A neural network-based approach to motion estimation with discontinuities

Mohamed BERKANE[1, 2], Patrick CLARYSSE[2], *Member, IEEE*,

Isabelle MAGNIN[2], *Member IEEE*

[1] *University of Larbi Ben Mhidi, O.E.Boughi, Algeria*
[2]*CREATIS-LRMN, CNRS UMR 5220, Inserm U630, INSA of Lyon, University of Lyon, Lyon, France*
{Mohamed.Berkane, Patrick.Clarysse, Isabelle.Magnin}@creatis.insa-lyon.fr

Abstract

A new neural network-based approach is proposed to estimate motion hierarchy in image sequences taking into consideration motion discontinuities. The network consists in an input layer, an intermediate layer and an output layer. In order to estimate the most likely displacement at each pixel, we have transposed the block matching approach into the neural network approach and add mechanisms to detect motion discontinuities. Information redundancy allows for parallel processing in view of real-time complex motion estimation tasks. Preliminary tests on synthetic and real images are very promising.

1. Introduction

Neural networks constitute an interesting approach to deal with, from new points of view, the problems of perception, memorization, learning and reasoning. Formal neural networks constitute also a very promising alternative to circumvent some limitations of conventional approaches. Thanks to their natural ability to parallel processing of information and their mechanisms based on nerve cells (biological neurons), they are able to deduce emerging properties to solve problems qualified as complex.

The motion detection and estimation methods are generally based on the assumption of *pixel brightness conservation* over all the image sequence. In other words, the intensity at each material point is supposed to be preserved during motion. The developed methods are generally evaluated according to the following criteria: accuracy, robustness against image noise, and computing time. In this paper, we propose a new approach based on neural networks which has the properties to be robust and parallelizable. The performances of this method are evaluated on both synthetic and real image sequences.

2. State of Art

The proposed method relies on a neural network formulation of a block matching strategy similar to that presented by Torok et al. [5] but with an ascendant approach from the pixels to the regions and objects, as in the Castellanos's method [6]. This latter approach is inspired from the motion perception in primates. The method presented by Seiffert and Michaelis used Kohonen maps to achieve a classification of moving objects [13,14]; this approach requires *a priori* information about the objects composing the scene. The method presented by Stocker [7,8] combines the model of Horn and Schunck [15] with an additional constraint which slightly bias to some *a priori* reference motion. On the contrary, our method does not require *a priori* knowledge on the objects composing the scene. One main interest of neural network based methods stands on the possibility to exploit parallelism in order to minimize computing time. Torok et al. have tested there method on a Cellular Neural/Non-linear Network Universal Machine (CNN-UM) [5] while Stocker has developed a hardware implementation of his method [7,8]. This is also one of our objectives.

3. Network structure

Motion is estimated in sequences with images of dimension $n \times m$. Our approach considers either image pairs or multiple consecutive frames up to the entire sequence. Drawing on vision system for humans, we propose a new neural network to estimate the movement at each pixel and also detect the dominant motion in a sequence. Parallel processing can be

978-0-7695-3326-1/08 $25.00 © 2008 IEEE
DOI 10.1109/HIS.2008.140

envisaged at several levels in order to speed up the motion analysis process.

3.1. Network structure and interconnections

At the level of the human brain, the organization of the visual system has several hierarchical layers called visual areas. The best known are the primary and secondary visual areas which are surrounded by several visual areas. For motion interpretation, there exists, in the primary visual area, neurons dedicated to the detection of directional structures and neurons dedicated to motion detection. By analogy, we have designed our neural network topology based on a layer hierarchy. Also inspired from the human visual system, 3D matrices record information about the direction and the magnitude of the motion at each pixel. This informations will be refined thereafter to detect motion discontinuities.

