Interpreting Symbols on Conceptual Spaces: the Case of Dynamic Scenarios

Antonio Chella†, Marcello Frixione‡ and **Salvatore Gaglio†**
†Dip. di Ingegneria Automatica e Informatica, Univ. of Palermo and CERE-CNR, Palermo, Italy
‡Dip. di Scienze della Comunicazione, Univ. of Salerno, Italy

Abstract

In (Chella, Frixione, & Gaglio 1997; 2000) we proposed a framework for the representation of visual knowledge, with particular attention to the analysis and the representation of scenes with moving objects and people. One of our aims is a principled integration of the models developed within the artificial vision community with the propositional knowledge representation systems developed within symbolic AI. In this paper we show how the approach we adopted fits well with the representational choices underlying one of the most popular symbolic formalisms used in cognitive robotics, namely the *situation calculus*.

Our model of visual perception: An overall view

In (Chella, Frixione, & Gaglio 1997; 2000) we proposed a theoretical framework for the representation of knowledge extracted from visual data. One of our aims is a principled integration of the approaches developed within the artificial vision community, and the propositional systems developed within symbolic knowledge representation in AI. It is our assumption that such an integration requires the introduction of a *conceptual level* of representation that is in some way intermediate between the processing of visual data and declarative, propositional representations. In this paper we argue that the conceptual representation we adopted for representing motion is compatible with one of the most influential symbolic formalisms used in cognitive robotics, namely the *situation calculus* (Reiter 2001). In the rest of this section we summarize the main assumptions underlying our model. Next section is devoted to a synthetic description of our conceptual level representation of motion. The third section shows how the situation calculus can be mapped on the conceptual representation we adopted. The fourth section describes a way to translate the symbolic representation we adopted in (Chella, Frixione, & Gaglio 2000) in the formalism of the situation calculus. A short conclusion follows.

On the one hand, the computer vision community approached the problem of the representation of dynamic scenes mainly in terms of the construction of 3D models, and of the recovery of suitable motion parameters, possibly in the presence of noise and occlusions. On the other hand, the KR community developed rich and expressive systems for representation of time, of actions and, in general, of dynamic situations.

Nevertheless, these two traditions evolved separately and concentrated on different kinds of problems. The computer vision researchers implicitly assumed that the problem of visual representation ends with the 3D reconstruction of moving scenes. The KR tradition in classical, symbolic AI usually underestimated the problem of grounding symbolic representations in the data coming from sensors.

It is our opinion that this state of affairs constitutes a strong limitation for the development of artificial autonomous systems (typically, robotic systems). It is certainly true that several aspects of the interaction between perception and behaviour in artificial agents can be faced in a rather reactive fashion, with perception more or less directly coupled to action, without the mediation of complex internal representations. However, we maintain that this is not sufficient to model all the relevant types of behaviour. For many kinds of complex tasks, a more flexible mediation between perception and action is required. In such cases the coupling between perception and action is likely to be "knowledge driven", and the role of explicit forms of representation is crucial. Our model can be intended as a suggestion to provide artificial agents with the latter of the above-mentioned functionalities.

The existing attempts to integrate visual perception with propositional KR are mostly oriented towards natural language interpretation, with particular emphasis on the aspects of man-machine interaction. They face only in a marginal way the general aspects of representation of knowledge. In addition, the existing proposals often concern specific domains of application, such as car traffic (Nagel 1994; Neumann 1989), biomedical images (Tsotsos *et al.* 1980; Tsotsos 1985), sport scenarios (Herzog & Wazinski 1994; Siskind 1994 5), simple assembly tasks (Kuniyoshi & Inoue 1993), visual surveillance (Buxton & Gong 1995).

