PERGAMON

AVAILABLE AT
www.ComputerScienceWeb.com
POWERED BY SCIENCE DIRECT®

Neural Networks 16 (2003) 205–222

Neural
Networks

www.elsevier.com/locate/neunet

On motion detection through a multi-layer neural network architecture

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

[a]Departamento de Informática, Universidad de Castilla-La Mancha, Escuela Politecnia Superior, Campus Universitario, 02071 Albacete, Spain
[b]Departamento de Inteligencia Artificial, UNED, c/Senda del Rey, 9, 28040 Madrid, Spain

Received 29 November 2001; accepted 10 October 2002

Abstract

A neural network model called lateral interaction in accumulative computation for detection of non-rigid objects from motion of any of their parts in indefinite sequences of images is presented. Some biological evidences inspire the model. After introducing the model, the complete multi-layer neural architecture is offered in this paper. The architecture consists of four layers that perform segmentation by gray level bands, accumulative charge computation, charge redistribution by gray level bands and moving object fusion. The lateral interaction in accumulative computation associated learning algorithm is also introduced. Some examples that explain the usefulness of the system we propose are shown at the end of this article.

Keywords: Multi-layer neural networks; Algorithmic lateral inhibition; Lateral interaction; Accumulative computation; Motion detection

1. Introduction

1.1. Biological motion detection

Motion detection is so important for the adaptation of most animals that only humans and some evolved primates can respond to objects with no motion. Many vertebrates (such as frogs) cannot see objects unless they are in motion. In humans this limitation persists in the outer part of the retina. We cannot detect any motion in the outlying ends of the visual field. Instead of it, a moving object in the periphery unchains an unconscious reflection that causes eye rotation, thus placing the moving object in the central visual field. Motion in the visual field could be detected by comparing the position of the images perceived in different moments.

The visual system's detectors only look at a small part of the visual field. The problem arises when assigning the true speed of an object starting from local measurements. In fact, motion on a single extended line segment does not determine motion of an object that contains that line segment (Adelson & Bergen, 1985; Fennema & Thompson, 1979; Hildreth, 1984; Horn & Schunck, 1981; Marr & Ullman, 1981; Wallach, 1976). Motion parallel to the line is invisible. This way, a set of possible motions can be the result of the detected movement. The solution to the so-called aperture problem is solved if at least two measurements of local component motions in a pixel exist, leading to the estimation of the velocity of a pattern. In a simple movement as translation in a plane, the problem is broadly resolved. Indeed, as 2D velocity is the same for the whole pattern, in most cases, more than two measurements of local components are present to estimate 2D velocity. This is not the case, however, for 3D and rotational motion, where the real 2D velocity varies from pixel to pixel. That is why, 3D motion measurement is ambiguous and some additional restrictions are required to find a unique solution.

When two or more objects move simultaneously in a limited region of the visual field, we need to distinguish between motion of the different parts of a particular object and motion of different objects. Current biological data suggest that there are several levels in motion analysis in the visual system (Albright, 1992; Allman, Miezin, & McGuinness, 1985; Andersen & Siegel, 1990; Morrone, Burr, & Vaina, 1995).

In first place, it is known that the aperture problem for the translation motion plane is solved in two levels. In the first level the local motion measurements extract the motion components that are perpendicular to the elements in

[*] Corresponding author. Tel.: +34-967-59-92-00; fax: +34-967-59-92-24.
E-mail address: caballer@info-ab.uclm.es (A. Fernández-Caballero).

0893-6080/03/$ - see front matter © 2003 Elsevier Science Ltd. All rights reserved.
PII: S0893-6080(02)00233-2

the image. The second level combines the local motion measurements of portions of the image with the purpose of calculating a smaller number of local translation estimates for the pattern. Finally, a third level integrates the local estimates of translation motion to calculate more complex non-local motions (i.e. global rotations). This way, at each level, motion information spatially located in an area seems to be combined to calculate less local but more complex motions (Sereno, 1993).

1.2. Problem statement

Motion plays an important role in our visual understanding of the surrounding environment (Mitiche & Bouthemy, 1996). From visual motion it is possible to gain insight about the 3D structure of the scene observed (Faugeras, 1993; Marr, 1982). It may be useful for the detection of shape (Faugeras, Lustman, & Toscani, 1987), and for providing information as the relative depth of moving objects (Tekalp, 1995), and supplying clues about the material properties of moving objects, such as rigidity and transparency (Shizawa, 1992). Motion information can also from the basis of predictions about time-to-impact and the trajectories of objects moving across a scene (Horn, 1986). Numerous psycho-visual studies have demonstrated that motion is a significant visual cue. For example, Ullman (1979) succinctly illustrated the shape from motion effect by generating a sequence of images corresponding to the projection of a set of random pixels on a pair of concentric cylinders rotating in opposite directions. Viewed individually, the images yield no 3D information, but when viewed all together they show that the human shape is recognizable from its characteristic motion. A video showing the motion of light sources attached to the ankles, knees and wrists of a person instantly convey the shape of the human form (Sekuler & Blake, 1994).

The problem we are stating is the discrimination of a set of non-rigid objects capable of holding our attention in a scene. These objects are detected from the motion of any of their parts. Detected in an indefinite sequence of images, motion allows obtaining the shapes of the moving elements. Whenever an element stops moving, it does no longer receive attention. Thus, interest on that particular shape declines, so that the shape does not belong to the discriminated objects. In real scenes, not all of the object's components move at the same time or may present no motion at all. For example, the human body (the object, in this case) is composed of a great number of members that do not move simultaneously. The system proposed can detect and even associate all moving parts of the objects present in the scene.

Thus, the particular problem we are dealing with is segmentation-from-motion by means of a model based on a neural architecture close to biology. The neurophysiological foundations of motion perception have been studied so far (Hildreth & Royden, 1998), as well as some models for performing this motion perception in biological systems implemented in artificial neural networks (Hatsopoulos & Warren, 1991; Sereno, 1993). But, such networks often embody a restricted formulation of the motion analysis problem. Another alternative to motion detection is self-organization (Marshall, 1998), where there is an extraction of basic local motion signals from image sequences, and an integration of multiple motion signals across the image.

A synthetic vision of the biological bases of our approach is given next. If we accept that in neural networks an important part of computation is associated to the shape of the receptive field and to the excitatory and inhibitory character of its center and periphery, the basic biological foundation of our approach is that the recurrent and non-recurrent lateral inhibition defines receptive fields whose operational description corresponds to kernels in differences. A spatio-temporal detection is intrinsically performed, which is tuned to the shape of the receptive field. This way, each receptive field has an optimal response to those stimuli that are in accordance to its shape. This occurs in retina, at lateral geniculate body level, and in cortex columns, where there are vertical structures tuned to different properties of the stimuli—spatial, spatio-temporal, orientation columns—(Mountcastle, 1979). The key point of our approach is that we have used the biological inspiration at organizational level and computational principles—what David Marr called computational theory (Marr, 1974)—but we have eliminated the restriction of the use of conventional analog operators in neural nets (weighted sums followed by sigmoid) and we have substituted the analog calculus by a set of inference rules, obtaining this way what we have called the algorithmic lateral inhibition, which gives a mayor computational capacity to the model. A comparison of our approach to others will be offered in Section 6.

2. Our approach: lateral interaction in accumulative computation

Some very interesting models based on biological evidence have been offered so far (Bülthoff, Little, & Poggio, 1989; Grossberg & McLaughlin, 1997; Grossberg & Rudd, 1989; Ross, Grossberg, & Mingolla, 2000; Yuille & Grzywacz, 1988).

A generic algorithm based on a neural architecture, with recurrent and parallel computation at each specialized layer, and sequential computation between consecutive layers, is presented. Each layer is composed of modules of the same type. The result of the activity of any layer can be considered as a classification associating input to output configurations. The latter are converted as well into input configurations of the following layer (Mira et al., 1995).

The model proposed is based on an accumulative computation function, followed by a set of co-operating lateral interaction processes performed on a functional receptive field organized as center-periphery over linear expansions of their input spaces (Gerstner, Ritz, & van Hemmen, 1993; Mira, 1993; Moreno-Diaz, Rubio, & Mira, 1969; Wimbauer, Gerstner, & van Hemmen, 1994). We will introduce both terms.

2.1. Lateral interaction models

The central nervous system is formed by common neural sets, containing very few or a great number of neurons. Inside a common group of neurons there is a great number of short nervous fibers, allowing the signals to spread horizontally from neuron to neuron inside the group. The dendrites of some neurons also ramify and are disseminated in the common set. The neuronal area stimulated by each nervous fiber is denominated stimulation field.

Let us remember, once again, that the stimulus that arrives to a neuron can be (1) exciting, also called threshold stimulus because it is above the necessary threshold for the excitement, or (2) a sub-threshold stimulus. A sub-threshold stimulus does not excite the neuron, but it makes it more excitable for impulses coming from other sources. The neuron that has become more excitable but that does not discharge is facilitated. The neural field area where neurons discharge at a given instant is termed threshold area. The area to each side where the neurons are facilitated but do not discharge is denominated facilitated area.

Many times a neural set receives input nervous fibers from diverse origins. We generally have a primary source and diverse secondary sources. Generally, the secondary sources are not enough to cause excitement, but they facilitate the neurons. Other times, the secondary sources highly inhibit the neuron set, so that a powerful signal of the primary source is needed to originate the normal discharge.

Most information is transmitted from one part of the nervous system to another through several successive neuronal sets. The neural set facilitation degrees are controlled by centrifugal nervous fibers. These undoubtedly help to control the fidelity of signal transmission. The space type tends to lose lucidity even before the signal begins to be transmitted across the pathway. However, in a pathway such as the visual one, lateral inhibitory circuits inhibit the peripheral neurons and they re-establish a true space disposition.

In lateral interaction models (Gilbert, Hirsch, & Wiesel, 1990; Mira, Delgado, Alvarez, de Madrid, & Santos, 1993; Mira, Delgado, Manjares, Ros, & Alvarez, 1996), there is a layer of modules of the same type with local connectivity. The response of a given element does not only depend on its own inputs, but also on the inputs and outputs of the element's neighbors. From a computational point of view, the aim of the lateral interaction nets is to partition the input space into three regions: center, periphery and excluded. The following steps have to be followed: (a) processing over the central region, (b) processing over the feedback of the periphery zone, (c) comparison of results from these operations and local decision generation, and (d) distribution over the output space.

2.2. Accumulative computation model

Information conversion and memory are also functions of the neurons that are related through a synchronous shot, as stated by Hebb's law. Every time a certain sensorial sign crosses a synapse series, these synapses are more and more able of transmitting the same sign next time. The memory helps to select the new sensorial information of importance and to deviate it toward appropriate areas of storage for future employment or toward areas that originate corporal responses.

At this point, we introduce the accumulative computation model (Fernandez & Mira, 1992; Fernandez et al., 1995). This model basically responds to a sequential module represented by its charge value. The accumulative computation process responds with an output called the module's charge value. The state value is also called the permanence value and is generally stored in a permanence memory. First of all, the module performs the sum of the charge value using the accumulative computation function. Note that the result from the previous operation has to fall between limits v_{dis} (discharged) and v_{sat} (saturated).

