An MRF Based Motion Detection Algorithm Implemented on Analog Resistive Network

Franck Luthon, George V. Popescu, Alice Caplier

LTIRF, INPG, 46 avenue Félix-Viallet, 38031 Grenoble cedex, FRANCE
email : luthon@tirf.grenet.fr fax : (33) 76 57 47 90

Abstract. We present an algorithm based on MRF modelling for motion detection in image sequences and give a modified version for implementation on analog resistive network. Energy minimization is realized by a network relaxing to its state of minimal power dissipation. It takes a few nanoseconds and replaces advantageously time consuming stochastic or suboptimal deterministic relaxation algorithms. The elementary cell of the network is presented along with the environment needed to feed it with the required inputs. Two network architectures are proposed, derived from CCD camera principle. Software simulations of a 128x128 network demonstrate the good behaviour of the modified algorithm on real sequences. Electrical simulations of a 16x16 network with ideal components give promising results. Implementation of the CMOS circuit with VLSI technology is under study at our laboratory.

1 Introduction

In this paper, we are concerned with motion detection at pixel level, based on Markov Random Field (MRF) modelling. Spatial and temporal interactions between pixels are modelled by a spatio-temporal MRF, constituting the *a priori model*. The solution sought corresponds to the most probable configuration of primitives (*labels*) according to data (*observations*). The processing consists in minimizing an energy function and requires only local and highly parallel computation, so that it may be implemented on an analog network. The idea of mapping an algorithm minimizing a cost function on an electrical circuit has already been proposed by Hutchinson, Koch and Mead [1-2], based on biological analogy. Our approach is different and the implementation also. Starting from an MRF-based sequential algorithm developped at our laboratory [3], we propose a modified version that lends itself to analog implementation. Section 2 outlines the MRF model for motion detection. Section 3 gives the electrical analogy and exhibits the elementary cell. The specific model with virtual neighbourhood intended for analog implementation is described in Section 4 along with software simulations. The complete cell and the network architecture are presented in Section 5. Section 6 reports the first results of electrical simulations.

2 MRF Modelling for Motion Detection

MRF modelling is a statistical tool for computing a field of labels given a field of observations [4]. Making assumption of constant illumination of the scene and static camera, we take as observations the intensity changes at each pixel (x, y, t) of the sequence : $o(x, y, t) = | I(x, y, t) - I(x, y, t-1) |$. The final label $e(x, y, t)$ that is to be

Lecture Notes in Computer Science, Vol. 800
Jan-Olof Eklundh (Ed.)
Computer Vision – ECCV '94

attributed to each pixel after processing should take one of the two values a ="1" (resp. b ="0") if the pixel belongs to a moving object (resp. static background). In our analog implementation, labels correspond to electrical potentials, so they may take *continuous* values in the range b to a. To get a *binary* label field after processing, a simple *thresholding* of these analogue labels is required.

Let S be the image of the sequence at time t, s a site of S i.e. a pixel (x, y, t), η_s the first order spatio-temporal neighbourhood of s shown on fig. 1.a, c any binary clique of η_s and C the set of all cliques in S. s is the current pixel which label is to be calculated. $r \in \eta_s$ denotes any neighbour of s, spatial or temporal.

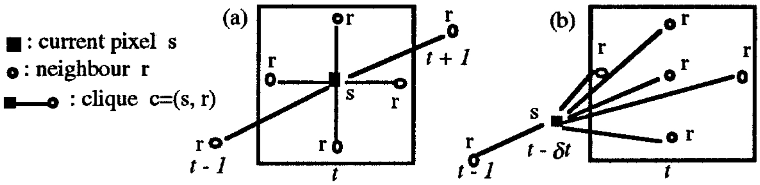


Fig. 1. Spatio-temporal neighbourhood η_s and associated binary cliques : (a) basic neighbourhood, (b) virtual neighbourhood.

