Hanging Around: Cognitive Inspired Reasoning for Reactive Robotics

Mihai POMARLAN [a], Stefano DE GIORGIS [b], Rachel RINGE [c], Maria M. HEDBLOM [d], Nikolaos TSIOGKAS [e]

[a] *Applied Linguistics Department University of Bremen, Bremen, Germany*
[b] *Institute of Cognitive Sciences and Technologies National Research Council, Catania, Italy*
[c] *Digital Media Lab University of Bremen, Bremen, Germany*
[d] *Department of Computing Jönköping School of Engineering, Jönköping, Sweden*
[e] *Department of Computer Science KU Leuven, Leuven, Belgium*

Abstract. Situationally-aware artificial agents operating with competence in natural environments face several challenges: spatial awareness, object affordance detection, dynamic changes and unpredictability. A critical challenge is the agent's ability to identify and monitor environmental elements pertinent to its objectives. Our research introduces a neurosymbolic modular architecture for reactive robotics. Our system combines a neural component performing object recognition over the environment and image processing techniques such as optical flow, with symbolic representation and reasoning. The reasoning system is grounded in the embodied cognition paradigm, via integrating image schematic knowledge in an ontological structure. The ontology is operatively used to create queries for the perception system, decide on actions, and infer entities' capabilities derived from perceptual data. The combination of reasoning and image processing allows the agent to focus its perception for normal operation as well as discover new concepts for parts of objects involved in particular interactions. The discovered concepts allow the robot to autonomously acquire training data and adjust its subsymbolic perception to recognize the parts, as well as making planning for more complex tasks feasible by focusing search on those relevant object parts. We demonstrate our approach in a simulated world, in which an agent learns to recognize parts of objects involved in support relations. While the agent has no concept of handle initially, by observing examples of supported objects hanging from a hook it learns to recognize the parts involved in establishing support and becomes able to plan the establishment/destruction of the support relation. This underscores the agent's capability to expand its knowledge through observation in a systematic way, and illustrates the potential of combining deep reasoning with reactive robotics in dynamic settings.

Keywords. Neurosymbolic Approaches, Image Schemas, Situated Robotics

1. Introduction

A complex tapestry of latent knowledge underpins every interaction between an agent and its environment. Aspect such as affordances, agent's own capabilities, and nuanced properties of the environment form the bedrock upon which cognitive agents perceive, interpret, and navigate their surroundings. The depth of this interaction is not merely a function of the agent's immediate sensory input but is influenced by a pre-existing, albeit latent, framework of knowledge. This framework includes stored interaction patterns and hard-wired relations that dictate the agent's engagement with its environment, guided by the rules that govern the world in which the agent is operating.

Despite impressive advancements in generative AI "foundational models," a significant gap remains in their understanding and representation of physical and spatial dynamics. Most recent OpenAI release, SORA[1], is a Language and Vision Model to generate video from text prompts. While the generated contents at this time are very good looking, they also showcase limitations such as a poor grasp of object permanence – entities may flick in and out of existence in implausible ways – and elementary physics laws [1].

Endorser of the "more data is all you need" claim that such errors will eventually be fixed, and this is not, a-priori, a vain hope: even large neural networks have to "compress" its training data, and in so doing, they will stumble upon regularities of the world. We are reluctant to endorse this view however. Generative AI models, while groundbreaking in generating coherent and contextually relevant linguistic or visual outputs, primarily operate on the basis of the statistical probability of sentence or image completion, with statistics obtained from a corpus of decontextualized recordings. This approach has a fundamental limitation: the lack of embodied grounding, informed by real-world sensorial data, obtained and interpreted by an agent in purposeful engagement with that world.

More than ever, SORA's impressive results raise questions like: what is that world knowledge we need, how could it be described, how would it be used in an autonomous agent interacting with some kind of world?

Furthermore, SORA and LLM's lack of embodied grounded knowledge is an echo of Moravec's paradox [2]: the observation that human intuitions for what is cognitively easy do not translate to machines. The problem of endowing practical know-how to artificial agents is of chief relevance in autonomous robotics, and it is from this perspective – that of artificial agents – that we approach the problem. Thus, though our system is cognitively inspired, we do not endeavour to obtain cognitive plausibility.

With the purpose to display how cognitive robotics, and in particular a neuro-symbolic architecture, is capable to perform commonsense reasoning on the world thanks to embodied cognition knowledge, our exploration in this work is driven by the following questions: which objects exist, where to place attention, and how may an agent enrich its knowledge, at different levels of abstraction, about the entities in the world and their interactions.

[1]SORA's technical report is available here:
`https://openai.com/research/video-generation-models-as-world-simulators`

Our contribution is threefold: (i) we represent, in a formal way, a semantic parsing of sensorimotor events that a cognitive system undergoes as it observes and interacts with an environment. This event segmentation is grounded in the paradigm of embodied cognition, specifically the notion of Image Schemas, as detailed in Section 2.

Second (ii), we propose the development of a modular reasoning system to identify specific situations, the entities involved, and the roles they play. This system is designed to operate at the intersection of neuro-symbolic processing, leveraging the continuous stream of perceptual data obtained from a neural architecture. The fusion of neural inputs with symbolic, ontology-based reasoning allows for the transformation of raw sensory signals into structured, actionable knowledge. This integration enables to dynamically update the system's internal knowledge to new information and environmental changes.