The synaptic vesicles contain a transmitted substance that, when liberated toward the synaptic fissure, excites or inhibits the neurons. The excitement or inhibition effect of a transmitter depends not only on its nature but also on that of the receiver in the post-synaptic membrane. Besides the inhibition caused by the button inhibitors acting at the synapse level—called post-synaptic inhibition—another inhibition type takes place before the signal arrives to the synapse. This inhibition type is called pre-synaptic inhibition. The pre-synaptic inhibition requires more time to develop than the post-synaptic, but once it happens it lasts much longer. This inhibition enforces the limits among the stimulated and not stimulated areas of the sensorial pathway, because it impedes the excessive dissemination from the sensorial signals to the not excited neurons. This process is also called increase of the contrast.

When the pre-synaptic terminals are continuously and repetitively stimulated, on a high frequency basis, the number of discharges at the post-synaptic neurons is very high at the beginning, but decrease with time. This is called the fatigue of the synaptic transmission. Fatigue is a very important characteristic of the synaptic function, because when areas of the central nervous system are overexcited, fatigue is able to cause this excessive excitation to disappear after a short period. The signal progressively weakening is usually denominated decrement conductivity. If an appropriate time of rest is allowed between the stimuli, the synapse conduction recovers after high level of fatigue.

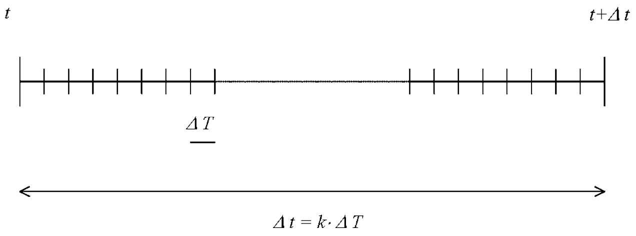

Fig. 1. Comparison between local and global time scales.

2.3. Double time scale

When an impulse is transmitted from a synaptic button to a post-synaptic neuron, a certain period of time is elapsed. This is due to several processes. First there is the substance discharge through the transmission button. Secondly, we have the diffusion of the transmitter to the sub-synaptic neural membrane. In third place, the action of the transmitter on the membrane, and, finally, the diffusion of the sodium toward the interior to elevate the excitation post-synaptic potential until the necessary value to discharge an action potential is reached. The minimum period required to perform all these steps is called the synaptic retard.

Obviously, one of the characteristics of the information transmitted is quantitative intensity. The different degrees of intensity can be transmitted using a larger number of parallel fibers or sending more impulses along one single fiber. These two mechanisms are spatial and temporal summation, respectively. Spatial summation is obtained by means of the effect of adding simultaneous post-synaptic potentials created by excitement of multiple buttons in very dispersed areas of the membrane, while temporal summation is obtained by quickly summing repetitive post-synaptic potentials.

The proposed algorithm also incorporates the notion of double time scale at accumulative computation level present at sub-cellular micro-computation (Fernandez et al., 1995). The following properties are applicable to this model: (a) local convergent process around each element, (b) semi-autonomous functioning, with each element capable of spatial-temporal accumulation of local inputs at time scale T, and conditional discharge, and (c) attenuated transmission of these accumulations of persistent coincidences towards the periphery that integrates at the global time scale t. Therefore there are two different time scales: (a) local time T, and (b) global time t ($T \ll t$). Fig. 1 shows the relationship between both time scales.

3. The multi-layer architecture

The present architecture is inspired by the schematic representation of the artificial vision as described by Mira, Delgado, Boticario, and Diez (1995).

Indeed, the architecture of the method described in this paper basically contemplates the low-level visual processing stage in a similar way to the representation offered in Fig. 2.

This work introduces the multi-layer architecture for the lateral interaction in accumulative computation model focused towards motion detection in an indefinite sequence of images. From the low level processing stage of Mira et al., the architecture includes cues like extraction of characteristics, segmentation and classification in its successive layers.

The lateral interaction model is not affected by the restrictions caused by the characteristics of the scenes analyzed as well as those of the high level process. We can and should consider the lateral interaction model applied to artificial vision as an isolated piece of any intelligent processing.

The general lateral interaction in accumulative computation model, as well as the entire multi-layer architecture that will be later commented, will be able to fit like a puzzle piece in a whole series of different scenes of the real world.

The following figure shows the complete modular computational solution. Fig. 3 shows the four layers that form the architecture of the lateral interaction in accumulative computation method.

The four layers are

(a) *Layer 0: segmentation by gray level bands*. This layer covers the need to segment the image in a preset group of n gray level bands. The input of each element in the layer will be the gray level value of the corresponding image pixel at each global time instant t. From each element, n values $\mathrm{GLL}_k(x,y,t)$ are output toward pixel (x,y) of the n sub-layers (as many as gray level bands established) at layer 1. These values indicate if the pixel corresponds to each of the gray level bands.

(b) *Layer 1: lateral interaction for accumulative computation*. This layer has been designed to obtain the permanence value $\mathrm{PM}_k(x,y,t)$ by decomposition in gray level bands. We will have n sub-layers and each one will memorize the value of the accumulative computation present at global time scale t for each element. Lateral interaction in this layer is thought to reactivate the permanence charge of those elements partially loaded and that are directly or indirectly connected to elements saturated. The permanence charge of each element will be offered to the following layer as output.

(c) *Layer 2: lateral interaction for charge redistribution by gray level bands*. Layer 2 is also formed of n sub-layers. It is handled by means of lateral interaction charge redistribution among all connected neighbors holding a minimum charge. Besides distributing the charge $C_k(x,y,t)$ in gray level

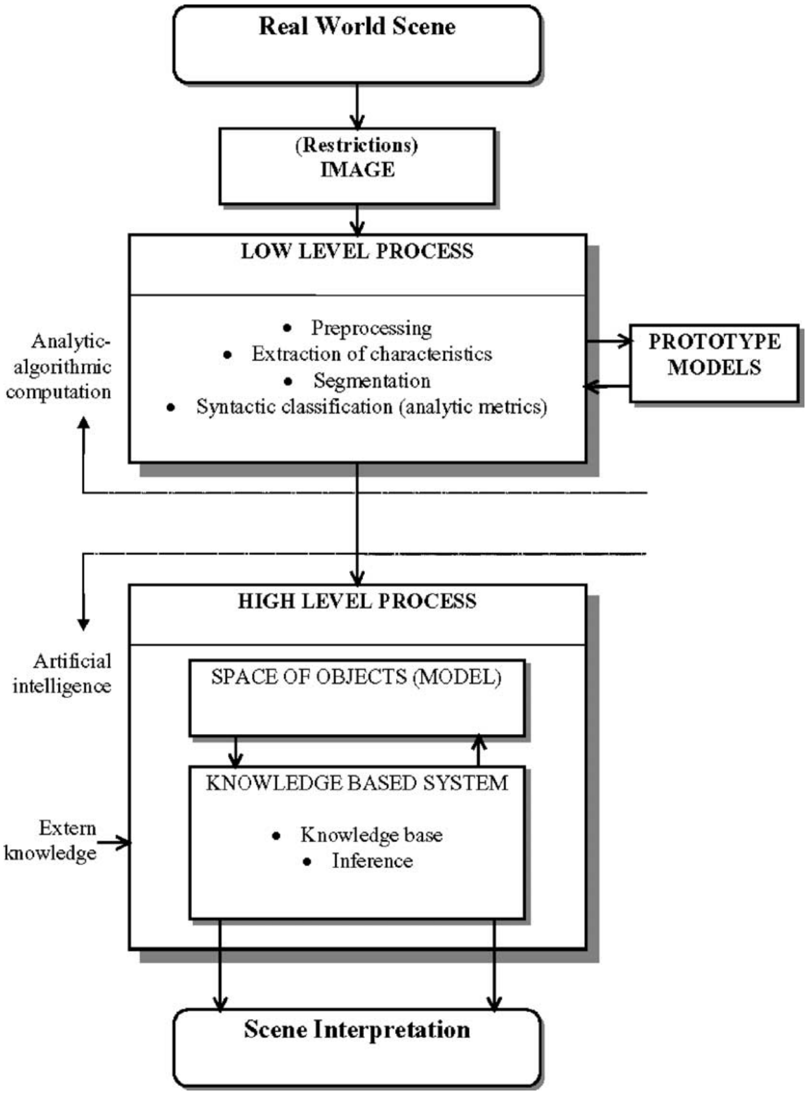

Fig. 2. Schematic representation of the artificial vision process.

bands, at this level, the charge due to the motion of the background is also diluted. The new charge obtained at this layer is offered as an output toward layer 3.

(d) *Layer 3: lateral interaction for moving object fusion*. Each element in this layer has an input from each corresponding element of the n sub-layers from layer 2. The aim in this layer is the fusion of the objects. The input charges of each gray level band are fused, obtaining all the moving objects of the original image. Output from layer 3 is a set of objects $S(x, y, t)$.

3.1. Layer 0: segmentation by gray level bands

An implementation by a modular computation form of the mechanisms described so far lead us to introduce a first layer of up to ij elements (one for each image pixel).

At this layer the external connections for each of the image pixels are those shown in Fig. 4.

Let $\mathrm{GL}(x, y, t)$ be the gray level of pixel (x, y) at time instant t. For each gray level band k, and for each image pixel (x, y), we have, at all instant t, the following algorithm

$$\mathrm{GLL}_k(x, y, t) = \begin{cases} 1, & \text{if } \mathrm{GL}(x, y, t) \in [(256/n)k, (256/n)(k+1)[\\ 0, & \text{otherwise} \end{cases}$$

where n is the total number of gray level bands, and, k is a specific gray level band.

