Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 58(72), Fascicola 1, 2013

# Eyes Detection by using Cellular Neural Networks

Corina Botoca[1]

**Abstract – This paper presents a novel method of eyes detection on color images of human faces, by using cellular neural networks. Based only on first order interconnections it is easily hard implementable.**
**Experiments made on a standard database show that the cellular neural networks algorithm is fast and reliable.**
**Keywords: eye detection, image processing, cellular neural networks**

## I. INTRODUCTION

Face recognition is a highly complex and challenging task, a research topic for the last decades that has been approached by different methods, categorized and evaluated in reference [1].
As a subtopic of face recognition, eyes detection might be used subsequently for a biometric identification in security systems, for a facial expression surveillance of a car driver, providing information on its mood or fatigue, as reference points in detection and tracking of faces in a dynamic environment, for autonomous robots.
Image processing is time consuming and it requires a high computing capacity. The characteristics of cellular neural networks (CNN) [2]: local interconnections, programmability and high processing speed, recommend them as a powerful tool for such a task.
There have been reported some studies for face detection but only on greyscale images that use the CNN technique, on the Cellular Neural Networks-Universal Machine (CNN-UM) [2], [3]. Reference [4] presents the extraction of facial features, such as nose, mouth and eyes by applying binary and morphological CNN operators on shades of gray images. Head and eyes detection on greyscale face images are used for human face normalization in reference [5], in order to transform images at the same dimension and position, as an intermediate step of an identification system. Features extraction of human faces by using CNN is studied also in [6] and compared with other techniques. A method for face detection and tracking based on CNN-UM is proposed in [7]. It was proven that using CNN in extracting facial features improves the recognition rate as well as the processing speed.

Advancing the greyscale image experiments referred in the literature [5] we implemented a fast and reliable algorithm suitable for eyes detection on color pictures of human faces. The CNN algorithm flow chart and its component parts are presented based on successions of interconnection templates [2], which play the role of instructions in a traditional computer. The interconnection templates used are only of first order, thus easily hard implementable on a CNN-UM chip [2]. The processed images are presented in different points, considered significant, along the algorithm steps.

## II. CELLULAR NEURAL NETWORKS

A CNN is a nonlinear dynamic system, formed by elementary processors, named cells, placed on a 3D grid, locally interconnected within a finite neighbourhood [2]. The local interconnection pattern, named cloning template or synaptic law, may be static or dynamic, linear or nonlinear, instantaneous or delayed. The CNN most important characteristics are their geometrical and electrical regular structure, the locality of interconnections between the processing elements and their programmability. The local interconnections between cells make CNN tailor made structures for various digital and analogical hard implementations, VLSI, optical, optoelectronic and neurobiological. The local interconnections and speed of processing are the great advantages of CNN over other neural networks. The invention of CNN-UM, the first spatio-temporal analog array computer [2], which processes with a trillion operations per second and of the Cellular Wave Computer, a brain inspired computer, capable of performing spatial temporal inference [3], open new perspectives to the CNN applications.
The dynamics of a bidimensional analogical CNN, represented in Fig.1.a) is described by the following equations:

$$C\dot{x}_{ij}(t) = -\frac{1}{R}x_{ij}(t) + \sum_{C_{kl}\in N_r(ij)} A_{ij,kl}\cdot y_{kl}(t) + \sum_{C_{kl}\in N_r(ij)} B_{ij,kl}\cdot u_{kl}(t) + I_{ij} \quad (1)$$

[1] Faculty of Electronics and Telecommunications, Communications Dept.
Bd. V. Parvan 2, 300223 Timisoara, Romania, corina.botoca@etc.upt.ro

BUPT

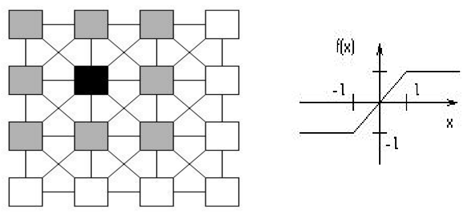
$$y_{ij}(t) = \frac{1}{2}\left(\left|x_{ij}(t)+1\right| - \left|x_{ij}(t)-1\right|\right) \quad (2)$$

where:

- $x_{ij}$ is the state of the $C_{ij}$ cell that satisfies the condition $\left|x_{ij}\right| \le 1$;
- $y_{ij}$ is the output f(x) of the $C_{ij}$ cell presented in Fig.1 b);
- $u_{kl}$ is the independent input in the $C_{kl}$ cell;
- $A_{ij,kl}$ is the weight of the feedback interconnection from the $C_{kl}$ cell to the $C_{ij}$ cell;
- $B_{ij,kl}$ is the weight of the control interconnection from the $C_{kl}$ cell to the $C_{ij}$ cell;
- $I_{ij}$ is the bias current of the $C_{ij}$ cell;
- C and R are positives constants;
- $N_r(ij)$ is the neighbourhood of $C_{ij}$ cell defined as :

$$N_r(i,j) = \left\{C_{k,l} \middle| \max(|k-i|, |l-j| \le r\right\} \quad (3)$$
$$1 \le k \le M, 1 \le k \le N$$

where r is the neighbourhood radius, a positive constant. The cells marked in gray in Fig.1.b) represent the neighbourhood of the black cell in the case of r=1, and is called a "3x3-neighborhood".




a) b)

Fig.1. a) A bidimensional CNN with a marked 3X3 neighbourhood b) A cell output function

The feedback interconnections $A_{ij,kl}$ and control interconnections $B_{ij,kl}$ of $C_{ij}$ cell are defined as matrices on a neighbourhood and they are called cloning templates, if they are the same for all the cells in a CNN. For example, for a neighbourhood of r=1,a set of 19 parameters completely defines the behaviour of the network for a given input and initial conditions. (9 coefficients for feedback interconnections $A_{k,l;i,j}$, 9 coefficients for control interconnections $B_{k,l;i,j}$ and 1 for bias I). This is the case of first order interconnections, recommended for hard implementations. Each pixel of an image is processed by a CNN cell, black corresponding to the cell state value of 1 and white to the cell state value of -1. All the other colours are normalised in the interval of [-1,+1] state values.

## III. THE CNN ALGORITHM AND EXPERIMENTS RESULTS

The dataflow is developed based on some properties of the human eyes, which are local, independent of the view perspective, THUS appropriate to be treated with CNN:

- The vertical and horizontal gradients are characteristic features;
- Eyes are black areas, relatively small and round;
- The eyes are two in a raw, located approximately in the same vertical position, in the interior of a face;

Consequently the CNN algorithm is compounded of the following processing parts, represented in Fig.2:

- Gradients processing;
- Black areas detection;
- Small and round black areas extraction;
- Interior of the face extraction;
- Two black areas on an horizontal line;

Each of the processing part of the CNN algorithm is a succession of templates, in its turn also an analogic CNN algorithm, as it will be further detailed. There were used basic image processing operators (as ccd_hor, gradient), spatial logical operators (as AND, OR) and morphological operators (as erosion, dilation) which templates can be found in [8], [9], [10].

