ARTIFICIAL NEURAL NETWORK FOR DISPLACEMENT VECTORS DETERMINATION

Dušan LEVICKÝ, Peter BÖHMANN
Dept. of Electronics and Multimedia Communications
Technical University in Kosice
Park Komenskeho 13, 040 01 Kosice
Slovak Republic

Abstract

An artificial neural network (NN) for displacement vectors (DV) determination is presented in this paper. DV are computed in areas which are essential for image analysis and computer vision, in areas where are edges, lines, corners etc. These special features are found by edges operators with the following filtration. The filtration is performed by a threshold function. The next step is DV computation by 2D Hamming artificial neural network. A method of DV computation is based on the full search block matching algorithms. The pre-processing (edges finding) is the reason why the correlation function is very simple, the process of DV determination needs less computation and the structure of the NN is simpler.

Keywords

neural network, displacement vector, block matching, motion estimation, image sequence analysis

1. Introduction

The importance of image sequence analysis is constantly growing. The aim of image sequence analysis is to detect the object movements on the scene with respect to the sensor (camera) or at least the projection of this motion onto the image plane. Many applications use motion estimation as their main image analysis tool or at least as an important tool. Among them, we can quote mobile cybernetics, medical imaging, video coding and frame rate conversion based on a temporal interpolation. According to the target application , the kind of scene shot, and the image quality (high or low definition, frame rate ...), the used methods may differ as much in their principle as in their computational complexity. Motion estimation can always be expressed in terms of matching. The algorithms attempt to match features present in consecutive images in order to deduce their movements. These features may be segments, characteristic points, pixels (that is in case for pel-recursive algorithms), regions of particular shape, blocks, raw image data, filtered image data including their derivatives, edges, lines corners or even 3 D physical objects etc. The way matching is carried out allows also to classify the methods. The algorithms for DV determination in the image sequences usually need many computations that's why the real time applications are very difficult. One of the way how this problem can by solved is application of neural networks (NN) . There exist some methods which use the neural network approach for DV computation. Most of them use the Hopfield NN [11,12,13,14]. An architecture of this NN is very suitable for this type of application. The positions of neurons and interconnections between neurons in the NN are similar as in the human visual cortex, where the neurons are arranged within complex and hypercomplex columns. Each neuron of the NN receives inputs from itself and other neurons at the neighbouring points, and a bias input from the outside world. There is presented another method for DV determination based on block matching algorithms in this paper. The method uses the 2D Hamming NN. The algorithm is divided into two steps. First, the characteristic edges are computed in images of image sequence. The edge selection is performed by gradient operators. 2D thresholding function is used for better computation of the edges. Second, the displacement vectors are determined by using 2D Hamming NN because this part has high computation requirements. NN reduces the time of computation and the type of input data causes that the architecture of the NN and the correlation function are not complicated. Therefore the results of NN are accurate and reliable and the time of DV determination is in a high degree reduced.

2. Pre - processing of Input Data

The choice of an input data type is very important for good and quickly obtained results by the NN. There are many types of input data in image processing. For example, there are very often raw image data or filtered image data including their derivatives, edges, lines, corners or characteristic points which are computed by various types of operators.

We tried to use more types of input data (lines computed by mask operators, edges computed by gradient operators and characteristic points computed by monotony operator)[3,4,6]. We obtained the best results if we used the edges like input data. An edge is defined as the boundary between two regions with relatively distinct gray-level properties. The Sobel gradient operators (Fig.1) were used for edges detection. First, gradient of the image f(i,j) is calculated by the following way.

$$A_{[1,0]} \approx \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \qquad A_{[-1,1]} \approx \begin{matrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{matrix}$$

$$A_{[0,1]} \approx \begin{matrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix} \qquad A_{[1,1]} \approx \begin{matrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{matrix}$$

Fig.1 Sobel operators (4 of 8)

$$g_{\mathbf{u}}(i,j) = \sum_{k=-1}^{1} a_{\mathbf{u}}(k,h) f(i+k, j+h) \tag{1}$$

$$\{f(i,j);\ i = 0,1,2 \ldots L_x - 1,\ j = 0,1,2 \ldots L_y - 1\} \tag{2}$$

$$A_{\mathbf{u}} = \{a_{\mathbf{u}}(k,h);\ k,h \in \{-1,0,1\}\} \tag{3}$$

$$\mathbf{u} = [x,y]; x,y \in \{-1,0,1\} \wedge [x,y] \neq [0,0] \tag{4}$$

$g_{\mathbf{u}}$(i,j) is the gradient of the image f(i,j) at location (i,j). It is two dimensional vector, therefore we computed it in 8 different directions Eq. (4). $A_{\mathbf{u}}$ are the masks (size 3x3) of the Sobel operators. Four masks of them are shown in Fig.1. The final gradient image is obtained by following equation