In other words, we have to determine in which gray level band a certain pixel falls. At this level, we are not

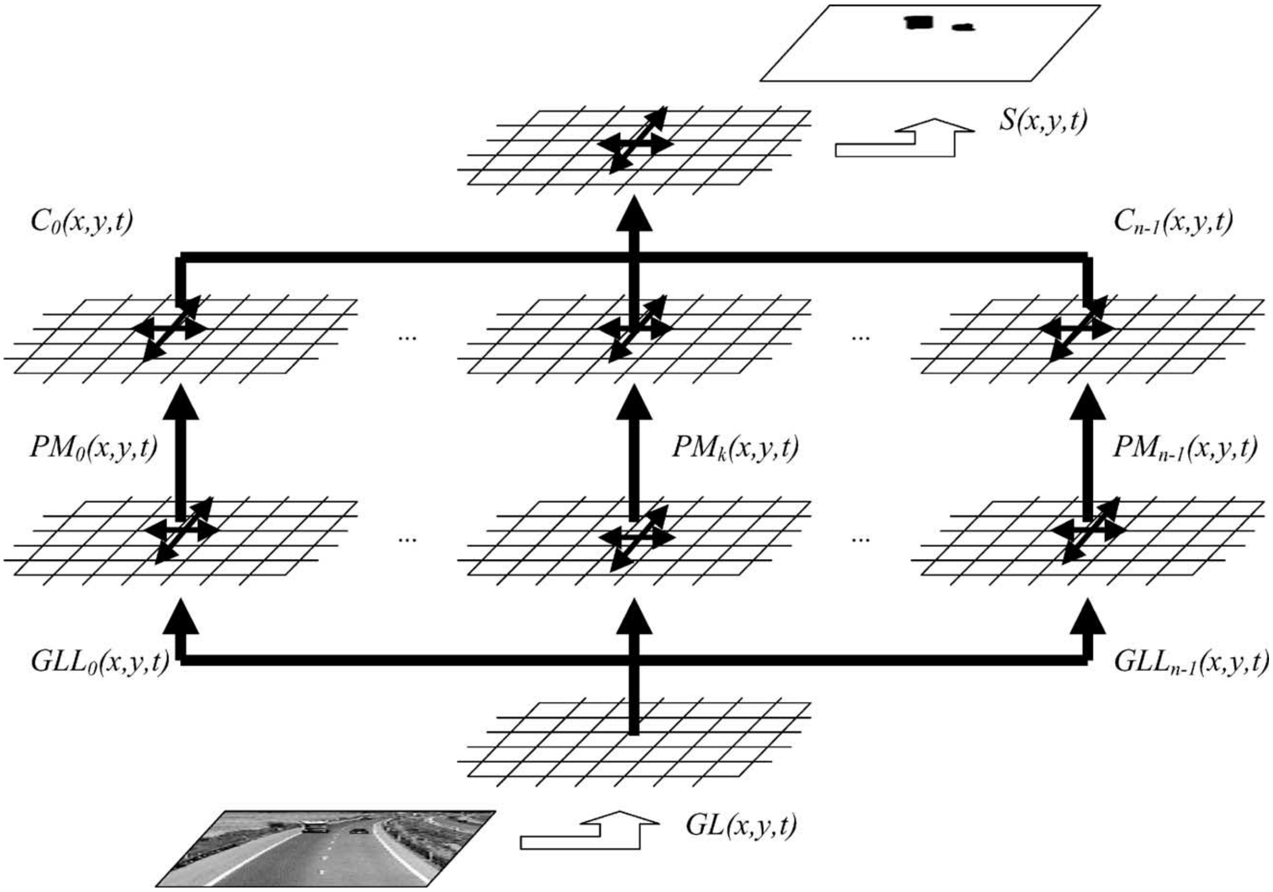

Fig. 3. Multi-layer configuration.

evaluating if there is motion in a gray level band for a given pixel. This task is performed in the following layer.

It must be clear that one, and only one output of all the detecting modules of the gray level bands can be activated at a given instant. This fact, although obvious, is of a great relevance at higher layers of the architecture, since it will prevent possible conflicts among the values offered by the different gray level bands. Indeed, only one gray level band will contain valid values.

3.2. Layer 1: lateral interaction for accumulative computation

At this layer a series of connections of modular structures in a mesh form are proposed (Fig. 5). This way all lines will be interconnected to each other, and so will the columns. It is also necessary to keep in mind that this layer is made up of n sub-layers (as many as chosen gray level bands).

Each node in the mesh can be considered as the basic structure. Lateral connections are called ACT1.

We can algorithmically formulate the behavior desired for our modules lying on five different steps.

3.2.1. Step 1

Step 1 is performed at global time scale t. Permanence memory charge or discharge is accomplished by motion detection. This information, given as an input from layer 0, is associated to sub-layer k in layer 1 (gray level

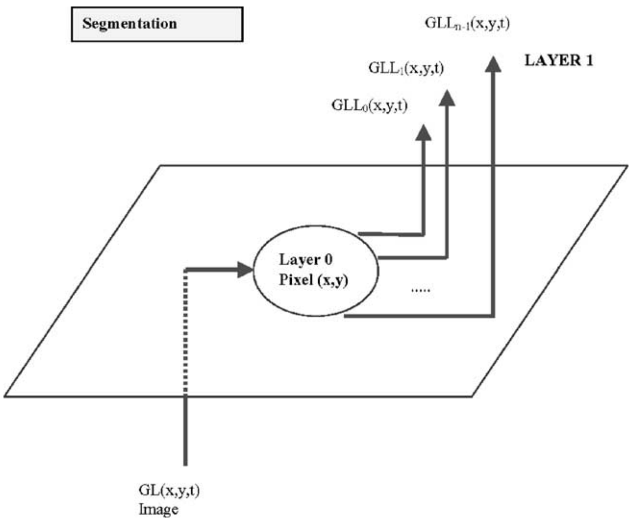

Fig. 4. Layer 0. External connections.

band k)

$$\mathrm{PM}_k(x,y,t) = \begin{cases} v_{\mathrm{dis}}, & \text{if}\,\mathrm{GLL}_k(x,y,t) \equiv 0 \\ v_{\mathrm{sat}}, & \text{if}\,\mathrm{GLL}_k(x,y,t) \equiv 1 \cap \mathrm{GLL}_k(x,y,t-\Delta t) \equiv 0 \\ \mathrm{PM}_k(x,y,t-\Delta t) - v_{\mathrm{dm}}, & \text{if}\,\mathrm{GLL}_k(x,y,t) \equiv 1 \cap \mathrm{GLL}_k(x,y,t-\Delta t) \equiv 1 \end{cases}$$

where v_{dm} is the discharge value due to motion detection, if this sub-layer k is informed that pixel (x,y) belongs to the gray level band k. Note that Δt determines the sequence frame rate and is given by the capacity of the model's implementation to process one input image. This sequence frame will greatly depend on the figure size.

There are three possibilities at each element (x, y)

- The sub-layer does not correspond to the gray level band of the image pixel, and the permanence value is discharged down to value v_{dis}.
- The sub-layer corresponds to the gray level band of the image pixel at time instant t, and it did not correspond to the gray level band at the previous instant $t - \Delta t$. The permanence value is loaded to the maximum of saturation v_{sat}.
- The sub-layer corresponds to the gray level band of the image pixel at time instant t, and it did also correspond to the gray level band at instant $t - \Delta t$. The permanence value is discharged the value $v_{>\mathrm{dm}}$ (discharge value due to motion detection); of course, the permanence value cannot be under a minimum value v_{dis}

$$\mathrm{PM}_k(x,y,t) = \begin{cases} \mathrm{PM}_k(x,y,t), & \text{if } \mathrm{PM}_k(x,y,t) > v_{\mathrm{dis}} \\ v_{\mathrm{dis}}, & \text{otherwise} \end{cases}$$

The discharge of a pixel by a quantity of v_{dm} is the way to stop paying attention to a pixel of the image that had captured our interest in the past. As it will be seen later on, if a pixel is not directly or indirectly bound by means of lateral interaction mechanisms to a maximally charged pixel (v_{sat}), it deceases to total discharge with time.

Step 1 also incorporates the setting at 1/0 of the variable OPEN_k

$$\mathrm{OPEN}_k(x,y,t) = \begin{cases} 1, & \text{if } v_{\mathrm{dis}} < \mathrm{PM}_k(x,y,t) < v_{\mathrm{sat}} \\ 0, & \text{otherwise} \end{cases}$$

The meaning of this variable is as follows

- The variable at 0 indicates that the structure has closed its input lateral interaction channels, and therefore, it will not accept any stimulus from the neighboring elements;

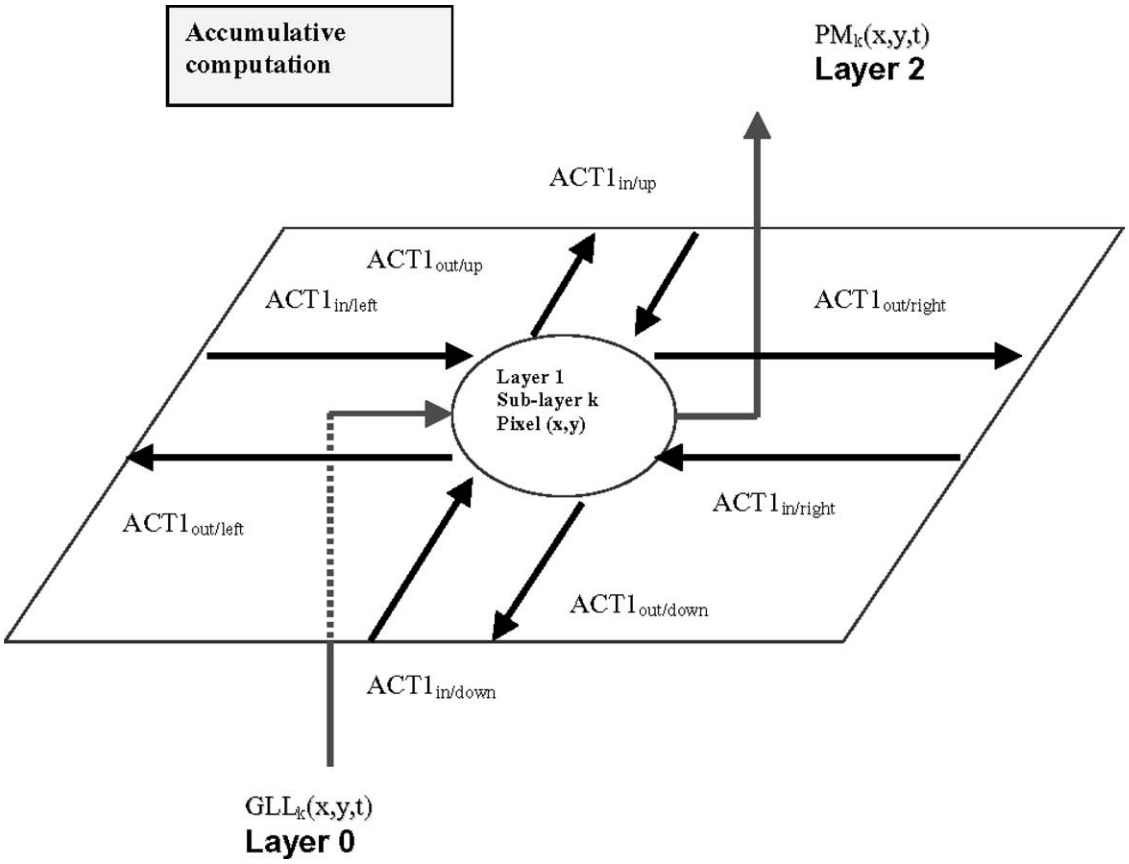

Fig. 5. Layer 1. Sub-layer k. External connections.

the variable takes this value when the permanence memory value is either totally charged or totally discharged.

- The variable at 1 indicates that the structure has opened its input lateral interaction channels to receive any stimulus from the neighboring elements; the variable takes this value when the permanence memory value is charged, but not saturated.

3.2.2. Step 2

In Step 2 those pixels with maximum permanence value (saturated pixels) inform their neighbors through specific channels, that is, the channels of type $\mathrm{ACT1}_{\mathrm{out}}$

$$\mathrm{ACT1}_{\mathrm{out/up}} = \mathrm{ACT1}_{\mathrm{out/down}} = \mathrm{ACT1}_{\mathrm{out/right}} = \mathrm{ACT1}_{\mathrm{out/left}} = \begin{cases} 1, & \text{if } \mathrm{PM}_k(x,y,t) \equiv v_{\mathrm{sat}} \\ 0, & \text{otherwise} \end{cases}$$

These two previous steps occur in normal time space t. The two following steps occur in an iterative way in a different space of time $T \ll t$. The value of ΔT will determine the number of times the mean value is calculated. Notice that the relation between ΔT and Δt will establish the influence outreach of saturated pixels.