We assume that a principled integration of the approaches of artificial vision and of symbolic KR requires the introduction of a missing link between these two kinds of representation. In our approach, the role of such a link is played by the notion of *conceptual space* (Gärdenfors 2000). A con-

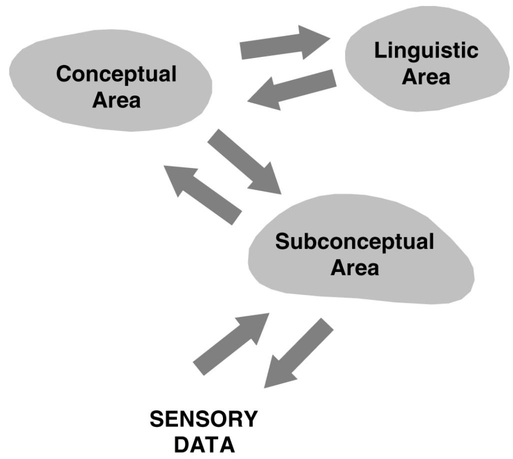

Figure 1: The three areas of representation, and the relations among them.

ceptual space (CS) is a representation where information is characterized in terms of a metric space defined by a number of *cognitive* dimensions. These dimensions are independent from any specific language of representation. According to Gärdenfors, 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, our architecture is organised in three *computational areas*. Figure 1 schematically shows the relations among them. The *subconceptual* area is concerned with the low level processing of perceptual data coming from the sensors. The term subconceptual suggests that here information is not yet organised in terms of conceptual structures and categories. In this perspective, our subconceptual area includes a 3D model of the perceived scenes. Indeed, even if such a kind of representation cannot be considered "low level" from the point of view of artificial vision, in our perspective it still remains below the level of conceptual categorisation.

In the *linguistic* area, representation and processing are based on a propositional KR formalism (a logic-oriented representation language). In the *conceptual* area, the data coming from the subconceptual area are organised in conceptual categories, which are still independent from any linguistic characterisation. The symbols in the linguistic area are grounded on sensory data by mapping them on the representations in the conceptual area.

In a nutshell, the operation of our model can be summarised as follows: it takes in input a set of images corresponding to subsequent phases of the evolution of a certain dynamic scene, and it produces in output a declarative description of the scene, formulated as a set of assertions written in a symbolic representation language. The generation of such a description is driven both by the "a priori", symbolic knowledge initially stored within the linguistic area, and by the information learned by a neural network component. The knowledge in the linguistic area plays a central role in determining the expectations of the system, that in their turn drive the scanning of the data coming from the sensors. Which high level information the system extracts from the sensory data (or, in other worlds, the way in which the system interprets the perceived scenes) crucially depends on a top down process partially directed by linguistic knowledge. The details of this process are described in (Chella, Frixione, & Gaglio 1997).

The purpose of this paper is to show that the conceptual representation we adopted for dynamic scenes (and that is described in details in (Chella, Frixione, & Gaglio 2000)) is consistent with the representational choices underlying one of the most popular symbolic formalisms used in cognitive robotics, namely the *situation calculus* (Reiter 2001). In this way, our dynamic conceptual spaces could offer a conceptual interpretation for the situation calculus, which could help in anchoring propositional representations to the perceptual activities of a robotic system.

Conceptual spaces for representing motion

Representations in the conceptual area are couched in terms of *conceptual spaces* (Gärdenfors 2000). Conceptual spaces provide a principled way for relating high level, linguistic formalisms on the one hand, with low level, unstructured representation of data on the other. 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 presuppose different conceptual spaces, and different conceptual spaces can be characterised by different dimensions. Examples of dimensions of a CS could be colour, pitch, mass, spatial co-ordinates, and so on. In some cases dimensions are strictly related to sensory data; in other cases they are more abstract in nature. Anyway, dimensions do not depend on any specific linguistic description. In this sense, conceptual spaces come before any symbolic-propositional characterisation of cognitive phenomena. In particular, in this note, we take into account a conceptual space devoted to the representation of the motion of geometric shapes.