Lastly, (iii) we investigate how the combination of neural network-based object recognition and reasoning can enlarge an artificial agents ontology. Our agent starts with a certain knowledge base: it knows a set of objects, understood as finite shapes. However, it can teach itself to recognize parts of these objects if those parts are involved in functional relationships, i.e. relationships between objects that constrain how a scene will unfold.

Thus, we introduce a "sense-making" system, dedicated to the identification and understanding of mereological affordances within the environment. This system, through the repeated observation of spatio-relational patterns, is adept at recognizing areas of interest that exhibit new *emergent properties*. These properties are not static but evolve based on the functional parts of the environment, thereby presenting a continuously shifting landscape of interaction possibilities.

Finally, from an ontological point of view, our work relies on the Description & Situation (DnS) pattern [3,4], based on the reification of intensional/extensional relations with recursive accessibility.

2. Related Work: Conceptual Modeling, Embodied Grounding, and Cognitive Robotics

In this section we will briefly review some conceptual tools and existing approaches we use to build our system. We start with a summary on image schemas and their hypothesized roles in human cognition, continue with frame semantics as employed to understand descriptions of world states, and end with a discussion on how image schemas have been previously formalized. Finally we provide references to previous approaches in robotics relying on cognitive paradigms.

Image Schemas To have an understanding of the space of possibilities for an artificial agent, it helps to look at what humans seem to do. The learning process in children, known as perceptual meaning analysis (PMA) [5], involves deriving generalized spatiotemporal patterns, called image schemas [6], from such interactions. Image schemas represent a finite set of relationships among objects, agents, and their environments that define their uses and the spaces of their affordances. Examples are CONTAINMENT, meaning one object can be/is contained

inside another one and SOURCE_PATH_GOAL, meaning objects can/do move along particular trajectories.

Image schemas are considered the foundation for reasoning [7], and were shown to evolve into cognitive functions such as natural language and conceptualizations of abstract entities [6,8] through grounded, experiential patterns. Furthermore, image schemas can combine in more complex structures. Take for instance the notion of "transportation". It does not rely on any object in particular, but can be generally understood as the "movement of object(s) from A to B". In image-schematic terms, it can be described as a combination of SOURCE_PATH_GOAL and SUPPORT or CONTAINMENT [9]. By combining these concepts in constellations and sequences (as state spaces) [10,11], it is possible to formally describe the structure of increasingly complex events.

Frame Semantics and Its Ontological Modelling For the conceptual modeling we rely on the Frame Semantics cognitive paradigm and reuse the notion of "frame", as in Minsky and Fillmore [12,13]. Frames are schematic abstract representations of recurrent situations. Each frame takes a set of semantic roles, namely the elements which participates to the frame situation. The minimum set of roles to realise a frame composes the "necessary roles". Frames are formalised as N-ary relations with a central node being the Event/Situation, and a number N of semantic roles participating to it. Note that the set of possible roles is much broader than the one of its necessary roles.

From an ontological modeling perspective, employing frames for representations adheres to good modeling practices through the adoption of the Description and Situation [3] Ontology Design Pattern. In this framework, each image schema discussed in our study is conceptualized as a semantic frame, represented as a *Description* that is satisfied by a *Situation*. In more detail, the *Situation* reflects a specific world state, encapsulating the essential roles required for its realization. If the *Situation* presents the necessary roles as formalized by a particular *Description*, then the *Situation satisfies* the *Description*. DnS as a formalization of Frame Semantics has been largely used in various projects (mainly in its OWL formalization) [14,15,16,17].

Image Schema Ontological Modeling Formally representing image schemas is a complex problem as they are conceived and treated as abstract "gestaltic entities" [6] without clear borders or structure. However, traditional methods in spatiotemporal reasoning have been proposed as representation approaches (e.g. [9]) and some of these logical languages and calculi, in particular Region Connection Calculus [18], Qualitative Trajectory Calculus [19] and Linear Temporal Logic [20] were combined into the modelling language the Image Schema Logic, ISL^{FOL} [21,22]. Being a description language in first-order logic, ISL^{FOL} can capture detailed spatiotemporal interactions and transformations that can be used to represent situations or events.

Complex events can be seen as compositions and co-occurrences of more elementary situations or 'scenes', which in turn can be represented using image-schematic representations [10]. In this way, it is possible to decompose events and even robotic action plans based on the sensorimotor input about a situation, exploiting flowing data from a perception module (see examples in [11,10,23,24]).

However, for most autonomous systems, this level of spatiotemporal modelling is too detailed to be used for actual real time situation analysis and decision making tasks. Therefore, the fundamental image-schematic representations have been proposed by transposing the methods into different types of description logics, e.g. EL++ [23] and OWL2[2] [25]). In [26], we introduced the image-schematic reasoning layer (ISRL) which is based on ISL2OWL, a simple ontological module of image-schematic components.

In ISL2OWL, acting as an ontological module, each image schema is modeled as a semantic frame. More precisely as an N-ary relation with central node the image schematic situation, where the spatial primitives forms the necessary roles. For example, a SUPPORT situation takes as necessary roles two elements: a SUPPORTER, and a SUPPORTED entity. The underlying theoretical assumption is derived from image schematic literature, and directly dependent on the Gestalt [27], frame-based nature of image-schemas [28]: if one of its roles (spatial primitives) is instantiated, this implies the activation of the whole image schema. This means that knowing there is a supporting entity also means knowing there is a supported one, and situation is a Support situation.

