ELSEVIER

Expert Systems with Applications xx (0000) xxx–xxx

Expert Systems with Applications

www.elsevier.com/locate/eswa

Knowledge modelling for the motion detection task: the algorithmic lateral inhibition method

José Mira[a], Ana E. Delgado[a], Antonio Fernández-Caballero[b,*], Miguel A. Fernández[b]

[a]Departamento de Inteligencia Artificial, Facultad de Ciencias and E.T.S.I. Informática, Universidad Nacional de Educación a Distancia, 28040 Madrid, Spain

[b]Departamento de Informática, Escuela Politécnica Superior, Universidad de Castilla-La Mancha, 02071 Albacete, Spain

Received 14 November 2003; revised 14 November 2003; accepted 19 January 2004

Abstract

In this article knowledge modelling at the knowledge level for the task of moving objects detection in image sequences is introduced. Three items have been the focus of the approach: (1) the convenience of knowledge modelling of tasks and methods in terms of a library of reusable components and in advance to the phase of operationalization of the primitive inferences; (2) the potential utility of looking for inspiration in biology; (3) the convenience of using these biologically inspired problem-solving methods (PSMs) to solve motion detection tasks.

After studying a summary of the methods used to solve the motion detection task, the moving targets in indefinite sequences of images detection task is approached by means of the algorithmic lateral inhibition (ALI) PSM. The task is decomposed in four subtasks: (a) thresholded segmentation; (b) motion detection; (c) silhouettes parts obtaining; and (d) moving objects silhouettes fusion. For each one of these subtasks, first, the inferential scheme is obtained and then each one of the inferences is operationalized. Finally, some experimental results are presented along with comments on the potential value of our approach.

Keywords: Knowledge modeling; Problem solving methods; Motion detection; Lateral inhibition

1. Knowledge modelling at the knowledge level

A central problem of applied artificial intelligence is to construct models of tasks and problem solving methods (PSMs) at the knowledge level and in the domain of the external observer (Ford, Bradshaw, Adams-Webber, & Agnew, 1993; Maturana, 1975; Mira & Delgado, 1987, 2003; Varela, 1979). Then we have to reduce these models of expertise from the domain of human experts to the domain of formal tools, both at the knowledge level. That is to say we have to go from natural language description of the task and the PSM used to solve this task, to a *formalism transformation* of this conceptual model in terms of formal tools (rules, neural nets). Finally a new rewriting of the formal model is made in terms of the primitives of a programming language to produce the program.

The usual approach to modelling at the knowledge level and to facilitating the subsequent model reduction of the model to the program has been to develop libraries of PSMs and *domain ontologies*. We talk about a reduction of the real model as information always remains at knowledge level (in the sense of Newell) and in the domain of the observer (in the sense of Maturana (1975), Mira and Delgado (1987) and Varela (1979)). Relevant examples of this approach include the CommonKADS methodology (Breuker & van de Velde, 1994; Eriksson, Shahar, Tu, Puerta, & Musen, 1995; Schreiber et al., 2001), the formal framework UPML (Fensel, Benjamins, Motta, & Wielinga, 1999), and the general-purpose framework Protégé-II (Eriksson et al., 1995; Mira, Alvarez, & Martinez, 2000). In this methodological context, the knowledge modelling process starts at the knowledge level and follows the next steps (Mira, Herero, & Delgado, 1998; Mira et al., 2000)

1. Describe in natural language the task you try to model and code, and disregard the terms that are not causal in the reasoning process.

* Corresponding author. Tel.: +34-967-599200; fax: +34-967-599224.
E-mail addresses: caballer@info-ab.uclm.es (A. Fernández-Caballero), jmira@dia.uned.es (J. Mira), adelgado@uned.es (A.E. Delgado), miki@info-ab.uclm.es (M.A. Fernández).

0957-4174/$ - see front matter © 2004 Published by Elsevier Ltd.
doi:10.1016/j.eswa.2004.01.006

ESWA 1181—10/2/2004—00:09—ADMINISTRATOR—94055— MODEL 5

2. Identify the entities of the domain knowledge. These entities play the same role as physical magnitudes in an analytical model. They represent separate *concepts* that the human expert considers necessary and sufficient to describe his/her knowledge concerning the solution of the specific task under consideration.
3. Identify the relations between these entities that appear explicitly or implicitly in the expert's description.
4. Search for inferential components of the reasoning, usually verbs (establish, refine, select, match, abstract), which are used by the human expert to describe his/her reasoning steps in natural language. These *inferences* are the components from which we will build the PSMs.
5. Describe, for each one of these inferential verbs, the *input* and *output roles* to be played by the domain entities.
6. Try to sketch the inferential circuit corresponding to the knowledge flow through the dynamic roles and the different inferences according to the sequence, concurrences, and loops that more closely represent the reasoning pattern followed by the expert. These reasoning patterns (PSMs) can sometimes be selected from a library of reusable components (Benjamins & Fensel, 1998; Breuker & van de Velde, 1994; Fensel, 1997; Schreiber et al., 2001) (*abstract-match-refine*, *establish-and-refine*, *propose-critique-modify*, *generate-and-test*, *cover-and-differentiate*), although additional knowledge is usually needed for adaptation of the PSM to the task (task–PSM bridge) and to the domain (PSM–domain bridge) (Taboada, Des, Mira, & Marin, 2001).

At the end of the last step we have

1. A set of *entities* and *relations* of the domain model.
2. A set of *inferences* with the corresponding input and output roles.
3. An *inferential circuit* connecting these inferences through dynamic roles.
4. A control *structure*.

That is to say, we have a conceptual model at the knowledge level to solve the task. The next step in the way to build the code is to make operational each one of these inferences (*abstract*, *select*, *classify*, *refine*). That is, to rewrite them in formal terms by selecting specific formal operators (symbolic rules, fuzzy rules, neural nets, Bayesian networks, and so on) for each one of the inferences. The criteria used in this selection process are always related with the balance between data and knowledge available for the specific inference under consideration. Also relevant is the sort of *knowledge* (precise, uncertain) and *data* (labelled, unlabelled) available.

We usually don't possess the whole knowledge to be able to only use knowledge-driven operators. Neither it is frequent to know nothing on the procedure used by human experts to solve that task, and then being forced to use date-driven methods. In real problems, most frequently the expert describes his method of solving the task in a *hybrid* way, with a *symbolic* part (rules) and a *connectionist* part (Fu & Fu, 1990; Hilario, Orsier, Rida, & Pellegrini, 1995; Sun & Alexandre, 1997). Conventionally, a method is said to be *symbolic* when it is essentially guided by knowledge which is made explicit in a declarative way and finishes being completely programmed. Alternatively, a method is called to be *connectionist* or neuronal if it possesses a modular fine grain architecture, with a local parametric function, and where an important part of the programming is substituted by a supervised or non-supervised learning mechanism. Essentially, a method is neuronal if it is data labelled. The idea of a *hybrid* system is used to describe those situations where not all data or knowledge necessary to solve the problem is available. Thus, the available knowledge may be firstly used to specify the initial skeletal model of a connectionist net and, afterwards, a supervised learning method to adjust the values of the parameters of this skeletal model is established.

In this work the concept of *hybrid* is used in the sense of the so called 'unified approach' (Hilario et al., 1995). That is to say, the structure of the connectionist net is maintained, while the calculation capacity of each node is augmented. This way there is a gap from the most usual model (weighted sum followed by sigmoid) to an inferential model that possesses the structure of a rule where the antecedent over the data field specified by the receptive field is evaluated. Next a look-up table (LUT) is used to select the most adequate action corresponding to each result of the evaluation of the antecedent of the rule. This is our approach in this paper for the task of silhouette obtaining of moving elements in a sequence of images.

2. The motion detection task

The global objective of the *task* is to obtain the silhouettes of all moving elements present in an indefinite sequence of images. This way, the task consists in observing, detecting, labelling and tracking the moving objects in the scene. These objects may be non-rigid and their detection is associated to the movement of any of the parts that compose them. This movement, captured from an indefinite sequence of frames, allows to gradually obtaining the silhouettes of the elements that offer any kind of motion. Fig. 1a shows one image of a satellite image. By taking in consideration motion detected in the proper image sequence, the silhouettes of all non-rigid moving objects present in the scene should be obtained. In the case of the present example, the optimal is given by the resulting image (Fig. 1b), where three different elements are detected. The problem faced is not limited to the observation, detection and tracking of a single non-rigid object in a scene, but

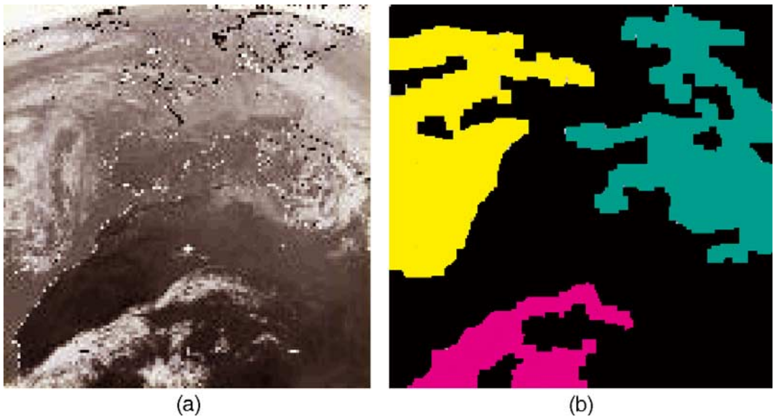

Fig. 1. (a) One meteorology satellite image; (b) result after 'motion detection' task.

rather it consists in discriminating all the objects that offer some kind of movement.

2.1. General motion detection

Motion detection in image sequences is applicable in multitude of fields—generally, where motion plays an important role in the definition of the problem of a given scene. In particular, there are cases of detection of elements with a certain velocity. Concrete cases can be found in traffic control, security, surveillance, and other similar fields. One of the most obvious applications in motion detection and analysis is possibly in the field of robotics (Horn, 1986). For autonomous robots, the visual movement is a source of rich information for sailing and route planning (Nair & Aggarwal, 1998; Wettergreen, Thomas, & Bualat, 1997). The techniques developed in robotics field are demonstrating their usefulness in more specific environments. Industrial arms, for example, can develop a great number of operations on objects passing on a conveyer belt (Lewis, Abdallah, & Dawson, 1993; Sternberg, 1985). Also, autonomous vehicles are able to follow the layout of a highway (Kuan, Phipps, & Hsueh, 1988).

