definitively more meaningfully connected to reality than another, however it is possible to draw conclusions and make useful comparisons about these individual characteristics of a system's groundedness.

One view of the development of robotic systems is, as a process, principally concerned with grounding: creating meaningful representations, maintaining those representations and acting upon them in an appropriate and timely manner. In theory, robots need not make use of any form of symbolic representation—they could, for example, be purely reactive—however, many practical robots in use and under development today make some use of symbolic representations. While these symbols may be 'second-class' entities that have been created on an ad-hoc basis and may be difficult to inspect, they are nevertheless abstract entities intended to refer to external entities or concepts. Groundedness with respect to these symbols is then the problem of how the symbols meaningfully correspond to real-life entities and concepts: the symbol grounding problem.

If the symbols in robotic systems are promoted to 'first-class' entities with independent existence and meaning, it then becomes possible to relate these symbols to concepts in a formal ontology. Doing so can enrich both the robotic system and the ontology. A robotic system enriched with a formal ontology is more grounded—the ontology can allow for greater adaptability, self-awareness, awareness of others, functionality, transparency and testability. Conversely, an ontology enriched with concepts of a robotic system has symbols whose meaning have been ground to reality via the robot: the robotic system fixes the meaning of the ontology, helps identify problematic constructions and can even be used as a reasoning mechanism in itself (a difficult problem may, in fact, be best solved by experimentation or exploration in the real world).

In this paper we introduce OBOC (Ontology Based Object Categorization), a combination of Semantic Web technologies and robotic systems that has been implemented on a Sony four-legged AIBO robot. While OBOC is narrowly focussed on the problem of recognition of objects and communication, related work by Vogt [27] and by Gärdenfors and Williams [8] has shown that categorisation is an essential process in understanding and constructing grounding capabilities. Categorisation is an important capability for robots because it allows them to deal with and efficiently store, reason with and communicate complex information captured by their internal and external senses. In OBOC, Semantic Web technologies enable categorization, communication and reasoning by providing standard protocols and languages for defining and sharing ontologies (using the Ontology Web Language, OWL). The result, a system combining the robot's physical and sensory capabilities with high-performance reasoning capabilities of the Racer inference engine, is a vast improvement over closed robotic systems that are unable to rapidly adapt to novel situations. And while our experiments with OBOC have, to date, focused on categorization and communication; there is no reason the framework would be unsuitable for extensions such as sharing actions and intentions.

2 Grounding, Symbol Grounding, Anchoring and Categorization

Grounding concerns managing the relationship between representations and the entities they represent in a *meaningful* fashion. This relationship is important because it affects the way an intelligent system can potentially behave, how it can interact with its environment, and what it is capable of achieving [28]. Related to grounding are sub-problems of symbol grounding, anchoring and categorization. Conceptualizing these sub-problems as restrictions on the general problem of grounding creates a clearer understanding of the issues.

It is important to realize that grounding or groundedness is a multidimensional process or capability: it is not possible to merely claim that a system's representations *meaningfully* correspond to external entities, for the definition of *meaningful* is a complex and loaded term. By conceptualizing grounding as a multidimensional and graded property, we can analyse different capabilities of systems and thereby compare different systems even though it may not be meaningful to claim that a system is objectively grounded or that one system is objectively more grounded than another entirely different system. Williams *et al.* [28] offer a framework for analysing groundedness; in their framework, the groundedness of a system is analysed in terms of a set of gradings along multiple dimensions. While they make no claims that their analysis is exhaustive, they offer sixteen dimensions: expressiveness, relevance, faithfulness, correctness, accuracy/precision, robustness, adaptability, timeliness, efficiency, self-awareness, awareness of others, functionality, transparency, testability and uncertainty management.

The Symbol Grounding Problem [9] is, in fact, a restriction on the general grounding problem (or conversely, grounding is a generalization of symbol grounding). Where grounding is concerned with the relationship between a system's representations and reality, the symbol grounding problem is concerned only with those systems whose high-level representations form a *symbolic system*. A symbolic system is a set of arbitrary tokens that are manipulated on the basis of explicit rules (that are also defined in terms of arbitrary tokens), and subject to a semantic interpretation. Intuitively, a symbolic system may be seen as systems that are implemented using high-level programming languages (C, C++, Java, Python, Lisp, etc.) or that resemble the logic-based approaches of "Good Old Fashioned AI" (GOFAI).

