Anchoring Action Representation to Perception in a Mobile Robot

Antonio Chella[1] and **Marcello Frixione**[2] and **Salvatore Gaglio**[3]

Abstract. The framework proposed in [3, 5] for the representation of inner perceptual knowledge of a robot is generalized to include a representation of actions, in order that a mobile robot can exploit its inner perceptual representations to anchor its actions to perception. Such extension is aimed to allow the robot to simulate its own future actions, to anticipate their consequences and evaluate them in order to choose the most appropriate action to perform. The proposed framework is illustrated by describing its performances in a *cat-mouse* scenario.

1 INTRODUCTION

In [3, 5] we propose a theoretical framework for the representation of knowledge about actions and dynamic scenes extracted from visual data. Among the aims of our proposal is a principled integration of the approaches developed within the artificial vision community on the one side, and the propositional systems developed within symbolic knowledge representation in AI on the other. Such an integration is based on the introduction of a *conceptual level* of representation, which is intermediate between the low-level processing of visual data and the symbolic representation. In addition, the conceptual level plays the role of an *inner environment* in the sense of Dennett [9] and of a *detached representation* in the sense of Gärdenfors [11].

Our approach is compatible with one of the most influential symbolic formalisms used in cognitive robotics for action representation, namely the *situation calculus* [15]. In this paper we discuss how *actions* and *fluents* of the situation calculus may be anchored [7] (for an up to date survey on different perspectives on anchoring see [6]) to representations in the conceptual level, which are generated starting from the robot perceptions. Moreover, the robot may simulate possible actions at the conceptual level and analyze their consequences, thus avoiding a costly and (possibly) dangerous trial-and-error behavior.

The following discussion is based on an experimental framework inspired by a *cat-mouse* task, in which a *cat* must catch a *mouse*. In our experiment, the cat is a RWI B21 robot equipped with stereo head (Fig. 1a), and the mouse is a Koala mini-robot (Fig. 1b). The mouse activities are driven by reactive behaviors: it wanders, avoids obstacles and escapes from the cat according to a schema-based architecture [1]. In addition to reactive behaviors, the *cat* is equipped with a symbolic KR, which is linked to the behaviors through a powerful conceptual level.

Next Section summarizes the main assumptions underlying the proposed architecture. Sects. 3 and 4 deal respectively with the subconceptual and the conceptual areas of the architecture, with particular emphasis on the representation of dynamic scenes. Sect. 5 suggests how the situation calculus can be mapped on the conceptual representation we adopted. Sect. 6 outlines how the conceptual area can be used as a simulation structure. Short conclusions follow.

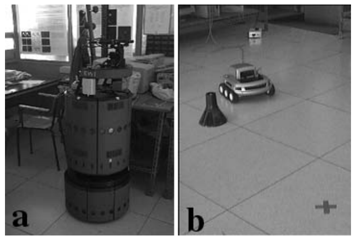

Figure 1. The employed robots: a) the cat, a RWI B21 robot; b) the mouse a Koala robot.

2 THE ROBOT INNER PERCEPTUAL KNOWLEDGE

We assume that a principled integration of the artificial vision representations and of symbolic KR requires the introduction of a missing link between the two kinds of representation [3, 5]. In our proposal the role of such a link is played by *conceptual spaces* [12]. A conceptual space (CS) is a representation in which information is characterized in terms of a metric space defined by a number of *cognitive* dimensions, which are independent from any specific language of representation. A CS acts as an intermediate representation between subconceptual knowledge (i.e., knowledge that is not yet conceptually categorized), and symbolically organized knowledge.

According to this view, the framework implemented in the *cat* robot is organized in three *computational areas*. Fig. 2 schematically shows the relations among them. The *subconceptual* area is concerned with the low level processing of perceptual data coming from the sensors. Here, information is not yet organized in terms of conceptual structures and categories. In the *linguistic* area, representation and processing are based on a logic based formalism, namely, the formalism of the *situation calculus*. In the *conceptual* area, the data coming from the subconceptual area are organized in conceptual categories, which

[1] Dipartimento di Ingegneria Informatica, Univ. of Palermo and CERE-CNR, Palermo, Italy, email: chella@unipa.it
[2] Dipartimento di Scienza della Comunicazione, Univ. of Salerno, and CERE-CNR, Palermo, Italy
[3] Dipartimento di Ingegneria Informatica, Univ. of Palermo and CERE-CNR, Palermo, Italy

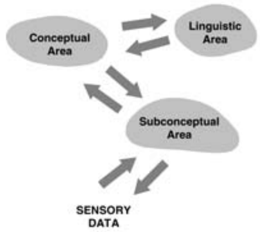

Figure 2. The three areas of representation, and the relations among them.

are still independent from any linguistic characterization. The symbols in the linguistic area are anchored on sensory data by mapping them on the representations in the conceptual area.