Therefore, given a certain state of the world, if a certain entity is retrieved as being in movement, that particular state of the world will be represented as a MOVEMENT situation, taking as participant the moving entity as MOVER. The same situation could, of course, be qualified by more than one image schematic relation, in a combinatorial increasing degree of complexity.

Furthermore, following [10], there are three possible image schematic combinations: Merge, Collection, and Structure. Thanks to the frame approach, more complex scenarios can be modeled as N-ary relations taking as roles more simple situations, for example, a TRANSPORTATION situation results from the co-occurrence of a MOVEMENT situation co-located with a SUPPORT situation by being axiomatised as taking as roles some MOVEMENT and SUPPORT. Thanks to the reasoning system described in the following, a MOVEMENT situation $S1$ having as participants x and y, and a SUPPORT situation $S2$, taking as participants the same x and y, is inferred as TRANSPORTATION situation.

While a lot of information is abstracted away from the original ISL^{FOL}, representing the image schemas in a computationally feasible way in ISL2OWL allows for them to be used in logical reasoners and as a consequence, we can represent them as semantic building components for the task descriptors.

Cognitive Robotics Allowing a robot to display an intelligent behaviour is the main goal of the field of cognitive robotics. It involves studying the knowledge representation and reasoning problems a robot faces in a dynamic and partially observable world [29]. In addition to representing knowledge and reasoning, cognitive robotics studies methods of learning through interaction with the environment [30,31].

For a cognitive robot to function in the dynamic and uncertain environment that the real world is, three main components are required: i) a source of knowledge regarding the environment, ii) a computational framework to process this

[2]See the full ISL2OWL graphs at `https://github.com/StenDoipanni/ISAAC/tree/main/ISL2OWL`

knowledge, and iii) a world representation that models the environment and the robot's behaviours. A combination of these components is named a *Semantic Reasoning Framework* (SRF) [32], from which we inherit the terminology to describe the architecture in the next section.

3. Semantic Reasoning Framework

In this section we describe the overarching goal of our agent and its modular structure. The fundamental goal of the agent is to gather information from the environment and interpret it in image schematic terms, so as to maintain an ongoing model of what it observes and engages with, and produce decisions on how to continue that engagement. A visual representation of the architecture can be seen in Fig. 1. We provide an informal, high-level descriptions of the various modules in the following subsections, to delineate what roles symbolic inferences ultimately play in our approach. We then describe the theories employed for the reasoning task in more formal detail in Section 4.

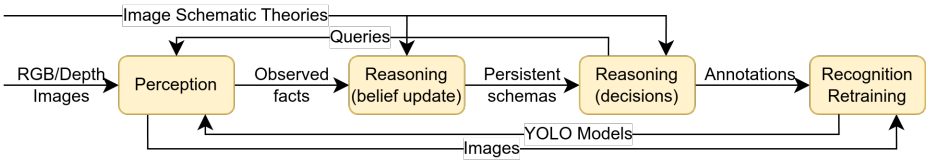

Figure 1. Architectural overview of the agent.

3.1. Towards Engagement with the World

A naive understanding of perception would be that, modulo errors that in principle can be eliminated, it constructs a truthful picture of the world out of facts independent of contextual factors such as the goals of the perceiver. Following Heidegger, AI critic Hubert Dreyfus argued against this view and that it is responsible for the stalling of early AI efforts [33]. In an attempt to simplify and translate a part of Dreyfus' critique in more engineering friendly terms[3], we would say that the fact of noise requires filtering, and filtering requires assumptions as to what is noise and what is meaningful. Thus, an agent is not a passive receiver of facts from the environment via a perception pipeline, i.e. a computational structure that feeds information in only one way.

Instead, an agent must actively choose what and how to look at, based on its current understanding of how it is embedded in a situation. This understanding includes beliefs about what the situation is and the agent's place in it, and how these may change in the future. Perception is then "taskable": reconfigurable

[3]The coauthors with a robotics background find such "translations" very necessary because it is otherwise hard from Dreyfus' philosophy to infer what to actually do in the context of AI research. Dreyfus may have intended to say AI is doomed, but as AI researchers we have to play that game anyway.

based on the agent's needs, in order to answer specific queries dictated by aspects such as what the agent expects of a situation.

Thus, one of our goals in implementing our system, is to investigate the knowledge structures which, if appropriately connected to sensoric and motor procedures, would enable an agent to have an understanding of a situation and its (plausible) evolution. For the sake of exposition we will use a shorthand and speak of a symbolic layer implementing reasoning on image schematic assertions, such as that some object supports another. It should be understood that the meaning of such a statement is not captured merely at the symbolic layer, but rather in how the symbolic inferences rewire sensors and actuators. "Support" is just a name, it gets its meaning from what the agent expects to observe and may decide to do and what outcomes this has in the world, including in the agent's own disposition about what to perceive and how to act.

Dreyfus' critique was arguably influential in the development of the related fields of reactive, embodied and situated robotics, a field which he himself later reviewed and described as "Heideggerian AI" [34]. We place our work in the field of reactive robotics too, so it is pertinent to notice that Dreyfus' verdict was this AI project also stalled – a conclusion he would probably maintain today as well.