In fact, symbol grounding relates to work that dates as far back as the ancient Greeks in their search for "true reality" and their study of metaphysics. More recently, prominent work by Kent [10], Searle [22] and Harnad [9] have raised the challenge in the context of AI. Searle [22] introduced the well known Chinese Room thought-experiment. Searle uses the Chinese Room to argue that computer programs are syntactic and lack the semantics that allow it to *understand*. Since no amount of syntax will ever produce semantics, Searle concluded that a purely symbolic system will never be able to understand what it is doing because of the lack of intentionality i.e., the inability to link internal representations to external objects or states [30]. Harnad [9] later identified a core challenge of the argument, and posed the question of how 'the semantic interpretation of a formal symbol system be made intrinsic to the system, rather than just parasitic on the meanings in our heads?' In other words, how can arbitrary symbols be grounded and given meaning? This is the challenge underlying the symbol grounding problem and the more general problem of grounding representations.

Harnad's [9] own response to the symbol grounding problem was to propose combining connectionism (sub-symbolic) methods with symbolism (symbolic systems) into a hybrid model. Harnad uses connectionism as a way of representing icons and categories—internal analog transformations of sensory data—that 'pick out' or 'distinguish' concepts. He sees the relationship of icons and categories to sensory input as beyond the need for semantic interpretation—the icons and categories are merely causal responses to sensations. These can then be fixed to the elementary symbols of a symbolic system: when one has a sufficient set of elementary symbols grounded via an iconic connectionist network, then the rest of the concepts and symbols in a complex language can be generated by symbol composition alone. This, Harnad argues, is a solution to the symbol grounding problem.

If we, however, view symbol grounding as a multidimensional property then one recognizes that the problem isn't merely a matter of finding a 'solution'. Different systems have different degrees of groundedness—a more grounded system might have the capability to introspect upon the relationships between its icons and sensations, or it might be based on the idea of anchoring meanings to external objects rather than sensations as in work by Vogt [27]. Furthermore, different choices of elementary symbols can result in systems that are more expressive, robust, adaptable, transparent and testable, and therefore more grounded. For example, a system that can represent the categorization of a 'soccer ball' in terms of its shape, texture and purpose would be more grounded than other system that treats 'soccer ball' as an elementary symbol.

3 Symbolic Systems, Ontologies and the Semantic Web

Many 'intelligent' systems today incorporate some form of symbolic system—even if those symbols and rules are tightly embedded as the atoms and constructs of the mainstream programming languages used to implement the system. For example, a soccer playing robot might use a set of tokens to represent the position of a ball: it may have a set of rules hard-coded as C++ functions to predict the path of the ball, and move to a strategic position. In seeking to create more grounded systems, we recognise however, that such hard-coded rules restrict our ability to adapt such systems to novel situations. By exposing the symbols of a system as first order constructs in and of themselves (rather than entities in the software engineer's mind) it is possible to create more grounded systems.

Formal ontologies, particularly ontologies described using expressive formal languages, is one approach to constructing symbolic systems. Such ontologies are logical structures intended to present and independent model of reality—each symbol is defined for consistent semantics, robust reasoning and communication.

In recent years, Semantic Web efforts have resulted in the development of standard languages for expressing, reasoning with and communicating ontologies. The Web Ontology Language (OWL) is an established standard with widely available and efficient reasoners. By integrating (or implementing) the high-level symbolic systems of a robot with Semantic Web technologies, it becomes possible to make use of widely available tools, communication protocols, reasoners and best practices, thereby enhancing and improving the groundedness of the robotic system.

The use of ontologies in robots is novel [8] and few, if any, implementations and evaluations have been reported. Perhaps the most related research is Schlenoff [19] who described the use of an ontology of obstacles to aid in path planning and obstacle avoidance. Other applications of ontologies have primarily been demonstrated in systems that are used for image and document classification [3,21,24], communication and object mapping in mobile robots [11,27] and also object learning [14].