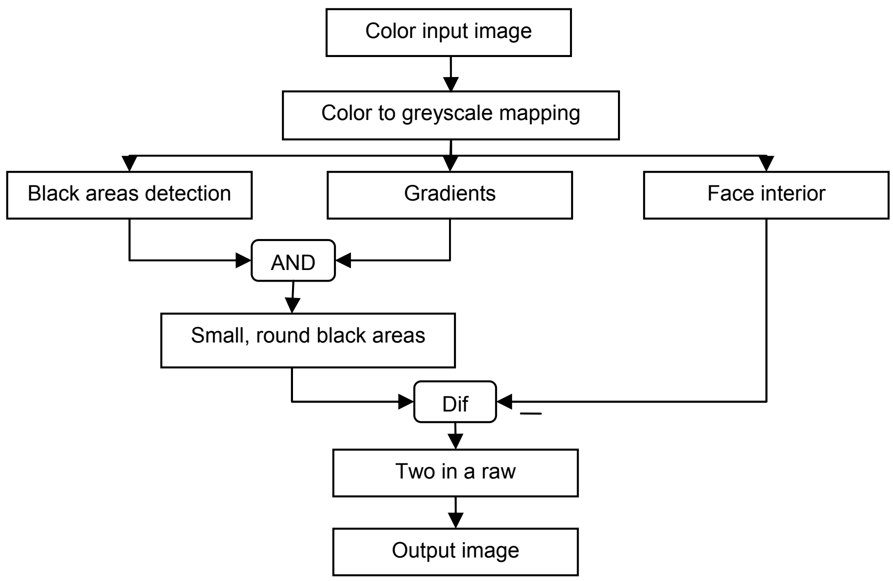



Fig. 2 CNN flow chart diagram for eye detection

The CNN templates that were used and their effects are presented in the following:

**Basic image processing:**

• threshold - marks the pixels that have values higher than a given level;

• gradient - generates a black pixel if the gradient intensity is above a selected value;

• grad_hor and grad_vert - detect areas where the directional gradient is larger than a given value;

• edge- detects the objects contour;

• ccd hor - indicates the number of objects in a row at the right side of the image, having as result some black pixels separated with one white pixel;

**Mathematical morphological operators** are followed by a sign denoting the structural element:

• dilation- makes the objects wider in the direction of the structural element;

• erosion – makes the objects thinner in the direction of the structural element;

• hollow - fills out the convex corners;

**Spatial logic operators:**

• junction - searches and joines dark fields with appropriate shape (e.g. a square dark field is not suitable) ;

• smkiller - deletes all separated pixels (white pixels in black surroundings and black pixels in white environment;

• logdiff (Dif) is a local boolean operator: the output pixel is black only if the state pixel is black and the input pixel is white;

• AND – generates the intersection of the independent input and state image;

• XOR – provides the logical disjunction (if the considered pixels are black and white, the resulted pixel is black;

• OR- provides the reunion of the independent input and state image;

• recall - reconstructs the marked objects;

• sh_left - spreads horizontally from right to left each black pixel;

Tests have been done on color images, randomly chosen frontal faces, from a standard database. It was used the CADETWin system [8] to develop and test the proposed CNN analogic algorithm.

First of all, the color image, used as input and as a fixed mask [2] has been converted into a greyscale image by using a succession of threshold and logical templates [9]. It was assumed that orange was the center of the color of the human skin, with a broad spectrum around it, from white to yellow, up to light brown.

The obtained image is represented in Fig.3. The greyscale obtained image was used as input to the other parts of the algorithm. Note that except the greyscale mapping, all the other parts of the CNN algorithm will generate binary output images, having only black and white pixels.

Fig.3. Color to greyscale image mapping

The gradients processing algorithm uses the succession of templates presented in Fig.4.

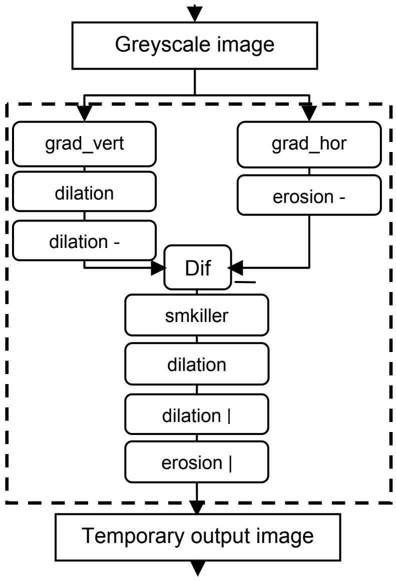



Fig.4 Templates flowchart for gradients detection

Vertical (grad_vert template) and horizontal gradients (grad_hor template) are computed in order to determine the head top, eyes, nose and mouth boundaries, respectively the left and right boundaries of face and nose as it can be seen in Fig.5.b) and Fig.5.c). In Fig.5.d) it can be seen the result of logical difference operator Dif (logdif template), which contains the possible areas of eyes.

a) b) c) d) e)

Fig. 5.Steps of gradients algorithm

BUPT

After eliminating small objects with smkiller, dilation and vertical dilation (dilation|) templates, it is obtained a first binary temporary image, represented in Fig.5.e).