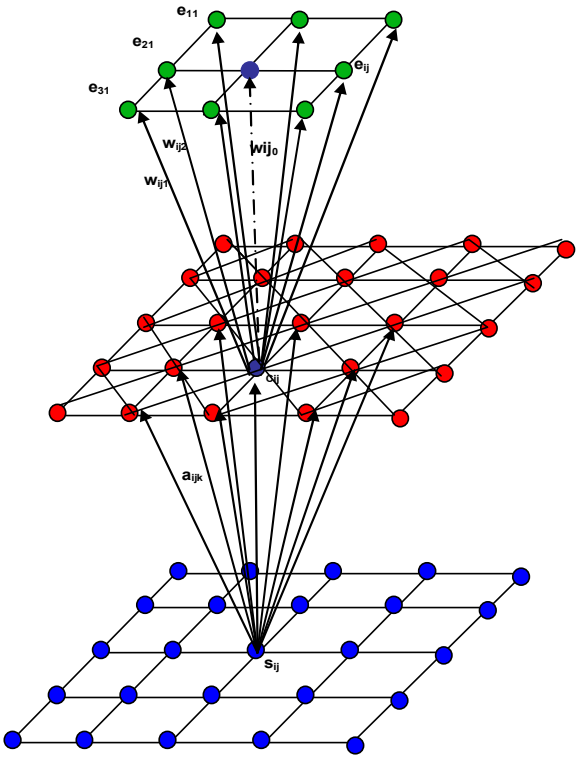


Figure 1. Network Topology

The network is composed of three layers (figure 1). An input layer composed of b neurons e_{ij}, where b represents the number of block elements. The second layer is an $n \times m$ matrix of neuron c_{ij} having the same size as the image of the sequence. Each neuron c_{ij} is connected to all the neurons from the input layer. An output layer is composed of s_{ij} neurons which has the same size as the previous layer. The s_{ij} neurons are connected to c_{ij} neurons and their neighbors.

3.2. Weight matrices

There are three weight matrices noted $\boldsymbol{W}$, $\boldsymbol{P}$, $\boldsymbol{A}$, respectively:

$\boldsymbol{W}=(w_{ij}^{k})$: $(i=1,..,n\ ;\ j=1,..,m\ ;\ k=0,..,8)$ is a three-dimensional matrix of weight connections between the neuron of the intermediate layer with neurons of the input layer. The weight w_{ij}^{k} consists in a triplet: the first component is the intensity of the k^{th} neighboring pixel of pixel ij; the second component represents the number of times where this neuron is moved into the direction of the k^{th} neuron. When considering the first order neighborhood, we have nine possible displacements: 8 elementary pixel displacements and the 9th for the absence of movement. The third value expresses the amount of displacement. The second and the third values are initialized to zero.

$\boldsymbol{P}=(p_{ij}^{k})$: $(i=1,..,n\ ;\ j=1,..,m,\ k=1..8)$ is the weight matrix of the neuron connections of the intermediate layer with their first order neighbor. Each neuron of the intermediate layer holds one vector of 8 elements (if we consider a 1st order neighborhood). The p_{ij}^{k} weight represents a force exerted by the neuron ij (i.e. the displacement contribution) onto the k^{th} neighbor (Figure 2.c) This notably allows for disregarding isolated neurons (Figure 2.c) and also detecting motion discontinuities between regions animated with different motions (Figure 2.b) at the operational phases (see section 4.3)

$\boldsymbol{A}=(a_{ij}^{k})$ $(i=1,..,n\ ;\ j=1,..,m,\ k=0,..,8)$ is the weight matrix for the connections of neurons of the output layer with the counterpart neurons of the intermediate layer and its neighbors. The weight a_{ij}^{k} is composed of two values. The first value represents the direction of the displacement at the pixel among the nine possibilities, while the second value represents the displacement magnitude at the same pixel. This matrix is mainly exploited for estimating the motion taking into account the discontinuities. Each neuron s_{ij} of the output layer is connected to a block of the intermediate layer. This block is constituted by the neuron c_{ij} and its neighbors. We have nine connections therefore nine pairs of values at each neuron (probable displacement direction and magnitude) . The idea is to exploit the information hold by the neurons neighbors of c_{ij} to influence the choice of the displacement direction and magnitude of this neuron. This will have an influence

when a neuron is isolated (Figure 2.*(c)*) or when it is part of a motion frontier (discontinuity, Figure 2.*(b)*). Finally, a single pair of values is selected for this pixel which gives both the motion direction and magnitude of the neuron s_{ij}.

Note that the size of the block plays an important role in the result accuracy. The higher the block size the better the accuracy. At another level, we have chosen to have redundant information in the matrix ***A*** in order to allow for parallel processing for all neurons a_{ij}.