4 Object Recognition in Robot Soccer

RoboCup is an international robot soccer initiative designed to advance the field of Robotics. It involves an annual competition involving soccer, rescue and household robotic systems where international teams of autonomous mobile robots compete against each other under different conditions.

The robot soccer domain is a complex and dynamic environment. In the 4-Legged League, Sony AIBO robots fight for possession of an orange ball in order to pass it to team-mates or shoot for goal. The playing field is 5.4m by 3.6m, made of green felt, and defined using white boundary lines and colour coded beacons. The four robots on each team are physically identical: principally interacting via their single camera, single chest-mounted range-finding sensor, joint actuators, joint-feedback sensors and wireless LAN (for robot-to-robot communication). The robots independently and autonomously process their motor-sensory data in order to construct a model of the current state of the field. A robot must be able to recognise the goals, the ball, other players, to reason about their position, and to select appropriate responses. This domain demands not merely on competent ball skills, but the fusion of sub-symbolic information into symbolic representations that guide strategy and that are shared with other robots to coordinate play.

We have targeted RoboCup as the domain to explore our ideas on object categorisation through ontologies and grounding because of the mixed demands for sophisticated object recognition, prediction, planning and communication in a dynamic environment. The UTS Unleashed! Robot Soccer System that competed in the Four-Legged RoboCup Competition in 2003 and 2004 (and has since been used for non-competitive research) was selected as the platform for development as it allowed us to focus entirely on the design and implementation of OBOC without having to redevelop the extensive infrastructure required for fundamentals such as locomotion and vision.

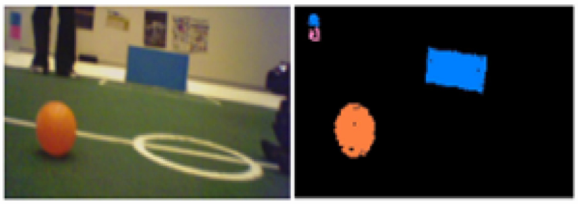

Figure 1: Raw camera image and extracted objects in the raw image

The RoboCup environment has been largely crafted to assist with the AIBO's sensory capabilities. Standardized colours and lighting conditions allow for objects to be identified using straightforward colour segmentation techniques. For example, the ball is usually perceived as a large, round area of orange colour (although this can be complicated by occluding objects and shadows). The robots ground their colour concepts to sensory stimuli derived from their cameras. Colour concepts effectively form icons and of the physical objects; the colour segmentation capabilities of the robot effectively form the causal link and elementary concepts of Harnad's [9] approach to symbol grounding. These elementary concepts can be used to build knowledge: combined into predicates that are used to describe the world and its behaviour within the robots knowledge base.

5 Applying OWL to RoboCup

The use of expressive ontologies in RoboCup allows for both feature-based and context-based categorisation. Furthermore, ontologies enable these concepts and categorisations to be shared with other robots. These are discussed in further detail below.

5.1 Feature Based Categorisation

Using OWL properties, we can define abstract concepts (such as the ball) from the elementarily grounded concepts. For example, a concept corresponding to **Ball** can have the necessary property **hasColor** constrained to the concept **Orange**. One can furthermore use OWL to define both sufficiency conditions: all objects with **hasColor Orange** are instances of **Ball**.

In practice, three factors govern the robot's performance with feature-based categorization:

1. The quality of the elementary symbols for describing terms in the ontology and distinguishing objects in the real world (can we distinguish colours, shapes, movements?)
2. The quality of the ontology in defining objects correctly and in sufficient detail
3. The scope of the robot's application and the similarity of objects that need to be distinguished

For example, no teams in the RoboCup 4-legged league would currently be able to distinguish an orange soccer ball from a piece of ripe orange fruit: given the elementary grounding of these systems, there is no attribute that can be used to distinguish them. In practice, these recognition problems can have a significant impact on play—stray orange objects in the background can sometimes be identified as balls.

We cannot, however, expect to be able to solve these problems completely. Even as human beings, our highly sophisticated object recognition system is perceptually limited and suffers from a range of mistakes including optical illusions, hallucinations and other phenomena such as change blindness. Some uncertainty in an ontology (with respect to the elementary grounding) is inevitable—we must accept this fact and handle it gracefully.