3 SUBCONCEPTUAL AREA

The subconceptual area is a repository of the behavior modules of the cat robot, which are responsible for its reactive activities. Some behaviors are purely reactive: they directly connect the sensors of the agent (in the present implementation, the camera and the odometer) to its actuators (the mobile base and the pan tilt), without interacting with conceptual and linguistic areas. Other behaviors process data coming from the sensors, and send the results to the conceptual space [4]. In particular, the 3D reconstruction module implements a simplified version of the RBC (Recognition By Component) approach [2] that employs superquadrics [13] as *geon*-like geometric primitives. The 3D reconstruction module employs both perceptive data coming out from the camera, and proprioceptive data coming from the robot odometry. The module is described in details in [3]. Fig. 3 shows the operation of 3D reconstruction module during camera calibration.

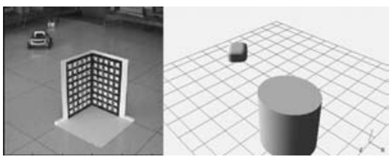

Figure 3. The operation of the 3D reconstruction module during camera calibration.

4 CONCEPTUAL AREA

As said before, representations in the conceptual area of the cat robot is couched in terms of *conceptual spaces* [12]. A conceptual space CS is a metric space whose dimensions are in some way related to the quantities processed in the subconceptual area. Different cognitive tasks may employ different conceptual spaces, and different conceptual spaces can be characterized by different dimensions.

We call *knoxel* a generic point in the conceptual space of the cat. A knoxel corresponds to an epistemologically primitive element at the considered level of analysis. In the case of static scenes [3], knoxels correspond to the superquadrics extracted by the above mentioned 3D reconstruction module.

In this example, in order to cope with the perception of dynamic scenes, an intrinsically *dynamic conceptual space* is adopted (see [5] for the details). Simple perceived motions are categorized in their wholeness, and not as sequences of static frames. According to this choice, every knoxel in the dynamic conceptual space of the cat corresponds to a simple motion of a 3D primitive. In other words, simple motions of superquadrics are assumed as the perceptual primitives for motion perception. Therefore, knoxels in the cat's conceptual area represent simple motions of the cat itself, of the mouse, of other objects, the surrounding obstacles, and so on (in this example static objects - e.g. obstacles - are assumed to be particular cases of simple motions).

In more details, a knoxel $\mathbf{k}$ in the cat conceptual space represents a *generalized* simple motion of a 3D primitive shape, where by *generalized* we mean that the motion is decomposed in a set of components each of them associated with a degree of freedom of the moving shape. In particular, the motions corresponding to each degree of freedom of a superquadric can be viewed as the result of the superimposition of the first low frequency harmonics, according to the Discrete Fourier Transform (DFT). We adopt the resulting functional space as the conceptual space of the cat for the representation of dynamic scenes [5].

The decision of which kind of motion can be considered *simple* so that it can be represented by a single *knoxel* is not straightforward, and is strictly related to the problem of motion segmentation. In the proposed framework, extending the approach followed in Marr and Vaina [14], a simple motions are characterized as the interval between two subsequent rest states. Such rest states may be instantaneous. Consider the cat moving towards the mouse, which is supposed to be at rest (Fig. 4a). When the cat is close enough, the mouse wakes up and tries to escape in some direction. As a consequence, the cat changes its own direction to chase the mouse (Fig. 4b).

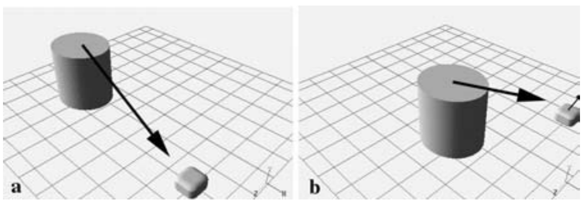

Figure 4. a) The cat moves towards the mouse. b) The mouse wakes up and escapes.

The first part of the trajectory of the cat is a simple motion represented in the conceptual space by a knoxel $\mathbf{k_a}$, which has been generated via its perceptive and proprioceptive sensors. When the mouse tries to escape, the cat abruptly changes its direction to pursuit it. The second part of the trajectory of the cat is another simple motion corresponding to a different knoxel, say $\mathbf{k_a'}$.