We note $E = \{ E(s), s \in S \}$ (resp. $O = \{ O(s), s \in S \}$) the label (resp. observation) field at time t, $E = e$ (resp. $O = o$) one particular realization of E (resp. O) and we use simplified notations : $e(x,y,t) = e(s) = e_s$ and $e(r) = e_r \; \forall r \in \eta_s$ (neighbourhood notation) ; $o(x,y,t) = o_t$ and $e(x, y, t) = e_t$ (temporal notation).

The main property of an MRF relatively to a neighbourhood system is that the probability for a pixel s to have a label e_s depends only on the labels of its neighbours, not on the whole image, thus involving only local computations. Moreover, an MRF being equivalent to a Gibbs distribution [4], there exists an explicit formulation for the a priori probability of the label field :

$$Pr\left[E = e\right] = \frac{1}{Z} exp\left(- U_m(e)\right)$$

Z is a normalizing constant. $U_m(e)$ is the spatio-temporal energy of the a priori model that should ensure spatio-temporal coherence of the masks and discard spurious labels due to noise. This model energy is a sum of elementary quadratic potential functions $V_c(e_s, e_r)$ associated to each clique c :

$$U_m(e) = \sum_{c \in C} V_c(e_s, e_r) \qquad where \qquad V_c(e_s, e_r) = \beta \left(e_s - e_r\right)^2 \qquad (1)$$

with β taking one of three positive values β_s, β_p and β_f, depending on the type of clique (spatial, past or future). These potentials favour homogeneous labelling since low potentials, resulting from identical labels for neighbours, induce a low energy, and thus correspond to a favourable (most probable) energy configuration. As regards temporal potentials, we take $\beta_f > \beta_p$ to favour the future since any innovation in motion is included in the future image.

The relation between observations and labels takes the generic form $o = \psi(e) + n$, where n is a Gaussian noise with zero mean and variance σ^2, and ψ a function of the labels that should model the observations. σ^2 is supposed to be constant and computed from the two first images of the sequence. Adapting the function ψ proposed in [5] for binary labelling, we define :

$$\psi(e_t, e_{t-1}) = m_2 (e_t - e_{t-1})(e_0 - e_{t-1}) \qquad (2)$$

where m_2 is a positive constant and $e_0 = (a+b)/2$. The term $(e_0 - e_{t-1})$ is introduced so that ψ is always positive, a condition required since ψ is supposed to model o_t. Indeed, the final label e_{t-1} obtained at time $t-1$ after binarization of the corresponding electrical potential may only take one of the two values a or b and the current label e_t may take continuous values during the relaxation process before binarization : $e_t \in [b;a]$. The adequation energy $U_a(e, o)$ is derived from ψ and a factor β_f / β_p is introduced to weight its influence :

$$U_a(e, o) = K \sum_{s \in S} \left[o_t - \psi(e_t, e_{t-1})\right]^2 \quad \text{where} \quad K = \frac{1}{2\sigma^2}\frac{\beta_f}{\beta_p} \qquad (3)$$

The total energy function is given by the sum : $U = U_m(e) + U_a(e,o)$ (4). $U_m(e)$ is a regularization term acting as a smoothness constraint on the label field. $U_a(e, o)$ is an adequation term that draws the label field towards the actual observations. The most probable configuration of the label field w.r.t. observations is given by the *Maximum A Posteriori criterion* (MAP) derived from Bayes theorem :

$$Pr\,[E = e\,/\,O = o\,]\ maxi \iff Pr\,[\,E = e,\ O = o\,]\ maxi \iff U\ mini$$

This minimum of energy may be computed using a stochastic or a deterministic relaxation algorithm (simulated annealing, ICM [6]). When choosing ICM, motivated by low computation cost, one has to be careful on the initialization of the present label field in order not to get trapped in a local minimum. Fig. 2.a summarizes the algorithm. It works on three frames. Suppose the past label field E_{t-1} has been determined as the result of previous relaxation, the present and future label fields $\hat{E}_t$ and $\hat{E}_{t+1}$ are initialized with binary fields derived from observations o_t and o_{t+1} using a test of maximum likelihood [7]. Given the two fields E_{t-1} and $\hat{E}_{t+1}$ and the initial field $\hat{E}_t$, the relaxation runs on E_t, taking into account the a priori model and the adequation with O_t. For each pixel s of the image at time t, the two possible labels a or b are tested and the one inducing the minimum local energy is kept. The visiting order of the sites may be sequential or random. The process iterates on the image until convergence.