5.2 Context Based Categorisation

Often contextual information is able to distinguish similar objects. For example, a round orange object in a fruit bowl is likely to be an orange or mandarin, whereas a round orange object on a RoboCup field is more likely to be a ball. In both cases the percept of round orange thing is similar, yet the classification is very different. An OWL ontology can be used to represent such contextual information. For example, we can express contextual information about the goal box in the following way:

$$\mathbf{GoalBox} \sqsubseteq \exists_1\mathbf{isBehindOf}.\mathbf{GoalKeeper} \sqcap \exists_1\mathbf{isNear}.\mathbf{OwnBeacon}$$

Using this technique, a robot would be able to use the ontology to determine the relationships between recognised objects and be able to successfully categorise them based on context. This is similar to how some systems use ontologies for image classification [3,21] and also for the use of Relational Object Maps [11].

5.3 Concept Learning and Sharing

OWL has been specifically designed for inter-operability among semantic web systems, and so is ideal for allowing robots to communicate. Ontologies allow robots to share knowledge about a single recognised object, even if they have different categorization capabilities. Ontologies not only allow two concepts to stand for the same object in respect to different groundings [25], but for the defined semantics and relationships to be compared, integrated and fused.

Consider the following example where Robot 1 fuses information from Robot 2:

Robot 1 (Initial):

$$\mathbf{Ball} \sqsubseteq \exists_1\mathbf{hasShape}.\mathbf{Round} \sqcap \exists_1\mathbf{isMovable}.\mathbf{True},$$
$$\mathbf{RoboCupBall} \sqsubseteq \mathbf{Ball}$$

Robot 2 (Initial):

$$\mathbf{PinkBall} \sqsubseteq \exists_1\mathbf{hasShape}.\mathbf{Round} \sqcap \exists_1\mathbf{isMovable}.\mathbf{True} \sqcap \exists_1\mathbf{hasColour}.\mathbf{Pink}$$

Robot 1 Fused (After):

$$\mathbf{Ball} \sqsubseteq \exists_1\mathbf{hasShape}.\mathbf{Round} \sqcap \exists_1\mathbf{isMovable}.\mathbf{True},$$
$$(\mathbf{RoboCupBall} \sqcup \mathbf{PinkBall}) \sqsubseteq \mathbf{Ball},$$
$$\mathbf{PinkBall} \sqsubseteq \exists_1\mathbf{hasColour}.\mathbf{Pink}$$

If Robot 2 recognises a round, pink object and categorises it as a **PinkBall** any robot that receives a message describing all the ontological features of **PinkBall** will be able to infer that this object is a type of Ball and respond accordingly. Obviously, a key assumption here is that overlapping symbols are jointly grounded or are defined in separate name-spaces—this is necessary, and implied in our use of ontologies that are intended to represent an 'objective' view of reality. If the meaning of **Round** is not known to be identical between Robot 1 and 2, then these should be given different symbols or name-spaces in the ontology: they should only be related or equated when more information is known. Note, also, that Robot 1 need not be able to perceive **Pink**—if it is merely looking for any **Ball** to kick, it can still communicate with Robot 2 about the existence of the **PinkBall** and reason about it as a ball, even if it cannot independently classify an object as being specifically a **PinkBall** without the assistance of Robot 2.

6 Ontology, System Design and Implementation

The aim of our research is to implement and evaluate an ontology-based approach to categorization. The architecture, design and development of this system, OBOC, is briefly described below.

6.1 RoboCup Ontology

Using Protégé for development, a domain specific ontology was created for the 4-legged robots. Represented in this ontology are both **ConcreteObjects** such as goals, players, balls, regions and beacons; and **AbstractObjects** such as colour, shape, heading and position. The robot's beliefs and perceptions about the transient state of the soccer field are maintained and shared as assertions (ABox), while the ontology (TBox) is used and shared for recognizing and classifying those objects as specific and defined concepts.

6.2 System Design

Figure 2 illustrates the foundation for the design model. The entities of the system are described below:

Object Categoriser: responsible for categorising objects based on properties and context.

