A CNN Computing Algorithm for Image Correlation

ȚEPELEA Laviniu[1], REIZ Romulus[1], GAVRILUȚ Ioan[1], GACSÁDI Alexandru[1]
TIPONUȚ Virgil[2]

[1]University of Oradea, Romania,
Department of Electronics and Telecommunications, Faculty of Electrical Engineering and Information Technology,
Universității Str. 1, 410087, Oradea, Romania, E-Mail: {ltepelea, rreiz, gavrilut, agacsadi}@uoradea.ro

[2]Politehnica University of Timișoara, Romania
Applied Electronics Department, Faculty of of Electronics and Telecommunications
B-dul Vasile Pârvan, No. 2, 300223, Timișoara, Romania, E-Mail: virgil.tiponut@etc.upt.ro

Abstract* – To computing the correlation coefficients between two images, this paper proposes an algorithm based on the use of cellular neural networks (CNNs), in which most operations (calculations) are achieved by parallel processing. Thus, on the one hand, we can reduce computing time; on the other hand, the computing time will not increase proportionally with increasing the size of the template images. By integrating the CNN algorithm on a emulated digital CNN-Universal Machine implemented on FPGA (Field Programmable Gate Array) there will be possible to perform some task in real time in the case of a system developed to assist people with visual impairments or in a medical diagnosis assistance system, for processing and analysis of computer tomography images

***Keywords:* cellular neural networks (CNN); image correlation; emulated digital CNN-Universal Machine; Field Programmable Gate Array (FPGA).**

I. INTRODUCTION

In the case of template matching or image correlation the main objective is to detect and locate in a test image similar objects to an object contained by a template image (or kernel correlation) by determining the extent to which a portion of the image is similar or different from the template image. The image content analysis algorithms based on matching with template images are used most often to identify simple objects (numbers, letters, and simple shapes).

In the case of binary images the object from the binary template is detected, in a current input of acquired image, which after basic processing is also binary. By using an appropriate metric, often only a concrete answer is offered, whether or not the searched object from the template exists in the current image without the need to locate the exact position. In case of the pixel by pixel comparison of binary images, the result is unambiguous, there is or no match between two corresponding pixels of the two images.

To make image correlation or template matching, in the approach of gray-scale and color images one could use the extraction of as much relevant information from original image by successive processing, including their binary conversion and followed by the simultaneous evaluation of all these primary information. But if the gray-scale images are directly compared it is obvious that it is practically impossible to perfectly match the gray levels so one must use the difference of gray levels to obtain a value indicating the matching of the compared images.

II. TEMPLATE MATCHING OR IMAGE CORRELATION

To analyze the degree of match between two gray-scale images can be used different metrics or procedures such as: Euclidean distance, Sum of Absolute Difference (SAD) (eq.1) and Mean Absolute Difference (MAD) (eq. 2) namely, Normalized Cross Correlation (NCC) (eq. 3):

$$SAD(i,j) = \sum_{p=1}^{P}\sum_{q=1}^{Q}\left|K(p,q) - \Lambda(p,q)\right| \tag{1}$$

$$MAD(i,j) = \frac{\sum_{p=1}^{P}\sum_{q=1}^{Q}\left|K(p,q) - \Lambda(p,q)\right|}{PQ} \tag{2}$$

where, K(p,q) represents the template image or correlation kernel K: $R^2 \rightarrow R$, and

$\Omega_K=\{(p,q): p\in[1,P], q\in[1,Q], P \text{ and } Q \in R^+\}$

respectively, $\Lambda(p,q)$ represents the current image compared to the template image, Λ: $R^2 \rightarrow R$, and

$\Omega_\Lambda=\{(p,q): p\in[1,P], q\in[1,Q], P \text{ and } Q \in R^+\}$.

Corresponding to each such metric, for each image it will be determined the maximum coefficient in relation to the template image (or the minimum difference coefficient), or the coefficients higher than a threshold value, resulting if the searched object is in the image and in which position.