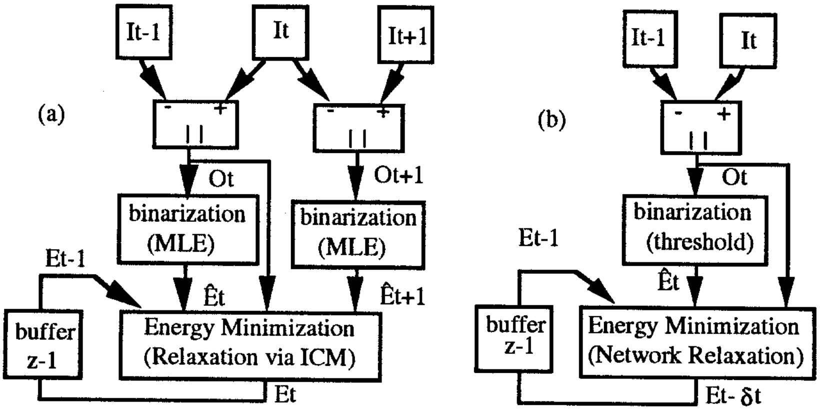


Fig. 2. The detection algorithm : (a) basic version, (b) modified algorithm. ($\hat{E}$ denotes a coarse estimate or initialization of field E).

3 Electrical Analogy

Instead of using a software relaxation scheme like ICM, we propose to efficiently implement the relaxation via an analog resistive network tending to equilibrium. Our purpose is to built a VLSI circuit that takes as input an image sequence and gives as output the masks of the moving objects. Using (1-4), we rewrite U :

$$U = \sum_{ij} \beta_s \, (e_{ij} - e_{i+1,j})^2 + \beta_s(e_{ij} - e_{i-1,j})^2 + \beta_s(e_{ij} - e_{i,j-1})^2 + \beta_s(e_{ij} - e_{i,j+1})^2$$
$$+ \beta_p(e_{ij} - p_{ij})^2 + \beta_f(e_{ij} - f_{ij})^2 + K\left[\, o_{ij} - m_2(e_{ij} - p_{ij})(e_0 - p_{ij})\right]^2$$

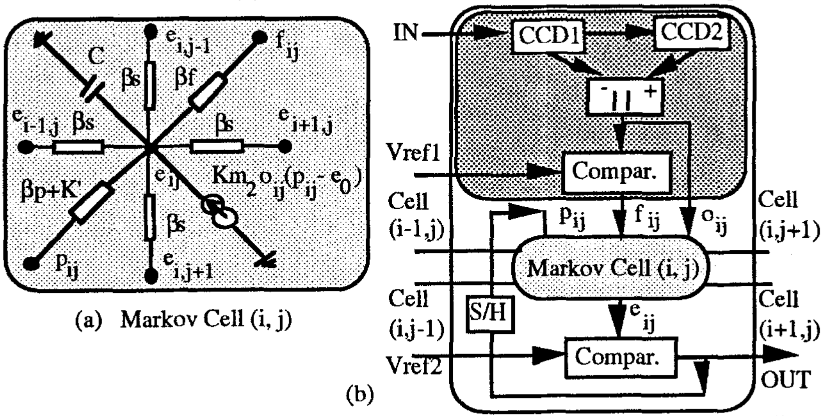
with digital notations : $o_{ij} = o(x,y,t)$, $e_{ij} = e(x,y,t) = e_t = e_S$; $p_{ij} = e(x,y,t\text{-}1) = e_{t\text{-}1}$ for the past ; $f_{ij} = e(x,y,t+1) = e_{t+1}$ for the future. The global minimum of U corresponds to null derivatives w.r.t. each e_{ij} :