Ontology Management: manages existing classes and creates new classes in the ontology. In addition it services queries from the Concept Learner and communicates with the Ontology Reasoning entity to categorise a class based on the properties and/or context.

Concept Learner: queries the ontology for categorised concepts to infer identifiable properties and features of the object that it represents.

Concept Merger: is responsible for identifying semantic relationships between concepts.

Ontology Reasoning: performs queries on the ontology using the reasoning services of a third party component—Racer, located on a server.

6.3 Ontology Based Categorisation Tool

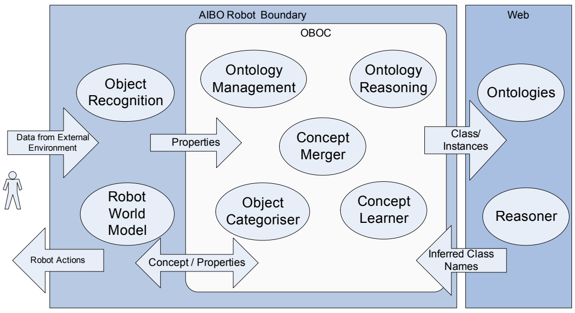

Figure 2: System boundary model of the OBOC system

A proof-of-concept tool has been developed and evaluated. The Protégé API was used to access and manage the ontologies, while the Racer server was used for ontological reasoning. The tool allows users to interact with the RoboCup Soccer ontology by enabling properties to be entered via a graphical user interface and immediately respond with categorized objects. Although this tool has been developed for a specific platform, the methods and techniques could be integrated into any robot infrastructure where high level object recognition capabilities is required for communication and collaboration.

7 Results

Our experimentation and interaction with OBOC does indeed confirm our claims:

OBOC successfully identified beacons, balls and other objects via the ontology. When colours were changed on the soccer field or in the ontology, OBOC responded appropriately.

While, not an ideal solution, simple qualitative heuristics proved sufficient for managing situations of uncertainty (other approaches are addressed as future work in the following section).

OBOC allowed human developers to richly query the state of the perceived world and rapidly understand the particular consequences of ontology modification.

OBOC responded in a timely and efficient manner. The theoretical upper bounds of ontological reasoning were not a practical problem.

While the current approach would violate of the rules of RoboCup (the use of an external reasoning server is prohibited), these results are very promising.

8 Generality and Future Work

The OBOC system is a starting point for a holistic approach to grounding robots. In this section, we focus on the feasibility of incorporating other extensions to OBOC. We claim that it is possible for future versions of OBOC to support tasks including reasoning, planning, ontology evolution, multi-agent knowledge sharing and coordination.

8.1 Reasoning Support

OBOC already incorporates advanced reasoning capabilities as a consequence of the use of logical formalisms. OWL is based on description logics (DL)—these have well-defined model-theoretic semantics and decisions procedures that provide a sound and complete support for concept satisfiability, concept subsumption and instance checking. In fact, efficient reasoners such as FaCT, Racer or KOAN2 offer highly practical performance on standard case problems.

Nonmonotonic reasoning can simplify the expression of domain knowledge, and allows an agent to deduce intuitive results in the case of uncertainty using defaults. While OBOC currently does not support nonmonotonic reasoning, extensions of description logics have been proposed based on defaults [1] and circumscription [2], or may be layered with a nonmonotonic logic providing higher-level reasoning. Furthermore, integration of description logics with logic programming rules [5, 15,18,29] can enrich the expressivity of an ontology and introduce closed world reasoning into otherwise open-world DL ontologies.

8.2 Planning

OBOC currently only supports categorization, but by extending the robot ontology to incorporate descriptions of actions, it would be possible to improve the groundedness of the robot's planning capabilities. Implementation need not be overly sophisticated: in DL-based ontologies, the existence of a plan can be reduced to reasoning on satisfiability of DL [13]. Planning with DLs has been very important in the composition and automatic execution of Semantic Web services. These very same techniques can be applied to autonomous robots for planning their goals.