$$g_c(i,j) = \max\{g_{\mathbf{u}}(i,j); \forall \mathbf{u}\}. \tag{5}$$

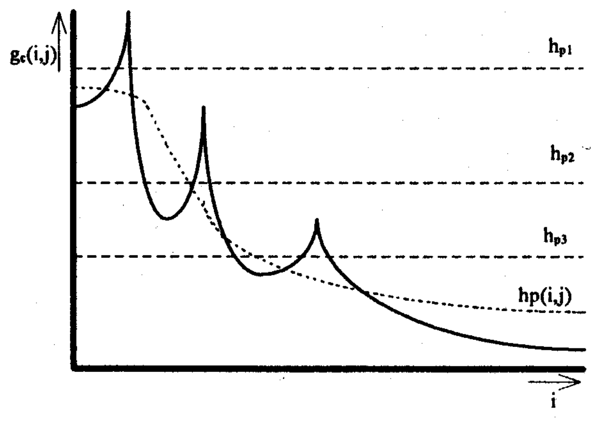


Fig.2. Cut of the thresholding function

The thresholding is one of the most effective methods how to obtain meaningful edges. Constant is usually used as a threshold (6). This way is not effective enough for following DV computation. Either we obtain the most important edges thin and others are lost or the second case is we obtain them all but the most important edges are too wide (Fig.2,3).

$$e(i,j) = \begin{cases} 1 & g_c(i,j) \geq h_p \\ 0 & g_c(i,j) < h_p \end{cases} \tag{6}$$

$$h_p(i,j) = \frac{1}{(2l+1)^2} \sum_{k=-1}^{1} \sum_{h=-1}^{1} g_c(i,j) \tag{7}$$

$$e(i,j) = \begin{cases} 1 & g_c(i,j) \geq h_p(i,i) + C \\ 0 & g_c(i,j) < h_p(i,i) + C \end{cases} \tag{8}$$

$$C = 0.1 \max(g_c(i,j)) \tag{9}$$

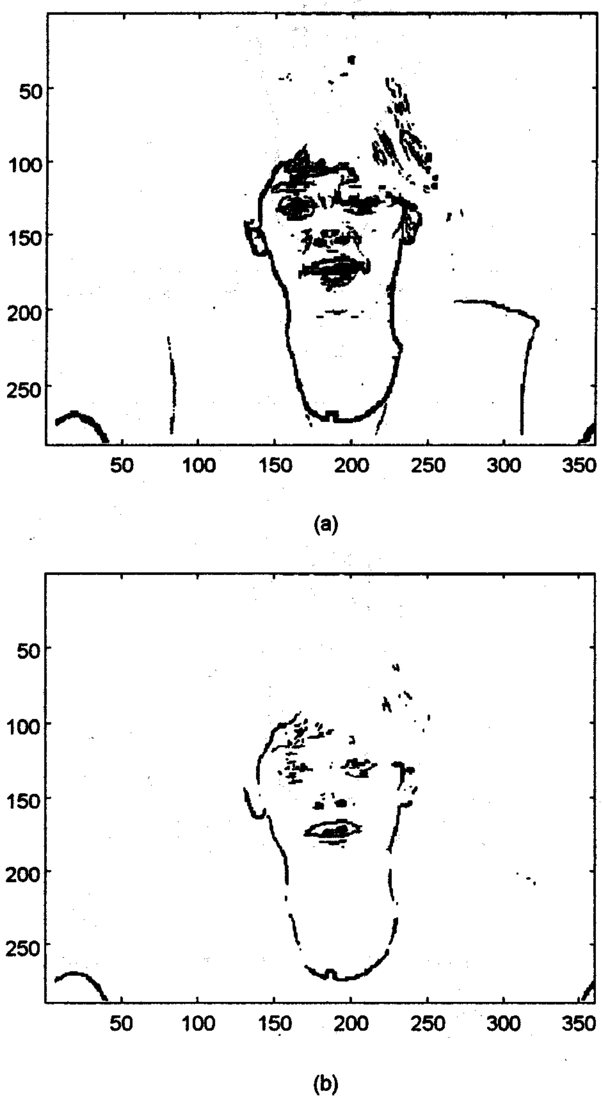
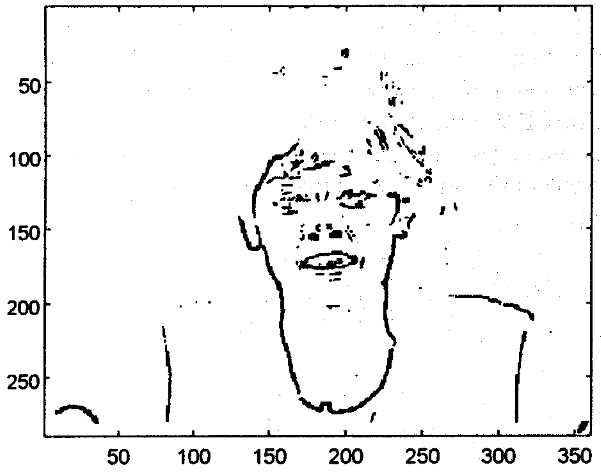
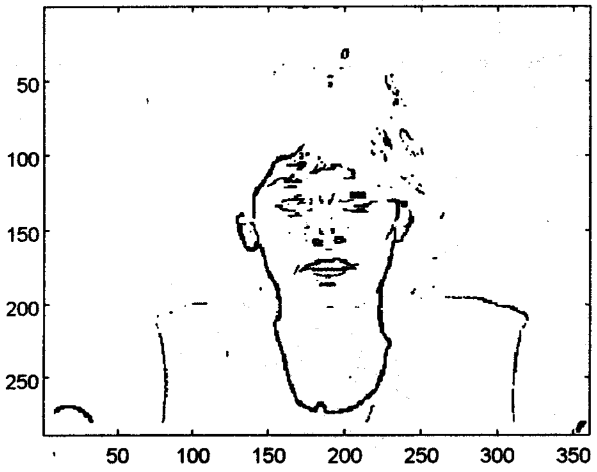