3.2.3. Step 3

In Step 3 an extra charge v_{rv} (charge value due to vicinity) is added to the permanence memory in those image pixels that receive an $\mathrm{ACT1}_{\mathrm{in}}$ stimulus from any of the four neighboring pixels. This can only be performed if a series of requirements are met. These conditions are met where lateral activation occurs. Evidently, the permanence memory cannot be loaded above the maximum value v_{sat}.

Notice that the permanence memory can only be recharged once. This fact is handled by means of the variable OPEN_k.

```
IF ($\mathrm{OPEN}_k(x,y,T) == 1$) THEN {
    $\mathrm{PM}_k(x,y,T) = \mathrm{PM}_k(x,y,T-\Delta T) + v_{\mathrm{rv}}$,
    $\mathrm{OPEN}_k(x,y,T) = 0$, if $\mathrm{ACT1}_{\mathrm{in/up}} \equiv 1 \cup \mathrm{ACT1}_{\mathrm{in/down}} \equiv 1 \cup \mathrm{ACT1}_{\mathrm{in/right}} \equiv 1 \cup \mathrm{ACT1}_{\mathrm{in/left}} \equiv 1$
    $\mathrm{PM}_k(x,y,T) = v_{\mathrm{sat}}$, if $\mathrm{PM}_k(x,y,T) > v_{\mathrm{sat}}$
}
```

This last recharge mechanism is the lateral interaction mechanism at layer 1 level (for each sub-layer of gray level band), and allows maintaining an active attention in the pixels with a certain charge. This mechanism is even able to reinforce the permanence memory value if the value of v_{rv} is greater than that of v_{dm}.

3.2.4. Step 4

Step 4 is similar to Step 2. The difference stands in that not only the maximally charged pixels are contemplated, but also those with an intermediate charge, and previously warned by the lateral input signals to retransmit the signals received

$$\mathrm{ACT1}_{\mathrm{out/up}} = \mathrm{ACT1}_{\mathrm{out/down}} = \mathrm{ACT1}_{\mathrm{out/right}} = \mathrm{ACT1}_{\mathrm{out/left}} \begin{cases} 1, & \text{if } \mathrm{PM}_k(x,y,T) \equiv v_{\mathrm{sat}} \cup \mathrm{PM}_k(x,y,T) > v_{\mathrm{dis}} \cap (\mathrm{ACT1}_{\mathrm{in/up}} \equiv 1 \cup \mathrm{ACT1}_{\mathrm{in/down}} \\ & \equiv 1 \cup \mathrm{ACT1}_{\mathrm{in/right}}(x,y,T) \equiv 1 \cup \mathrm{ACT1}_{\mathrm{in/left}} \equiv 1) \\ 0, & \text{otherwise} \end{cases}$$

3.2.5. Step 5

Again at global time scale t, the permanence value at each pixel (x,y) is thresholded and sent to the next layer

$$\mathrm{PM}_k(x,y,t) = \begin{cases} \mathrm{PM}_k(x,y,t), & \text{if } \mathrm{PM}_k(x,y,t) > \theta_{\mathrm{per}} \\ \theta_{\mathrm{per}}, & \text{otherwise} \end{cases}$$

In order to explain the central idea of layer 1, we will say that the activation toward the lateral modular structures (up, below, right and left) is based on the following basic ideas

1. All modular structures with maximum permanence value v_{sat} (saturated) inform their neighbors (they output the charge toward the neighbors).
2. All modular structures with non-saturated charge value that have been activated from some neighbor, allow this information to pass through them (they behave as transparent structures to the charge passing).
3. The modular structures with minimum permanence value v_{dis} (discharged) stop the passing of the charge information toward the neighbors (they behave as opaque structures).

Therefore, an explosion of lateral activation takes place starting at the structures with permanence memory set at v_{sat}, and it spreads in the direction of its four closer neighbors, until a structure with discharged permanence memory appears in the pathway.

Table 1 shows how to appropriately use the relationship between the values of v_{dm} and v_{rv} depending on the objectives proposed.

3.3. Layer 2: lateral interaction for charge redistribution by gray level bands

Starting from the values of the permanence memory in each pixel on a gray level band basis, we will experience how it is possible to obtain all the parts of an

Table 1
Appropriate use of relationship between v_{dm} and v_{rv}

Relationship between v_{dm} and v_{rv}	Explanation
$v_{dm} \leq v_{rv}$	All pixels with a permanence value between v_{dis} and v_{sat} and directly or indirectly connected to pixels with value v_{sat}, take a new value v_{sat}. The pixel is part of the object while any pixel of the object moves
$v_{dm} > v_{rv}$	All pixels with a permanence value between v_{dis} and v_{sat}, and directly or indirectly connected to pixels with value v_{sat}, will discharge slowly. The pixels that are far away from the maximally charged pixels (motion 'center') will be slowly disassociated from the object
$v_{dm} \gg v_{rv}$	All pixels with a permanence value between v_{dis} and v_{sat}, and directly or indirectly connected to pixels with value v_{sat}, will discharge quickly. The object will be formed by the pixels that have moved recently

object in movement. An object part concretely means the union of pixels that are together and in a same gray level band.

The discrimination of each part composing the object is equally obtained by lateral cooperation mechanisms. Again we will connect the modular structures of this layer in a mesh form. Once again, notice that there are as many sub-layers in this layer 2 as gray level bands defined.

At layer 2 the charge will be homogenized among all the pixels in the same gray level band that are directly or indirectly connected to each other.

Thus, a double objective will be satisfied

1. Diluting the charge due to the false image background motion along the other pixels of the background. This way, there should be no presence of background motion, but we will keep motion of the objects present in the scene.
2. Obtaining a parameter common to all the pixels of the part of the object with the same gray level band. This common parameter will be sent to higher levels (layer 3, in principle) for processing purposes.

The modular structure connections at this level can be seen just as they are shown in Fig. 6, where the lateral connections are called ACT2.

The algorithms of layer 2 are also to be explained in four different steps. Steps 1 and 4 occur on time scale *t*, whereas steps 2 and 3 are in time scale *T*.

3.3.1. Step 1

Initially, the charge value at each pixel (x, y) and at each sub-layer k is given the value of the permanence value from

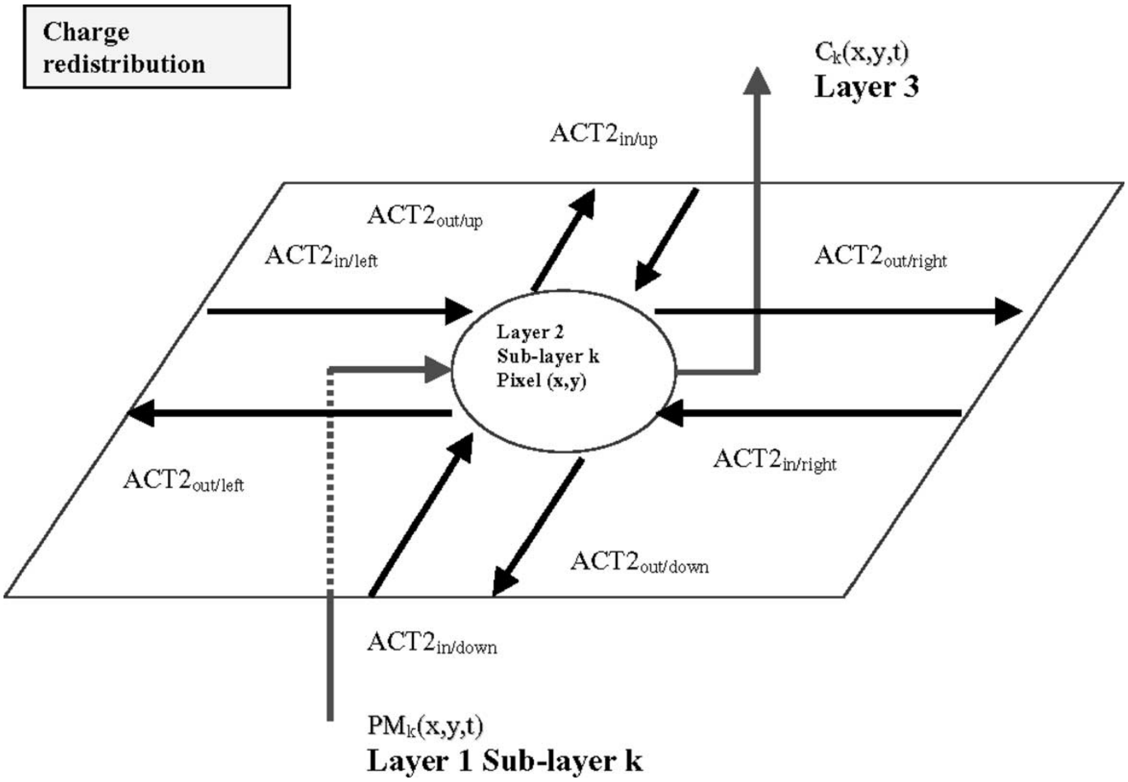

Fig. 6. Layer 2. Sub-layer *k*. External connections.

the previous layer

$$C_k(x,y,t) = \mathrm{PM}_k(x,y,t)$$

3.3.2. Step 2

Recursively, the charge value is spread toward the four neighbors

$$\mathrm{ACT2}_{\mathrm{out/up}} = \mathrm{ACT2}_{\mathrm{out/down}} = \mathrm{ACT2}_{\mathrm{out/right}} = \mathrm{ACT2}_{\mathrm{out/left}} = C_k(x,y,T)$$

3.3.3. Step 3

Provided that the neighbor input charge values are high enough, the center element (x,y) calculates the mean of its value and the neighbors partially charged

$$C_{\mathrm{up}} = \begin{cases} \mathrm{ACT2}_{\mathrm{in/up}}, & \text{if } \mathrm{ACT2}_{\mathrm{in/up}} > \theta_{\mathrm{per}} \\ 0, & \text{otherwise} \end{cases}$$

$$C_{\mathrm{down}} = \begin{cases} \mathrm{ACT2}_{\mathrm{in/down}}, & \text{if } \mathrm{ACT2}_{\mathrm{in/down}} > \theta_{\mathrm{per}} \\ 0, & \text{otherwise} \end{cases}$$

$$C_{\mathrm{right}} = \begin{cases} \mathrm{ACT2}_{\mathrm{in/right}}, & \text{if } \mathrm{ACT2}_{\mathrm{in/right}} > \theta_{\mathrm{per}} \\ 0, & \text{otherwise} \end{cases}$$

$$C_{\mathrm{left}} = \begin{cases} \mathrm{ACT2}_{in/left}, & \text{if } \mathrm{ACT2}_{\mathrm{in}/left} > \theta_{\mathrm{per}} \\ 0, & \text{otherwise} \end{cases}$$

$$C_k(x,y,T) = \mathrm{mean}(C_k(x,y,T-\Delta T) + C_{\mathrm{up}} + C_{\mathrm{down}} + C_{\mathrm{right}} + C_{\mathrm{left}})$$

3.3.4. Step 4

Back to global time scale t, the charge value at each pixel (x,y) is threshold and sent to the next layer

$$C_k(x,y,t) = \begin{cases} C_k(x,y,t), & \text{if } C_k(x,y,t) > \theta_{\mathrm{ch}} \\ \theta_{\mathrm{ch},} & \text{otherwise} \end{cases}$$

3.4. Layer 3: lateral interaction for moving object fusion

Up to this moment, attention has been captured on any moving object in the scene by means of cooperative calculation mechanisms in all gray level bands. Motion due to the background has also been eliminated. Now a new objective must be set to clearly distinguish the different objects as a whole. Properly spoken this is not a classification preceded by a previous supervised learning, but rather an auto-classification based on the characteristics found on layer 2. In other words, it is non-supervised learning.