$$\frac{\partial U}{\partial e_{ij}} = 0 \Leftrightarrow \beta_s \nabla^2 e_{ij} + [\, \beta_p + K'] \; (e_{ij} - p_{ij}) + \beta_f \, (e_{ij} - f_{ij}) + K m_2 \, (p_{ij} - e_0) o_{ij} = 0 \qquad (5)$$

where $\nabla^2 e_{ij} = [4 \, e_{ij} - e_{i+1,j} - e_{i-1,j} - e_{i,j-1} - e_{i,j+1}]$ and $K' = K \, m_2{}^2 \, (e_0 - p_{ij})^2$ is a constant since $\forall p_{ij} \in \{a;\, b\}$, $(e_0 - p_{ij})^2 = (a\text{-}b)^2/4$. Now we can make an analogy between (5) and Kirchhoff's law at a node. Each term may be interpreted as a current converging to node ij which electrical potential is e_{ij}. Labels correspond to electrical potentials and label differences to voltages. $Km_2(p_{ij} - e_0)o_{ij}$ is a current generator driven by a voltage. β_S, $(\beta_p{+}K')$ and β_f represent conductances. Introducing a leakage capacitance C specific of the circuit dynamics yields :

$$\forall ij, \;\; C\frac{\partial e_{ij}}{\partial t} = \beta_s \nabla^2 e_{ij} + [\, \beta_p + K'] \; (e_{ij} - p_{ij}) + \beta_f \, (e_{ij} - f_{ij}) + K m_2 \, (p_{ij} - e_0) o_{ij} \qquad (6)$$

where the left member of (6) becomes zero when the network reaches equilibrium.



(a) Markov Cell (i, j)

(b)

Fig. 3. The cell of the analog network : (a) elementary cell, (b) complete cell including pre-processing stage (*N.B.* the upper part is removed if pre-processing is done outside the cell)

The linear system of equations (6) may be solved via a simple resistive network, made of elementary cells as the one shown on fig. 3.a. It contains only resistors and a current generator. All these components are *scalable* which is very important for the purpose of implementation. This elementary cell implements the local energy minimization. When electrical potentials p_{ij}, f_{ij} and command voltages $(p_{ij} - e_0)o_{ij}$ are set up, the network will relax, following Kirchhoff's law, until it dissipates its minimum of energy. The static output electrical potentials correspond to the solution of (6), i.e. the optimal continuous label field. To get the final binary labelling, a simple threshold is applied on each electrical potential e_{ij}. There exists always one unique and stable solution, even if current sources are negative. So we have achieved to map the minimization algorithm on a resistive network.

4 Modified Algorithm and Functional Simulation

Some important modifications are introduced in the algorithm for the purpose of implementation. First, with the basic algorithm, it was necessary to initialize carefully the present label field $\hat{E}_t$ in order not to get trapped in a local minimum with ICM. But for analog implementation, we dot not need to initialize the network at the beginning of the relaxation process at each time t. In fact we only need to initialize the future label field $\hat{E}_{t+1}$, not the present label field $\hat{E}_t$ since the network will relax to its state of minimal energy dissipation wathever the initial conditions are at each node. Taking into account this remark, we redefine the concept of past, present and future and propose the use of a *virtual neighbourhood.* What we call the *present* corresponds now to a virtual time $t - \delta t$ where $0 < \delta t < 1$ and we redefine spatial and temporal neighbours of *virtual pixel s* as shown on fig. 1.b. So the modified algorithm is greatly simplified since it does not need any longer to work on three frames : two frames are sufficient for the relaxation process to run. This simplification induces a drastic reduction (33 %) of the circuit dimension. Another simplification concernes the binarization of O_t to get a coarse estimate of the "future" $\hat{E}_t$: instead of a maximum likelihood estimate [7], we use a simple threshold. These two simplifications lead to the synoptic given in fig. 2.b.