Again simplifying and "translating", Dreyfus observes that reactive robotic systems are nonetheless trapped by their ontology. Reactive robots have a given, finite, inventory of concepts and no ability to produce new ones regardless of any interactions with the world they may experience. Simplifying even more, Dreyfus' challenge is to have an agent able to teach itself to see new things.

To address this challenge we endow our system with the ability to store snapshots of sensor data and automatically annotate them as exemplars of concepts created at the image schematic layer. These "new concepts" have a given structure – "an object that can play a particular role in a situation satisfying some description" – but can in principle be arbitrarily complex based on how intricate the role and situation descriptions are. Simply creating a new concept expression is of course not enough to produce something meaningful, which is why the stored exemplars are used to retrain perception – literally, training it to see new patterns. Thus, the new concepts become grounded in new perception procedures to recognize them in the world, and in ways to use the new objects once discovered – the concepts describe what roles they can play.

This is made possible precisely by the interplay of a symbolic layer maintaining an agent's understanding of its situation, and its subsymbolic sensorimotor apparatus. The sensors can partition the world in an infinity of ways – in other words, the pixels of an image can be clumped arbitrarily – while the image schematic understanding picks out which ways may be meaningful. This allows the system to somewhat escape the ontology trap seen by Dreyfus, because it is not limited to an initial set of objects it can recognize and which it is forced to treat as atomic. Instead, it can discover "functional parts", i.e. partition objects based on how the objects play the roles they play in a situation.

3.2. The Perception Module

The perception module offers the agent a conversion between numeric sensor data obtained through an RGB and depth camera and qualitative descriptions. It fulfils

this task by answering queries such as object, relative movement, and contact detection. A query is represented as a triple of form (p, s, o) where p indicates the type of query, and s and o are objects. Note that o can also be left blank, in which case the query is interpreted as asking about all objects that move relative to/contact object s.

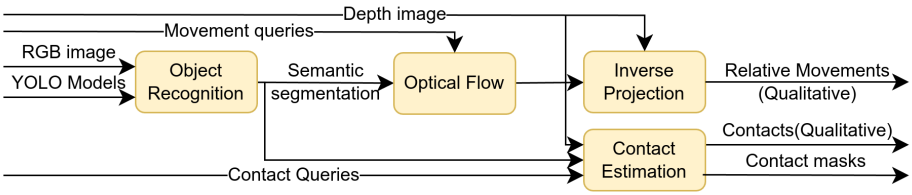

Figure 2. Overview of the perception module.

As shown in Fig. 2, the perception module produces annotations of pixels in the image. Neural networks – YOLOv8 models [35] – are used to annotate pixels as belonging to one of several classes of interesting objects; see Fig. 3. These annotations are referred to as "segmentations" (of the image into objects). Another annotator is the contact region annotator, which flags pixels close to an area where two objects of interest are in contact.

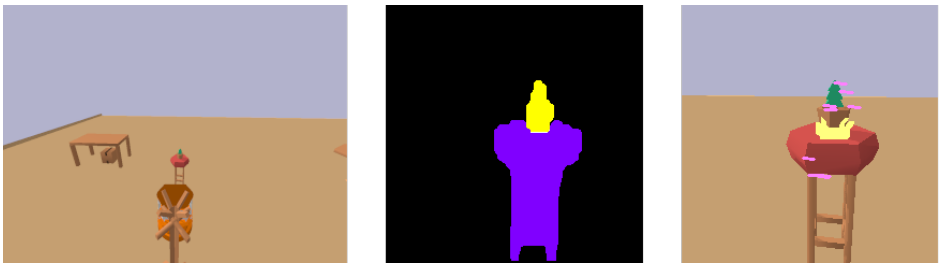

Figure 3. *Left:* a 3rd person view of the turtlebot in the scene. *Middle:* segmentation masks from YOLO. *Right:* optical flow points (purple) and contact masks (yellow) superimposed on the robot's RGB image.

The main perception output is a set of qualitative descriptions expressed as triples of forms (pso), $(-pso)$ where p can be *contacts*, *approaches*, *departs*, *stillness*, and $-p$ can be $-contact$ (objects not in contact), and a set of contact masks. For perception to assert any triple, or produce contact masks, it has to be asked to look. Without queries, there will be no perception results.

3.3. The Reasoning Module

The reasoning component is responsible for maintaining a belief state about the situation and deciding what to do based on that belief, see Fig. 4 for an overview. Most of the reasoning is done with (defeasible) rules. For the work in this paper, defeasibility was not yet used and the inferences can be implemented via SWRL.

The main constituent of the robot's belief is a set of persistent (image) schemas, i.e. assertions about relationships between objects such as CONTACT,

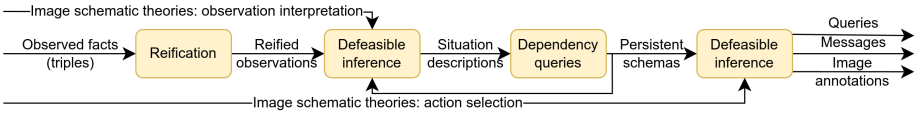

Figure 4. Overview of the reasoning components.