Object discrimination is achieved equally by lateral cooperation mechanisms. The modular structures at this layer are again connected in a mesh form. Nevertheless, it is not arranged in sub-layers, but rather all the information of the n sub-layers of layer 2 ends up in a single layer.

At layer 3, the charge values are homogenized among all the pixels that contain some charge value over a minimum threshold and that are physically connected to each other. Lateral connections are called ACT3 in this layer.

The connections of all modular structures of this level can be seen just as they are shown in Fig. 7.

The algorithmic behavior of the modular structure for each one of the image pixels is described next.

3.4.1. Step 1

Initially, we define the silhouette charge value at each pixel (x,y) to be the charge value of the only charged sub-layer k from the previous layer

$$\mathrm{FOR}(k = 1 \text{ to } n) S(x,y,t) = \max(C_k(x,y,t))$$

3.4.2. Step 2

Recursively, the charge value is spread toward the four neighbors

$$\mathrm{ACT3}_{\mathrm{out/up}} = \mathrm{ACT3}_{\mathrm{out/down}} = \mathrm{ACT3}_{\mathrm{out/right}} = \mathrm{ACT3}_{\mathrm{out/left}} = S(x,y,T)$$

3.4.3. Step 3

Provided that the neighbor input charge values are high enough, the center element (x,y) calculates the mean of its value and the neighbors partially charged

$$C_{\mathrm{up}} = \begin{cases} \mathrm{ACT3}_{\mathrm{in/up}}, & \text{if } \mathrm{ACT3}_{\mathrm{in/up}} > \theta_{\mathrm{ch}} \\ 0, & \text{otherwise} \end{cases}$$

$$C_{\mathrm{down}} = \begin{cases} \mathrm{ACT3}_{\mathrm{in/down}}, & \text{if } \mathrm{ACT3}_{\mathrm{in/down}} > \theta_{\mathrm{ch}} \\ 0, & \text{otherwise} \end{cases}$$

$$C_{\mathrm{right}} = \begin{cases} \mathrm{ACT3}_{\mathrm{in/right}}, & \text{if } \mathrm{ACT3}_{\mathrm{in/right}} > \theta_{\mathrm{ch}} \\ 0, & \text{otherwise} \end{cases}$$

$$C_{\mathrm{left}} = \begin{cases} \mathrm{ACT3}_{\mathrm{in/left}}, & \text{if } \mathrm{ACT3}_{\mathrm{in/left}} > \theta_{\mathrm{ch}} \\ 0, & \text{otherwise} \end{cases}$$

$$S(x,y,T) = \mathrm{mean}(S(x,y,T-\Delta T) + C_{\mathrm{up}} + C_{\mathrm{down}} + C_{\mathrm{right}} + C_{\mathrm{left}})$$

3.4.4. Step 4

Back to global time scale t, the silhouette charge value at each pixel (x,y) is thresholded and sent to the next layer

$$S(x,y,t) = \begin{cases} S(x,y,t), & \text{if } S(x,y,t) > \theta_{\mathrm{obj}} \\ \theta_{obj}, & \text{otherwise} \end{cases}$$

4. Learning algorithm in lateral interaction in accumulative computation

Learning in lateral interaction in accumulative computation starts from the knowledge of the influence of the basic

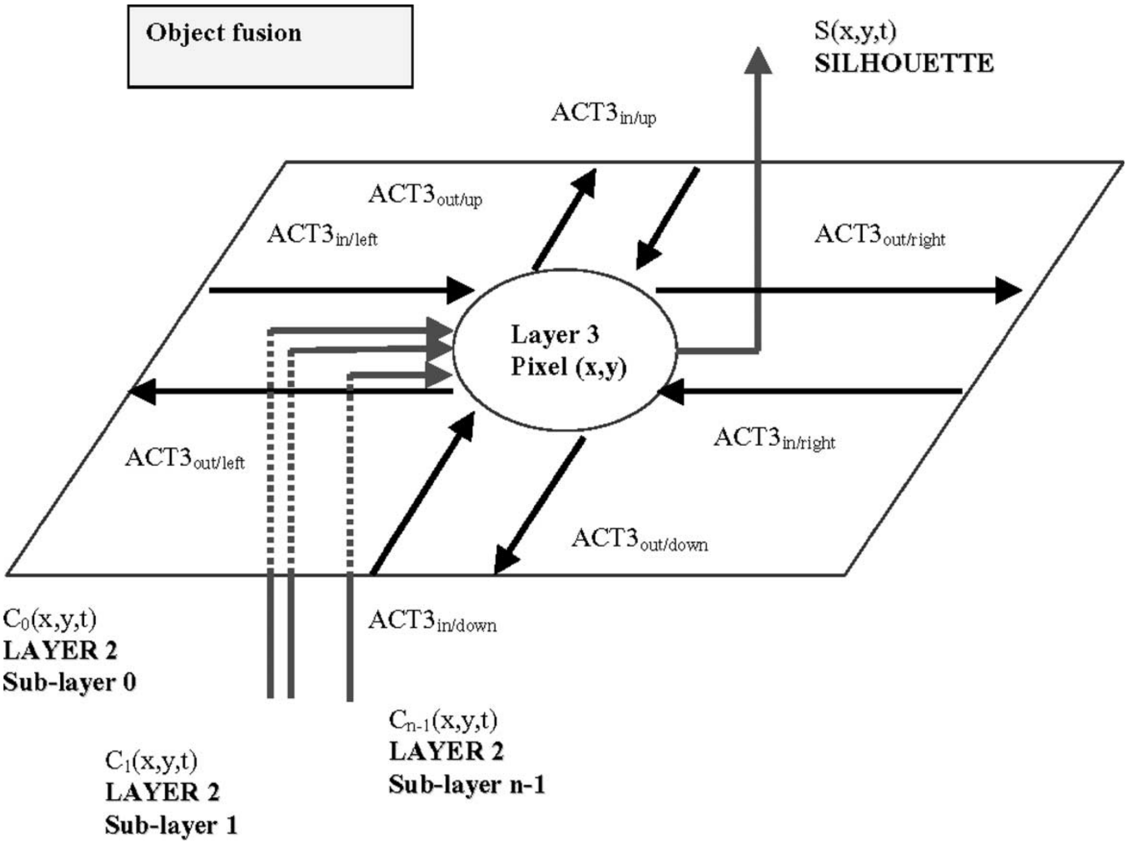

Fig. 7. Layer 3. External connections.

parameters of the model. Learning in lateral interaction in accumulative computation model consists in adjusting the parameters of the diverse layers to offer the best processing result of the image sequence when obtaining the silhouettes of moving elements present in the scene.

During the learning process, previous to the normal operation process, the architecture is offered an input image sequence, as well as the following reinforcement parameters (see Fig. 8):

- *Number of moving elements*. (S_{m}) to be detected in the sequence. This parameter is fixed for a scene and must be given by the user.
- *Maximum size of a silhouette*. ($S_{\max}$) to be detected in the sequence
- *Minimum size of a silhouette*. ($S_{\min}$) to be detected in the sequence.

Due to its simplicity, it does not seem necessary to explain the reinforcement parameter *Number of moving elements* (S_{m}). The other two parameters arise from the domain knowledge of lateral interaction in accumulative computation model. It is indispensable to introduce parameters *Maximum size of a silhouette* ($S_{\max}$) and *Minimum size of a silhouette* ($S_{\min}$) to capture the attention on those objects whose silhouette falls between these two magnitudes. Notice that by varying these two parameters it is possible to obtain very different results. One may, for example, center the attention on pedestrians or on cars in a same visual surveillance scene.

Learning turns, in our case, into an iterative process where, for a given scene, the model is nurtured by the same image sequence, just modifying the basic parameters until the number of silhouettes obtained at layer 3 is close enough to Number of moving elements (S_{m}). The output obtained at layer 3 is *called Number of Detected Silhouettes* (S_{d}).

The basic parameters of lateral interaction in accumulative computation model have been classified into two groups:

(a) *Parameters with constant values that do not evolve during the learning phase*. These are v_{dis} (minimum permanence value) and v_{sat} (maximum permanence value) at layer 1.
(b) *Parameters with values that do evolve during the learning phase*. These are: n (number of gray level bands) at layer 0; v_{dm} (discharge value due to motion detection), v_{rv} (recharge value due to vicinity), and, θ_{per} (threshold) at layer 1; θ_{ch} (threshold) at layer 2; θ_{obj} (threshold) at layer 3.

Thus, we use an error minimization function. The problem is now to find a procedure of estimating a set of

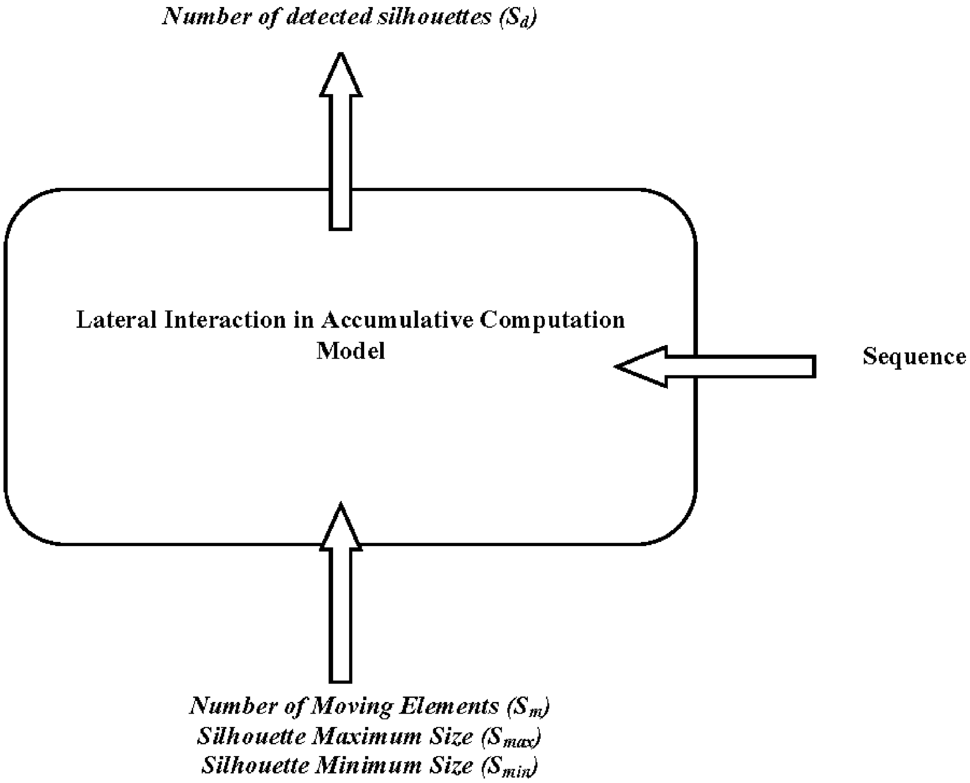
Number of detected silhouettes (S_d)
Lateral Interaction in Accumulative Computation Model
Sequence
Number of Moving Elements (S_m)
Silhouette Maximum Size (S_{max})
Silhouette Minimum Size (S_{min})

Fig. 8. Inputs and outputs during learning phase.

values that best leads to the desired solution. In other words, we have to look for a set of optimal values

$$C^* = (n^*, v^*_{\mathrm{dm}}, v^*_{\mathrm{rv}}, \theta^*_{\mathrm{per}}, \theta^*_{\mathrm{ch}}, \theta^*_{\mathrm{obj}})$$

that minimize error function

$$E = \left| S_{\mathrm{m}} - \frac{1}{k} \sum_{t=0}^{k} S_{\mathrm{d}}(t) \right|$$

where k is the number of images that form the learning sequence, S_{m} is the number of moving elements to be detected (constant through the whole training sequence), $S_{\mathrm{d}}(t)$ is the number of detected silhouettes at time instant t.