To test the modified algorithm before running electrical simulations, we wrote a program in C language, running on a Sun Sparc workstation, to make a software simulation of the network. The main difference between the analog circuit relaxation and the software developped to simulate energy minimization is that the first one is a continuous and parallel process while the other one is a discrete sequential process. From (5), we derive an iterative scheme to calculate e_{ij} from its neighbourhood :

$$e_{ij} = \frac{\beta_s(e_{i,j-1}+e_{i,j+1}+e_{i-1,j}+e_{i+1,j}) + (\beta_p + K')\, p_{ij} + \beta_f f_{ij} + K m_2\, (e_0 - p_{ij}) o_{ij}}{4\beta_s + \beta_f + \beta_p + K'} \qquad (7)$$

We use a frame recursive method to compute the best configuration of the electrical potentials at each time t : we scan the image and calculate each electrical potential following (7). When a complete scan of the image is accomplished, we update all electrical potentials at the same time. We iterate the scanning until convergence. Only few iterations per image are needed before convergence (less than 10). Various tests have been made on synthetic and real 8-bit image sequences of size 128x128, demonstrating the robustness of the model parameters β_s, β_p, β_f and m_2 which may be adjusted once and for all. The values $\beta_s = 40$, $\beta_p = 5$, $\beta_f = 30$, $m_2 = 1$ give satisfying results in all cases. This is worth to mention since it implies that we do not need adjustable but only constant resistors. Still, parameters K and K' depend on σ^2. In our first simulations, we use a constant value (e.g. $K=60$) but it limits the grey level dynamics of observations. The detection algorithm has a good behaviour in the presence of noise : in a few iterations per image, it may clean up very noisy observations and reconstruct the masks of moving objects.

An example of motion detection is shown on fig. 4. We used a standard output threshold value e_0 so that the mask of the moving car is a little too large. But if we take a better value for the threshold, we obtain a mask closer to the shape of the car.

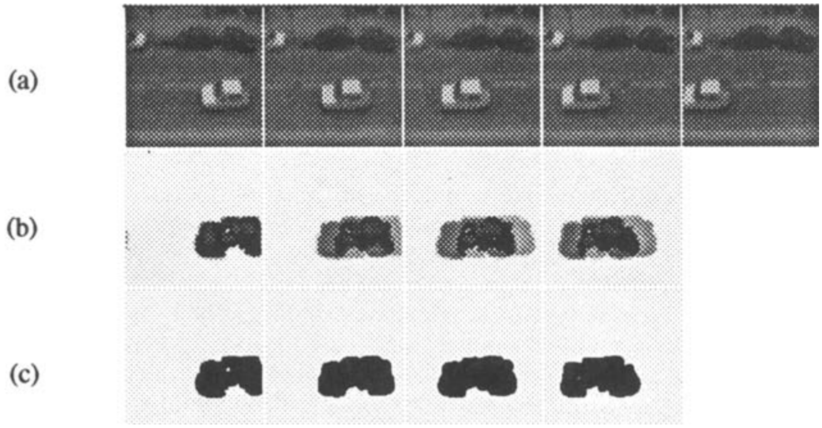


Fig. 4. Result of motion detection : (a) original real sequence with a white car in the foreground moving from right to left and others cars in the background being static, (b) grey level representation of continuous label field before thresholding, (c) final binary label field after thresholding exhibiting the mask of the moving car.

5 Network Architecture

We shall specify two different ways for supplying the elementary cell with the required inputs and for integrating it in the network.