In the longer term, future research will explore the representation of even broader ranges of concepts. Ideally, it would be possible to represent not just a plan of the current robot, but also the intentions and likely plans of other robots (both team-mates and opposition). This level of sophistication would be cued by the opposition's behaviours (i.e., inferring intention from behaviour) and is necessary to anticipate behaviour and perform actions such as deliberate (as opposed to opportunistic) intercepts.

8.3 Ontology Evolution

While the ontology-based approach used in OBOC allows for rapidly evolving ontologies and sharing knowledge, in richly dynamic environments the underlying ontologies may be subject to on-line revision: humans may supply new knowledge, robots may merge knowledge bases and rules of the game may change. In classical formalisms, such changes introduce inconsistencies and the effect is explosive: any conclusion can be obtained from reasoning, and the ontology turns out to be trivial. The agent must either support nonmonotonic reasoning or, in the case of more dramatic change, revise its own ontology to allow the consistent addition of new knowledge.

There has been much recent work on inconsistency handling in DL ontologies. Although standard belief revision applies for the propositional case, generalizing these results to description logics can present a challenge. Flouris [6,7] proposes a DL version of AGM postulates that serve as rational postulates for ontology contraction and revision operators, and Qi *et al.* [17] offer a model-theoretic version of AGM postulates. These works lay the foundation for ontology revision. Revision methods include those of Qi [17], Meyer [12], Scholbach [20]. These methods allow robots to consistently revise their own knowledge in the face of new information.

Furthermore, ontology debugging and diagnosis may be useful for identifying and correcting (possibly with human assistance) the axioms that 'cause' inconsistency/incoherence [16,20]. Ontology debugging systems such as Swoop [16] can be readily employed in OBOC for improving the testability and transparency (and therefore groundedness) of the system.

8.4 Multi-agent Communication

While OBOC currently assumes that the ontologies shared between robots are in some way consistent or readily merged, in future it may be possible to support more loosely coupled coordination through techniques such as ontology mapping [23] to create translations or bridge axioms [26]. One solution is to have an ontology alignment protocol that can be interleaved with any other agent interaction protocol and would be triggered upon the receipt of an unrecognized message from a foreign ontology. Agents meeting each other for the first time would be able to negotiate the matching of terms in their respective ontologies and to translate the content of the message they exchange with the help of the alignment.

9 Conclusions

Building on Harnad's [9] 'solution' to the symbol grounding problem, we have designed an architecture for constructing robot systems that is attentive to the richer understanding of grounding and groundedness. In OBOC, very simple iconic and categorical representations are causally connected to the robot's sensory subsystems—providing an elementary grounding upon which Semantic Web technologies are applied. The result is a richer, more flexible, more adaptive and therefore more grounded robotic system. While OBOC is currently limited to object categorization on the basis of features and context, the OBOC architecture and the ontology-based reasoning systems can be readily extended with greater capabilities.

There is much scope for future work. Outside the robot soccer field, we obviously do not have the ability to mark or colour code every object. Of course, the key point in OBOC is that the grounding still takes place via the sensors, even if those sensors are in-fact unable to distinguish or identify complex objects outside of the soccer field. Aside from improving the reasoning capabilities of the system, there is also therefore much scope for improving the richness of the elementary grounded symbols of the system. The long term objective is to gradually enrich this elementary grounding, improve the reasoning capabilities, tool-set and detail of the system's ontology—that is, to improve the system's grounding—so that it can respond to new objects and learn from other systems in unknown situations outside of the crafted soccer field.

10 References

[1] Baader, F. & Hollunder, B., 'Embedding defaults into terminological knowledge representation formalisms', *J. Autom. Reasoning*, 14(1):149–180, (1995).

[2] Bonatti, P., Lutz, C. & Wolter, F., 'Description logics with circumscription', *Proc. of KR-06* (2006).

[3] Breen, C., Khan, L. & Ponnusamy, A., 'Image classification using neural networks and ontologies', *The 13th Int. Wksp. on Database and Expert Systems Applications*, pp. 98-102, (2002).

[4] Coradeschi, S. & Saffiotti, A., 'Special issue on perceptual anchoring symbols to sensor data in single and multiple robot systems', *Robotics and Autonomous Systems*, 43 (2-3), (2003).