5. Results

We shall demonstrate the usefulness of our neural network architecture in four layers with some examples. Some input sequences have been obtained from our own research team. The rest are image sequences available from some educational Internet web sites. Note that only three

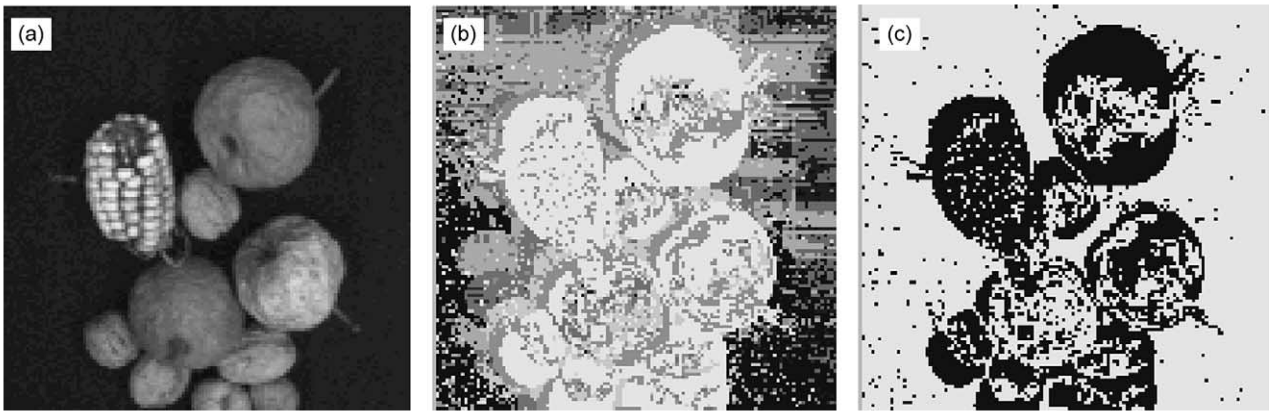

Fig. 9. (a) One image of the Pears and nuts image sequence from the MOVI Image Base; (b) result after Layer 1; (c) result after Layer 3.

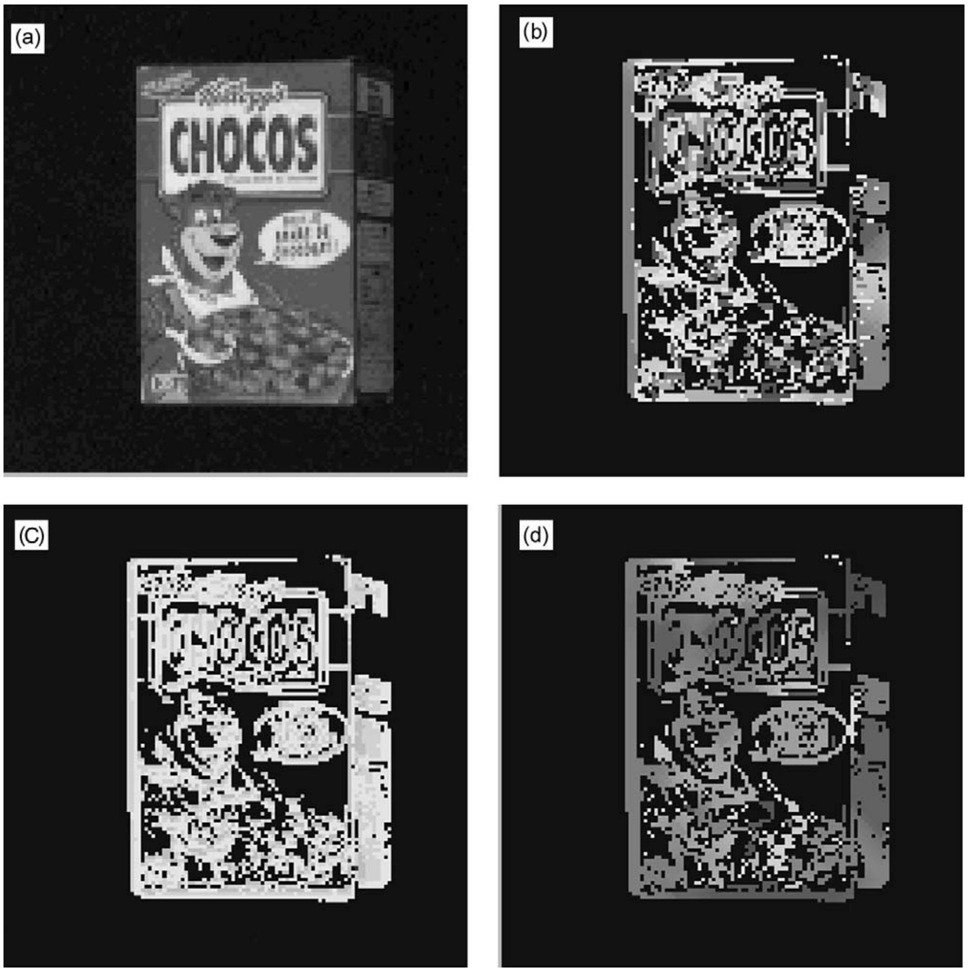

Fig. 10. (a) One image of the Chocos image sequence from the MOVI Image Base; (b) result after Layer 1; (c) result after Layer 2; (d) result after Layer 3.

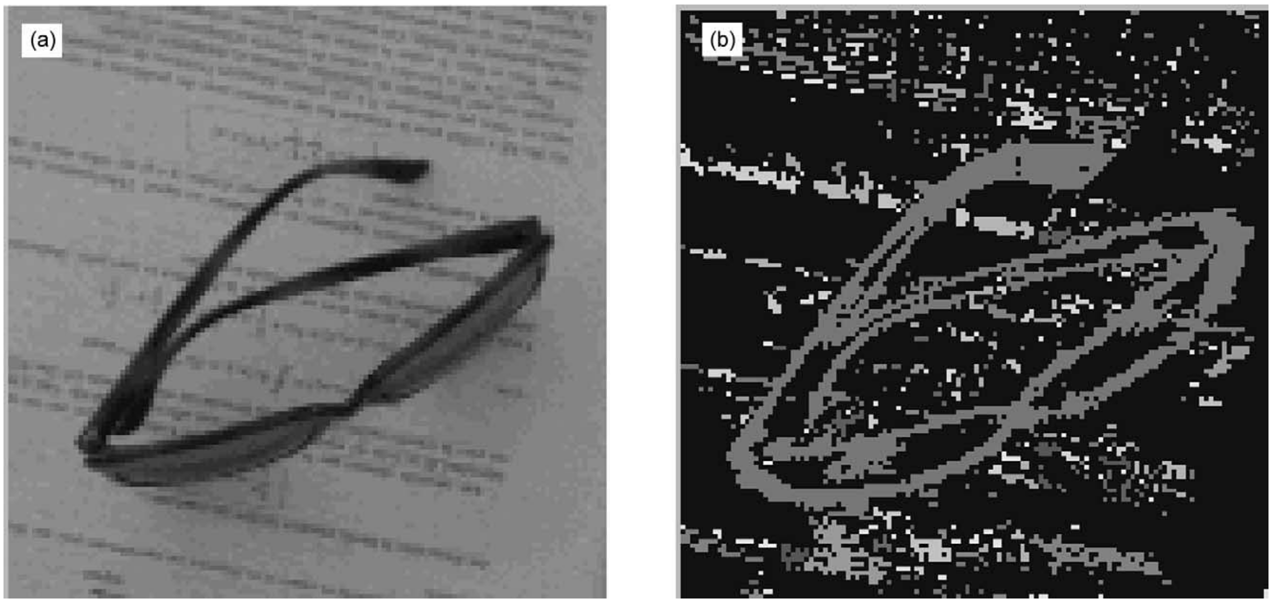

Fig. 11. (a) One image of the *Sun glasses over printed paper* image sequence from the MOVI Image Base; (b) result after Layer 3.

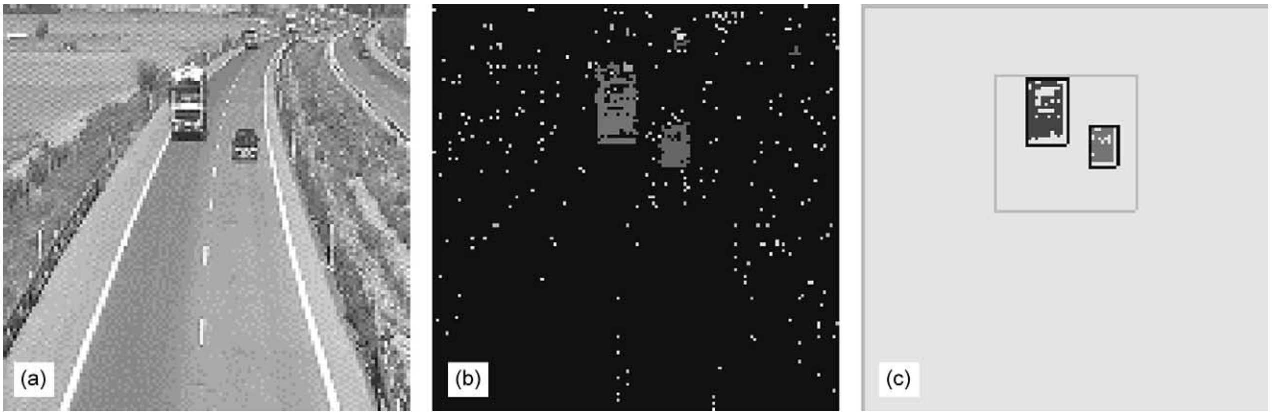

Fig. 12. (a) Road-traffic monitoring image; (b) result after Layer 3; (c) high-level processing.

frames are needed to obtain accurate segmentation results for any of the following study cases. Study cases 1 to 3 make use of some image sequences from the *MOVI Image Base*, which offer complex motion situations (translation, or translation plus rotation) due to camera movement. These study cases demonstrate the usefulness and versatility of our method to differentiate figure from ground (case 1), to obtain the parts of an object (case 2), or to obtain an object as a whole (case 3), just by using different parameter values to segment from inherent motion of the image sequences. Notice, nevertheless, the proposed method offers its best results when working with a stationary camera (study cases 4–7). This does not mean that the background must be stationary, as we will appreciate in the examples offered.