In other motion detection applications, three-dimensional (3D) vision is not the main objective. Among these applications, there is the interpretation of images taken from a satellite or astronomical images, such as the analysis of the formation of clouds in weather prediction (Colet, Quinquis, & Boucher, 1992). Of a great interest are restoration and image enhancement (Irani & Peleg, 1993). Another example is noise elimination in old cinema movie (Vlachos & Thomas, 1996). Another area where motion detection is of a great importance resides in medical images, where it is used, for instance, to monitor motion patterns of the heart starting from MR images (Prince & McVeigh, 1992), or to improve and to interpret scanned ultrasound images (Quistgaard, 1997). Motion analysis is also finding a growing use in multimedia systems (Idris & Panchanathan, 1997). In the field of videotape data compression, motion information is used to exploit temporary redundancies in the data.

A high level approach that incorporates some of the 3D vision techniques previously mentioned, is the codification based on models, where a 3D geometric model is built in a limited scene. The model consists, for example, of the head and the shoulders of a person, and may be used for videoconferences. Once the model is known in the reception and transmission nodes, transmitting the coded motion data (Li, Rovainen, & Forchheimer, 1993) can animate it. As related, applications that benefit from estimation, analysis and tracking starting from motion detection are very diverse.

2.2. Segmentation from motion

Segmentation from motion is already a classical problem in computer assisted artificial vision. The most popular general methods of moving object extraction are based in (a) optic flow and (b) image differences. The first set of techniques in motion segmentation is based on the optic flow calculation. The velocity field is segmented to identify the different objects in movement in the image. There basically exist two approaches to calculate the disparity map between two frames. The continuity (or gradient based) approach uses the spatio-temporal variation according to the famous motion restriction equation described by Horn and Schunck (1981). This approach is completed with three complementary techniques, that is, a technique of local optimisation (Thompson & Barnard, 1981), a technique of global optimisation (Horn & Schunck, 1981) and an approach to the obtention of classes (Fennema & Thompson, 1979). The discrete (characteristics based) approach to calculate the optic flow consists on extracting those characteristics that correlate two consecutive frames. The second set of segmentation from motion techniques is based on image differences. Again, we are in response to two categories of image difference techniques. The difference between two images may be obtained from any frame of an image sequence and a reference frame. You may also calculate the difference between any couple of consecutive frames of an image sequence.

When dealing with non-rigid objects, the motion detection problem is much more complex. The different approaches to the problem differ basically in the way they model the shape and the movement, as well as in the adopted method of optimisation. The formulations may be continuous or discrete, deterministic or statistical, parametric or not. In general, we can affirm that great attention has been paid to techniques based on active contours (snakes) of non-rigid surfaces (Kass, Witkin, & Terzopoulos, 1988). The use of snakes based techniques in the context of the estimation of non-rigid motion is mainly interesting in objective tracking, whenever there is a precise prediction step (Bascle, Bouthemy, Deriche, & Meyer, 1994). An alternative method consists on the use of parametric 2D patterns, wrapping a global compact parameterisation to represent the shapes of interest (Yuille, 1992). Also parametric models of B-spline type under forced deformations have been tested (Bascle et al., 1994). These models appear to be more general and more robust than those previously mentioned. In the studies of the non-rigid motion, it is important to keep in mind as much the global deformations as the local ones. The statistical models like the Markov random field (MRF) model are very well adjusted to this purpose (Amit, Grenander, & Piccioni, 1991). The articulated movement is of a special interest in the analysis of human movement. The quantitative study of the human movement (facial movement, gestures, etc.) is useful in multitude of applications, including clinical rehabilitation, sports bio-mechanics, new man–machine

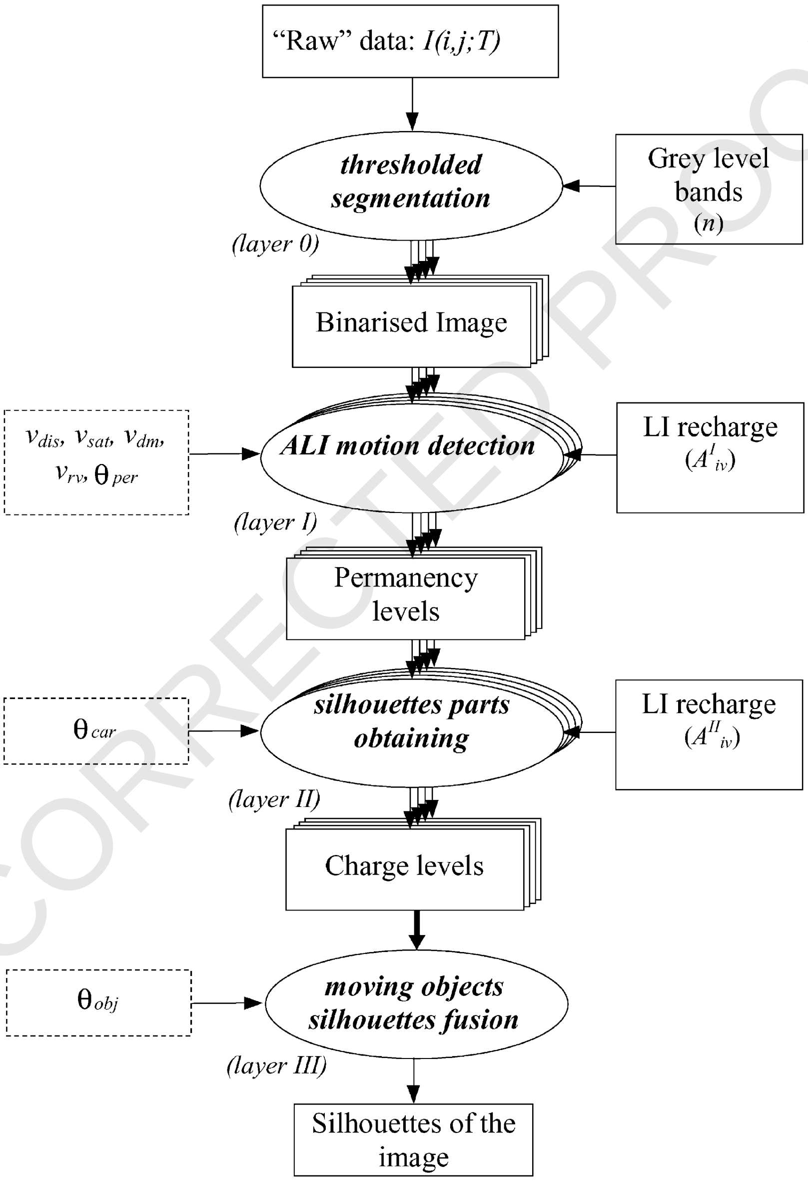

Fig. 2. Inferential scheme of the ‘motion detection’ task.

interfaces in virtual reality systems design, visual surveillance, etc. (Rohr, 1994). With no doubt, the biggest source of data representative of complex non-rigid motion resides in biomedical imagery. Research in this field is really important, mainly in elastic 3D models. There are also 2D models, as for example in X-ray or ultrasound image processing (Cootes, Hill, Taylor, & Haslam, 1994). The computer processing of fluid motion in image sequences is still a recent topic, although one can already speak of some first pioneer intent (Maurizot, Bouthemy, Delyon, Iouditski, & Odobez, 1995). It may also be focused toward the exploitation of satellite images in meteorology or oceanography (Cootes & Taylor, 1994).

2.3. Motion detection through ALI

In our proposal, the *method* used for the decomposition of each one of the *subtasks* is a version of the algorithmic lateral inhibition (ALI) (Delgado, Mira, & Moreno-Diaz, 1989; Fernández-Caballero, Fernández, Mira, & Delgado, 2003a; Fernández-Caballero, Fernández, Mira, & Delgado, 2003b; Fernández-Caballero, Mira, Delgado, & Fernández, 2003c; Fernández-Caballero, Mira, Fernández, & López, 2001; Mira & Delgado, 2001). It is based on the selective accumulation of properties detected in the temporary expansion of the receptive field of the neuronal units of the *layers* associated to the different *subtasks* by means of which we decompose the global task. The ALI method maintains the conceptual aspects of lateral inhibition (LI) and the skeletal model, but we change the type of operators used to make computational each one of the inferences. That means that we move from analytics (adders, multipliers, sigmoids, and so on) to inferential rules of parametric nature. The first decomposition in subtasks of the moving objects silhouettes obtaining task provides the following subtasks

(a) Thresholded segmentation (layer 0). Subtask ‘thresholded segmentation gets as input data the values of the 256 grey level input pixels and generates as output n binary images corresponding to n levels defined by role *bands*. The output space has a FIFO memory structure with two levels, one for the current value and another one for the previous instant value.
(b) ALI motion detection (layer I). The aim of this subtask is to detect the temporal and local (pixel to pixel) contrasts at each sub-layer associated to the n grey level images binarised at layer 0. The couple of binarisation values at each band constitute the input space. The output space is formed by the permanency levels accumulated in the local memory of element (i,j) at band k after dialogue processing with neighbouring elements.
(c) Silhouettes parts obtaining (layer II). The aim of this sub-layer is to obtain the silhouette of all components of a moving object. The layer considers the union of pixels that are physically together and at a same grey level band to be a component. n parallel channels also form this layer, one for each grey level. At each channel a set of concurrent LI processes are performed to distribute the charge among all neighbours that possess a certain minimum load and are physically connected.
(d) Moving objects silhouettes fusion (layer III). The purpose of this last layer is ‘to fuse’ or to juxtapose the silhouettes of the moving objects detected by the different grey level bands. The method used is again LI, but now there are not n sub-layers or channels processing in parallel, but rather the n channels of layer II converge in one single layer through a multiplexing operation on the n channels, where only one of them has a charge value different from zero for each co-ordinate (i,j).