SUPPORT, as well as assertions about the robot's "goals". Reification steps are necessary because one can assert statements about image schematic relationships, e.g. the presence of an image schema may prevent a goal from being fulfilled. Thus, a statement coming from perception that (*contact a b*) must be converted into a set of statements about the existence of an entity of type *Contact*, with participants *a* and *b* – and this *Contact* entity is then further related to other image schemas perceived by the agent. Reifications also introduce new entities: the rule engine our agent uses does not allow existential rules, however the inference that certain special, "reifiable" predicates hold for a pair of entities will trigger the creation of a new entity and relationships to that predicate's arguments.

The reasoning module handles statements represented as triples, so these can be said to describe a graph. Some queries are convenient to handle with dedicated graph connectivity tests as opposed to rule-based inference, and thus received their own submodule: "dependency queries", which find entities/relationships necessary for some path to exist between certain other entities.

4. Image-Schematic Knowledge For a Naive Theory of Support

We now present part of our agent's theory related to the image schema *Support*. Our agent uses a rule engine and reification in its reasoning and thus we will give the axioms here not as rules (all variables universally quantified), but as FOL expressions. Variables are expressed in bold font.

We first describe a theory of *Support* situations, which we take to be that an object is supported if and only if it does not fall. We allow ourselves to make use of two categories of atypical objects – the *floor*, which exerts gravity from a distance, and *Fixed* objects, unmovable by force application. Other *Object*s are "typical": they can be moved by forces, and do not exert forces remotely. *Force*s act on objects and have *Direction*s, and directions may be opposite each other. For space reasons, we leave out disjointness of *Fixed* and *Object* and their common superclass of physical entities that can exert forces, and domain and range axioms that can be filled in by the reader from the informal glosses below.

A force that affects an object is exerted by some object (axiom 1). Gravity is a force with direction down (ax. 2), exerted by the floor, and acting on all objects (ax. 3). If an object other than the floor exerts a force on another, the two are in contact (ax. 4); if two objects are in contact, they exert forces on one another (ax. 5). The floor can exert an upward force only when in contact (ax. 6).

$$\forall \mathbf{f}, \mathbf{o} : aff(\mathbf{f}, \mathbf{o}) \rightarrow (\exists \mathbf{o2} : exrt(\mathbf{o2}, \mathbf{f})) \tag{1}$$

$$\forall \mathbf{f} : Grv(\mathbf{f}) \rightarrow Frc(\mathbf{f}) \wedge dir(\mathbf{f}, down) \tag{2}$$

$$\forall \mathbf{o} : Obj(\mathbf{o}) \rightarrow (\exists \mathbf{f} : exrt(floor, \mathbf{f}) \wedge Grv(\mathbf{f}) \wedge aff(\mathbf{f}, \mathbf{o})) \tag{3}$$

$$\begin{aligned} \forall \mathbf{o1}, \mathbf{o2}, \mathbf{f} :& (\mathbf{o1} \neq floor) \wedge exrt(\mathbf{o1}, \mathbf{f}) \wedge aff(\mathbf{f}, \mathbf{o2}) \rightarrow \\ & (\exists \mathbf{c} : Con(\mathbf{c}) \wedge hasPrtcp(\mathbf{o1}) \wedge hasPrtcp(\mathbf{o2})) \end{aligned} \tag{4}$$

$$\begin{aligned} \forall \mathbf{c}, \mathbf{o1}, \mathbf{o2} :& Con(\mathbf{c}) \wedge hasPrtcp(\mathbf{o1}) \wedge hasPrtcp(\mathbf{o2}) \rightarrow \\ & (\exists \mathbf{f} : exrt(\mathbf{o1}, \mathbf{f}) \wedge aff(\mathbf{f}, \mathbf{o2})) \end{aligned} \tag{5}$$

$$\begin{aligned} \forall \mathbf{f}, \mathbf{o} :& exrt(floor, \mathbf{f}) \wedge aff(\mathbf{f}, \mathbf{o}) \wedge dir(\mathbf{f}, up) \rightarrow \\ & (\exists \mathbf{c} : Con(\mathbf{c}) \wedge hasPrtcp(\mathbf{c}, floor) \wedge hasPrtcp(\mathbf{c}, \mathbf{o})) \end{aligned} \tag{6}$$

If a "typical" object does not move in the direction of a force exerted on it, then another force acts on that object in opposite direction (ax. 7). If an object exerts an upward force on another - through contact - then it is below (ax. 8).

$$\begin{aligned} \forall \mathbf{o}, \mathbf{f}, \mathbf{d} :& Obj(\mathbf{o}) \wedge aff(\mathbf{f}, \mathbf{o}) \wedge dir(\mathbf{f}, \mathbf{d}) \wedge \neg movDir(\mathbf{o}, \mathbf{d}) \rightarrow \\ & (\exists \mathbf{f2}, \mathbf{d2} : aff(\mathbf{f2}, \mathbf{o}) \wedge dir(\mathbf{f2}, \mathbf{d2}) \wedge opp(\mathbf{d2}, \mathbf{d}) \\ & \neg exrt(\mathbf{o}, \mathbf{f2})) \end{aligned} \tag{7}$$

$$\forall \mathbf{o1}, \mathbf{o2}, \mathbf{f} : exrt(\mathbf{o1}, \mathbf{f}) \wedge aff(\mathbf{f}, \mathbf{o2}) \wedge dir(\mathbf{f}, up) \rightarrow below(\mathbf{o1}, \mathbf{o2}) \tag{8}$$