We use the term *knoxel* to denote a point in a conceptual space. A knoxel is an epistemologically primitive element at the considered level of analysis. For example, in (Chella, Frixione, & Gaglio 1997) we assumed that, in the case of static scenes, a knoxel coincides with a 3D primitive shape, characterised according to some constructive solid geometry (CSG) schema. In particular, we adopted superquadrics (Pentland 1986; Solina & Bajcsy 1990) as a suitable CSG schema. However, we do not assume that this choice is mandatory for our proposal. Our approach could be reformulated by adopting different models of 3D representation.

The entities represented in the linguistic area usually do not correspond to single knoxels. We assume that *complex entities* correspond to sets of knoxels. For example, in the case of a static scene, a complex shape corresponds to the set of knoxels of its simple constituents.

In order to account for the perception of dynamic scenes, we choose to adopt an intrinsically *dynamic conceptual space*. It has been hypothesised that simple motions are categorised in their wholeness, and not as sequences of static frames. According to this hypothesis, we define a dynamic conceptual space in such a way that every knoxel corresponds to a simple motion of a 3D primitive. In other words, we assume that simple motions of geometrically primitive shapes are our perceptual primitives for motion perception. As we choose superquadrics as geometric primitives, a knoxel in our dynamic conceptual space correspond to a simple motion of a superquadric.

Of course, the decision of which kind of motion can be considered "simple" is not straightforward, and is strictly related to the problem of motion segmentation. (Marr & Vaina 1982) adopted the term *motion segment* to indicate such simple movements. According to their SMS (State-Motion-State) schema, a simple motion is characterised by the interval between two subsequent overall rest states.

Such rest states may be instantaneous. Consider a person moving an arm up and down. According to Marr and Vaina's proposal, the upward trajectory of the forearm can be considered a simple motion, that is represented in CS by a single knoxel, say $\mathbf{k_a}$. When the arm reaches its vertical position, an instantaneous rest state occurs. The second part of the trajectory of the forearm is another simple motion that corresponds to a second knoxel, say $\mathbf{k'_a}$. The same holds for the upper arm: the first part of its trajectory corresponds to a certain knoxel $\mathbf{k_b}$; the second part (after the rest state) corresponds to a further knoxel $\mathbf{k'_b}$.

In intrinsically dynamic conceptual spaces, a knoxel $\mathbf{k}$ corresponds to a *generalised* simple motion of a 3D primitive shape. By *generalised* we mean that the motion can be decomposed in a set of components x_i, each of them associated with a degree of freedom of the moving primitive shape. In other words, we have:

$$\mathbf{k} = [x_1, x_2, \ldots, x_n]$$

where n is the number of degrees of freedom of the moving superquadric. In this way, changes in shape and size are also taken into account. Parameters of superquadrics are given with respect to an absolute reference system. As a consequence, also motion properties of knoxels is described in absolute terms.

In turn, each motion x_i corresponding to the i-th degree of freedom can be viewed as the result of the superimposition of a set of elementary motions f^i_j:

$$x_i = \sum_j X^i_j f^i_j$$

In this way, it is possible to define a set of basis functions f^i_j, in terms of which any simple motion can be expressed. Such functions can be associated to the axes of the dynamic conceptual space as its dimensions. In this way, the dynamic CS results in a functional space. The theory of function approximation offers different possibilities for the choice of basic motions: trigonometric functions, polynomial functions, wavelets, and so on.

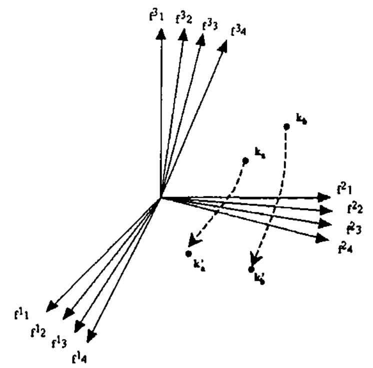

Figure 2: A dynamic conceptual space.