The *inferential scheme* corresponding to this decomposition is shown in Fig. 2, where input and output roles of each subtask and the parallelism inherent to the concurrent calculation for each grey level band in which the initial image breaks down are included. In Fig. 3 we show the results of each subtask as well as the global transformation.

3. The algorithmic lateral inhibition as a co-operative PSM

When looking for inspiration in Biology, and when studying the way the nervous system processes information in the visual pathway, from the photoreceptors, amacrines, horizontal, bipolar, and ganglions cells in the retina up to the associative cortex where, presumably, images are interpreted, one observes that there is a modular architecture that repeats again and again:

1. Computation is modular, of small grain and recursive (synapse, neuron, layer, column,).
2. Computation elements take their data from their receptive fields.
3. There are overlaps of receptive fields at input (shared data), as well as at output (dialogue among neighbouring elements).
4. There coexist a double organisation, horizontal (layers) and vertical (parallel calculation channels). In the horizontal organisation there is all LI processes which occur at the same time at all layers. The vertical organisation corresponds to multiple channels in parallel, as, for instance, in the visual pathway from ganglion cells to columns in cortex.
5. Networks work in two time scales. There is a local time, in general of analogical calculation, and a global one, a slower time and in general of digital nature with a clock defined by the inverse of the synaptic retard.

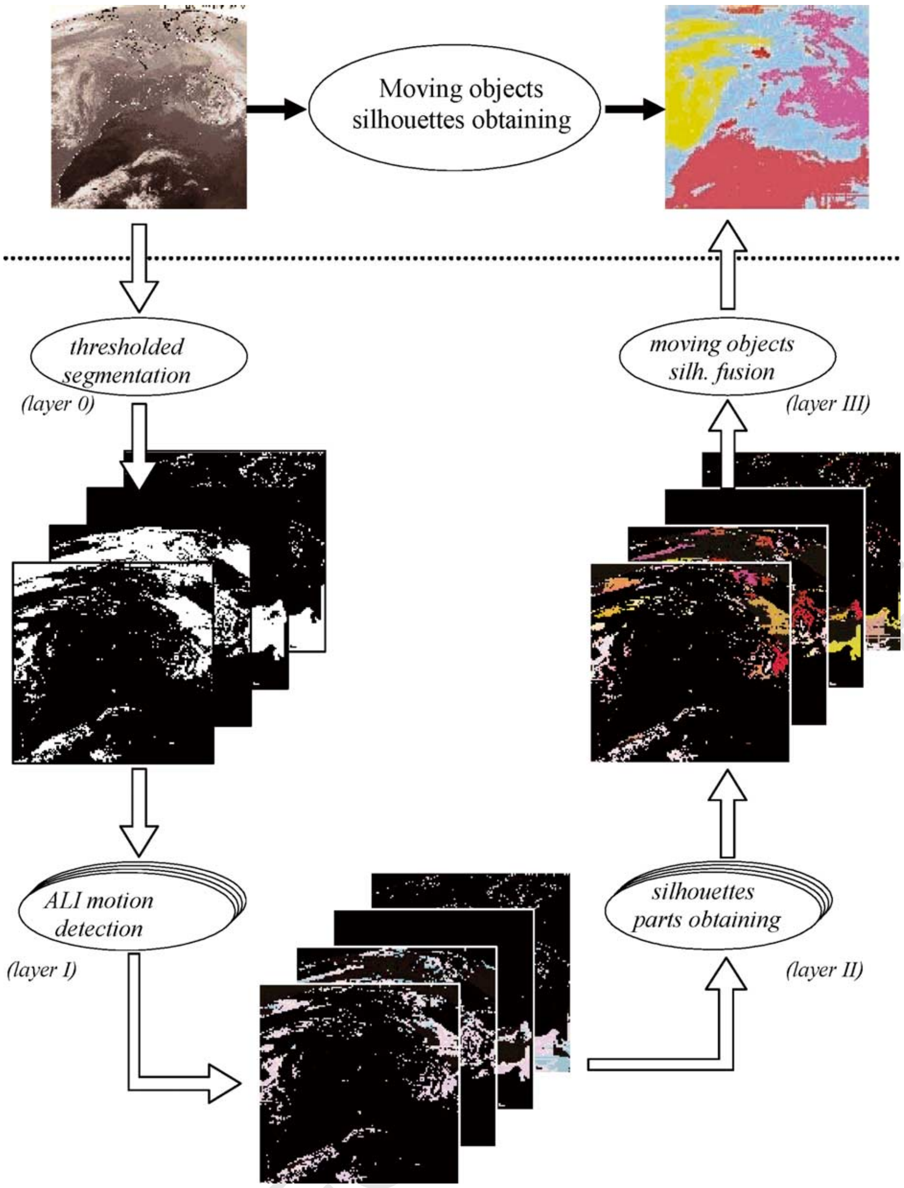

Fig. 3. Result images of the diverse motion detection subtasks.

6. Finally, the local calculation model possesses an invariant structure that in most cases can be formulated in terms of recurrent and/or non-recurrent *LI architecture.*

Anatomically, the LI circuits correspond to schemes such as the one shown in Fig. 4. The unit response not only depends on its own inputs, but on the inputs and responses of the neighbouring neurons. In general, the interaction is of inhibitory type such that the activity of one neuron diminishes where its neighbours are active. We can distinguish between recurrent and non-recurrent actions and, in both cases, between additive and multiplicative–divisive actions, according to the mathematical operation which best adjusts experimental results.

From the analytical modelling point of view, LI circuits can be described with the following assumptions

- Each element of calculus performs a partition of the input space into three regions: centre, periphery and excluded. It does the same with the feedback from the output space and in both cases it carries out a local process over the central zone and another one over the peripheral zone. Subsequently it analytically compares the results of these processes and generates an output.
- These processes can be represented by means of interaction factors, $K(x, y; \alpha, \beta)$. If we name $I(\alpha, \beta)$ the input signal on the element located at coordinates (α, β) and $\Phi(x, y)$ the signal at the output of the element located in position (x, y), we can formulate LI as

Non-recurrent

$\Phi(x, y) =$ Accumulation of direct excitation $I(x, y)$ with that coming from the interaction with the neighbouring

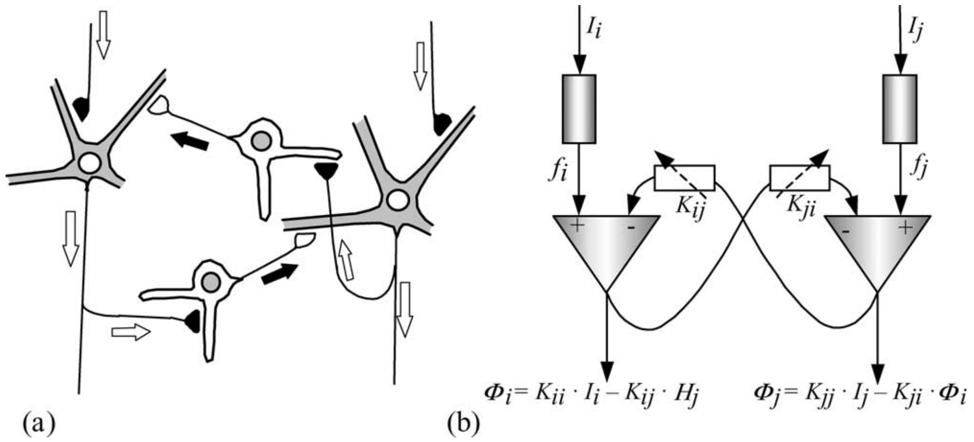

Fig. 4. (a) Recurrent LI circuit. (b) Analytical model.

elements, $I(\alpha, \beta)$ through the interaction factors $K(x, y; \alpha, \beta)$.

$$\Phi(x, y) = \int\int_R K(x, y; \alpha, \beta) I(\alpha, \beta) \mathrm{d}\alpha \mathrm{d}\beta$$

Recurrent

$\Phi(x, y) =$ Accumulation of direct response $\Phi(x, y)$ with that coming from the interaction with the neighbouring elements, $\Phi(\alpha, \beta)$ through the interaction factors $K^*(x, y; \alpha, \beta)$.

$$\Phi(x, y) = \int\int_{R^*} K^*(x, y; \alpha, \beta) \Phi(\alpha, \beta) \mathrm{d}\alpha \mathrm{d}\beta$$

The specific shape, size, structure and adaptive changes of these spatio-temporal convolution kernels convey a relevant part of the LI computation. So, as we detail the shape and position of the ON and OFF volumes, we are designing the filter. This analogical formulation corresponds to a band-pass spatio-temporal recursive filter of order n. That is to say, LI is a detector of spatio-temporal contrast complemented with the possibilities of (1) non-linear expansions of the input and output spaces (multiplicative and divisive inhibition) (2) 'tissue recruitment' covered by dynamic reconfiguration of the ON and OFF volumes (Delgado & Mira, 2002; Mira & Delgado, 2001; Mira, Delgado, Boticario, & Díez, 1995).

If we change the physical input/output spaces by spaces of representation, the integral by generic inference *evaluate*, and the non-linear decision function (the sigmoid) by inference *select*, we have an inferential scheme abstracted from the LI circuit. We call this scheme ALI.

For the non-recurrent (input driven) case we obtain the scheme of Fig. 5. Each calculation element samples its data in the central (C) and periphery (P) part of the volume that its receptive field (RF) specifies in the input space V. On these two data fields (dynamic roles), the calculation element carries out evaluation inferences (*evaluate*) and results comparison (*compare*). This comparison inference is made according to a set of criteria (comparison frame) to generate a set of 'difference values' (discrepancy classes) that play a static role in the final inference *select*, where the output is obtained from the set of outputs associated with the different discrepancy classes, according to the specific discrepancy classes generated by the previous *compare* inference.