"Typical" objects only have "typical" parts (ax. 9). Forces exerted on/by objects are exerted on/by parts of them (ax. 10, 11).

$$\forall \mathbf{o}, \mathbf{p} : Obj(\mathbf{o}), hasPrt(\mathbf{o}, \mathbf{p}) \rightarrow Obj(\mathbf{p}) \tag{9}$$

$$\forall \mathbf{o}, \mathbf{f} : exrt(\mathbf{o}, \mathbf{f}) \rightarrow (\exists \mathbf{p} : hasPrt(\mathbf{o}, \mathbf{p}) \wedge exrt(\mathbf{p}, \mathbf{f})) \tag{10}$$

$$\forall \mathbf{o}, \mathbf{f} : aff(\mathbf{f}, \mathbf{o}) \rightarrow (\exists \mathbf{p} : hasPrt(\mathbf{o}, \mathbf{p}) \wedge aff(\mathbf{f}, \mathbf{p})) \tag{11}$$

A *Support* situation has supportee and supporter (ax. 12). A supportee does not fall (ax. 13), a supporter exerts an upwards force on the supportee (ax. 14).

$$\forall \mathbf{s} : Supp(\mathbf{s}) \rightarrow (\exists \mathbf{e}, \mathbf{r} : suppee(\mathbf{s}, \mathbf{e}) \wedge supper(\mathbf{s}, \mathbf{r})) \tag{12}$$

$$\forall \mathbf{s}, \mathbf{e} : suppee(\mathbf{s}, \mathbf{e}) \rightarrow \neg movDir(\mathbf{e}, down) \tag{13}$$

$$\begin{aligned} \forall \mathbf{s}, \mathbf{e}, \mathbf{r} :& Supp(\mathbf{s}) \wedge suppee(\mathbf{s}, \mathbf{e}) \wedge supper(\mathbf{s}, \mathbf{r}) \rightarrow \\ & (\exists \mathbf{f} : exrt(\mathbf{r}, \mathbf{f}) \wedge aff(\mathbf{f}, \mathbf{e}) \wedge dir(\mathbf{f}, up)) \end{aligned} \tag{14}$$

Our agent actually uses descriptions of situations, so we need axioms to tell it what to query from perception to check that a Support description still applies,

and upon what perceptual results it should come to believe a Support description applies. I.e., what are consequences of a Support situation in the above theory become expectations to be had if a Support description is believed to be satisfied (ax. 15), and symptoms to diagnose as a Support description applying (ax. 16).

$$\forall \mathbf{s}, \mathbf{e} : DSupp(\mathbf{s}) \wedge suppee(\mathbf{s}, \mathbf{e}) \rightarrow qrelMov(\mathbf{e}, floor) \wedge qCon(\mathbf{e}) \tag{15}$$

$$\begin{aligned} \forall \mathbf{e}, \mathbf{r}, \mathbf{c} : & Con(\mathbf{c}) \wedge hasPrtcp(\mathbf{c}, \mathbf{e}) \wedge hasPrtcp(\mathbf{c}, \mathbf{r}) \wedge below(\mathbf{r}, \mathbf{e}) \wedge \\ & \neg movDir(\mathbf{e}, down) \rightarrow (\exists \mathbf{s} : DSupp(\mathbf{e}) \wedge suppee(\mathbf{s}, \mathbf{e}) \wedge supper(\mathbf{s}, \mathbf{r})) \end{aligned} \tag{16}$$

The descriptions an agent believes, and the perception results they are based on/applied to, are attached to one iteration of its perception-action loop. Perception queries produced at one iteration constrain available results at the next.

5. Functional Object Parts: Define, Recognize, Use

We now describe how the agent's theories described in Section 4 and the perception system pick out a new object concept, which is then used to teach perception to recognize the object, and how it makes motion planning queries feasible.

Predicates such as $qCon(\mathbf{e})$ are a trigger for perception to look for objects in contact with $\mathbf{e}$. Perception returns not only a statement of two objects being in contact, but a "contact mask": points near where this contact occurs. The theory of support asserts parts in contact also exert and are affected by forces relevant in the support situation. For an object class, e.g. *Mug*, which the agent observes supported by a *Hook*, a functional part, used to support the mug using the hook, is one for which there is an observation of the part playing the supportee role:

$$\begin{aligned} \forall \mathbf{x} : & MugSuppbyHook(\mathbf{x}) \leftrightarrow \\ & (\exists \mathbf{c}, \mathbf{s}, \mathbf{m}, \mathbf{h} : Con(\mathbf{c}) \wedge DSupp(\mathbf{s}) \wedge Hook(\mathbf{h}) \wedge Mug(\mathbf{m}) \wedge hasPrt(\mathbf{m}, \mathbf{x}) \\ & \wedge suppee(\mathbf{s}, \mathbf{m}) \wedge supper(\mathbf{s}, \mathbf{h}) \wedge hasPrtcp(\mathbf{c}, \mathbf{x}) \wedge hasPrtcp(\mathbf{c}, \mathbf{h}) \wedge below(\mathbf{h}, \mathbf{x})) \end{aligned} \tag{17}$$

Treating the definition of *MugSuppByHook* as a new concept allows the agent to collect images and contact masks that are observations of its instances, and retrain the neural network responsible for object detection. Thus, *MugSuppByHook* becomes a class of "object" the network can recognize, like *Mug* and *Hook* are in this example. Crucially, the network can recognize a *MugSuppByHook* even outside of a "supported by *Hook*" situation. What object detection labels as *MugSuppByHook* is such that it can play a supportee role in a possible situation.