Although some of these procedures are relatively easy to implement using hardware methods, they require high processing time. In fact, the first of the above metrics, are preferred instead of computing the correlation coefficients, precisely because of the computational cost reduction, of course, sometimes to the detriment of accuracy.

On the other hand, the CNNs (Cellular Neural Networks) proved to be very useful regarding real-time image processing [1,2]. The reduction of computing time, due to parallel processing, can be obtained only if the algorithm can be implemented on a processor array [3,4,5,6]. In general, the CNN based implementation of pattern recognition methods is not a purpose in itself. In this way computing time can be reduced because of full parallel processing.

For some of these metrics (SAD and MAD) there were developed algorithms that were implemented using CNN, in either analog way, using the CNN chip, or an emulated digital CNN-UM (CNN-Universal Machine) implemented on FPGA (Field Programmable Gate Array). Even in these conditions, choosing the most appropriate way depends on the concrete application: the nature of processed images (binary images, gray-scale or color) image size, etc..

Thus, in [5] it is shown that using CNN binary image processing there is the possibility to compute the Hamming distance between objects. Another known distance metric is the Hausdorff metric which is more tolerant to shift error. The proposed new approach is based on a map generation corresponding to nonlinear wave propagation. Information on the similarities and differences between objects that are compared or classified can be easily extracted using local CNN operators, so the proposed algorithm could be easily implemented on a CNN hardware structure and provides an increased tolerance to noise effects.

Taking into account the limitations of the FPGA devices and special parameterization of the mandatory wave front sensors, in [6] the (SAD) method is used to implement the required correlation like processing. Several efficient FPGA implementations of the SAD algorithm have emerged, due to the claims of the FPGA implementation of motion image compression algorithms.

The system to assist visually impaired persons, called Bionic Eyeglass [7], which uses the CNN implementation of image processing is an integrated environment that takes into account many requirements. For this purpose, the recommendations of visually impaired persons were taken into account, and the following functions have been defined: recognition of clothes based on color and texture detection, recognition of pedestrian crossings, bill recognition, recognition of public transport signs and transport route numbers, recognition of the elevator sings and recognition of escalators movement directions. The development of the Bionic Eyeglass assistance system has taken into consideration the existence of three types of environments in which assistance is offered: at home, at work and the traveling between these two. Approaches of different functions fulfilled by CNN procedures are based upon the hardware and software infrastructure that includes, on one hand, from an architectural and visual dual Bi-i, on the other hand, an emulated digital CNN-UM implemented on an FPGA.

It may be noted that the macroinstructions developed for this system can facilitate the effective implementation of some metrics used also in pattern recognition and classification of objects. The introduction of additional simplifying criterions eases the achieving of the objective. Even so, in the case of visual travel assistant systems for people with visual impairments, the reported results solve particular classes of problems. Of course these architectures are in continuous development, both independently and by interconnecting them, to improving performance, in terms of increased accuracy and decreased processing time. The complexity of the situations encountered in reality, justify the existence of multiple CNN detection algorithms. Choosing of the most suitable algorithm for a given task remains a difficult task.

III. IMAGE CORRELATION

Let us take a gray-scale image $\Phi(m,n)$,

where Φ: $R^2 \rightarrow R$, and

$\Omega_\Phi = \{(m,n)$: $m \in [1,M]$, $n \in [1,N]$, M and N $\in R^+\}$, represents a test image, respectively K(p,q), a template image or correlation kernel), where K: $R^2 \rightarrow R$, and

$\Omega_K = \{(p,q)$: $p \in [1,P]$, $q \in [1,Q]$, P and Q $\in R^+\}$.

The test image is scanned pixel-by-pixel so that the template image to completely overlap the test image and the matching degree of each pixel is calculated. The matching degree with the template K(p,q) of a region $\Lambda(p,q)$ from the source image is obtained by computing the correlation coefficient (a numerical index) which indicate how well the pattern matches the contents of that region (compared image). Thus it results the correlation image Corr(i,j) or target image, Fig. 1.