For our purposes, we are not interested in the representation of any possible motion, but only in a compact description of perceptually relevant kinds of motion. In the domain we are facing here, 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 well-known Discrete Fourier Transform (DFT), see (Oppenheim & Shafer 1989) for an in-depth description of the topic. According to this approach, given a generic parameter of a superquadric, e.g., a_x, let be $a_x(t)$ the function that returns for each instant t the corresponding value of a_x. This function can be viewed as a superimposition of a discrete number of trigonometric functions and it can then be represented by a point in a discrete functional space, whose basis functions are the trigonometric functions. By a suitable composition of the time functions of all superquadric parameters, the overall motion of the superquadric is represented in its turn by a point in a discrete functional space. We adopt the resulting functional space as the conceptual space for the representation of dynamic scenes.

The choice of DFT for motion representation seems promising from out point of view also because Fourier analysis has been proposed as a psychologically plausible model for representation of human and animal motion (Gallistel 1980; Johansson 1973). See (Chella, Frixione, & Gaglio 2000) for greater details.

Figure 2 is an evocative representation of a dynamic conceptual space. 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.

In this way, each knoxel in a dynamic conceptual space represents a *simple motion*, i.e. the motion of a simple shape occurring within the interval between two subsequent rest states. We call *composite simple motion* a 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 knoxels corresponding to the motions of its components. For example, the first part of the trajectory of the whole arm shown in figure 2 is represented as a composite motion made up by the knoxels $\mathbf{k_a}$ (the motion of the forearm) and $\mathbf{k_b}$ (the motion of the upper arm). 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. We assume that the configurations of knoxels within a single structured process are separated by instantaneous changes. In the transition between two subsequent different configurations, there is a change of at least one of the knoxels in the CS. We call a "scattering" such a transition from one knoxel to another. This corresponds to a discontinuity in time, and is associated with an instantaneous event.

In the example of the moving arm, a scattering occurs when the arm has reached its vertical position, and begins to move downwards. In a CS representation this amount to say that knoxel $\mathbf{k_a}$ (i.e., the upward motion of the forearm) is replaced by knoxel $\mathbf{k'_a}$, and knoxel $\mathbf{k_b}$ is replaced by $\mathbf{k'_b}$. In fig. 2 such scattering is graphically suggested by the dotted arrows connecting respectively $\mathbf{k_a}$ to $\mathbf{k'_a}$, and $\mathbf{k_b}$ to $\mathbf{k'_b}$.

Mapping situation calculus on conceptual spaces

In (Chella, Frixione, & Gaglio 2000) the formalism adopted for the linguistic area was a KL-ONE like semantic net, equivalent to a subset of the first order predicate language. In this note we propose the adoption of the *situation calculus* as the formalism for the linguistic area. Indeed, the kind of representation it presupposes is in many respects homogeneous to the conceptual representation described in the previous section. In this section we suggest how a representation in terms of the situation calculus could be mapped on the conceptual representation presented above.

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. It was primarily developed by (McCarty 1968); for an up to date and exhaustive overview see (Reiter 2001).

The basic idea behind the situation calculus is that the evolution of a state of affairs can be modeled in terms of a sequence of situations. The world changes when some *action* is performed. So, given a certain situation s_1, performing a certain action a will result in a new situation s_2. Actions are the sole sources of change of the word: if the situation of the word changes from, say, s_i to s_j, then some action has been performed.

The situation calculus is formalised 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: $do(a, s)$ denotes the new situation obtained by performing the action a in the situation s.

Actions occur in time. According to (Reiter 1996), we introduce a one argument symbol function $time$. If a is an action, $time(a)$ is a real number corresponding to the occurrence time of a.

Classes of actions can be represented as functions. For example, the one argument function symbol $pick\_up(x)$ could be assumed to denote the class of the actions consisting in picking up an object x. Given a first order term o denoting a specific object, the term $pick\_up(o)$ denotes the specific action consisting in picking up o.