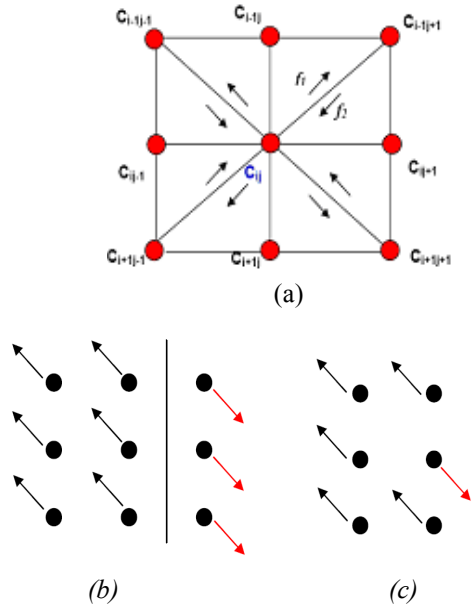


Figure 2. Neuronal interactions at the intermediate layer (a) $f1 = p_{ij}^{k}$ represents the force exerted by c_{ij} onto c_{i-1j+1}, the k^{th} neighbor of c_{ij} . $f2 = p_{i-1j+1}^{k'}$ represents the force exerted by c_{i-1j+1} on c_{ij}, the k'^{th} neighbor of c_{i-1j+1}. (b) the case of motion discontinuity (c) isolated neuron.

4. Network mechanism

4.1. Initialization phase

In a first step, the matrices of neuronal interaction ***P*** and the weight matrices ***W, A*** are initialized (eq.(1)). In a second step, we construct the learning set where each image at time point t is represented by EP, the set of blocks X_{ij}^{t} with a fixed size (eq.(2) and figure 3). The construction of the learning set results in redundant information. This redundancy allows, during the learning phase, to separate the block matching process for every pixel in the search window. This independence allows for parallel processing on a parallel computer or the development of an electronic circuit dedicated to this specific neural network architecture.

- $a_{ij}^{k} = 0$,
- $p_{ij}^{k} = 0$
- $w_{ij}^{0} = I_{ij}$ *Intensity of pixel ij*
- $w_{ij}^{k} = I_{i\pm1,j\pm1}$, $k \neq 0$

Intensity of neighboring pixels **(1)**

$EP=\{X_{ij}^{t},\ i=1..n;\ j=1..m\ ;\ t=1..T\}$ **(2)**

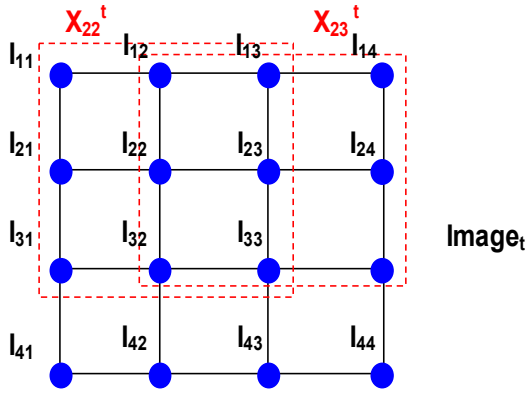


Figure 3. Decomposition example of an image into blocks of 9 pixels

4.2. Learning phase

In the learning phase, an element (X_{ij}^{t}) of the learning set EP is presented to the input layer at each iteration (Figure 4). We seek for the winner neuron by applying the block matching principle between the values brought by the neurons of the input layer and the intensity values of one neuron and its neighbors such that this neuron is an element of the search window. The winner neuron c^{*} is the neuron which presents the minimal intensity difference (eq. (3-4)). Its weight is then updated according to the chosen direction as in (5). This process is repeated for all elements of the learning set. As a result, we obtain a matrix which associates to each direction of each neuron one weighting. The weighting represents the probability that the corresponding pixel moved to the associated direction. The most probable direction for each neuron can be derived from the highest probability value.