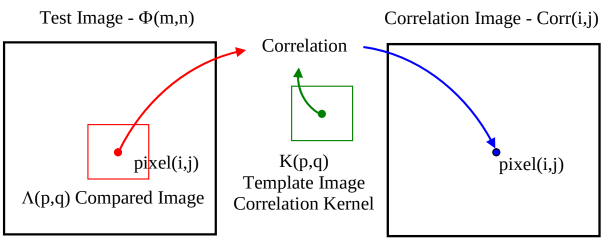

Fig. 1. The image correlation principles.

The correlation coefficient is a metric that expresses the similarity (the matching) between two images (the template image and a region of the test image), but not necessarily in values but in the overall behavior.

For image comparison the normalized cross correlation (NCC) is used on large scale defined as: (3)

$$\mathrm{Corr}(i,j) = \frac{\sum_{p=1}^{P}\sum_{q=1}^{Q}\left(K(p,q)-\overline{K}\right)\cdot\left(\Lambda(p,q)-\overline{\Lambda}\right)}{\sqrt{\sum_{p=1}^{P}\sum_{q=1}^{Q}\left(K(p,q)-\overline{K}\right)^2}\cdot\sqrt{\sum_{p=1}^{P}\sum_{q=1}^{Q}\left(\Lambda(p,q)-\overline{\Lambda}\right)^2}}$$

where, K(p,q) represents the template image,

$\Lambda(p,q)$ represents the current image compared with the template image with central coordinates (i,j).

$\overline{K}$ is the mean intensity of the template image,

$\overline{\Lambda}$ is the mean intensity of the test image in the region centered at (i, j).

The correlation coefficient has the value Corr(i,j)=1 if the two images are absolutely identical, Corr(i,j)=0 if they are completely uncorrelated, and Corr(i,j) = -1 if they are completely anti-correlated, for example, if one image is the negative of the other. Typically, the correlation coefficient is used to compare two images of the same object (or scene), taken at different times. Higher values of the correlation coefficients represent a better match between the two images (the template image and compared regions in the test image). For each image the maximum coefficient (or coefficients higher compared with a threshold value) will be searched. The choice of threshold value used for comparison is dependent on the application and is often between 0.3 and 0.8.

IV. CNN COMPUTING ALGORITHM FOR IMAGE CORRELATION

The computing time for obtaining a correlation coefficient of coordinates (i,j) is dependent on the template image's dimensions, increasing proportionally with them. On the other hand it is desirable that the template image size should be large enough to contain relevant information.

By using CNN parallel processing algorithm for calculating the correlation coefficient between two images the computing time can be reduced and it will not increase proportionally with increasing the template image size.

Relation (3) can be rewritten as: (4)

$$\mathrm{Corr}(i,j)=\frac{\overline{\left(K(p,q)-\overline{K}\right)\cdot\left(\Lambda(p,q)-\overline{\Lambda}\right)}}{\sqrt{\overline{\left(K(p,q)-\overline{K}\right)^2}}\cdot\sqrt{\overline{\left(\Lambda(p,q)-\overline{\Lambda}\right)^2}}}$$

That is, when calculating the correlation coefficient between two images of the same size several operations are used as: mediation, adding (subtracting), multiplying, operations which in the CNN field are achievable by parallel processing without the computing time depending on the image's size. The only two operations that cannot be made in parallel way, direct in CNN domain, are division and square root extraction. In some development environments there are already CNN algorithms (macros) for such operations, being implemented in parallel way using DSP.

Given the significance and value domain of the resulted correlation coefficients, there may be applications where the eq. (4) can be written in the form:

$$\mathrm{Corr}^2(i,j)=\frac{\left[\overline{\left(K(p,q)-\overline{K}\right)\cdot\left(\Lambda(p,q)-\overline{\Lambda}\right)}\right]^2}{\overline{\left(K(p,q)-\overline{K}\right)^2}\cdot\overline{\left(\Lambda(p,q)-\overline{\Lambda}\right)^2}} \qquad (5)$$

Using this transformation, the square root operation can be eliminated by replacing it with a multiplication of two images, (computing the square of an image), operation that can be done in a parallel way by the CNN. In this case we must consider that, in fact, high values of correlation coefficient will result even in the situation where the two images are completely anti-correlated.