In an analogous manner we can obtain the inferential scheme abstracted from the recurrent IL circuits, as shown in Fig. 6. Now each element of calculus starts to infer from data sampled in the central (C^*) and periphery (P^*) parts of its feedback receptive fields in the output space. The values in C^* (individual opinion before dialogue) are compared with the evaluation of the 'opinions' of all the elements in the periphery. This comparison is made according to a set of 'rules for consensus' (consensus criteria) to produce a 'discrepancy class'. Finally, as in the non-recurrent case, this discrepancy class plays the static role of a *select* inference to provide the consensued output.

When first specifying the nature of inferences *compare*, *evaluate* and *select* using decision rules, then specifying the formal expression of these rules (differential operators logical-relational rules) we obtain the different operationalizations of the ALI method. In order of growing difficulty, beginning with the analytic operators and finishing with the inferential ones, we get the following functions

1. Temporal recursive and non-recursive filtering (temporal characteristics extraction and temporal harmonic analysis).
2. Spatial recursive and non-recursive filtering (spatial characteristics extraction and spatial harmonic analysis).
3. Spatio-temporal filtering for motion detection (e.g. direction, velocity, objects size).
4. Colour detection and coding.
5. Cooperation-competition processes.

Now we will explain how this LI method is applied in levels 3 and 5 (spatio-temporal filtering and formulation of algorithmic cooperation-competition processes) to

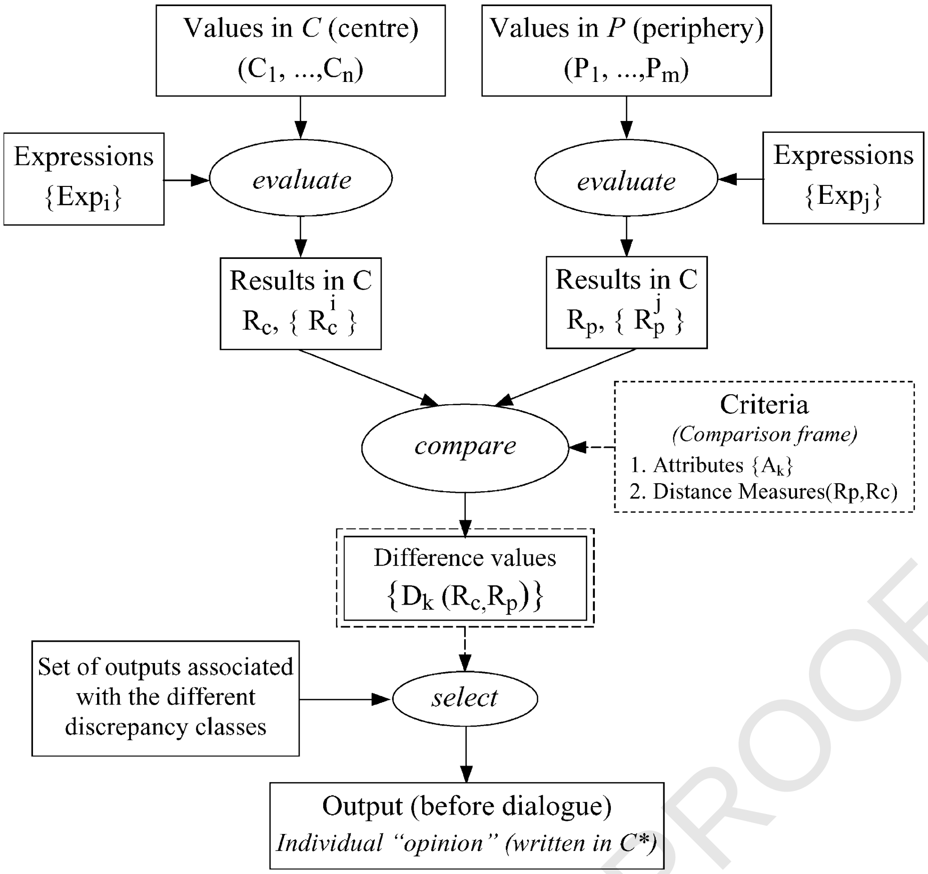

Fig. 5. Inferential scheme for the non-recurrent ALI (data-driven).

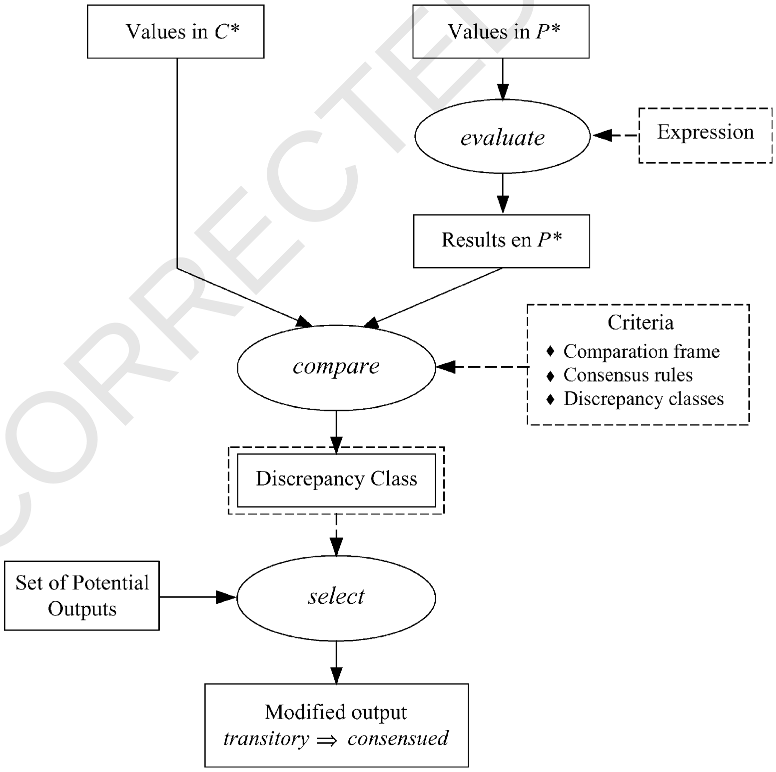

Fig. 6. Inferential scheme for the recurrent ALI (output-driven dialogue).

decompose the four subtasks associated to the task of obtaining the silhouettes of moving objects (thresholded segmentation, movement detection, silhouettes parts obtaining and moving objects silhouettes fusion).

4. Thresholded segmentation subtask (layer 0)

Subtask *thresholded segmentation* gets as input data the values of the 256 grey level input pixels $I(i,j;T)$ and generates as output n binary images, $x_k^o(i,j;t)$, corresponding to n levels defined by role *bands*. The output space has a FIFO memory structure with two levels, one for the current value and another one for the previous instant value. Thus, for n bands, there are $2n$ binary values for each input pixel, $x_k^o(i,j;t)$ and the previous value $x_k^o(i,j;t-\Delta t)$.

$$x_k^o(i,j;t) = \begin{cases} 1, & \text{if } I(i,j;t) \in \left[k\dfrac{256}{n}, \quad (k+1)\dfrac{256}{n} - 1 \right], \\ 0, & \text{otherwise} \end{cases}$$

where $k = 0, \ldots, n-1$, is the band index.

This image binarisation in n bands expands the inferential scheme in n parallel processes, one for each band. The calculation elements of the neuronal inferential network of this layer do not need lateral interaction. In Fig. 7 we show an example of the results of this subtask.

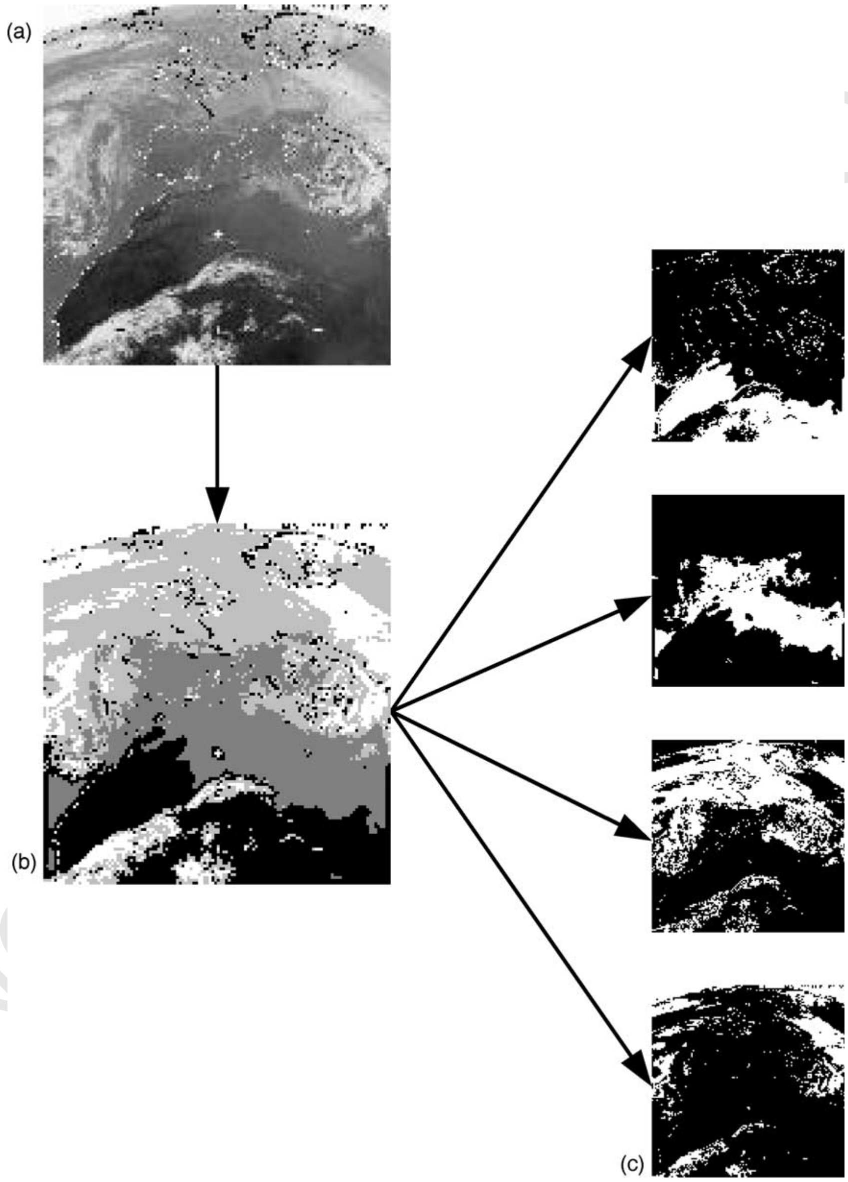

Fig. 7. An example for subtask ‘thresholded segmentation’. (a) Input image from a satellite. (b) Image segmented in four grey level bands (superposition of the four bands in one single image). (c) Result images for each grey level band.