The algorithm for black areas extraction is performed by the succesion of templates from Fig.6.

Junction template marks these areas, smkiller template erases small fragmented objects and the rest is expanded with dilation template. The result is presented in Fig.7.a.

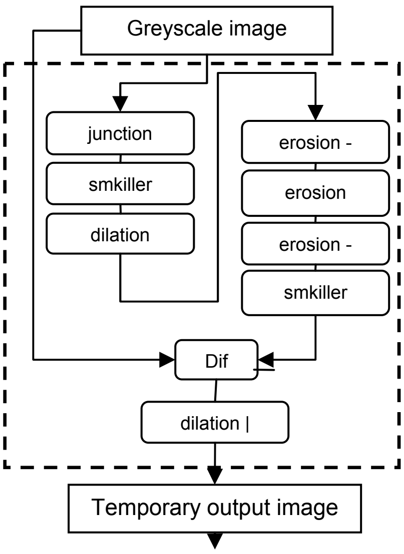



Fig.6 Templates flowchart for black areas detection algorithm

The next succesion of erosion and smkiller templates from Fig.6 delete the small areas of black pixels inside the face and only large homogeneous black areas will remain, as it is represented in Fig.7 b). Then this image is sustracted from the input original image and is obtained Fig.7.c).The two temporary images previously processed are used together with the greyscale original image as inputs of the small and round areas algorithm, with the flowchart of templates presented in Fig.8, to generate the possible areas of the eyes.

a) b) c)

Fig.7 Steps of black areas detection

Fig.9.a) represents the obtained image after AND operator. Edge template extracts the objects contour and hollow fills out the convex corners to obtain round objects. Smkiller template deletes the remained edges after filling and the close, fragmented objects are joined with dilation erosion and dilatation is obtained the image presented in Fig.9.b). Fig. 9.c) represents the greyscale original image processed with erosion template and Fig.9.d) the temporary output image of the small and round areas algorithm.

Than, the interior of the face algorithm is performed on the greyscale image, with the flowchart of templates presented in Fig.10.