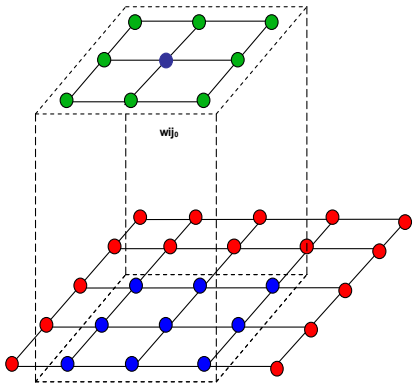


Figure 4. Setting up correspondence between input neurons and one block of the search window

$$\Delta_{c^*} = \min(\Delta_{c_{ij}}) \quad \textbf{(3)}$$

$$\Delta_{c_{ij}} \leftarrow \sum_{k=0}^{8} (w_{ijk} - e_k)^2 \quad for\ c_{ij} \in research\ window \quad \textbf{(4)}$$

$$\begin{cases} p_{ij}^{c^*}(1) \leftarrow p_{ij}^{c^*}(1) + 1 \\ p_{ij}^{c^*}(2) \leftarrow p_{ij}^{c^*}(2) + q^{c^*} \end{cases}$$

with q^{c^*} *the force quantity exerted on the winner neuron* c^* **(5)**

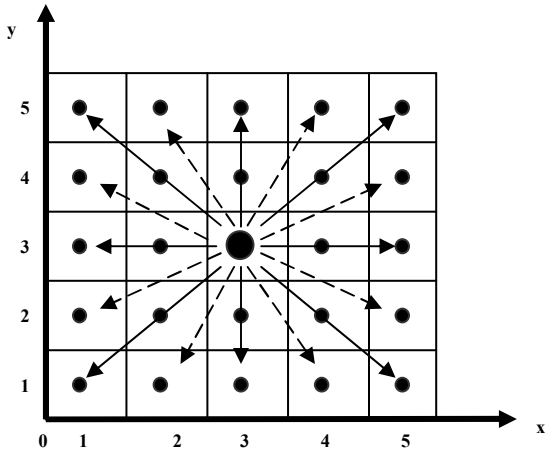


Figure 5. Search window with a second order neighborhood

4.3. Operational phase

In the operational phase, we operate on the inter-neuronal interactions in order to detect the discontinuities. The interactions are represented by the movement associated with each pixel. Each given direction for a pixel represents a force exerted onto the corresponding neighbor. Neighbors with the same displacement direction create a displacement field, (6) (7). Taking into account these forces and these fields, we can deduce the discontinuities over the whole image. We have therefore the matrix ***A*** which has two values for each pixel: the direction (8) and the magnitude of displacement (9).

$$\lambda_{ij} \leftarrow k^* / p_{ij}^{k^*}(1) = \max(p_{ij}^{k}(1)) \ with\ k = 0..9 \quad \textbf{(6)}$$

$$\beta_{ij} \leftarrow \lambda^* \ / \left[p_{ij}^{\lambda^*}(1) = \max \sum_{(i,j) \in w} p_{ij}^{k^*}(1) \right] \quad \textbf{(7)}$$

$$\begin{cases} a_{ij}^{\beta_{ij}}(1) \leftarrow 1 \\ a_{ij}^{k}(1) \leftarrow 0 \quad With\ k = 0,..9\ and\ k \neq \beta_{ij} \end{cases} \quad \textbf{(8)}$$

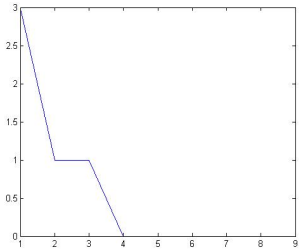
$$\begin{cases} a_{ij}^{\beta_{ij}}(2) \leftarrow p_{ij}^{\beta_{ij}}(2) \\ a_{ij}^{k}(2) \leftarrow 0 \quad With\ k = 0,..9\ and\ k \neq \beta_{ij} \end{cases} \quad \textbf{(9)}$$