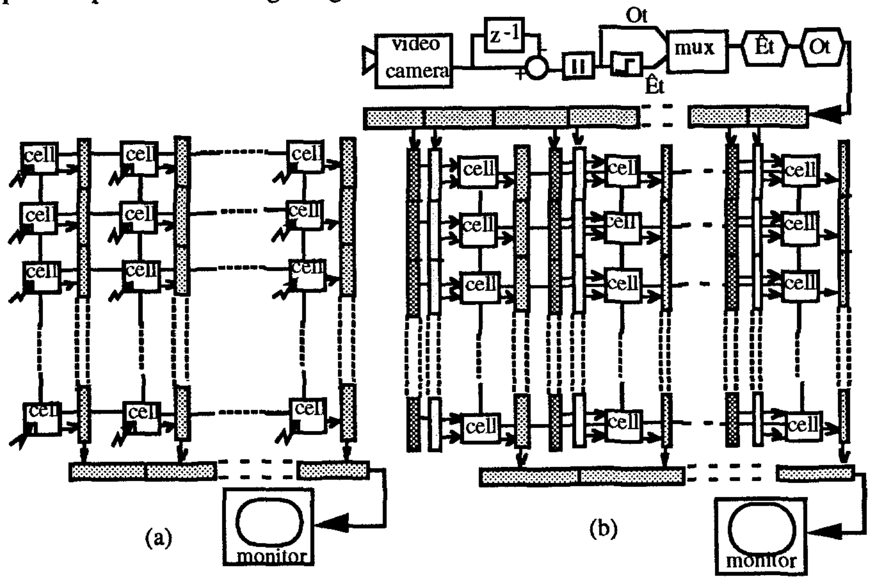


Fig. 5. Network architecture : (a) direct parallel inputs, (b) serial inputs.

If one photoreceptor is included in each cell (direct parallel input to the network), all the pre-processing stage (image difference, module, threshold) for computing the inputs must be implemented on the cell itself (fig. 3.b). This option requires a CCD register to store the values *I(x,y,t)* and *I(x,y,t-1)*, a circuit implementing the difference and the absolute value to get o_{ij}, plus an input comparator (threshold) to get f_{ij}. Note that the past label p_{ij} should not be considered as a supplementary input since it may be obtained via a sample and hold circuit (S/H) included in the cell itself. We also need an output comparator to get the final binary (thresholded) label from the continuous electrical potential after relaxation. The corresponding network is shown on fig. 5.a. Inputs are parallel (integrated matrix of photoreceptors) and outputs are serial. We propose an architecture using vertical and horizontal CCD registers. The network architecture is quite simple but the complete cell is more cumbersome. This solution gives a dedicated circuit that could be used for motion remote control.

The other option is to feed the network with data coming from a camera (serial input). In this case, the pre-processing is done outside the network. This simplifies the cell (fig. 3.b without the upper part). But feeding the cells with the inputs is more complicated, leading to the network shown on fig. 5.b. We need an analog demultiplexer to supply the network with parallel inputs. Thus one horizontal CCD register must be added to feed the network with the data plus two vertical CCD registers per column of cells to feed each cell with o_{ij} and f_{ij}. This solution may be chosen for applications in image compression and coding for TV transmission.

6 Results of Electrical Simulation

Two main caracteristics of the cell are its surface and its power dissipation. We may rescale all the model parameters β_s, β_p, β_f, m_2 so that the order of magnitude of all required resistors may be of about *10 kΩ*. This leads to a power dissipation of about *0.5 μW* per cell for a supply voltage of *100 mV*. As regards the surface, the more cumbersome component is the capacitance of about *4 pF* required for the S/H circuit, occupying a surface of *4000 μm*^{2}. About 50 transistors are needed to implement the complete cell of fig. 3.b. This leads to an acceptable dimension for the cell.

We have simulated the electrical behaviour of a 16x16 network with an electrical simulation software *Eldo*. As expected, spatial resistors act as spatial smoothing (low-pass filter) while current generators inject high currents at nodes corresponding to transition areas (static/moving), increasing their electrical potential. The power dissipated in the network is about *0.5 mW*, while current intensities range from *10 nA* to *1 μA*. Of course, the time needed for the circuit to relax is very short : about *10 ns*. That is the reason why such an implementation is interesting : we can developp a motion detector working at video rate.