As a state of affairs evolves, it can happen that properties and relations change their values. In the situation calculus, properties and relations that can change their truth value from one situation to another are called (relational) *fluents*. An example of fluent could be the property of being red: it can happen that it is true that a certain object is red in a certain situation, and it becomes false in another. Fluents are denoted by predicate symbols that take a situation as their last argument. For example, the fluent corresponding to the property of being red can be represented as a two place relation $red(x, s)$, where $red(o, s_1)$ is true if the object o is red in the situation s_1.

Different sequences of actions lead to different situations. In other words, it can never be the case that performing some action starting form different situations can result in the same situation. If two situations derive from different situations, they are in their turn different, in spite of their similarity. In order to account for the fact that two different situations can be indistinguishable from the point of view of the properties that hold in them, the notion of *state* is introduced. Consider two situations s_1 and s_2; if they satisfy the same fluents, then we say that s_1 and s_2 correspond to the same state. That is to say, the state of a situation is the set of the fluents that are true in it.

In the ordinary discourse, there are lot of actions that have a temporal duration. For example, the action of walking from certain point in space to another takes some time. In the situation calculus all actions in the strict sense are assumed to be instantaneous. Actions that have a duration are represented as processes, that are initiated and are terminated by instantaneous actions (see (Pinto 1994) and Chap. 7 of (Reiter 2001)). Suppose that we want to represent the action of moving an object x from point y to point z. We have to assume that moving x from y to z is a process, that is initiated by an instantaneous action, say $start\_move(x, y, z)$, and is terminated by another instantaneous action, say $end\_move(x, y, z)$. In the formalism of the situation calculus, processes correspond to relational fluents. For example, the process of moving x from y to z corresponds to a fluent, say $moving(x, y, z, s)$. A formula like $moving(o, p_1, p_2, s_1)$ means that in situation s_1 the object o is moving from position p_1 to position p_2.

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 conceptual space CS corresponds to an (instantaneous) *action*. A knoxel corresponds to a particular *process*. A configuration of knoxels in CS corresponds to a *state*.

Please, note that, according to the situation calculus terminology we adopt here, a state (in the technical sense) does not necessarily correspond to a static configuration of entities. Rather, as we said in the previous paragraph, a state is defined as the set of fluents that are true in a given situation, and we represent processes (i.e., actions that have a duration) as a particular kind of relational fluents. Therefore, a state can correspond to a set of processes.

Note also that a configuration of knoxels cannot correspond to a situation, because, at least in principle, the same configuration of knoxels could be reached starting from different configurations, and it can never happen that the same situation (in the technical sense) can be obtained starting from different from different states of affairs.

As an example, consider a scenario in which a certain object o moves from position p_1 to position p_2, and then rests in p_2. When the motion of o from p_1 towards p_2 starts, a scattering occurs in the conceptual space CS, and a certain knoxel, say k_1, becomes active in it. Such a scattering corresponds to an (instantaneous) action that could be represented by a term like $start\_move(o, p_1, p_2)$. The knoxel k_1 corresponds in CS to the process consisting in the motion of the object o. During all the time in which o remains in such a motion state, the CS remains unchanged (provided that nothing else is happening in the considered scenario), and k_1 continue to be active in it. In the meanwhile, the fluent $moving(o, p_1, p_2, s)$ remains true. When o's motion ends, a further scattering occurs, k_1 disappears, and a new knoxel k_2 becomes active. This second scattering corresponds to an instantaneous action $end\_moving(o, p_1, p_2)$. The knoxel k_2 corresponds to o's rest. During the time in which k_2 is active in CS, the fluent $staying(o, p_2, s_1)$ is 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. This is tantamount to say that several instantaneous actions occur concurrently. For example, according to our terminology (as it has been established in the preceding section), a composite simple motion is a 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 knoxels corresponding to the motions of its components. For example, the trajectory of a whole arm moving upward (the example in the previous section) is represented as a composite motion made up by the knoxels $\mathbf{k_a}$ (the upward motion of the forearm) and $\mathbf{k_b}$(the upward motion of the upper arm).