5. Results

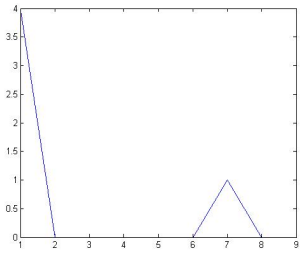
The method has been tested on synthetic and real image sequences. The first synthetic sequence (Figure 6) is composed of six images of a thorax that underwent a translation along the diagonal at a constant speed. Our approach allowed deducing the exact dominant motion into the sequence. The result in figure 8.a is obtained using a second order neighborhood (Figure 5) and a search window with size 7x7. In a second sequence, a non rigid dilation is applied to the same thorax image (6 images). The result for this case is displayed in figure 8.b. We used the same neighborhood search window size as with the previous sequence. Our method has been also tested on two real standard sequences. The first one comes from the Rubik's cube sequence. In this 2-image sequence, the cube exhibits a slight rotation. The results are depicted in Figure 9.a. The second sequence is composed of two images representing three cars (sequence known as Hambourg taxi) where each of them has a different motion direction (Figure 9.b). Within those various sequences that contain different objects with different movements (translation, rotation, homothety, etc), the method achieved very satisfactory results.

Figure 6. First image of the thorax test sequence

The graphs in Figure 7 represent motion vote for two pixels taken as an example. The graph abscissa represents the 9 possible motion directions (the lack of motion corresponds to the value 5). The ordinate indicates the number of times the given pixel/neuron has a preferred direction. For example, in Figure 7.b, the pixel (150x150), has two peaks, one in (1,4) and the other one in (7,1). In other words, during the processing of the 6 images of the sequence, motion in direction «1» has been selected 4 times while motion in direction «7» has been selected only once. So the probability that it moved in direction «1» is 0.8.

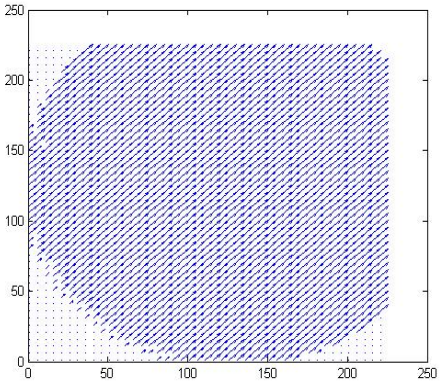


(a)

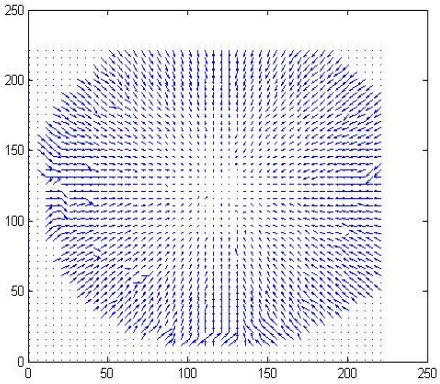


(b)

Figure 7. (a) Dominant motion of pixel «150x150» (b) Dominant motion of pixel «200x200» from the 6 images of sequence 8(a)

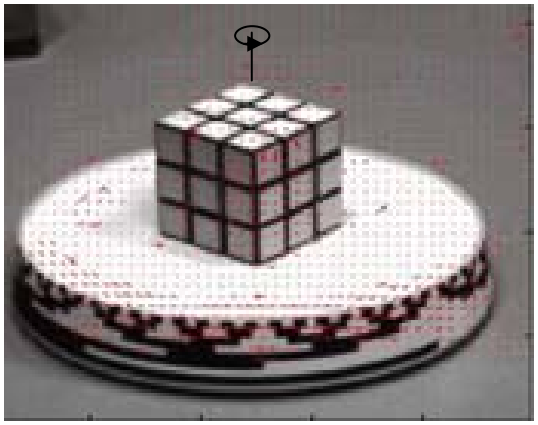


(a)

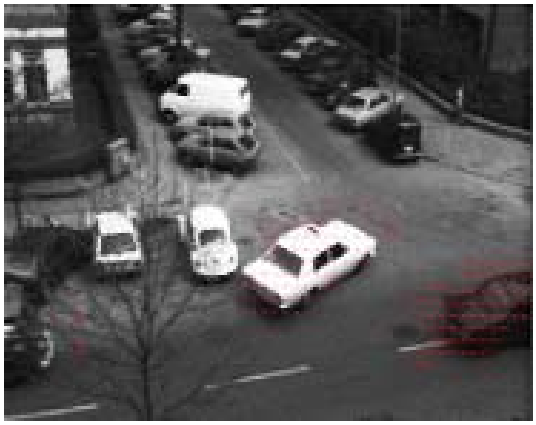


(b)