5. ALI motion detection (layer I)

The aim of this subtask is to detect the temporal and local (pixel to pixel) contrasts at each sub-layer associated to the n grey level images binarised at layer 0, taking into account the problems associated to contrasts not related to silhouette motion. From now on, we shall only speak a one generic sub-layer associated to grey level k.

To decompose this subtask we use first a temporal version of the non-recurrent ALI introduced in Fig. 5. Later, we will use the recurrent ALI method of Fig. 6 to cope with the need for dialogue among the neighbouring elements of calculus in the periphery of the output space, (P^*).

The couple of binarisation values at each bands, $x_k^o(i,j;t)$ and $x_k^o(i,j;t-\Delta t)$ constitute the input space of the temporal non-recurrent ALI. The output space (before dialogue) is the result of the individual calculus phase in each element, as shown in Fig. 8.

Inference *compare* receives observable $x_k^o(i,j;t)$ and $x_k^o(i,j;t-\Delta t)$ and the current charge value that initially is $v_{\rm dis}$. It also receives as static role the comparison rule and the numerical coding of the different discrepancy classes $(D1, D2, D3)$. The output role (dynamic) is the class of discrepancy selected at this time, $D(t)$. This class now plays the static role of a *select* inference in charge of filtering a specific charge value (before dialogue) from a set of potential values. These potential values are $v_{\rm dis}$, $v_{\rm sat}$ and max $\{v-v_{dm}, v_{\rm dis}\}$, where v_{dm} is the decrement value applied when no motion is detected between two frames, $v_{\rm dis}$ is the minimum charge value and $v_{\rm sat}$ is the maximum charge value. Value $v_{\rm sat}$ is obtained either when an object just enters the receptive field, or when movement has been detected by any of the pixel's neighbours.

The output selected constitutes the charge value accumulated before dialogue, $v(t+k\Delta t)$, complemented by a label, A, that, when $A=1$, denotes the fact that a movement has been locally detected by this element. This information is used for dialogue in the recurrent part of the calculus.

These values of charge accumulated before dialogue are written in the central part of the output space of each element (C^*) that now enters in the dialogue phase according to recurrent ALI inferential scheme of Fig. 6, instantiated for this task in Fig. 9. The data in the periphery of receptive field in the output space of each element (P^*) contains now the individual calculi of the neighbours. Then, each element takes into account this set of individual calculus, $\{v_j(t+k\Delta\tau), A_j\}$, by means of an *evaluate* inference that uses as static role the logical union of the labels $A_j, A_{P^*}(\tau)=\bigcup_j A_j(\tau)$. This results, $A_{P^*}(\tau)$, is now compared with A_{C^*}, giving rise to one of

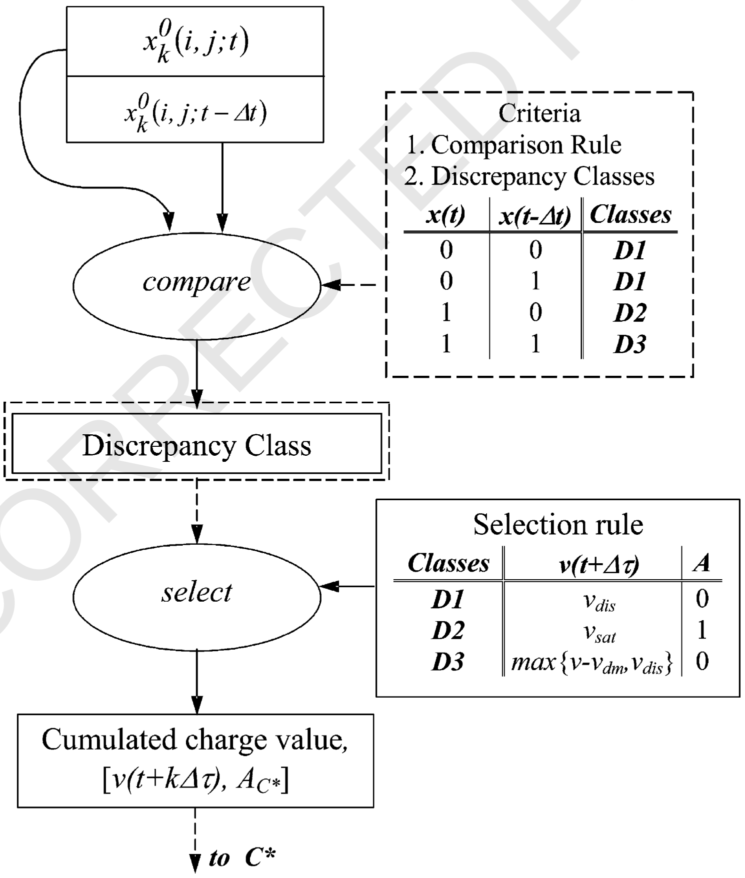

Fig. 8. Instantiation of the temporal non-recurrent ALI method used to decompose the first phase of the motion detection subtask (individual calculus).

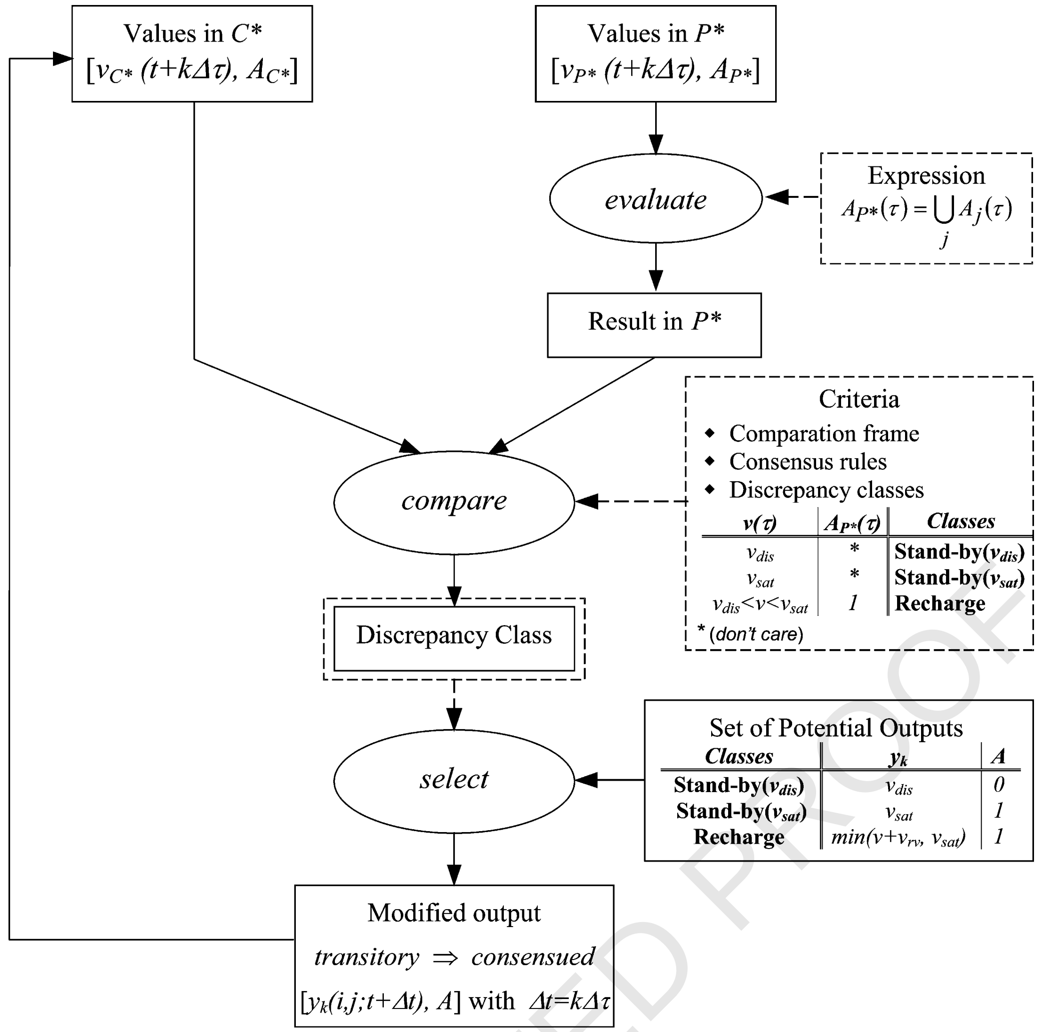

Fig. 9. Inferential scheme of the dialogue phase of subtask 'motion detection' by means of recursive ALI.

two discrepancy classes (recharge or stand-by) and, subsequently, the class activated plays the static role of selection criteria in the next *select* inference that outputs the new consensued charge value after dialogue, $y_k(i,j;t+\Delta t)$, with $\Delta t = k\Delta\tau$, being k the number of iterations in the dialogue phase, a function of the size of the receptive field. The purpose of this last inference of selection is to fix a minimum object size in each grey level band. In Fig. 10 we show an example of the result of 'motion detection' subtask.

6. Silhouettes parts obtaining (layer II)

The aim of this sub-task is to obtain the silhouette of all components of a moving object. The layer considers the union of pixels that are physically together and at a same grey level band to be a component. As at the previous layer, also n parallel channels form this layer, one for each grey level. At each channel, a set of non-recurrent lateral interaction processes are performed to distribute the charge among all neighbours that possess a certain minimum load and are physically connected. A double objective is aimed

1. To dilute the charge due to the image background motion among other points of the own background, so that only moving objects are detected.
2. To obtain a parameter common to all pixels of the part of the object that belongs to the same grey level band. This parameter will again be processed in layer III.