The values of the most important parameters for these experiments were $\Delta t = 0.42$ s (reached frame rate), Δt ranging from 8 to $64\Delta T$, $v_{\text{dis}} = 0$, and $v_{\text{sat}} = 255$. The learning phase performance has taken an average of 5 min for a sequence of 50, 256 × 256 pixels image.

5.1. Study case 1

The first study case shows the capacity of our model to separate moving objects from background. The *pears and nuts* image sequence from the MOVI Image Base offers complex motion to test segmentation algorithms. This sequence contains images where the camera position and orientation varies slowly from one image of a sequence to the next one. Our neural network architecture is capable of segmenting the images into figures and ground (Fig. 9). Of course, this segmentation capacity may be considered as trivial, as the background is black. But, this is just one possibility of our implementation. Next cases will show more possibilities of our method.

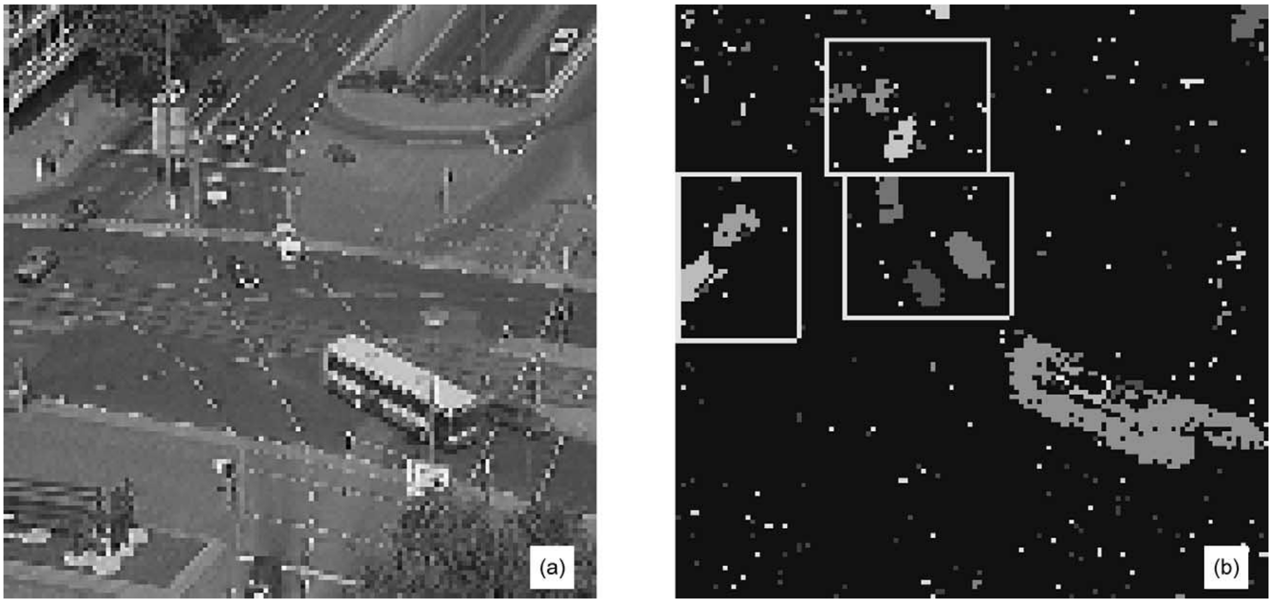

Fig. 13. (a) An image of the traffic intersection sequence at the Ettlinger-Tor in Karlsruhe; (b) result after Layer 3.

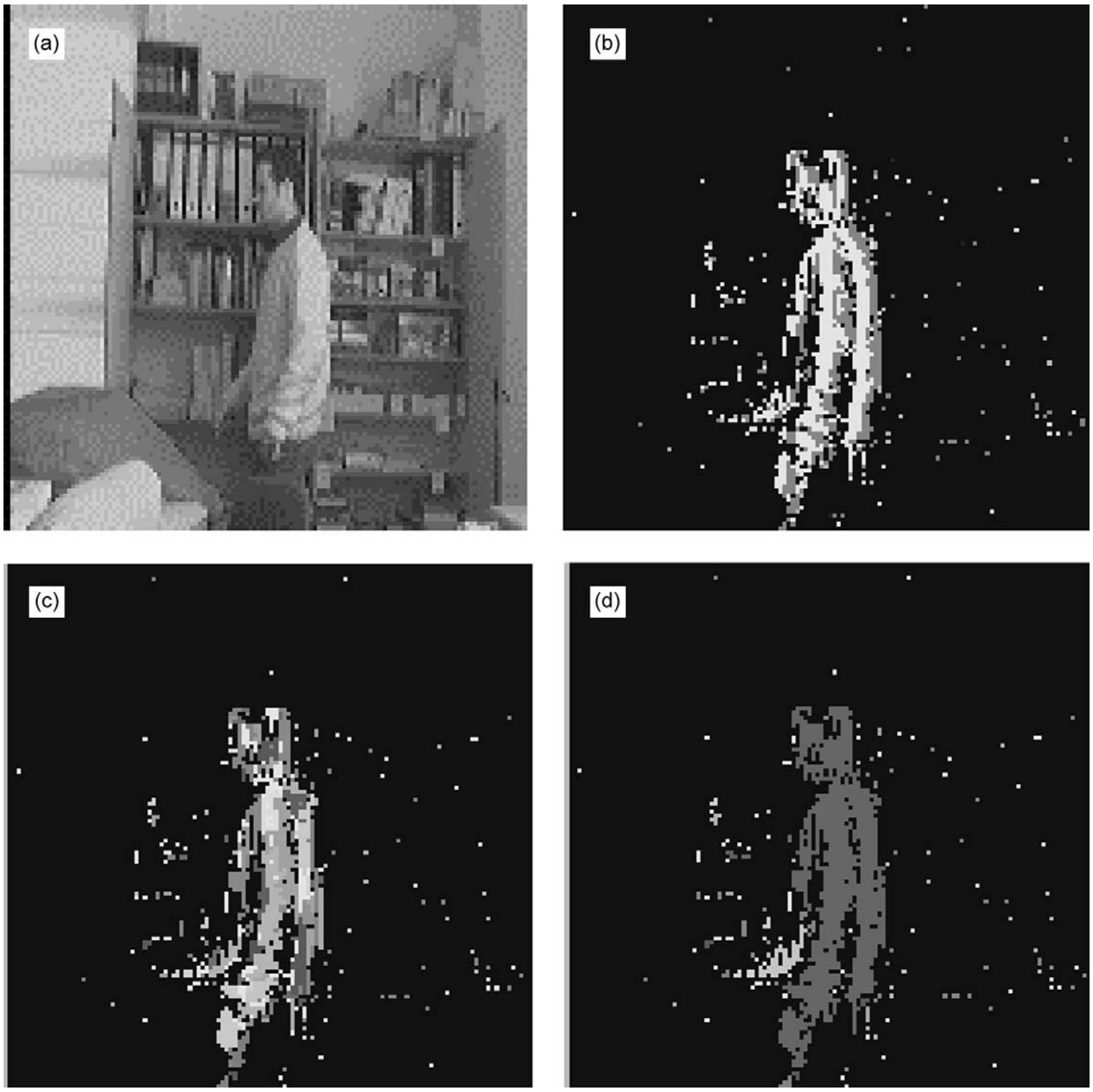

Fig. 14. (a) A man walking in a laboratory; (b) result after Layer 1; (c) result after Layer 2; (d) result after Layer 3.

5.2. Study case 2

Objects as a *Chocos* cereal box performing complex motion (rotations) may be segmented in its constituent parts by means of our lateral interaction in accumulative computation (Fig. 10). Again, this sequence contains images where camera position and orientation varies slowly in a complex way (translation plus rotation) from one image of a sequence to the next one. This study case shows the versatility of our implementation for segmenting moving elements as a whole or as segmenting moving elements constituent parts. The degree of decomposition depends on the set of values used in a specific implementation. In this case, the results obtained are probably non-sense. This way, we show the importance of the learning algorithms to adapt the method's parameters correctly. Notice that by varying the values of the method's parameters it is possible to get more or less details of the object's parts. Thus, we might offer as result of our motion detection algorithms a wide range of possibilities, going from simple image difference (pixels that have moved from one image to the next) up to object silhouettes.

5.3. Study case 3

The third study case shows the robustness of our architecture for discriminating objects from the motion of the entire environment. In this sequence (*Sun glasses over printed paper*), the camera position varies slowly from one image of a sequence to the next one by performing a linear translation along the optical axis. This discrimination, as already mentioned, is related to the connectivity of the constituent parts of the objects.

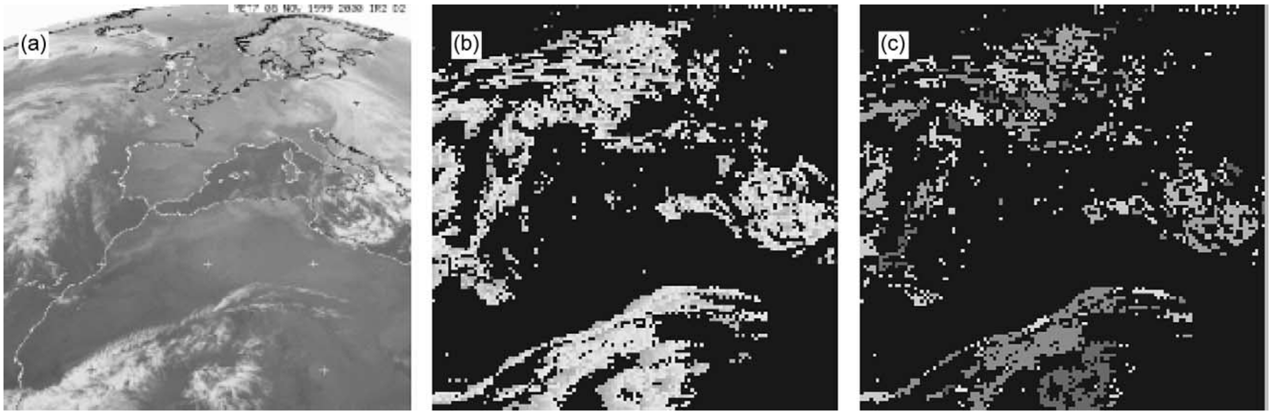

Fig. 15. (a) A METEOSAT satellite image; (b) result after Layer 1; (c) result after Layer 3.