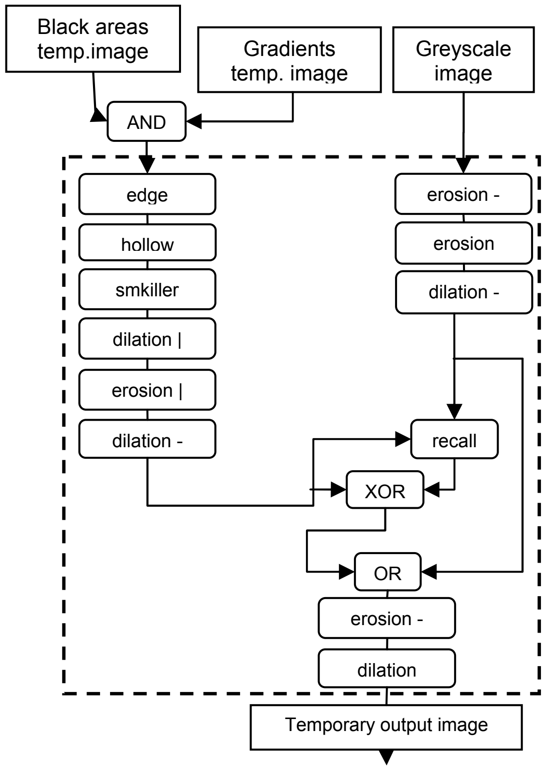



Fig.8 Templates flowchart for small and round areas extraction algorithm

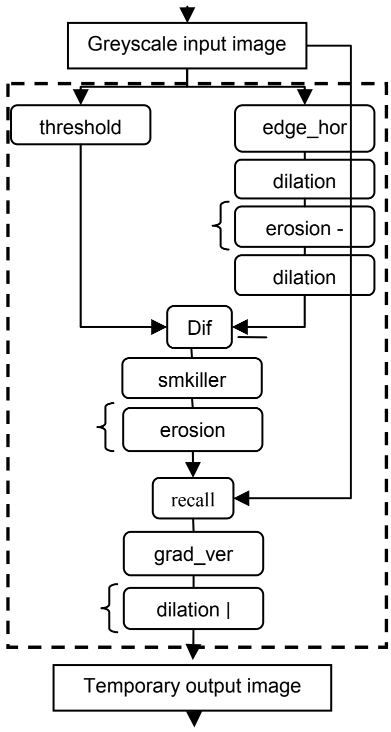
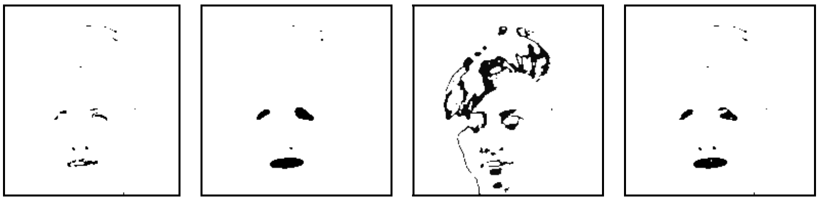



Fig.10 Templates flowchart for interior of the face extraction algorithm



a) b) c) d)

Fig. 9 Steps of small and round black areas extraction algorithm

a) b) c) d)

Fig.11 Steps of interior of the face extraction algorithm

The threshold template generates the dark areas, represented in Fig.11.a) which represents the background. The edge_hor template detects the horizontal edges of the initial image, which after a dilatation and erosion templates processing were subtracted from the background, providing the image from Fig.11.c). Glasses didn't disturb the CNN algorithm performance because the frame of glasses is larger than the eyes areas. The algorithm works also for semi closed eyes.

Smkiller and erosion templates erase small and medium sized objects. The remaining objects are recalled from the original greyscale image and the inverted face mask is obtained as temporary output image, as it can be seen in Fig.11.d)

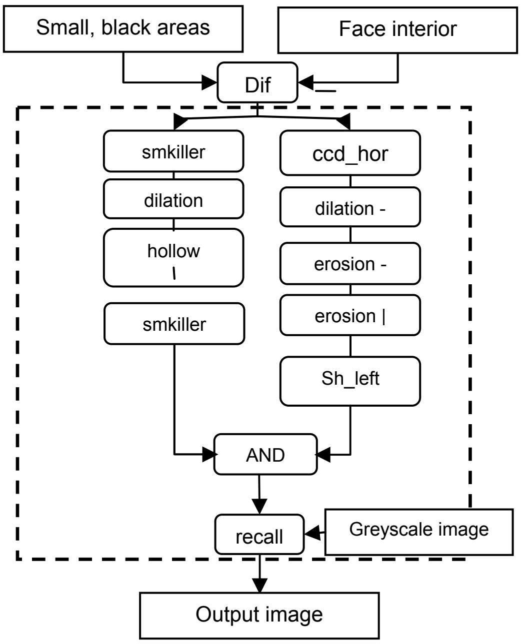



Fig.12 The flowchart of the two black areas on a horizontal line

The temporary interior of the face image is inverted and subtracted from the small and round areas temporary image and the result is applied as input image (Fig.13.a)) to two black areas in a raw, on a horizontal line processing part, as it can be seen in Fig.12. We have considered that the picture is a frontal view of a person. If the angle of head rotation view is only some degrees the eyes of a person are two in a row, located approximately in the same vertical position. Smkiller, dilatation and hollow compacts and magnifies the obtained black areas, possible location of the eyes. The ccd_hor template indicates the number of objects in a raw. If the number of connected components is one, then the row is deleted by erosion template, if not, as it is the case represented in Fig.13, the obtained black pixels are spread horizontally with the sh_left template and it is produced the picture from Fig.13.c). Then the possible location of the eyes is marked by using an AND template. By using a recall template the eyes are recalled on the original greyscale image, which means that they are marked with white spots, as represented in Fig.13.d).