Fig. 11 shows the inferential scheme of this subtask, which is similar to the generic non-recurrent ALI of Fig. 5. Charge values, $y_k(i,j;t+\Delta t)$, offered by layer I are now evaluate in the centre and in the periphery. In inference *evaluate* of P^* we have the average of those neighbours that have charge values different from zero (that is to say, those had charge values superiors to threshold value θ_{per} in the previous layer). Observe that in this subtask the dialogue (inferences *compare* and *select*) again needs k iterations of clock τ, being k

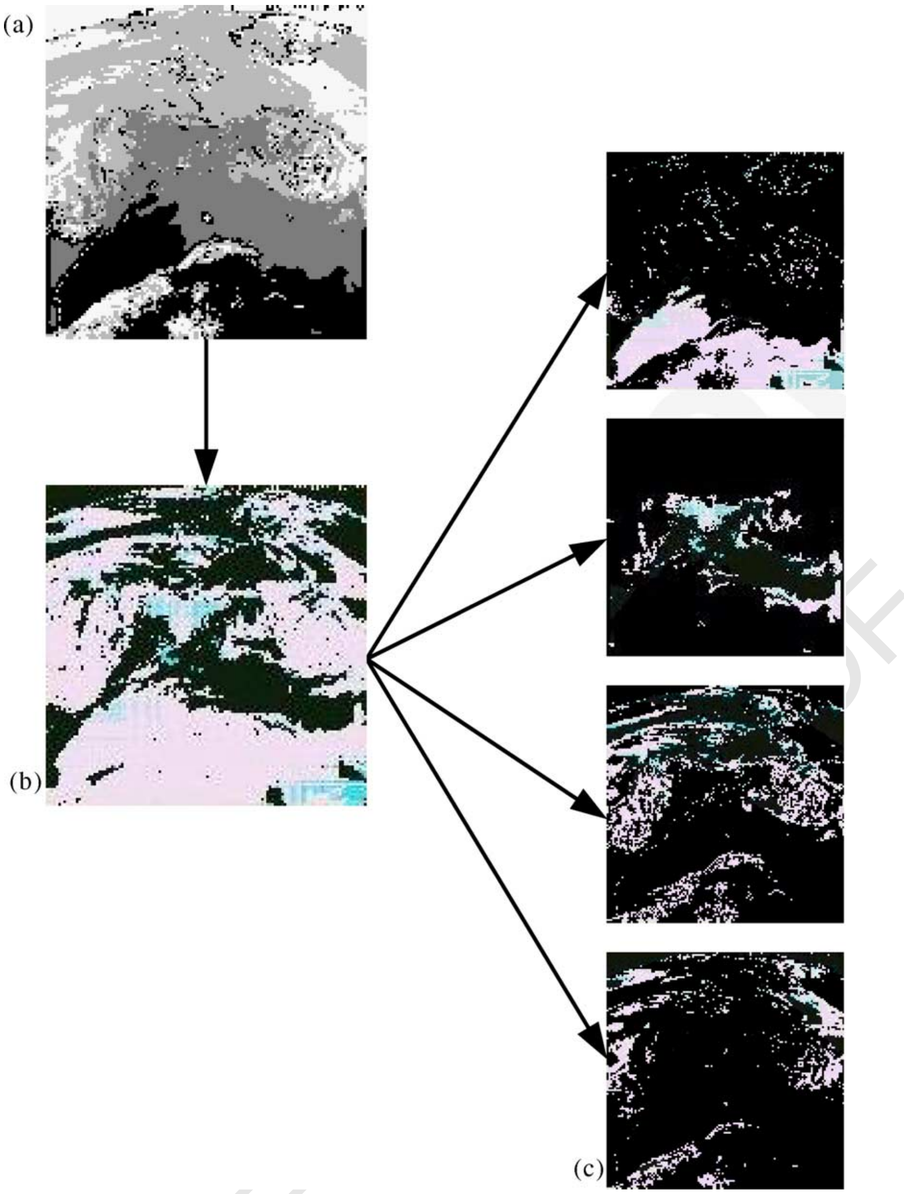

Fig. 10. An example for subtask ‘motion detection’. (a) Image segmented in four grey level bands (superposition of the four bands in one single image); (b) motion image (superposition of the four bands in one single image); (c) result images for each grey level band.

a function of the size of the receptive field. Thus, inference *compare* compares the result of the individual value (C) with the mean value in (P) and produces a discrepancy class according with layer II threshold, (θ_{car}), and passes to layer III the mean charge values that overcome that threshold ($\{z_k(i,j;t\}|z_k \geq \theta_{\text{car}}$).

Later it ‘waits’ for a new image, at the end of Δt. In Fig. 12 we show the results of this subtask.

7. Moving objects silhouettes fusion (layer III)

The previous layers have detected the image elements that are moving in some of the grey level bands and they have tried to eliminate by means of two thresholds (θ_{per} and θ_{car}) all components of motion due to the background. The purpose of this last layer is ‘to fuse’ or to juxtapose the silhouettes of the moving objects detected by the different grey level bands. The method used is again lateral interaction, but now there are not n sub-layers or channels processing in parallel, but rather the n channels of layer II converge in one single layer through a multiplexing operation (selection of maximum) on the n channels, where only one of them has a charge value different from zero for each co-ordinate (i,j) at each t.

The inferential scheme of this layer is the same as that of layer II (Fig. 11) changing the domain elements that play the corresponding dynamic and static roles. Now the input to each element (i,j) is the maximum value of the outputs of the corresponding pixels of each sub-layer of layer II, $z_k(i,j;t)$, $k = 1,\dots,n$, at each Δt. Then, this maximum value is averaged with the values of

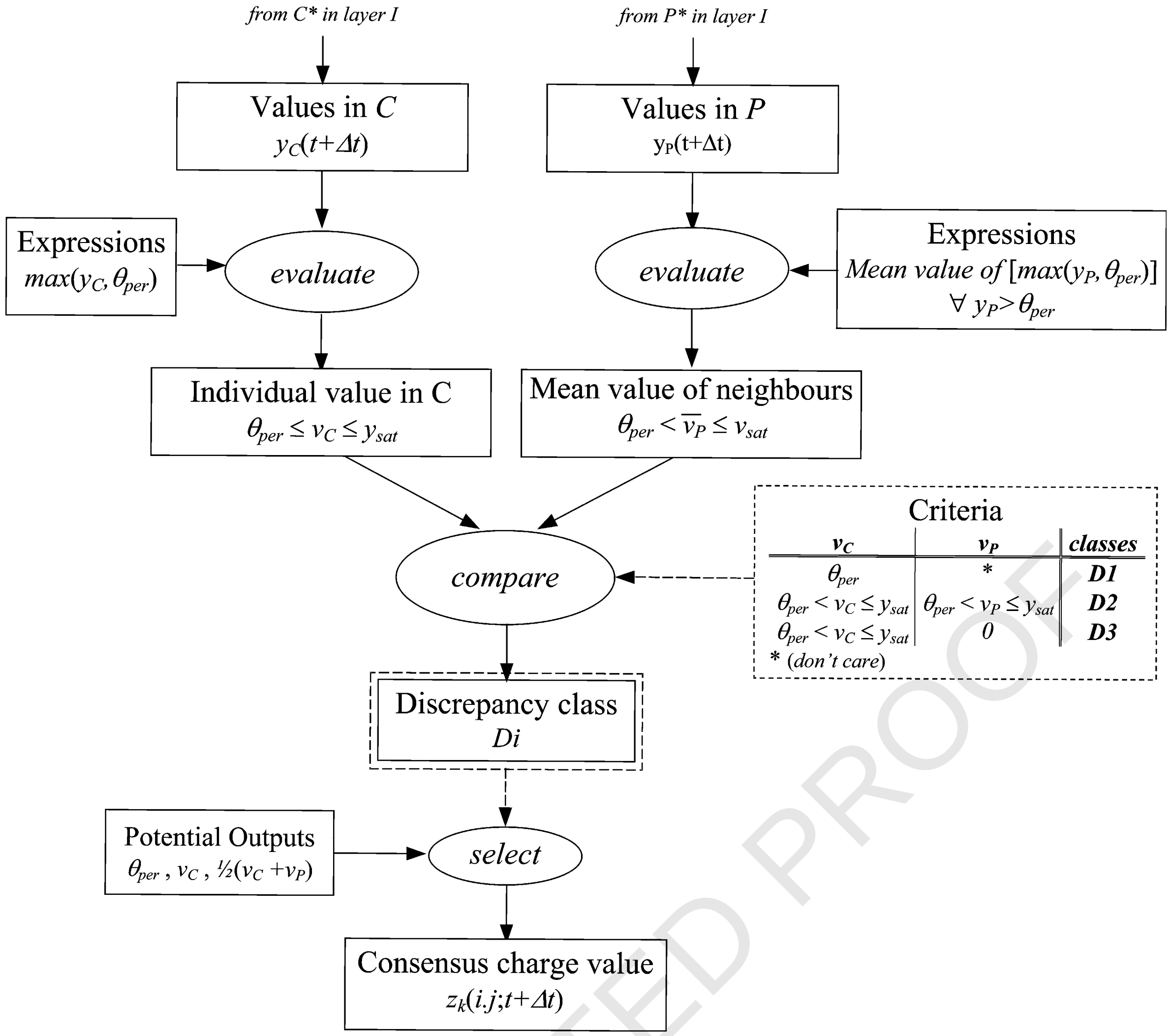

Fig. 11. Inferential scheme of layer II.

the periphery of the receptive field that have overcome a certain threshold value. This is the way the silhouettes $O(i,j;t+\Delta t)$ are obtained. Finally, an illustration of the outputs of the different layers in the *motion detection* task is summarized in Fig. 13.