Figure 8. (a): Displacement field obtained with the synthetic translation sequence. (b): Displacement field obtained with the dilation sequence



(a)



(b)

Figure 9. (a) Displacement field obtained with the Rubik's cube image pair. (b) Displacement Field obtained with the Hambourg taxi image pair. (KOGS/IAKS Universität Karlsruhe)

6. Conclusion

We proposed a new method for motion estimation in image sequences based on a new neural network architecture. This network has a specific topology dedicated to motion estimation inspired from the human vision system. It combines the advantages of both the block matching method and the neural networks. Instead of saying that all the neighboring pixels move in the same direction, we introduced a new concept of neuronal force interaction. This information enabled us to better express the relationship between the neighboring pixels which helps to detect motion discontinuities between neighboring regions. The duplication of information at two levels of the network allows for parallel processing. In addition, considering that the proposed network has local connections, a hardware implementation can be developed and is susceptible to present very interesting performances.

7. References

[1]. B. Delhay, P. Clarysse, and I.E. Magnin. Locally adapted spatio-temporal deformation model for dense motion estimation in periodic cardiac image sequences. In *Functional Imaging and Modeling of the Heart,* volume LNCS 4466, Salt Lake City, UT, USA, pages 393-402, June 2007.

[2]. Planat, A. C." Estimation de mouvement par maillage actif multi échelle avec prise en compte de la discontinuités: Application a l'imagerie cardiaque en résonance magnétique". PhD Thesis, Creatis-LRMN Laboratory. Lyon, INSA Lyon: 201. 1999.

[3]. Delhay, B. "Estimation spatio-temporelle de mouvement et suivi de structures déformables. Application à l'imagerie dynamique du coeur et du thorax". PhD Thesis, Creatis-LRMN Laboratory, INSA, Lyon, pp. 217. 2006.

[4]. C. Grava. "Compensation de mouvement par réseaux neuronaux cellulaires. Application en imagerie médicale". Doctor, Claude-Bernard LYON I University, 2003

[5]. Torok, L. "Stability, Optical Flow and Stochastic Resonance in Cellular Wave Computing". PhD Thesis. Computer and Automation Research Institute Hungarian Academy of Sciences. 103pp, 2005

[6]. Castellanos Sanchez, C. C. "Modèle connexionniste neuromimétrique pour la perception embarquée du movement". PhD Thesis, , Université Henri Poincaré, Nancy, France 168pp, 2005.

[7]. Stocker, A. A. "Constraint optimization Networks for visual motion perception analysis and synthesis". PhD Thesis, Zurich, Swiss Federal institute of technology: 182. 2001.

[8]. Stocker, A. A. "Analog VLSI Focal-Plane Array With Dynamic Connections for the Estimation of Piecewise-Smooth Optical Flow". IEEE Transactions on circuits and systems I, Vol. 51, n° 5, May 2004.

[9]. Kohonen T., Lagus K., Salojarvi J., Honkela J., Paatero V., Saarela A. "Self organization of a massive document collection". IEEE Trans on Neural Networks, vol. 11, pp.574-585, 2000.

[10]. Kohonen T. "Self-Organizing Maps". 3 edn. Springer-verlag, Berlin Heidelberg New York, pp. 501, 2001.

[11]. Milanova M. G., Campilho A. C., Correia M. V. : Cellular Neural Networks for Motion Estimation. ICPR : pp. 3827-3830. 2000.

[12]. Bertram E. Shi, : Gabor-Type Filtering in Space and Time with Cellular Neural Networks. IEEE Transactions on circuits and Systems Vol. 45, February pp : 121-132. 1998.

[13]. Seiffert. U., Michaelis B. : Growing Multi-Dimentional Self-Organizing Maps. International Journal of Knowledge-Based Intelligent Engineering Systems. Vol 2, pp: 42-48 January 1998.

[14]. Michaelis. B., Schnelting. O., Seiffert. U., Mecke. R.: Motion Estimation Using A Compounded Self Organizing Map-Multi Layer Perceptron. WCNN '95. World Congress on Neural Networks. Washington USA July 1995.

[15]. B. Horn and B. Schunck, "Determining optical flow." Artificial Intelligence, vol. 17, pp. 185–203, 1981.