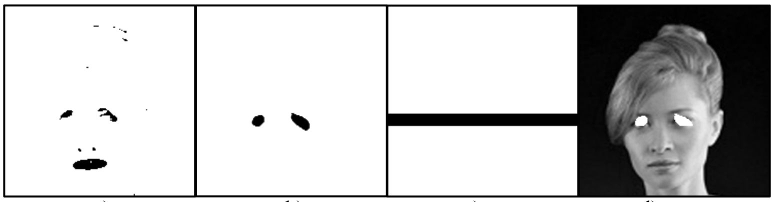
Tests were performed on 20 images, randomly selected faces from a standard database of faces. Only in two cases the algorithm didn't correctly find the eyes. The results are similar to those reported in the literature, where only greyscale images have been used [2]. In recent CNN literature there are no similar studies to specifically compare with.

## IV. CONCLUSIONS

The proposed analogic CNN algorithm detects the eyes on a color face image, an important step forward in face recognition.

The use of only 3x3 neighborhood templates make this method quite appropriate for hard implementations, using the CNN-UM chip.

BUPT



a) b) c) d)

Fig.13 Steps of the two in a raw algorithm

The processing time can be estimated to be of an order of ms [2] which is technically real time processing.

This recommends the method as a step towards a real time processing identification task.

Future research should also consider different angles of orientation of the faces and various lighting conditions.

## REFERENCES


[1] M.-H. Yand, D.J. Kriegman and N. Ahuja, “Detecting Faces in Images: A Survey”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2002, Vol. 24. pp. 34-58.

[2] L. O. Chua & T. Roska, *Cellular neural networks and visual computing, Foundations and applications*, Cambridge University Press, 2002.

[3] T. Roska, “Circuits, computers, and beyond Boolean logic”, *Int. J. Circ. Theor. App*l, Tom 35, 2007, pp.485–496.

[4] Z. Szlávik, T. Szirányi, “Face analysis using CNN-UM”, *Proc. of CNNA’04*, IEEE, , Budapest, sept, 2004, pp. 190-196.

[5] D. Balya and T. Roska, “Face and Eye Detection by CNN Algorithms”, *Journal of VLSI Signal Processing*, 1999, Vol. 2., pp. 497-51.

[6] H.D Vankayalapati, K. Kyamakya, “Nonlinear feature extraction approaches with application to face recognition over large databases”, *Proceedings of INDS'09 Second International Workshop on Nonlinear Dynamics and Synchronization, 2009* (INDS'09), pp.44-48.

[7] Mc Raven, M. Scheutz, Gy. Cserey, V. Andronache and W. Porod, “Fast Detection and Tracking of Faces in Uncontrolled Environments for Autonomous Robots Using the CNN-UM”, *Proceedings of the 8th IEEE International Workshop on Cellular Neural Networks and their Applications (CNNA 2004*), Budapest, 2004.

[8] “*CADETWin, User's Manual*”*,* Analogical and Neural Computing Laboratory, Computer and Automation Institute, Hungarian Academy of Sciences, Budapest, 1999.

[9] T. Roska, A. Zarandy and L.Chua, “Color image processing using CNN”, *Proc.Eur. Conf. on Circuit Theory and Design*, Amsterdam, 1993, pp.57-62.

[10] *Cellular Wave Computing Library*, (Templates, Algorithms, And Programs) Version 2.1, Budapest, 2007.


BUPT