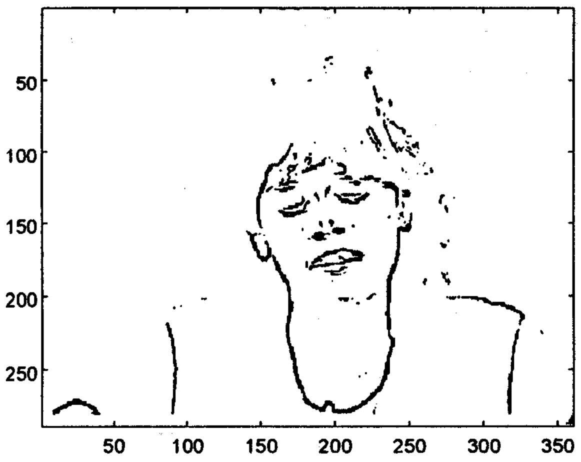
Fig.3 Edges computed by the constant threshold ($h_{p(a)} > h_{p(b)}$)



(a)



(b)



(c)

Fig.4 Meaningful edge in the image sequence 'Miss America'
(a) 1st image
(b) 8th image
(c) 64th image

If we have wide edges the correspondence problem grows. If we lose edges somewhere in the image, where the changes of the gray-level function are small, we lose information about a movement. Therefore we use the 2D thresholding function to obtain edges in the image (7,8,9). The method is more effective than a method with a constant threshold (Fig.2). The meaningful edges in the image sequence ' Miss America ' are shown in Fig.4.

3. Displacement Vectors Computation

The importance of the NN applications grows very enormously in the last years. Producers try to create processors based on NN with the higher effectiveness than conventional processors. The NN applications are very useful in cases if algorithm needs many computations. One of them is the image sequence processing [5,8,9,10]. Although the Hopfield NN is first of all used for DV determination in image sequence, we tried to use a different way. The 2D Hamming neural network is employed in this work. NN contains four layers (Fig.5):

- input layer
- computational layer
- competition layer
- output layer

The NN works according to the rules of correlation methods [7]. A basic feature of DV determination algorithms which use correlation methods is they are divided into two steps.