Something similar holds for the mouse: its rest state corresponds to a knoxel $\mathbf{k_b}$ in the conceptual space of the cat; then, a further knoxel $\mathbf{k_b'}$ describes its escape.

Fig. 5 is an evocative representation of the dynamic conceptual space of the cat. In the figure, each group of axes f^i corresponds to the i-th degree of freedom of a simple shape; each axis f^i_j in a group f^i corresponds to the j-th component pertaining to the i-th degree of freedom.

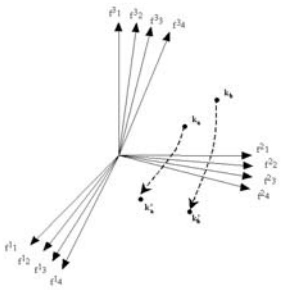

Figure 5. The dynamic conceptual space of the cat.

We call *composite simple motion* a motion in which more than one superquadric is involved, such as, for example, the motion of a composite object (i.e., an object approximated by more than one superquadric). A composite simple motion is represented in the CS by the set of the knoxels corresponding to the motions of its components. An example of composite simple motion could the chase of the mouse by the cat considered as a whole. The chase is a composite simple motion made up by the knoxels $\mathbf{k_a}$ (the motion of the cat) and $\mathbf{k_b}$ (the motion of the mouse). Note that in composite simple motions the (simple) motions of their components occur simultaneously. That is to say, a composite simple motion corresponds to a single configuration of knoxels in the conceptual space.

In order to consider the composition of several (simple or composite) motions arranged according to some temporal relation (e.g., a sequence), we introduce the notion of *structured process*. A structured process corresponds to a series of different configurations of knoxels in the conceptual space of the cat. We assume that the configurations of knoxels within a single structured process are separated by instantaneous changes. The transition between two subsequent different configurations involves the change of at least one knoxel in the CS. We call *knoxel scattering* the change of the configuration of knoxels in CS [5].

For example, when the cat reaches the mouse, a knoxel scattering occurs in the CS of the cat, due to the abrupt change of the knoxels describing the motion state of the two robots.

It should be noted that a knoxel scattering occurs in the cat CS also when an object appears or disappears; e.g., this is the case when the mouse escapes outside the cat field of view, or when it is hidden by a large obstacle.

5 MAPPING SITUATION CALCULUS ON CONCEPTUAL SPACES

In the present work, the *situation calculus* is proposed as a formalism for the linguistic area. In this Section we suggest how a representation in terms of the situation calculus could be mapped on the conceptual representation presented above. Such a mapping is based on the mechanisms described in [3, 5]; in particular, it is based on the combination of both bottom up (data driven) and top down (knowledge driven) processes.

The situation calculus is a logic based approach to knowledge representation, developed in order to express knowledge about actions and change using the language of predicate logic. In the following, we will refer to the exhaustive introduction to the situation calculus by Raymond Reiter [15].

The basic idea behind the situation calculus is that the evolution of a dynamic system can be modelled in terms of a sequence of situations. The world changes only when some *action* is performed. So, given a certain situation S_i, performing a certain action a will result in a new situation S_{i+1}. The situation calculus is formalized using the language of predicate logic. Situations and actions are denoted by first order terms. The two place function do takes as its arguments an action and a situation: $S_{i+1} = do(a, S_i)$ denotes the new situation S_{i+1} obtained by performing a in the situation S_i. Classes of actions can be represented as functions. For example, the one argument function symbol $start\_move(r, x)$ could denote the class of the actions consisting in moving the robot r towards x. As the dynamic system evolves, properties and relations change their values. Properties and relations that can change their truth value from one situation to another are called (relational) *fluents*. All actions in the strict sense are assumed to be instantaneous. Actions that have a duration are represented as *processes* (which are particular fluents) that are initiated and are terminated by instantaneous actions.

This approach is analogous to the representation of actions adopted in the dynamic conceptual spaces described in the preceding section. In a nutshell, a scattering in the cat conceptual space CS corresponds to an (instantaneous) *action*. A knoxel corresponds to a particular *process*. The sequence of the CS configurations obtained when the cat perceives a sequence of actions corresponds to a *situation*.