Suppose that we want to represent such a motion within the situation calculus. According to what stated before, moving an arm is represented as a process, that is started by a certain action, say $start\_move\_arm$, and that is terminated by another action, say end_move_arm. (For sake of simplicity, no arguments of such actions - e.g. the agent, the starting position, the final position - are taken into account here). The process of moving the arm is represented as a fluent $moving\_arm(s)$, that is true if in situation s the arm is moving. The scattering in CS corresponding to both $start\_move\_arm$ and end_move_arm involves two knoxels, namely $\mathbf{k_a}$ and $\mathbf{k_b}$, that correspond respectively to the motion of the forearm and to the motion of the upper arm. Consider for example $start\_move\_arm$. It is composed by two concurrent actions that could be named $start\_move\_forearm$ and $start\_move\_upper\_arm$, both corresponding to the scattering of one knoxel in CS (resp. $\mathbf{k_a}$ and $\mathbf{k_b}$).

Extensions of the situation calculus that allows for a treatment of concurrency have been proposed in the literature (Reiter 1996; 2001; Pinto 1994). (Pinto 1994) adds to the language of the situation calculus a two argument function $+$, that, given two actions as its arguments, produces an action as its result. In particular, if a_1 and a_2 are two actions, $a_1 + a_2$ denotes the action of performing a_1 and a_2 concurrently. According to this approach, an action is *primitive* if it is not the result of other actions performed concurrently. If a is a complex action such that $a = a_1 + a_2 + \ldots + a_n$, then, for each i such that $1 \leq i \leq n$, we say that $a_i \in a$.

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

Therefore, according to Pinto's notation, the representation of the arm motion example in the formalism of the (concurrent) situation calculus involves four primitive actions:

$$\begin{array}{l} start\_move\_forearm \\ start\_move\_upper\_arm \\ end\_move\_forearm \\ end\_move\_upper\_arm \end{array}$$

and two non primitive actions:

$$\begin{array}{rcl} start\_move\_arm & = & start\_move\_forearm + \\ & + & start\_move\_upper\_arm \\ end\_move\_arm & = & end\_move\_forearm + \\ & + & end\_move\_upper\_arm \end{array}$$

In addition, the three following fluents are needed:

$moving\_arm$ (a formula like this is true if in the CS configuration corresponding to the situation s both $\mathbf{k}_a$ and $\mathbf{k}_b$ are present);

$moving\_forearm$ (a formula like this is true if in the CS configuration corresponding to the situation s $\mathbf{k}_a$ is present);

$moving\_upper\_arm$ (a formula like this is true if in the CS configuration corresponding to the situation s $\mathbf{k}_b$ is present).

From a terminological taxonomy of processes to the situation calculus

In (Chella, Frixione, & Gaglio 1997; 2000), we adopted a linguistic representation based on a hybrid formalism in the KL-ONE tradition (on this kind of formalisms see (Brachman & Schmoltze 1985; Nebel 1990). These formalisms are

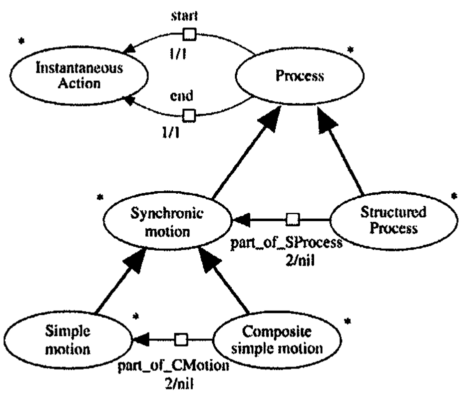

Figure 3: A fragment of the terminological KB describing some of the most general concepts.

hybrid in the sense that they are constituted by two different components: a *terminological component* for the description of concepts, and an *assertional component*, that stores information concerning a specific context. In the domain of dynamic scene representation, the terminological component contains the description of significant concepts such as various types of motions, of processes and of actions. The assertional component stores the assertions describing specific perceived dynamic situations.