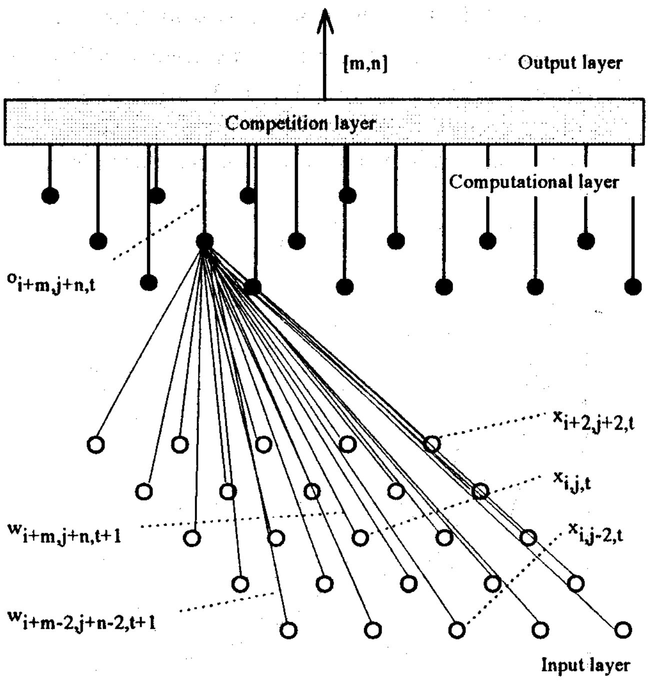


Fig.5 2D Hamming Neural Network

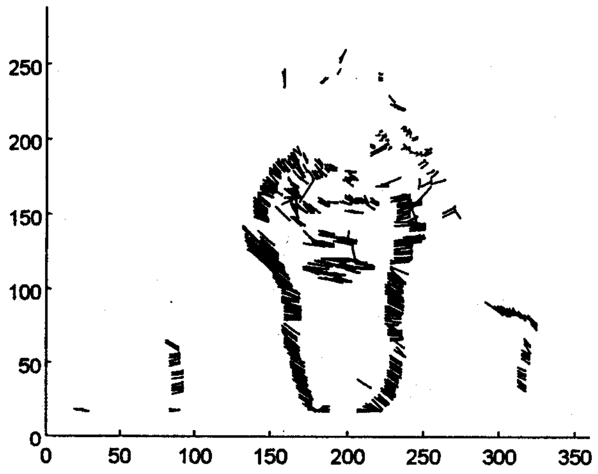
First, the suitable features are found in the images of the image sequence (in our case the edges). Second, it is necessary to find a characteristic feature from the first image in the second one. Cross-correlation coefficient is suitable for usage as a similarity criterion (Eq. 10)[4]. It is zero for totally dissimilar (orthogonal) patterns and it reaches a maximum for similar features. Cross-correlation coefficient can be expressed in the form

$$r^2(i+m,j+n)=$$
$$=\frac{\left(\sum_{h=-1}^{1}\sum_{h=-1}^{1}e(i+k,j+h,t)e(i+k+m,j+h+n,t+1)\right)^2}{\sum_{h=-1}^{1}\sum_{h=-1}^{1}e^2(i+k,j+h,t)\sum_{h=-1}^{1}\sum_{h=-1}^{1}e^2(i+k+m,j+h+n,t+1)} \tag{10}$$

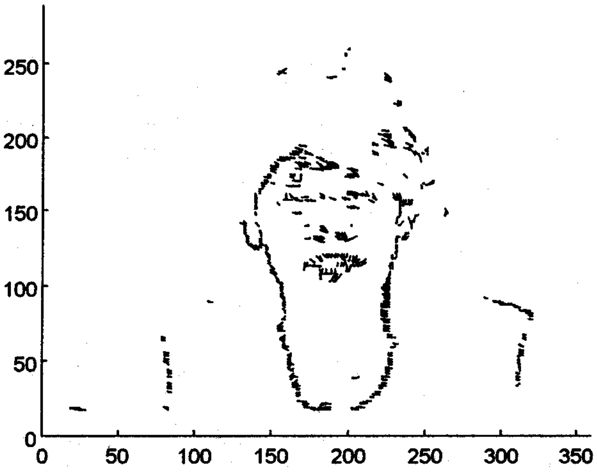
where $[i,j]$ are co-ordinates of a start point of the DV in the first image and co-ordinates of the block of the first image, $[m,n]$ are co-ordinates of the DV and relative co-ordinates of the compared block in a second image, $t, t+1$ are order numbers of images, $e(i,j,t)$ is a pixel in the edge image, $(2l+1)\mathrm{x}(2l+1)$ is size of compared blocks.
The computation requirements are very high if the cross-correlation coefficient is used. Therefore there are looked for simpler similarity criteria for real time applications (for example, mean square error or mean absolute value [2]). Our similarity criterion is simpler (Eq. 11,12)

$$o(i+m,j+n,t)=$$
$$=\sum_{k=-l}^{l}\sum_{h=-l}^{l}e(i+k,j+h,t)w(i+k+m,j+h+n,t+1) \tag{11}$$

$$w(i,j,t)=e(i,j,t) \tag{12}$$



(a)



(b)