[5] Eiter, T. *et al.*, 'Combining answer set programming with description logics for the semantic web', *Proc. of KR04* (2004).

[6] Flouris, G., Plexousakis, D. & Antoniou, G., 'On applying the AGM theory to DLs and OWL', *Proc of ISWC-05* (2005).

[7] Flouris, G. *et al.*, 'Inconsistencies, negations and changes in ontologies', *Proc. of AAAI-06* (2006).

[8] Gärdenfors, P. & Williams, M-A., 'Building rich and grounded robot world models from sensors and knowledge resources: a conceptual spaces approach', *Proc. of the Int. Symp. on Autonomous Mini-robots for Research and Edutainment* (2003).

[9] Harnad, S., 'The symbol grounding problem', *Physica D: Nonlinear Phenomena*, 42: 335-346. , (1990).

[10] Kent, W., *Data and Reality*, North Holland, (1978).

[11] Limketkai, B., Liao, L. & Fox, D., 'Relational object maps for mobile robots', *Proc. of the IJCAI2005*, pp. 1471-1476, (2005).

[12] Meyer, T., Lee, K. & Booth, R., 'Knowledge integration for description logics', *Proc. of AAAI-05* (2006).

[13] Milicic, M., 'Planning in action formalisms based on DLs: first results', *Proc. of DL-07* (2007).

[14] Modayil, J. & Kuipers, B., 'Bootstrap learning for object discovery', *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 742-747, (2004).

[15] Motik, B. & Rosati, R., 'A faithful integration of description logics with logic programming', *Proc. of IJCAI2007* (2007).

[16] Parsia, B., Sirin, E. & Kalyanpur, A., 'Debugging OWL pntologies', *Proc. of WWW-05* (2005).

[17] Qi, G., Liu, W. & Bell, D.A., 'Knowledge base revision in description logics', *Proc. of JELIA-06* (2006).

[18] Rosati, R., 'DL+log: tight integration of description logics and disjunctive datalog', *Proc. of KR2006* (2006).

[19] Schlenoff, C., 'Linking sensed images to an ontology of obstacles to aid in autonomous driving', *Proc. of the 18th Nat. Conf. on Artificial Intelligence: Wksp. on Ontologies for the Semantic Web* (2002).

[20] Schlobach, S. & Cornet, R., 'Non-standard reasoning services for the debugging of description logic terminologies', *Proc. of IJCAI2003* (2003).

[21] Schober, J.P., Hermes, T. & Herzog, O., 'Content-based image retrieval by ontology-based object recognition', *Proc. of the KI-2004 Wksp. on Applications of Description Logics (ADL-2004)* (2004).

[22] Searle, J.R., 'Minds, brains and programs', *Behavioural and Brain Sciences*, 3 (3): 417-457, (1980).

[23] Shvaiko, P. & Euzenat, J., 'A survey of schema-based matching approaches', *J. of Data Semantics (JoDS)*, IV, (2005).

[24] Song, M.H., 'Ontology based automatic classification of web pages', *Int. J. of Lateral Computing*, 1 (1), (2004).

[25] Sowa, J.F., 'Ontology, metadata, and semiotics', viewed 2/8/05 <http://www.isye.gatech.edu/faculty/Leon_McGinnis/8851/Sources/Ontology/SemioticsSowa.htm>, (2000).

[26] van Eijk, R. *et al.*, 'On dynamically generated ontology translators in agent communication', *Int. J. of Intelligent Systems*, 16:587-607, (2001).

[27] Vogt, P., 'Anchoring of semiotic symbols', *Robotics and Autonomous Systems*, 43 (2): 109-120, (2003).

[28] Williams, M-A. *et al.*, 'A framework for evaluating groundedness of representations in systems: from brains in vats to mobile robots', *IJCAI2005 Workshop on Agents in Real-Time and Dynamic Environments* (2005).

[29] Yang, F. & Chen, X., 'DLclog: a hybrid system integrating rules and description logics with circumscription', *Proc. of DL-07* (2007).

[30] Ziemke, T., 'Rethinking grounding', *Austrian Soc. for Cognitive Science, Proc. of New Trends in Cognitive Science* (1997).