Note that both in the situation calculus and in our conceptual approach instances of fluents (or, respectively, knoxels) can correspond either to actions with a temporal duration, or to static states of affairs. However, in our approach, information about the kinematic properties of knoxels exists at the conceptual level, and, if needed, it can be mirrored by means of suitable predicates at the linguistic level.

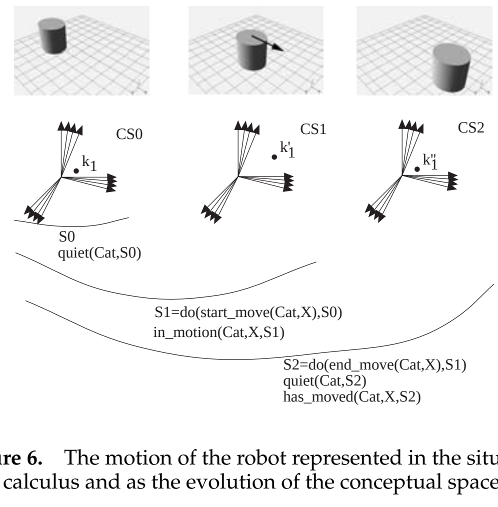

Figure 6. The motion of the robot represented in the situation calculus and as the evolution of the conceptual space.

To clarify these concepts, consider Fig. 6, representing the motion of the cat towards a certain position X, as perceived by itself via its proprioceptive and perceptive sensors. The initial situation S_0 corresponds to the initial configuration CS_0 in the cat conceptual space, in which k_1 corresponds to the cat robot at rest. In this situation, the fluent $quiet(Cat, S_0)$ is true. When the sensors of the robot perceive its own motion, then a scattering occurs in its conceptual space, and a new configuration CS_1 is generated, in which k_1 scatters to k_1'. In the linguistic area this scattering corresponds to an instantaneous action $start\_move(Cat, X)$.

The new situation $S_1 = do(start\_move(Cat, X), S_0)$ (resulting from performing in S_0 the action $start\_move(Cat, X)$) corresponds in CS to the sequence of configurations (CS_0, CS_1).

During all the time in which the cat remains in such a motion state, its CS configuration remains unchanged (provided that nothing else is happening in the considered scenario). In the meanwhile, the fluent $in\_motion(Cat, X, S_1)$. remains true. When the motion of the cat ends, k_1' scatters to k_1'', corresponding to the robot's rest. This second scattering corresponds to another instantaneous action $end\_move(Cat, X)$. This results in a new situation $S_2 = do(end\_move(Cat, X), S_1)$, corresponding to the sequence of configurations (CS_0, CS_1, CS_2). In S_2 both the fluents $quiet(Cat, S_2)$ and $has\_moved(Cat, X, S_2)$ are true.

In its traditional version, the situation calculus does not allow to account for concurrency. Actions are assumed to occur sequentially, and it is not possible to represent several instantaneous actions occurring at the same time instant. For our purposes, this limitations is too severe. When a scattering occurs in a CS it may happen that more knoxels are affected, i.e., several instantaneous actions occur concurrently. For example, the trajectory of two robots moving concurrently is represented as a composite motion made up by the knoxels $\mathbf{k_a}$ (the motion of the cat) and $\mathbf{k_b}$(the motion of the mouse).

Extensions of the situation calculus that allows for a treatment of concurrency and the related problems have been proposed in the literature, see Shanahan for a review [16]. In what follows, the *concurrent non-temporal* extension of situation calculus proposed by Reiter [15] is adopted. In particular, if a_1 and a_2 are two actions, the set $\{a_1, a_2\}$ denotes the action of performing a_1 and a_2 concurrently. An action is *primitive* if it is not the result of other actions performed concurrently.

In our approach, the scattering of a single knoxel in CS corresponds to a primitive action; a set of knoxels scattering at the same time is a complex action resulting from concurrently performing the corresponding set of different primitive actions.

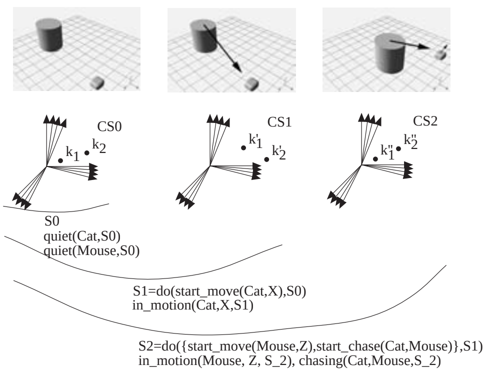

Figure 7. The chasing operations of the cat.