Fig. 3 shows a fragment of the terminological knowledge base adopted in (Chella, Frixione, & Gaglio 2000), describing the most general concepts representing motions, processes and actions. The graphical notation is similar to that of Brachman and Schmoltze (Brachman & Schmoltze 1985). In this section, we show how a taxonomy like this can be translated in the formalism of the situation calculus.

Intuitively, *Processes* are the most generic motions corresponding to a temporal evolution of the knoxels in the CS. The beginning and the end of a *Process* is determined by *Instantaneous_Actions*, that have no temporal duration and correspond to a scattering of knoxels in the CS. *Instantaneous_Actions* are analogous to the *actions* of the situation calculus. Therefore, the concept *Instantaneous_Action* corresponds to a one place predicate, that is true of the individuals in the domain that correspond to actions.

As we anticipated before, *Processes* are analogous to *fluents*. In order to keep a greater correspondence with the semantic network representation, we adopt the choice of reifying processes. That is to say, we assume that in our domain individual entities exist, which correspond to specific instances of processes. Therefore, a certain kind of process P_1 (i.e., a subconcept of the *Process* concept of the fig. 3) is represented as a two place relational fluent $P_1(x, y)$. An atomic formula $P_1(p, s)$ means that the individual p is a process of kind P_1, and that it holds in the situation s.

Processes may be arranged according to different temporal relations: they may be consecutive, may overlap, and so on.

A *Synchronic_motion* is a *Process* involving the motion of one or more objects, that corresponds to one arrangement of knoxels simultaneously occurring in CS. In other words, within a synchronic motion no scattering occurs. A *Simple_motion* is a synchronic motion which has no parts: it corresponds to a single knoxel in CS. A *Composite_simple_motion* is a *Synchronic_motion* with at least two parts that are simple motions occurring simultaneously. That is to say, all the parts of a composite simple motion start and end at the same time. Therefore, if the simple motions $sm_1 \ldots sm_n$ are the parts of a certain composite simple motion csm, and $a_1 \ldots a_n$, a are the instantaneous actions that initiate, respectively, $sm_1 \ldots sm_n$ and csm, then we have that $a = a_1 + \ldots + a_n$. In other words:

$$\forall x_1, x_2, y_1, y_2, s(Composite\_simple\_motion(x_1, s) \wedge \\ \wedge part\_of\_CMotion(x_1, x_2) \wedge start(x_1, y_1) \wedge \\ \wedge start(x_2, y_2) \rightarrow y_2 \in y_1)$$

A similar constraint holds for the actions terminating a composite simple motion.

A *Structured_process* is a motion involving a temporal evolution (i.e., a scattering) in CS; a *Structured_process* has at least two parts that are *Synchronic_motions* not occurring at the same time. In the case of structured processes, the following condition holds:

$$\forall x_1, x_2, y_1, y_2, s(Structured\_Process(x_1, s) \wedge \\ \wedge part\_of\_Sprocess(x_1, x_2) \wedge start(x_1, y_1) \wedge \\ \wedge start(x_2, y_2) \rightarrow time(y_1) \leq time(y_2))$$

That is to say, if x_1 is s a structured process that is initiated by the action y_1, and x_2 is a part of x_1 that is initiated by the action y_2, then y_2 must temporally follow or coincide with y_1.

A symmetric condition constraints the end of structured processes and of their parts:

$$\forall x_1, x_2, y_1, y_2, s(Structured\_Process(x_1, s) \wedge \\ \wedge part\_of\_Sprocess(x_1, x_2) \wedge end(x_1, y_1) \wedge \\ \wedge end(x_2, y_2) \rightarrow time(y_1) \geq time(y_2))$$

Some conclusions

In the above section we suggested a possible interpretation of the language of the situation calculus in terms of conceptual spaces. In this way the situation calculus could be chosen as the linguistic area formalism for our model, with the advantage of adopting a powerful, well understood and widespread formal tool. Besides this, we maintain that 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.