Fig. 6 exhibits results for the same sequence as in fig. 4.a. Only three images are shown (a). We were limited to a 16x16 network because *Eldo* couln't simulate a bigger one (too much signals to handle). So we decimated the data to reduce image size from 128x128 to 16x16. This leads of course to a drastic decrease in spatial resolution : we show in (b) the final masks (obtained after thresholding of electrical potentials). They are of poor quality. But more interesting to see is the state of the network after relaxation but before binarization of the electrical potentials (c) : electrical potential at each node reflects motion information. Nodes corresponding to static pixels have a null electrical potential while nodes corresponding to moving pixels have a high electrical potential. Of course, the result for the first image of the sequence is of poor quality since we do not have any past image at our disposal. Fig. 6.d illustrates the influence of two parameters, K and β_p : increasing K yields sharper transitions in the electrical potentials. Decreasing β_p reduces the trail at the back of the car (due to the past position of the car).

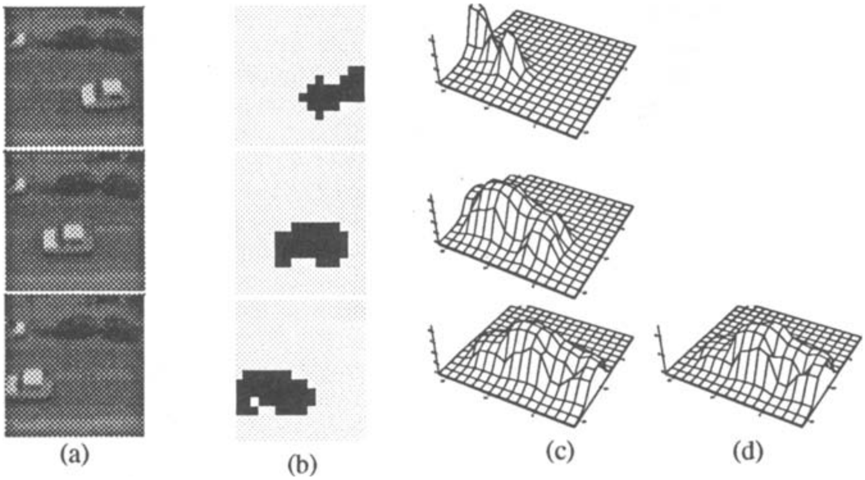


Fig. 6. Processing of a real sequence : (a) real sequence, (b) binary labelling after thresholding, (c) state of the network obtained for some standard values of K and β_p, (d) state of the network for the last image when K is increased and β_p is decreased.

7 Conclusion and Perspectives

Starting from an MRF-based motion detection algorithm developped at our laboratory and implemented on software, we propose a modified version that may be implemented on hardware. Energy minimization is realized via an analog resistive network relaxing to its state of minimal dissipation. Electrical potentials at each node of the network correspond to MRF labels. The first electrical simulations of the network with ideal components are promising. Presently we are testing the cell and the network with real components. The circuit will be implemented with CMOS technology. It is well suited for image compression via motion coding. This application is now being investigated with industrial partners.

References

1. Hutchinson J., Koch C., Luo J., Mead C., Computing motion using analog and binary resistive networks. Computer 21, 52-63 (1988)
2. Koch C., Marroquin J., Yuille A., Analog "neuronal" networks in early vision. Proc. Natl. Acad. Sci. USA, Biophysics 83, 4263-4267 (1986)
3. Luthon F., Caplier A., Motion detection and segmentation in image sequences using Markov Random Field modelling. Eurographics 93 Animation and Simulation Workshop, Barcelona, 265-275 (1993)
4. Geman S., Geman D., Stochastic relaxation, Gibbs distributions, and the bayesian restoration of images. IEEE Trans. PAMI 6, No 6, 721-741 (1984)
5. Lalande P., Bouthemy P., A statistical approach to the detection and tracking of moving objects in an image sequence. Signal Processing V: Theories and Applications, Elsevier Ed., Proc. of EUSIPCO, Barcelona, 947-950 (1990)
6. Besag J., On the statistical analysis of dirty pictures. J. Royal Statist. Soc. B 48, No 3, 259-302 (1986)
7. Hsu Y.Z., Nagel H.H., Rekers G., New likelihood test methods for change detection in image sequences. Comput. Vis. Graph. Im. Proces. 26, 73-106 (1984)