8. Conclusions

We have presented the knowledge modelling approach for the moving objects in image sequences detection task, trying to show three methodological items.

1. The convenience of modelling knowledge of tasks and methods in terms of a library of reusable components (inferential verbs *evaluate*, *compare* and *select*) and a set of input and output roles played by the entities of the application domain. This way, we contribute to approach knowledge engineering to electronic engineering, where the inherent advantage of the reusable character of the same basic circuits is evident.
2. The potential utility of seeking for inspiration in Biology. In this case, we have used a widespread version of the LI circuits to model a vision task, starting from the certain fact that this circuit type is the one that repeats in an insistent way in the visual pathway of the vertebrates. The distinctive character of our approach is that we have introduced an abstraction. We have passed from the signals level, where LI acts as a spatio-temporal band-pass filter (contrast detection), to the knowledge level, where LI becomes a generic PSM built up on the inferences *evaluate*, *compare* and *select* that samples data from a partition of the external inputs space and from the outputs space (feedback).
3. The convenience to use hybrid PSM to solve problems in artificial vision where the final configuration of a PSM is always dependent on the particular balance between data and knowledge available for the specific case under consideration. In the motion detection task, we have used the available knowledge to specify the architecture of the net. Then we have enhanced

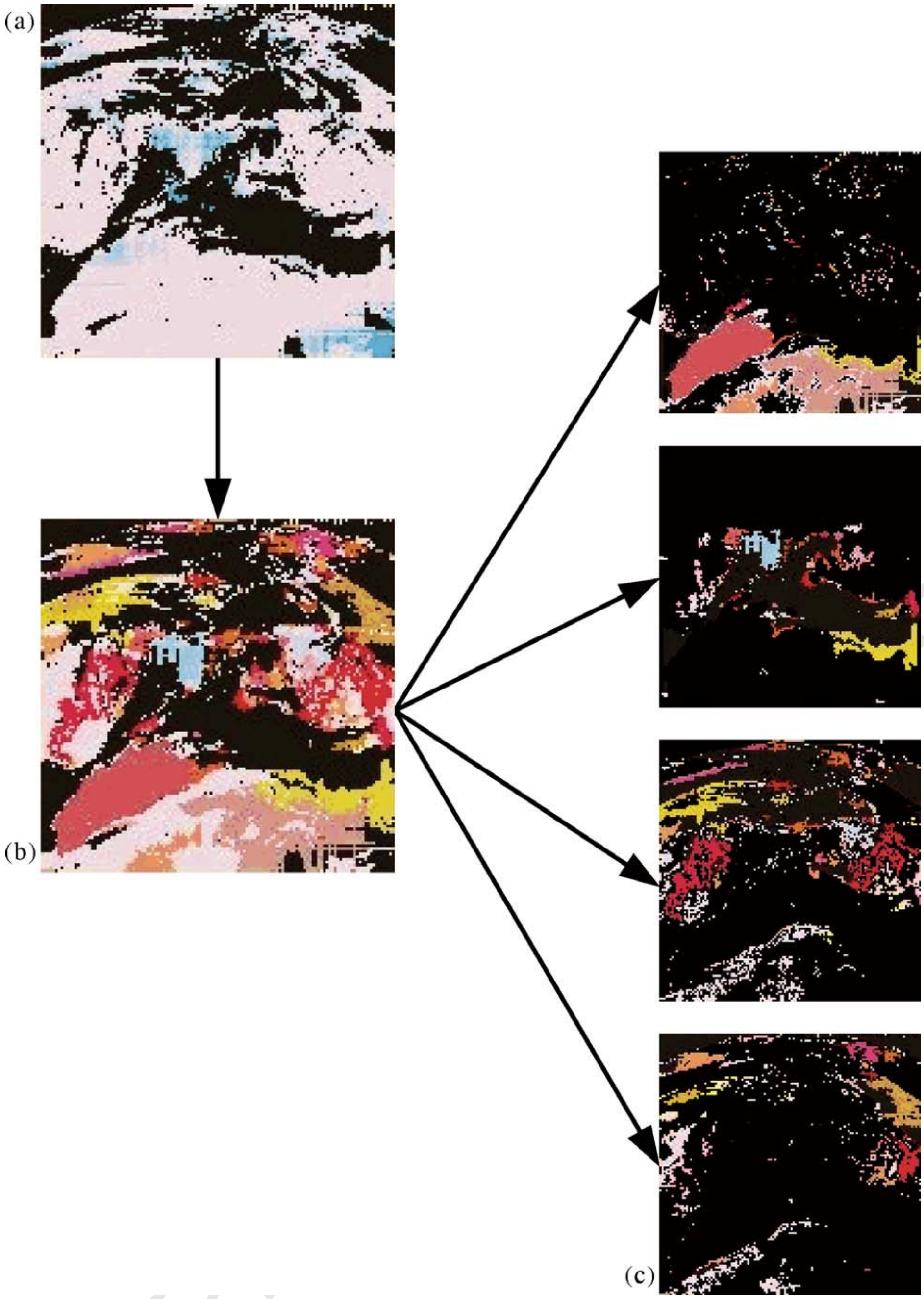

Fig. 12. An example for subtask 'silhouettes parts obtaining'. (a) Motion image (superposition of the four bands in one single image); (b) image composed of silhouette elements (superposition of the four bands in one single image); (c) result images for each grey level band.

the computational power of the artificial neurons that become inferential rules of parametric nature. In this way we can use supervised methods of connectionist learning to adjust the values of these parameters.

For each one of the subtasks we have illustrated the results of the inferential scheme. Finally, we conclude with a summary of advantages and deficiencies of our approach in comparison with others well established alternatives.

Our model applied to motion detection is a 2D approach to motion estimation. In these kinds of approaches, motion estimates are obtained from 2D motion of intensity patterns. In these methods there is a general restriction: the intensity of the image along the motion trajectory must be constant, that is to say, any change through time in the intensity of a pixel is only due to motion. This restriction does not affect our model at all. This way, our algorithms are prepared to work with lots of situations of the real world, where changes in illumination are of a real importance.

The gradient-based estimates have become the main approach in the applications of computer vision. These methods are computationally efficient and satisfactory

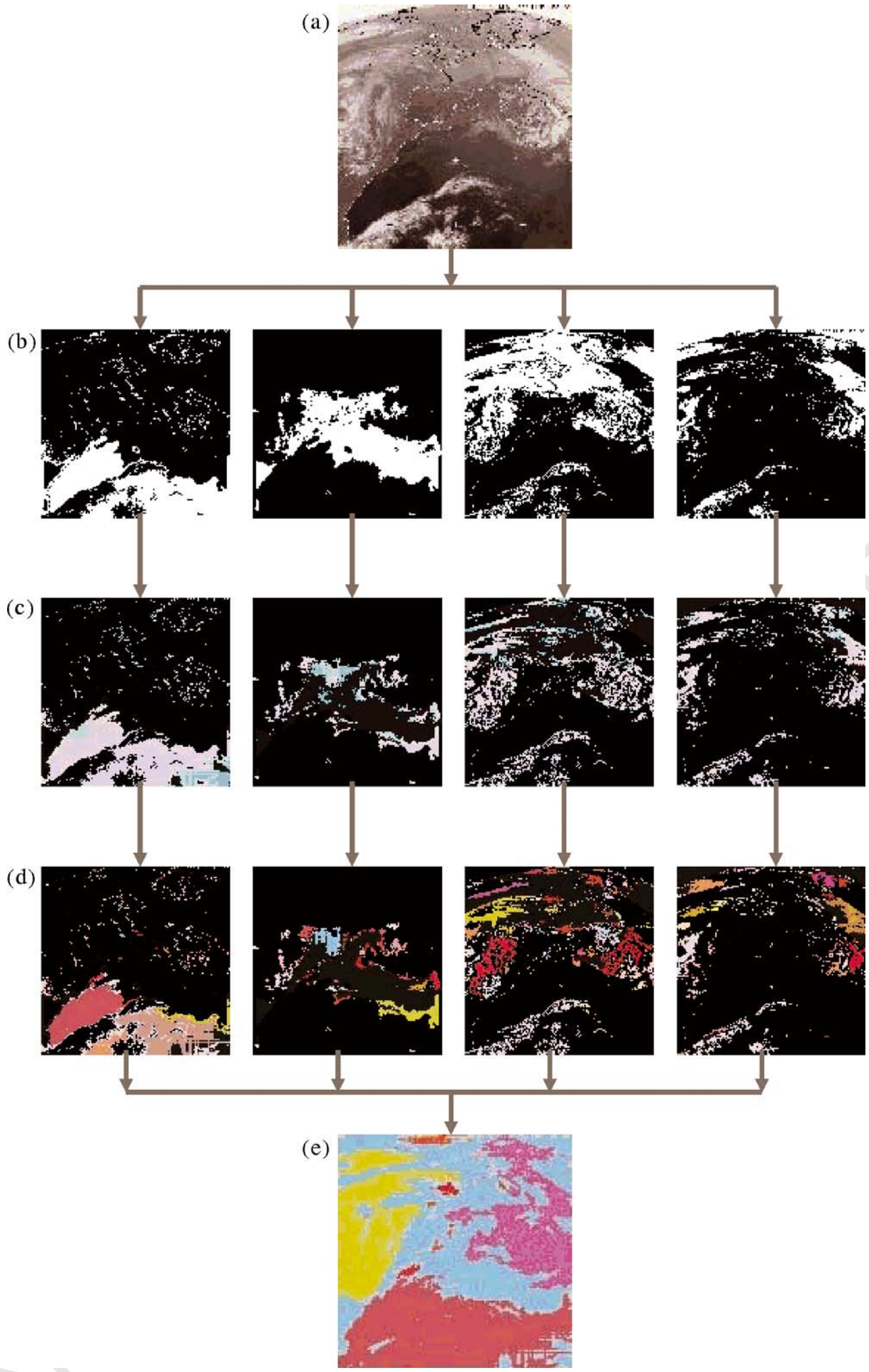

Fig. 13. Description of 'motion detection' task. (a) Input image; (b) result of subtask 'thresholded segmentation'; (c) result of subtask 'ALI motion detection'; (d) result of subtask 'silhouettes parts obtaining'; (e) result of subtask 'moving objects silhouettes fusion'.

motion estimates of the motion field are obtained. Unfortunately, the gradient-based methods always present some restrictions, but our method does not. The disadvantages common to all methods based on the gradient also arise from the logical changes in illumination.