Consider again the example of the cat chasing the mouse. When the cat points towards the mouse, the formulas in the linguistic area are similar to those of the previous example: the cat moves towards the point X where the mouse rests. Now, the new situation S_2 is characterized by the fact that both the cat and the mouse start to move. Therefore, a scattering occurs in the CS of the cat, involving both the knoxels representing the motion of the cat itself and of the escaping mouse.

In the concurrent non-temporal situation calculus formalism, this new situation can be described by the formula: $S_2 = do(\{start\_move(Mouse, Z), start\_chase(Cat, Mouse)\}, S_1)$, in which the actions $start\_move(Mouse, Z)$ and $start\_chase(Cat, Mouse)$ are performed concurrently. In S_2, the following fluents hold: $in\_motion(Mouse, Z, S_2)$ and $chasing(Cat, Mouse, S_2)$.

Fig. 7 shows the chasing operation expressed both in the situation calculus and in terms of the evolution of the cat CS configurations.

6 ACTION SELECTION IN CONCEPTUAL SPACES

The described framework has been extended to allow the cat robot to use the conceptual representations in order deliberate its own sequences of actions. The forms of planning that are more directly related to perceptual information can take great advantage from the representations in the robot conceptual area. In this perspective, the preconditions of a planned action can be simply verified by geometric inspections of the CS; also the effects of an action can be checked by examining the *expected* CS configurations resulting from the *imagined* execution of the action itself.

In order to illustrate the spirit of this approach, in the following we shortly describe a simple example. Let us suppose that the cat (through the mapping mechanisms described in [3, 5]) has recognized that the current situation falls under the same description of the situation S_0 of the previous section (Fig. 8): as in that example, the cat is aiming at the mouse, and the chase is going to start. However, now there is a big box near the mouse, and at a certain point, the mouse goes behind the box, disappearing from the sight of the cat (and from the cat's CS representation). The cat interprets this event as a consequence of a *disappears* action performed by the *mouse*.

Now, the cat generates a set of expected CS configurations $\{CS_2', CS_2'', \ldots, \}$ by means of an associative device (in the current implementation a recurrent neural network of the Elman type [10]). The operation of such an associative device is analogous to the mechanism that generates associative expectations in shape and action recognition [3, 5]. Each CS_2 in this set is the simulated *effect* of some possible (simple or complex) action a_i in a set $\{a_1, a_2, \ldots, \}$, where each a_i is geometrically compatible with the current situation S_1.

The robot chooses an action a_i according to some criteria; e.g., a_i is the action whose expected effects have the minimum distance in CS from the "goal", which, in this case, is to reach the mouse. Once that a_i has been chosen, the cat can execute it. Then it updates the current situation according to the new perceptions, and restart the mechanism of generation of expectations.

In the current example, the expectation of the cat is that the mouse is not really disappeared, but it is continuing to move behind the box. Then, the cat simulates in its conceptual space the action of going beyond the box, where it expects to find the *mouse* and to restart the chase.

These *associative* expectations are at the basis of a simple form of planning based on expectations in CS: perceived situations can "reactively" recall some expected effect of an action. This process of action selection has the effect of "situate" the robot in its environment, by firmly anchoring the actions choice to the robot perceptions.

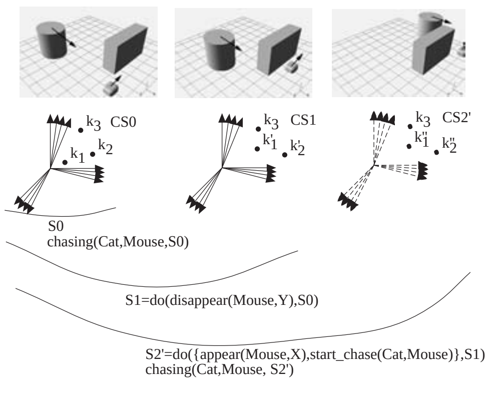

Figure 8. The cat figures out a possible behavior of the mouse and chooses a suitable action.

7 CONCLUSIONS

In this paper we suggest a possible way of anchoring symbols representing actions to representations in conceptual spaces. We choose the situation calculus as the symbolic formalism, because it is a powerful, well understood and widespread formal tool. In addition, the situation calculus is particularly well suited to be mapped on dynamic conceptual spaces. A conceptual interpretation of the situation calculus would be interesting in itself. Indeed, it could be considered complementary with respect to traditional, model theoretic interpretations for logic oriented representation languages.