Analyzing eq. (5) the proposed CNN algorithm for calculating the correlation coefficient is based upon the following observations:

- in the CNN field, the calculus of the mean value of an image can be done in parallel way (by wave computing) and it results an image having pixels with constant values that represent even the mean value of the initial image;
- to this purpose, there are more CNN templates that have the same smoothing effect on the processed image. To test the proposed CNN algorithm in this paper, we used the template aintpol2.tem [9] characterized by high propagation speed of the mediation wave (B=0, z=0): (6)

A =

0	0.25	0
0.25	0	0.25
0	0.25	0

- for reducing the computational time required to obtain the correlation coefficient generally is known that the mean value of the template image, respectively the image, must be calculated only once being considered a known constant; similarly, can be calculated for each position (i,j) the mean value of the compared image from the test image, respectively can be calculated simultaneously the difference image; in the field of CNN, when adding or subtracting two images one must take into consideration that the result of these operations do not lead to saturation. Therefore, always ahead of such operations input images are weighted by 0.5, taking into account this weight loss the value of the resulting correlation coefficient is not distorted;
- also multiplying two images in CNN domain can be done fully parallel (for example $\left(K(p,q)-\overline{K}\right)\cdot\left(\Lambda(p,q)-\overline{\Lambda}\right)$); this way the processing time is the same regardless of the image's dimensions; of course the images that are multiplied pixel by pixel must have the same sizes;
- using template aintpol2.tem, the calculus of the mean value of an image is similar as in case $\left(\overline{K}\right)$ or $\overline{\Lambda}$, for example $\overline{\left(K(p,q)-\overline{K}\right)\cdot\left(\Lambda(p,q)-\overline{\Lambda}\right)}$ and $\overline{\left(K(p,q)-\overline{K}\right)^2}$ and $\overline{\left(\Lambda(p,q)-\overline{\Lambda}\right)^2}$ can be calculated;
- depending on the implementation environment of the proposed CNN algorithm the images dividing operation can be done; so if the whole algorithm will be homogeneously implemented in CNN environment, there will be no time spent to pass from one domain to another. For example, the maximum correlation coefficient (maximum value of correlation image) can be determined also by CNN processing or one can obtain the higher coefficients compared with a threshold value.

V. TESTING THE CNN ALGORITHM FOR IMAGE CORRELATION

The proposed CNN algorithm to obtain the correlation coefficient between two images was tested by using the following simulation environments: InstantVision Integrated Software Environment [10], the CadetWin (CNN Application Development Environment and Toolkit under Windows) [9], and the Matlab Tools and Development Environment.

In this section the simulated experimental results are presented. For correlation coefficient computing, relation (5) was used.

An example for correlation image determination by using synthetic images, without noise, is shown in Fig. 2.

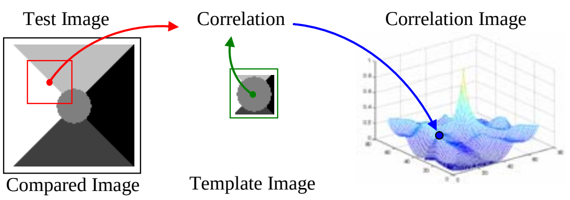

Fig. 2. Testing the CNN algorithm for image correlation.

In Fig. 3. was used real images, which might appear in case that the proposed CNN algorithm is included in a system to assist visually impaired persons [8].

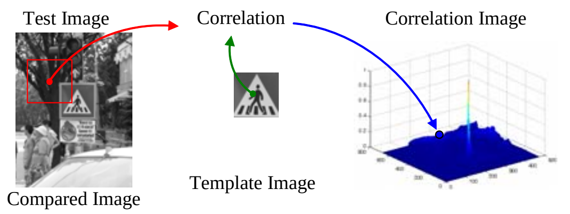

Fig. 3. Testing the CNN algorithm for image correlation.