The *MugSuppByHook* object is then useful when the agent is given a goal to support the mug from a hook. Such a goal needs motion planning, but before

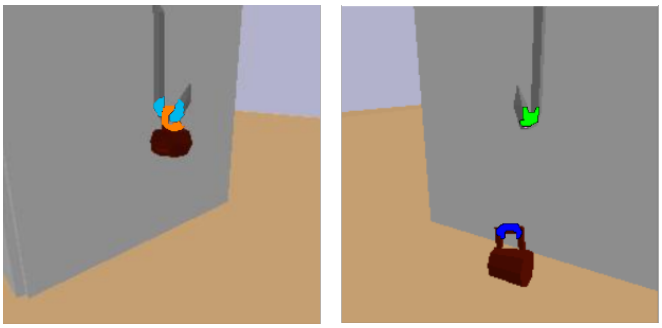

Figure 5. Functional parts. Left: a frame stored for training, with annotations of functional parts. Right: frame where the newly trained network is used to recognize functional parts.

one can apply search one needs to know what mug target pose would prevent its falling away from the hook. The theory of support provides necessary conditions: if the mug is supported by the hook, then they must be in contact, with the mug above; another constraint says the regions occupied by mug and hook should not overlap. Such a description is usable by a constraint solver to find a satisfying pose[4]. Unfortunately, many of the satisfying poses will not result in the mug being supported, e.g. having just the outside bottom of the mug touch the hook is not a stable configuration and the mug will fall.

Using the part of a mug labeled as *MugSuppByHook*, instead of the whole mug, as the entity for which to solve constraints reduces the search space, and also makes it virtually guaranteed that if a pose satisfying the constraint is found, then the mug is in fact supported by the hook.

6. Discussion, Conclusions, and Future Work

In this work we presented a neuro-symbolic architecture for cognitively inspired reactive roboticsand we escaped Dreyfus' ontology trap via learning new functional properties of existing objects and creating new concepts starting from sensor data. We have shown how through a combination of symbolic inference driven by image schematic knowledge, and numerical procedures such as perception algorithms and geometric constraint solvers, an artificial agent is able to recognize from observed situations examples of "functional parts", i.e. parts of objects that are relevant for particular image schemas. Through associating the concept expression for a functional part to a set of observations it is then possible to train perception to recognize a new kind of object. Further, because the new object is a functional part it assists in the solution of motion planning tasks, because it focuses the search for arrangements conducive to manifesting an image schema.

We are pursuing several avenues for continuation. One is to expand the role of geometric inference and physics simulation so as to incorporate anticipation. The main direction however is to expand the depth of time that the agent can consider.

[4]Since we have depth data, what the agent sees as labeled objects are sets of 3D voxels which can be moved and checked for collisions to find coordinates satisfying some symbolic constraint.

Its current operation attends only to the current moment, with the previous visual image used to compute relative movements. Schemas are persistent only in the sense that a set of triples that held at a previous step may still hold now, and previously captured images with annotations of functional parts are treated as independent of each other. However, functional parts involved in a situation often become obscured from vision by performing their role; e.g. the container part of a spoon sinking into a soup. Thus, it is necessary to annotate functional parts on frames where they do not yet perform the function, which requires, at the symbolic level, an understanding of a sequence of frames as observing a process with image-schematic consequences, and at the numeric level techniques to match frame parts assumed to exist at a particular location but invisible.

Acknowledgements

This work was supported by the Future Artificial Intelligence Research (FAIR) project, code PE00000013 CUP 53C22003630006, the German Research Foundation DFG, as part of Collaborative Research Center (Sonderforschungsbereich) 1320 Project-ID 329551904 EASE - Everyday Activity Science and Engineering, subproject "P01 – Embodied semantics for the language of action and change: Combining analysis, reasoning and simulation", and by the FET-Open Project #951846 "MUHAI Meaning and Understanding for Human-centric AI" by the EU Pathfinder and Horizon 2020 Program.

References

[1] Marcus G. Sora's surreal physics; 2024. Available from: `https://garymarcus.substack.com/p/soras-surreal-physics`.
[2] Moravec H. Mind children: The future of robot and human intelligence. Harvard University Press; 1988.
[3] Gangemi A, Mika P. Understanding the semantic web through descriptions and situations. In: OTM Confederated International Conferences" On the Move to Meaningful Internet Systems". Springer; 2003. p. 689-706.
[4] Gangemi A. Norms and plans as unification criteria for social collectives. Autonomous Agents and Multi-Agent Systems. 2008;17:70-112.
[5] Mandler JM. The Foundations of Mind: Origins of Conceptual Thought: Origins of Conceptual Though. New York: Oxford University Press; 2004.
[6] Johnson M. The Body in the Mind Metaphors. University of Chicago Press; 1987.
[7] LAKOFF G. The Invariance Hypothesis: is abstract reason based on image-schemas? Cognitive Linguistics. 1990;1(1):39-74. Available from: `https://doi.org/10.1515/cogl.1990.1.1.39`.
[8] Lakoff G, Núñez R. Where mathematics comes from. vol. 6. New York: Basic Books; 2000.
[9] Kuhn W. An image-schematic account of spatial categories. In: International Conference on Spatial Information Theory. Springer; 2007. p. 152-68.
[10] Hedblom MM, Kutz O, Peñaloza R, Guizzardi G. Image schema combinations and complex events. KI-Künstliche Intelligenz. 2019;33:279-91.
[11] St Amant R, Morrison CT, Chang YH, Cohen PR, Beal C. An image schema language. In: International Conference on Cognitive Modeling (ICCM); 2006. p. 292-7.
[12] Minsky M. A framework for representing knowledge. MIT, Cambridge; 1974.