Obviously, a way of solving the former limitations of gradient-based methods is to consider image regions instead of pixels. In general, these methods are less sensitive to noise than gradient-based methods. Our particular approach takes advantage of this fact and uses all available neighbourhood state information as well as the proper motion information. On the other hand, our method is not affected by the greatest disadvantage of region-based methods. Our model does not depend on the pattern of translation motion. In effect, in region-based methods, regions have to remain quite small so that the translation pattern remains valid.

The most important limitation of the method applied to motion detection is the impossibility to differentiate among objects that are seen as a whole due to occlusions.

References

Amit, Y., Grenander, U., & Piccioni, M. (1991). Structural image restoration through deformable templates. *Journal of the American Statistical Association*, *86*, 376–387.

Bascle, B., Bouthemy, P., Deriche, N., & Meyer, F. (1994). *Tracking complex primitives in an image sequence. In Proceedings of the 12th International Conference on Pattern Recognition, Jerusalem, Israel*, (pp. 426–431).

Benjamins, R., & Fensel, D. (1998). Editorial: problem-solving methods. *International Journal on Human–Computer Studies*, *49*(4), 305–313.

Breuker, J., & van de Velde, W. (1994). CommonKADS library for expertise modelling. Amsterdam: IOS Press.

Colet, C., Quinquis, A., & Boucher, J. M. (1992). *Cloudy sky velocity estimation based on optical flow estimation leading with an entropy criterion. ICPR-C*, (pp. 160–163).

Cootes, T. F., Hill, A., Taylor, C. J., & Haslam, J. (1994). Use of active shape models for locating structures in medical images. *Image and Vision Computing Journal*, *12*(6), 355–365.

Cootes, T. F., & Taylor, C. J. (1994). Modelling object appearance using the grey-level surface. *British Machine Vision Conference*, *2*, 423–433.

Delgado, A. E., & Mira, J. (2002). Algorithmic lateral inhibition as a generic method for visual information processing with potential applications in robotics. In P. Da Ruan D'Hondt, & E. E. Kerre (Eds.), *Computational intelligent systems for applied research* (pp. 477–484). Singapore: World Scientific.

Delgado, A. E., Mira, J., & Moreno-Diaz, R. (1989). A neurocibernetic model of modal co-operative decision in the Kilmer–McCulloch space. *Kybernetes*, *18*(3), 48–57.

Eriksson, H., Shahar, Y., Tu, S. W., Puerta, A. R., & Musen, M. A. (1995). Task modeling with reusable problem-solving methods. *Artificial Intelligence*, *79*(2), 239–326.

Fennema, C. L., & Thompson, W. B. (1979). Velocity determination in scenes containing several multiple moving objects. *Computer Graphics and Image Processing*, *9*, 301–315.

Fensel, D. (1997). The tower-of-adapter method for developing and reusing problem-solving methods. In E. Plaza, & V. R. Benjamins (Eds.), *Knowledge acquisition, modeling and management* (pp. 97–112). *Proceedings of the 10th European Workshop, EKAW'97, LNAI, 1319*, Berlin: Springer.

Fensel, D., Benjamins, V. R., Motta, E., & Wielinga, B. (1999). *UPML: A framework for knowledge system reuse. In Proceedings of the17th International Joint Conference on Artificial Intelligence, IJCAI99*, San Francisco: Morgan Kaufmann.

Fernández-Caballero, A., Fernández, M. A., Mira, J., & Delgado, A. E. (2003a). Spatio-temporal shape building from image sequences using lateral interaction in accumulative computation. *Pattern Recognition*, *36*(5), 1131–1142.

Fernández-Caballero, A., Fernández, M. A., Mira, J., & Delgado, A. E. (2003b). On motion detection through a multi-layer neural network architecture. *Neural Networks*, *16*(2), 205–222.

Fernández-Caballero, A., Mira, J., Delgado, A. E., & Fernández, M. A. (2003c). Lateral interaction in accumulative computation: a model for motion detection. *Neurocomputing*, *50C*, 314–364.

Fernández-Caballero, A., Mira, J., Fernández, M. A., & López, M. T. (2001). Segmentation from motion of non-rigid objects by neuronal lateral interaction. *Pattern Recognition Letters*, *22*(14), 1517–1524.

Ford, K., Bradshaw, J. M., Adams-Webber, J. R., & Agnew, N. M. (1993). Knowledge acquisition as a constructive modeling activity. *International Journal of Intelligent Systems*, *8*, 9–32.

Fu, L. M., & Fu, L. C. (1990). Mapping rule-based systems into neural architecture. *Knowledge-Based Systems*, *3*(1), 48–56.

Hilario, M., Orsier, B., Rida, A., & Pellegrini, C. (1995). *Integration of model-based reasoning and neural processing. Report of MIX Project*, CUI, University of Geneva.

Horn, B. K. P. (1986). Robot vision. Cambridge: The MIT Press.

Horn, B. K. P., & Schunck, B. G. (1981). Determining optical flow. *Artificial Intelligence*, *17*, 185–203.

Idris, F., & Panchanathan, S. (1997). Review of image and video indexing techniques. *Journal of Visual Communication and Image Representation*, *8*(2), 146–166.

Irani, M., & Peleg, S. (1993). Motion analysis for image enhancement: resolution, occlusion and transparency. *Journal of Visual Communication and Image Representation*, *4*, 324–335.

Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: active contour models. *International Journal of Computer Vision*, *1*(4), 321–331.

Kuan, D., Phipps, G., & Hsueh, A. C. (1988). Autonomous robotic vehicle road following. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *10*(5), 648–658.

Lewis, F., Abdallah, C., & Dawson, D. (1993). Control of robot manipulators. New York: Macmillan.

Li, H. B., Roivainen, P., & Forchheimer, R. (1993). 3-D motion estimation in model-based facial image coding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *15*(6), 545–555.

Maturana, H. R. (1975). The organization of the living: a theory of the living organization. *International Journal of Man-Machine Studies*, *7*, 313–332.

Maurizot, M., Bouthemy, P., Delyon, B., Iouditski, A., & Odobez, J. M. (1995). *Determination of singular points in 2D deformable flow fields. In Proceedings of the second IEEE International Conference on Image Processing, Washington DC*.

Mira, J., Alvarez, J. R., & Martinez, R. (2000). Knowledge edition and reuse in DIAGEN: a relational approach. *IEE Proceedings-Software*, *147*(5), 151–162.

Mira, J., & Delgado, A. E. (1987). Some comments on the antropocentric viewpoint in the neurocybernetic methodology. *In Proceedings of the Seventh International Congress of Cybernetics and Systems*, *2*, 891–895.

Mira, J., & Delgado, A. E. (2001). What can we compute with lateral inhibition circuits. In J. Mira, & A. Prieto (Eds.), (pp. 38–46). *Connectionist models of neurons, learning processes and artificial intelligence, LNCS, 2084*, Berlin: Springer.

Mira, J., Delgado, A. E., Boticario, J. G., & Díez, F. J. (1995). Aspectos Básicos de la Inteligencia Artificial. Madrid: Editorial Sanz y Torres.

Mira, J., Herrero, J. C., & Delgado, A. E. (1998). *Where is knowledge in computational intelligence? On the reduction of the knowledge level to the level below. In Proceedings 24th EUROMICRO Conference*, IEEE, II, (pp. 723–732).

Mira, J., & Delgado, A. E. (2003). Where is knowledge in robotics? Some methodological issues on symbolic and connectionist perspectives of AI. In C. Zhou, D. Maravall, & Da Rua (Eds.), *Autonomous robotic systems, Chap. 1* (p. 334) Heidelberg: Physica/Springer.

Nair, D., & Aggarwal, J. K. (1998). Moving object detection from a navigating robot. *IEEE Transactions on Robotics and Automation*, *14*(3), 404–416.

Prince, J. L., & McVeigh, E. R. (1992). Motion estimation from tagged MR image sequences. *IEEE Transactions on Medical Imaging*, *11*, 238–249.

Quistgaard, J. U. (1997). Signal acquisition and processing in medical diagnostic ultrasound. *IEEE Signal Processing Magazine*, *14*(1), 67–74.

Rohr, K. (1994). Toward model-based recognition of human movements in image sequences. *Computer Vision, Graphics, and Image Processing: Image Understanding*, *59*(1), 115.

Schreiber, A., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., van de Welde, W., & Wielinga, B. (2001). *Knowledge engineering and management*. *The CommonKADS methodology*, Cambridge: MIT Press.

Sternberg, S. R. (1985). *Vision guided robot for automotive assembly*. *In Proceedings of the IEEE International Conference on Robotics and Automation*, (pp. 686–690).

Sun, R., & Alexandre, F. (1997). Connectionist—Symbolic integration. London: LEA Publishers.

Taboada, M., Des, J., Mira, J., & Marin, R. (2001). Diagnosis systems in medicine with reusable knowledge components. *IEEE Intelligent Systems*, *16*(6), 68–73.

Thompson, W. B., & Barnard, S. T. (1981). Lower-level estimation and interpretation of visual motion. *IEEE Computer*, *14*, 20–28.

Varela, F. J. (1979). *Principles of biological autonomy*. *The North Holland series in general systems research*, New York: North-Holland.

Vlachos, T., & Thomas, G. (1996). *Motion estimation for the correction of twin-lens telecine flicker*. *In Proceedings of the IEEE Conference on Image Processing, 1, Lausanne, Swiss*.

Wettergreen, D., Thomas, H., & Bualat, M. (1997). *Initial results from vision-based control of the Ames Marsokhod rover*. *In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Grenoble, France*.

Yuille, A. L. (1992). Feature extraction from faces using deformable templates. *International Journal of Computer Vision*, *8*(2), 90–111.