The sunglasses from camera motion are perfectly segmented (Fig. 11).

5.4. Study case 4

Evidently, when testing our proposed model with images with a quite static background, the results are astonishingly good. A stationary camera on a highway permits to obtain all vehicles running on the scene (Fig. 12). When adding high-level processing dependent on the kind of application, the neural architecture may be exploited with excellent results (Fig. 12(c)). As our architecture is independent from image understanding, it may be used for many different image analysis applications.

5.5. Study case 5

Next, we offer the results obtained for the traffic intersection sequence recorded at the Ettlinger-Tor in Karlsruhe by a stationary camera.

This example shows the usefulness of our neural architecture for traffic monitoring in complex intersection situations. Note also that there is a lot of noise due to the vibration of the stationary camera. Nevertheless, the results are excellent. Fig. 13(a) shows one image of the sequence. You can observe the existence of ten cars and one bus driving in three different directions. At the bottom of the image there is another car, but this one is still. Fig. 13(b) shows the result of applying our model to some images of

Table 2
Comparison to other approaches

Approach	Description/comparison	Reference
Image difference approaches	Up to some extent, our method can be generically classified into the models based on image difference. But our method is much stronger than simple image difference, and even cumulative image difference. Compared to both algorithms, our lateral interaction in accumulative computation model offers more accurate and less noisy results	Fernandez and Mira (1992) and Simoncelli (1993)
Gradient-based approaches	The gradient-based estimates have become the main approach in the applications of computer vision. These methods are computationally efficient and satisfactory 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. 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	Fennema and Thompson (1979), Horn and Schunck (1981), Lawton (1989) and Marr and Ullman (1989)
Region-based approaches	These approaches work with 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	Adams and Bischof (1994), Horowitz and Pavlidis (1976), Revol and Jourlin (1997) and Zucker (1976)

the traffic intersection sequence. As you may observe, the system is perfectly capable of segmenting all the moving elements present on Fig. 13(a).

5.6. Study case 6

Our system has also been tested as a visual surveillance tool. Fig. 14 shows the possibility of obtaining the silhouettes of people walking through a scene.

5.7. Study case 7

Note the versatility of our architecture. Any high-level application founded basically on motion detection can make use of our lateral interaction in accumulative computation model in its low-level stages. Here we offer the possibility to manage satellite images (Fig. 15).

6. Discussion

A model based on a neural architecture close to biology has been proposed in this paper. A simple algorithm of lateral interaction in accumulative computation is capable of detecting all rigid and non-rigid moving objects in an indefinite sequence of images in a robust and coherent manner. The method has been tested on a wide range of real images. The results are especially relevant when applied to image sequences taken from a stationary camera. In fact, only very simple segmentations can be achieved when using a moving camera. A general comparison to other approaches is offered in Table 2.

Compared to all other approaches, our proposed model has no limitation in the number of non-rigid objects to differentiate. Our system facilitates object classification by taking advantage of the object charge value, common to all pixels of the same moving element. Thanks to this fact, any higher-level operation will decrease in difficulty.

We conclude stating that the proposed neuronal lateral interaction in accumulative computation mechanisms offer an excellent tool for image segmentation as a first approach to pattern recognition. Currently, we are studying the usefulness of our algorithms for very different real world applications such as traffic monitoring, people surveillance, and medical imaging.

Acknowledgements

The authors wish to thank the INRIA Rhône-Alpes for their courtesy in offering the MOVI Image Base available at web site http://www.inrialpes.fr/movi/pub/Images/. Traffic intersection sequence at the *Ettlinger-Tor* in *Karlsruhe*, Copyright© 1998 by H.-H. Nagel, Institut für Algorithmen und Kognitive Systeme, Fakultät für Informatik, Universität Karlsruhe (TH), Postfach 6980, D-76128 Karlsruhe, Germany, http://i21www.ira.uka.de/image_sequences/. The satellite images have been used from http://www.eumetsat.de/ under Copyright© 2001 EUMETSAT.

References

Adams, R., & Bischof, L. (1994). Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *16*, 641–647.

Adelson, E. H., & Bergen, J. R. (1985). Spatiotemporal energy models for the perception of motion. *Journal of the Optical Society of America A*, *2*, 284–299.

Albright, T. (1992). Form-cue invariant motion processing in primate visual cortex. *Science*, *255*, 1141–1143.

Allman, J., Miezin, F., & McGuinness, E. (1985). Direction- and velocity-specific responses from beyond the classical receptive field in the middle temporal visual area (MT). *Perception*, *14*, 105–126.

Andersen, T., & Siegel, R. (1990). Motion processing in the primate cortex. In G. Edelman, W. Gall, & W. Cowan (Eds.), *Signal and sense: Local and global order in perceptual maps* (pp. 131–141). New York: Wiley-Liss.

Bülthoff, H., Little, J., & Poggio, T. (1989). A parallel algorithm for real-time computation of optical flow. *Nature*, *337*, 549–553.

Faugeras, O. (1993). *Three-dimensional computer vision—A geometric pixelview*. Cambridge, MA: MIT Press.

Faugeras, O., Lustman, F., & Toscani, G. (1987). Motion and structure from motion from pixel and line matches. *Proceedings of the First International Conference on Computer Vision*, 25–34.

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

Fernandez, M. A., & Mira, J. (1992). *Permanence memory: A system for real time analysis in image sequences* (*92*). *IAPR Workshop on Machine Vision Applications MVA'92*, pp. 249–252.

Fernandez, M. A., Mira, J., Lopez, M. T., Alvarez, J. R., Manjares, A., & Barro, S. (1995). Local accumulation of persistent activity at synaptic level: Application to motion analysis. In J. Mira, & F. Sandoval (Eds.), *From natural to artificial neural computation* (pp. 137–143). *LNCS 930*, Berlin: Springer, IWANN'95.

Gerstner, W., Ritz, R., & van Hemmen, J. L. (1993). Why spikes? Hebbian learning and retrieval of time-resolved excitation patterns. *Biological Cybernetics*, *69*, 503–515.

Gilbert, C. D., Hirsch, J. A., & Wiesel, T. N. (1990). Lateral interactions in the visual cortex. *Cold Spring Harbor Symposium of Quantitative Biology*, *55*, 663–677.

Grossberg, S., & McLoughlin, E. (1997). Cortical dynamics of 3-D surface perception: Binocular and half-occluded scenic images. *Neural Networks*, *10*, 1583–1605.

Grossberg, S., & Rudd, M. (1989). A neural architecture for visual motion perception: Group and element apparent motion. *Neural Networks*, *2*, 421–450.

Hatsopoulos, N. G., & Warren, W. H. (1991). Visual navigation with a neural network. *Neural Networks*, *4*, 303–317.

Hildreth, E. C. (1984). *The measurement of visual motion*. Cambridge, MA: MIT Press.

Hildreth, E. C., & Royden, C. S. (1998). Motion perception. In M. Arbib (Ed.), *Handbook of brain theory and neural networks* (pp. 585–588). Cambridge, MA: MIT Press.

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

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

Horowitz, R. M., & Pavlidis, T. (1976). Picture segmentation by a tree traversal algorithm. *Journal of the ACM*, *23*, 368–388.

Lawton, T. B. (1989). Outputs of paired Gabor filters summed across the background frame of reference predict the direction of movement. *IEEE Transactions on Biomedical Engineering*, *36*, 130–139.

Marr, D. (1974). The computation of lightness by the primate retina. *Vision Research*, *14*, 1377–1388.

Marr, D. (1982). *Vision*. San Fransisco, CA: W.H. Freeman.

Marr, D., & Ullman, S. (1981). Directional selectivity and its use in early visual processing. *Proceedings of the Royal Society of London B*, *211*, 151–180.

Marshall, J. A. (1998). Motion perception: Self-organization. In M. Arbib (Ed.), *The handbook of brain theory and neural networks* (pp. 589–591). Cambridge, MA: MIT Press.

Mira, J. (1993). Computación neural en el camino visual. *Notas de Visión y Apuntes sobre la IngenierÍa del Software. Colección Estudios*, *24*, 175–197.

Mira, J., Delgado, A. E., Alvarez, J. R., de Madrid, A. P., & Santos, M. (1993). Towards more realistic self contained models of neurons: High-order, recurrence and local learning. In J. Mira, J. Cabestany, & A. Prieto (Eds.), *New trends in neural computation* (pp. 55–62). *LNCS 686*, LNCS: Springer, IWANN'93.

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

Mira, J., Delgado, A. E., Manjares, A., Ros, S., & Alvarez, J. R. (1996). Cooperative processes at the symbolic level in cerebral dynamics: Reliability and fault tolerance. In R. Moreno-Diaz, & J. Mira (Eds.), *Brain processes, theories and models* (pp. 244–255). Cambridge, MA: MIT Press.

Mitiche, A., & Bouthemy, P. (1996). Computation and analysis of image motion: A synopsis of current problems and methods. *International Journal of Computer Vision*, *19*(1), 29–55.

Moreno-Diaz, R., Rubio, F., & Mira, J. (1969). Aplicación de las transformaciones integrales al proceso de datos en la retina. *Revista de Automática*, *5*, 7–17.

Morrone, M., Burr, D., & Vaina, L. (1995). Two stages of visual processing for radial and circular motion. *Nature*, *376*, 507–509.

Mountcastle, V. B. (1979). An organizing principle for cerebral function: The unit module and the distributed system. In F. O. Schmitt, & F. G. Worden (Eds.), *The neuroscience fourth study program* (pp. 1115–1139). Cambridge, MA: MIT Press.

Revol, C., & Jourlin, M. (1997). A new minimum variance region growing algorithm for image segmentation. *Pattern Recognition Letters*, *18*, 249–258.

Ross, W. D., Grossberg, S., & Mingolla, E. (2000). Visual cortical mechanisms of perceptual grouping: Interacting layers, networks, columns, and maps. *Neural Networks*, *13*, 571–588.

Sekuler, R., & Blake, R. (1994). *Perception*. New York: McGraw-Hill.

Sereno, M. E. (1993). *Neural computation of pattern motion*. Cambridge, MA: MIT Press.

Shizawa, M. (1992). On visual ambiguities due to transparency in motion and stereo. *Lecture notes in computer science*, *599*, 411–419.

Simoncelli, E. P (1993). *Distributed representation and analysis of visual motion*. PhD dissertation, MIT.

Tekalp, A. M. (1995). *Digital video processing*. Englewood Cliffs, NJ: Prentice Hall.

Ullman, S. (1979). *The interpretation of visual motion*. Cambridge, MA: MIT Press.

Wallach, H. (1976). On perceived identity: 1. The direction of motion of straight lines. In H. Wallach (Ed.), *On perception*. New York: Quadrangle.

Wimbauer, S., Gerstner, W., & van Hemmen, J. L. (1994). Emergence of spatio-temporal receptive fields and its application to motion detection. *Biological Cybernetics*, *72*, 81–92.

Yuille, A. L., & Grzywacz, N. (1988). A computational theory for the perception of coherent visual motion. *Nature*, *333*, 71–74.

Zucker, S. W. (1976). Region growing: Childhood and adolescence. *Computer Graphics and Image Processing*, *5*, 382–399.