If there are several regions of interest characterized by correlation coefficients that pass the sensibility threshold, the correlation coefficients are calculated in each of these regions through a procedure that ensures a high accuracy. The development of some algorithms with multiple conditioning thresholds can help the recognition of some shapes. If the processing time is sufficiently reduced, the recognition accuracy of a template can be increased.

V. CONCLUSIONS

In this paper the authors propose a CNN algorithm for computing the correlation coefficients between two images. In general, the computing time for the determination of the correlation coefficient between two images is high and dependent on the size of the template image, increasing proportionally with it. On the other hand it is desirable that the template image size should be large enough to contain relevant information.

To computing the correlation coefficient between two images, the majority of operations which are included in the proposed CNN algorithm are achievable by parallel processing. Thus, on the one hand, we can reduce the computing time; on the other hand, this time will not increase proportionally with increasing the size of the template image.

Depending on the concrete application where the proposed algorithm will be used, it is possible to determine the position of the compared image (region) of the test image, which shows the maximum correlation coefficient with the template image, or higher correlation coefficients compared with a threshold value. A special importance is the evaluation of the algorithm in regard to its integration into semiautomatic or automatic medical diagnosis systems, which provide results in real time that can be used in everyday, therefore an emulated digital CNN-UM implemented on an FPGA is necessary [4]. The proposed CNN algorithm for calculating the correlation coefficient will be integrated and tested in a system to assist visually impaired persons, respectively in a system to assist medical diagnosis using cellular neural networks to computer tomography image processing and analysis.

ACKNOWLEDGMENT

This work was supported by CNCSIS – UEFISCSU, project number PNII – IDEI 668/2008.

REFERENCES

[1] L.O. Chua, L. Yang: “Cellular Neural Networks: Theory and Applications”, IEEE Trans. on Circuits and Systems, Vol. 35, pp.1257-1290, 1988.

[2] T. Roska, L.O. Chua, “The CNN Universal Machine: An Analogic Array Computer”, IEEE Trans. on Circuits and Systems, Vol. 40, pp.163-173, 1993.

[3] G.L. Cembrano, A. Rodríguez-Vázquez, S. Espejo-Meana, R. Domínguez-Castro, “ACE16k: A 128×128 Focal Plane Analog Processor with Digital I/O“, Int. J. Neural Syst. 13(6), pp. 427-434, 2003.

[4] Z. Nagy, Zs. Vörösházi, P. Szolgay, “Emulated Digital CNN-UM Solution of Partial Diferential” Int. Journal of Circuit Theory and Applications Vol. 34, Issue 4, pp.445-470, 2006.

[5] I. Szatmári , Cs. Rekeczky , T. Roska, “A Nonlinear Wave Metric and its CNNImplementation for Object Classification”, Journal of VLSI Signal Processing, 23, pp.437-447, 1999.

[6] Z. Kincses, L.Orzó, Z. Nagy, G. Mező and P. Szolgay, “High-Speed, SAD Based Wavefront Sensor Architecture Implementation on FPGA”, Journal of Signal Processing Systems, Springer, New York, 06 May, 2010.

[7] T. Roska, D. Balya, A. Lazar, K. Karacs, and R. Wagner, “System aspects of a bionic eyeglass,” in Proc. IEEE International Symposium on Circuits and Systems (ISCAS’06), 2006.

[8] V. Tiponut, D. Ianchis, Z. Haraszy, “Assisted Movement of Visually Impaired in Outdoor Environments”, Proc. WSEAS Intern. Conf on Systems, pp.386-391, Rodos, Greece, July 22-24, 2009.

[9] *** CadetWin-CNN application development environment and toolkit under Windows”, Version 3.0, Analogical and Neural Computing Laboratory, Comp. and Aut. Inst. Hungarian Academy of Sciences, Budapest,1999.

[10] *** Bi-i V301F-Vision System, InstantVision Integrated Software Environmentv 3.1, User's Manual, AnaLogic Computers Ltd.