In this perspective, it is our opinion that an interpretation of symbols in terms of conceptual spaces of the form sketched in the above pages 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.
- A kind of interpretation that accounts for the ontological differences between the entities denoted by symbols belonging to the same syntactic category. 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 that are closer to the lower levels of the cognitive architecture (reactive behaviours, and so on).

Acknowledgements

This work has been partially supported by MURST "Progetto Cofinanziato CERTAMEN".

References

Brachman, R., and Schmoltze, J. 1985. An overview of the KL-ONE knowledge representation system. *Cognitive Science* 9(2):171–216.

Buxton, H., and Gong, S. 1995. Visual surveillance in a dynamic and uncertain world. *Artif. Intell.* 78:371–405.

Chella, A.; Frixione, M.; and Gaglio, S. 1997. A cognitive architecture for artificial vision. *Artif. Intell.* 89:73–111.

Chella, A.; Frixione, M.; and Gaglio, S. 2000. Understanding dynamic scenes. *Artif. Intell.* 123:89–132.

Gallistel, C. 1980. *The Organization of Action: A New Syntesis.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Gärdenfors, P. 2000. *Conceptual Spaces.* Cambridge, MA: MIT Press, Bradford Books.

Herzog, G., and Wazinski, P. 1994. VIsual TRAnslator: Linking parceptions and natural language descriptions. *Artif. Intell. Review* 8(2-3):175–187.

Johansson, G. 1973. Visual perception of biological motion and a model for its analysis. *Perception and Psychophysics* 14:201–211.

Kuniyoshi, Y., and Inoue, H. 1993. Qualitative recognition of ongoing human action sequences. In *Proc. 13th IJCAI*, 1600–1609.

Marr, D., and Vaina, L. 1982. Representation and recognition of the movements of shapes. *Proc. R. Soc. Lond. B* 214:501–524.

McCarty, J. 1968. Situations, actions and causal laws. In Minsky, M., ed., *Semantic Information Processing.* Cambridge, MA: MIT Press. 410–417.

Nagel, H. 1994. A vision of "vision and language" comprises action: An example from road traffic. *Artif. Intell. Review* 8(2-3):189–214.

Nebel, B. 1990. *Reasoning and Revision in Hybrid Representation Systems*, volume 422 of *Lecture Notes in Artificial Intelligence.* Berlin: Springer-Verlag.

Neumann, B. 1989. Natural language description of time-varying scenes. In Waltz, D., ed., *Semantic Structures.* Hillsdale, NJ: Lawrence Erlbaum.

Oppenheim, A., and Shafer, R. 1989. *Discrete-Time Signal Processing.* Englewood Cliffs, N.J.: Prentice Hall, Inc.

Pentland, A. 1986. Perceptual organization and the representation of natural form. *Artif. Intell.* 28:293–331.

Pinto, J. 1994. Temporal reasoning in the situation calculus. Technical report, Dept. of Computer Science, Univ. of Toronto, Toronto, CA.

Reiter, R. 1996. Natural actions, concurrency and continuous time in the situation calculus. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Fifth International Conference (KR'96).*

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

Siskind, J. 1994-5. Grounding language in perception. *Artif. Intell. Review* 8(5-6):371–391.

Solina, F., and Bajcsy, R. 1990. Recovery of parametric models from range images: The case for superquadrics with global deformations. *IEEE Trans. Patt. Anal. Mach. Intell.* 12(2):131–146.

Tsotsos, J.; Mylopoulos, J.; Covvey, H.; and Zucker, S. 1980. A framework for visual motion understanding. *IEEE Trans. Patt. Anal. Mach. Intell.* 2(6):563–573.

Tsotsos, J. 1985. Knowledge organisation and its role in representation and interpretation for time-varying data: the ALVEN system. *Computational Intelligence* 1:16–32.