Fig.6 Displacement vectors between two images
(a) 1st and 64th
(b) 1st and 8th

Its simplicity is caused due to simplicity of input data (binary) and due to the architecture of NN (Fig.5). Blocks of the first image enter in the inputs of neurons. The number of inputs depends on the size of compared blocks $((2l+1)\mathrm{x}(2l+1))$. The number of neurons in the NN depends on maximal assumed size of displacement of the objects in the picture. The weights of neurons are the values of pixels of the compared blocks in the second image. Each neuron computes the value of the similarity criterion for certain DV. Outputs of the neurons continue to the competition layer. In this layer the neuron is chosen which has computed the maximal value of the similarity criterion. Its co-ordinates are the output of the NN. They are the co-ordinates of the DV. The experimental results are presented in Fig.6. The size of the blocks was 15x15 pixels and the maximal size of the displacement was 15 pixels. Two cases (two different distances between corresponding features) were chosen for the better illustration. A point at the end of the DV indicates the displacement direction.

4. Conclusion

A new algorithm for a DV computation is described in this paper. It is built on a DV detection based on a full search block matching algorithm. We used the 2 D Hamming neural network to reduce the time of the computation. The input data were meaningful edges of the original image sequence (in this case the image sequence 'Miss America'). Fast, accurate and reliable results are obtained due to the neural network parallel computation capability and the correlation function simplicity. The similarity criterion can by modified according to the type of logical circuits, which is used for design of the NN (for example, the equality function instead of the AND function, etc.) We can achieve an additional time reduction by parallel connection of more neural networks. The presented algorithm has the best results with an image sequence with a relatively small motion. Then the important edges are better obtained and shapes of objects are better recognised.

References

[1] Böhmann,P.: Detekcia pohybu v obrazových sekvenciách, Rigorózna práca, TU Košice, 1996

[2] Gamec,J.: Metódy estimácie pohybu v obraze pri medzisnímkovom kódovaní, Rigorózna práca, TU Košice, 1992

[3] Gonzales,R.C.: Digital image processing, Addison-Wesley Publishing Company, 1987

[4] Hlaváč,V- Šonka,M.: Počítačové vidění, Grada a.s., 1992

[5] Chmúrny,J.-Čižmár,A.: Modely neurónových sietí na spracovanie signálov, Elektrotechnický časopis, 42, č. 9-10, 1991, s. 531-542

[6] Jähne,B.: Digital image processing, Springer-Verlag, 1994

[7] Komárek,T,-Pirsch,P., Array architectures for blocks matching algorithms, IEEE Transactions on Circuits and Systems, vol. 36., no. 10, October 1989, pp.1301-1308

[8] Kostelník,J.: Využitie neurónových sietí pri analýze obrazových sekvencií, Diplomová práca, TU Košice, 1995

[9] Levický,D.-Böhmann,P.: Artificial neural network for displacement vectors determination, Proceedings of the International Conference Artificial Neural Network and their Application Possibilities, Košice, 1996

[10] Levický,D.-Král,P.: Neural Networks in Visual Pattern Image Coding, Neural Networks World, 2/1995, pp. 163-169

[11] Skrzypkowiak,S.S.-Jain,V.K.: Affinity-weighted neural network motion estimation for MPEG coding, Digital Signal Processing 5, 1995, pp. 149-159

[12] Szostakovitch,J-Stajniak,A.: Displacement estimation by artificial neural networks, Second Conference Neural Networks and their Applications, Szczyrk, 1996, pp. 471-476

[13] Zhou,Y.T.-Chellappa,R.: Computation of optical flow using a neural network, in Proc. IEEE International Conference on Neural Networks, vol.2., 1988, pp. 71-78,

[14] Zhou,Y.T.-Chellappa,R.: A neural network for motion processing, Neural Networks for Perception, vol. 1., Human and Machine Preception, Academia Press, Inc., 1992, pp. 493-514

About authors...

Dušan LEVICKÝ was born in 1948 in Slanec, Slovak Republic. He received the M. Sc. degree and Ph.D. degree from Faculty of Electrical Engineering, of Technical University in Košice. He is currently working as professor with the Department of Electronics and Multimedia Communications at the Faculty of Electrical Engineering and Informatics, Technical University in Košice. His research interests include digital image processing, image transmission and neural networks.

Peter BÖHMANN was born in Handlová, Slovak Republic. He received M. Sc. degree in radioelectronics from Faculty of Electrical Engineering, of Technical University in Košice. Now a days he is a Ph.D. student at the Department of Electronics and Multimedia Communications the Faculty of Electrical Engineering and Informatics of the Technical University in Košice. His research interest concerns digital image processing and neural networks.