[13] Fillmore CJ. Frame semantics. In: Linguistics in the Morning Calm. Seoul: Hanshin; 1982. p. 111-38.
[14] Scherp A, Franz T, Saathoff C, Staab S. F–a model of events based on the foundational ontology dolce+ DnS ultralight. In: Proceedings of the fifth international conference on Knowledge capture; 2009. p. 137-44.
[15] Eschenbach C, Gruninger M. Formal ontology in information systems: proceedings of the Fifth International Conference (FOIS 2008). vol. 183. IOS Press; 2008.
[16] Höffner S, Porzel R, Hedblom MM, Pomarlan M, Cangalovic VS, Pfau J, et al. Deep understanding of everyday activity commands for household robots. Semantic Web. 2022;13(5):895-909.
[17] Beretta F. A challenge for historical research: making data FAIR using a collaborative ontology management environment (OntoME). Semantic Web. 2021;12(2):279-94.
[18] Randell DA, Cui Z, Cohn AG. A spatial logic based on regions and connection. KR. 1992;92:165-76.
[19] Van de Weghe N, Cohn A, De Tre G, De Maeyer P. A qualitative trajectory calculus as a basis for representing moving objects in geographical information systems. Control and cybernetics. 2006;35(1):97-119.
[20] Kröger F, Merz S. Temporal Logic and State Systems. Texts in Theoretical Computer Science. An EATCS Series. Springer; 2008. Available from: https://doi.org/10.1007/978-3-540-68635-4.
[21] Hedblom MM. Image Schemas and Concept Invention: Cognitive, Logical, and Linguistic Investigations. Cognitive Technologies. Springer Computer Science; 2020.
[22] Hedblom MM, Kutz O, Mossakowski T, Neuhaus F. Between Contact and Support: Introducing a logic for image schemas and directed movement. In: Esposito F, Basili R, Ferilli S, Lisi FA, editors. AI*IA 2017: Advances in Artificial Intelligence; 2017. p. 256-68.
[23] Hedblom MM, Pomarlan M, Porzel R, Malaka R, Beetz M. Dynamic Action Selection Using Image Schema-Based Reasoning for Robots. In: The 7th Joint Ontology Workshops (JOWO). Bolzano, Italy; 2021. .
[24] Dhanabalachandran K, Hassouna V, Hedblom MM, Küempel M, Leusmann N, Beetz M. Cutting Events: Towards Autonomous Plan Adaption by Robotic Agents through Image-Schematic Event Segmentation. In: Proceedings of the 11th on Knowledge Capture Conference; 2021. p. 25-32.
[25] De Giorgis S, Gangemi A, Gromann D. Imageschemanet: Formalizing embodied commonsense knowledge providing an imageschematic layer to framester. Semantic Web Journal, forthcoming. 2022.
[26] Pomarlan M, De Giorgis S, Hedblom M, Diab M, Tsiogkas N. Thinking in front of the box: Towards intelligent robotic action selection for navigation in complex environments using image-schematic reasoning. In: The 8th Joint Ontology Workshops (JOWO). Jnkping, Sweden; 2022. .
[27] Verstegen I. Gestalt psychology in Italy. Journal of the History of the Behavioral Sciences. 2000;36(1):31-42.
[28] Lakoff G, Johnson M. Metaphors we live by. University of Chicago press; 1980.
[29] Levesque H, Lakemeyer G. Cognitive robotics. Foundations of artificial intelligence. 2008;3:869-86.
[30] Moldovan B, Moreno P, Van Otterlo M, Santos-Victor J, De Raedt L. Learning relational affordance models for robots in multi-object manipulation tasks. In: 2012 ieee international conference on robotics and automation. IEEE; 2012. p. 4373-8.
[31] Lara B, Astorga D, Mendoza-Bock E, Pardo M, Escobar E, Ciria A. Embodied cognitive robotics and the learning of sensorimotor schemes. Adaptive Behavior. 2018;26(5):225-38.
[32] Liu W, Daruna A, Patel M, Ramachandruni K, Chernova S. A survey of Semantic Reasoning frameworks for robotic systems. Robotics and Autonomous Systems. 2023;159:104294.
[33] Dreyfus HL. What Computers Still Can't Do. Revised edition ed. MIT Press; 1992.
[34] Dreyfus HL. Why Heideggerian Ai Failed and How Fixing It Would Require Making It More Heideggerian. Philosophical Psychology. 2007;20(2):247-68.
[35] Jocher G, Chaurasia A, Qiu J. Ultralytics YOLO; 2023. Available from: https://github.com/ultralytics/ultralytics.