Model theoretic semantics (in its different versions: purely Tarskian for extensional languages, possible worlds semantics for modal logic, preferential semantics for non monotonic formalisms, and so on) has been developed with the aim of accounting for certain metatheoretical properties of logical formalisms (such as logical consequence, validity, correctness, completeness, and son on). However, it is of no help in establishing how symbolic representations are anchored to their referents.

In addition, the model theoretic approach to semantics is "ontologically uniform", in the sense that it hides the ontological differences between entities denoted by expressions belonging to the same syntactic type. For example, all the individual terms of a logical language are mapped onto elements of the domain, no matter of the deep ontological variety that may exist between the objects that constitute their intended interpretation. Consider the situation calculus. According to its usual syntax, situations, actions and objects are all represented as first order individual terms; therefore, they are all mapped on elements of the domain. This does not constitute a problem given the above mentioned purposes of model theoretic semantics. However, it becomes a serious drawback if the aim is that of anchoring symbols to their referents through the sensory activities of an agent.

Typed versions of model theoretic semantics are of little help from this point of view. Instead of a single domain, the interpretation of the language is given in the terms of different sets; each of them, however, is still a collection of unstructured set-theoretical individuals.

In this perspective, the anchoring of actions symbols in terms of conceptual spaces could offer a kind of interpretation that does not constitute only a metatheoretic device allowing to single out certain properties of the symbolic formalism; rather, it is assumed to offer a further level of representation that is, in some sense, closer to the data coming from sensors, and that, for this reason, can help in anchoring the symbols to the external world.

Moreover, the proposed anchoring operation accounts for the ontological differences between the entities denoted by symbols belonging to the same syntactic category: situations are mapped on sequences of CS configurations, (instantaneous)actions on changes in the CS, on so on. This would result in a richer and finer grained model, that stores information that is not explicitly represented at the symbolic level, and that therefore can offer a further source of "analog" inferences, offering at the same time a link between deliberative inferential processes, and forms of inference closer to the lower levels of the robot architecture as the behaviors that actively controls the robot sensors [8].

REFERENCES

[1] R. Arkin, *Behavior-Based Robotics*, MIT Press, Cambridge, MA, 1998.

[2] I. Biederman, 'Human image understanding: recent research and a theory', *Computer Vision, Graphics and Image Processing*, **32**, 29–73, (1985).

[3] A. Chella, M. Frixione, and S. Gaglio, 'A cognitive architecture for artificial vision', *Artif. Intell.*, **89**, 73–111, (1997).

[4] A. Chella, M. Frixione, and S. Gaglio, 'An architecture for autonomous agents exploiting conceptual representations', *Robotics and Autonomous Systems*, **25**(3-4), 231–240, (1998).

[5] A. Chella, M. Frixione, and S. Gaglio, 'Understanding dynamic scenes', *Artif. Intell.*, **123**, 89–132, (2000).

[6] S. Coradeschi and A. Saffiotti (eds.), *AAAI Fall Symposium on Anchoring Symbols to Sensor Data in Single and Multiple Robot Systems*, AAAI Press, Menlo Park, CA, 2001.

[7] S. Coradeschi and A. Saffiotti, 'Perceptual anchoring of symbols for action', in *Proc. IJCAI-01*, pp. 407–412, Seattle, WA, (2001).

[8] J.L. Crowley and H.I. Christensen, *Vision as process*, Springer-Verlag, Berlin, 1995.

[9] D. Dennett, *Brainstorms: Philosophical Essays on Mind and Psychology*, MIT Press, Cambridge, MA, 1978.

[10] J.L. Elman, 'Finding structure in time', *Cognitive Sciences*, **14**, 179–211, (1990).

[11] P. Gärdenfors, 'Cued and detached representations in animal cognition', *Behavioural Processes*, **35**, 263–273, (1996).

[12] P. Gärdenfors, *Conceptual Spaces*, MIT Press, Cambridge, MA, 2000.

[13] A. Jaklič, A. Leonardis, and F. Solina, *Segmentation and Recovery of Superquadrics*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000.

[14] D. Marr and L. Vaina, 'Representation and recognition of the movements of shapes', *Proc. R. Soc. Lond. B*, **214**, 501–524, (1982).

[15] R. Reiter, *Knowledge in Action: Logical Foundations for Describing and Implementing Dynamical Systems*, MIT Press, Cambridge, MA, 2001.

[16] M. Shanahan, *Solving the frame problem*, MIT Press, Cambridge, MA, 1997.