Applied Intelligence 18, 195–213, 2003

Image Processing Using RBF like Neural Networks: A ZISC-036 Based Fully Parallel Implementation Solving Real World and Real Complexity Industrial Problems

KUROSH MADANI
Intelligence in Instrumentation and Systems—I^{2}S, SENART Institute of Technology, University PARIS XII, Avenue Pierre POINT-F-77127 LIEUSAINT, France
madani@univ-paris12.fr

GHISLAIN DE TRÉMIOLLES AND PASCAL TANNHOF
IBM France, Laboratoire d'Etude et de Developpement, 224, boulevard John Kennedy, F-91105 Corbeil Essonnes Cedex, France
tremiollesg@fr.ibm.com
tannhof@fr.ibm.com

Abstract. The present article concerns neural based image processing and solutions developed for industrial problems using the ZISC-036 neuro-processor, an IBM hardware processor which implements the Restricted Coulomb Energy algorithm (RCE) and the K-Nearest Neighbor algorithm (KNN). The developed neural based techniques have been applied for image enhancement in order to restore old movies (noise reduction, focus correction, etc.), to improve digital television images, or to treat images which require adaptive processing (medical images, spatial images, special effects, etc.). We also have developed and implemented on ZISC-036 neuro-processor, a neural network based solution for visual probe mark inspection in VLSI production for the IBM Essonnes plant. The main characteristics of such systems are real-time control and high reliability in detection and classification tasks. Experimental results, validating presented concepts, have been reported showing quantitative and qualitative improvement as well as the efficiency our solutions.

Keywords: image processing, neural networks, ZISC-036 neuro-processor, real-time processing, industrial application

1. Introduction

As a result of their adaptability, artificial neural networks (ANN) present good solutions for an ever-increasing range of problems [1–3]. Even though their usefulness has already been confirmed, very few papers deal with real applications of this kind of technology.

In this paper we present a set of neural network based image processing techniques and their implementation on ZISC-036 [4–7] (Zero Instruction Set Computer), an IBM hardware implementation of the Restricted Coulomb Energy algorithm (RCE) and the K-Nearest Neighbor algorithm (KNN). Our methods take advantage from the power given by the high degree of parallelism provided by a hardware implementation of ANN. Each recognition process is performed using the SIMD (Single Instruction Multiple Data) characteristics of ANN thus improving the efficiency of the system without disturbing the response time. Another important point of the use of ANN, and particularly incremental ANN, lies in their adaptability: learning can be performed on-line to improve the response of the system.

The developed ZISC-036 based neural image processing technique has been applied to improve two different classes of applications. The first category of application concerned image enhancement in order to restore old movies (noise reduction, focus correction, etc.), to improve digital television, or to handle images which require adaptive processing (medical images, spatial images, special effects, etc.). Obtained results pointed out the efficiency of the developed solutions from the point of view of both quality and speed improvement. On the side of the processing speed improvement, obtained performances are approximately ten times higher compared to conventional systems.

The second kind of applications concerned the visual probe mark inspection dilemma in VLSI production. A neural network based solution has been developed and implemented on ZISC-036 neuro-processor, for the IBM Essonnes plant. The main advantages of developed solutions are real-time control and high reliability in detection and classification tasks.

The paper has been organized as following: Section 2 presents the RCE neural model, related learning algorithms and describes the basic properties of the IBM ZISC-036 component. In this section we also analyze a set of main parameters related to the RCE like neural based image processing and it's ZISC-036 based implementation. Noise reduction, image enhancement and visual probe mark inspection in VLSI production, using the ZISC-036 neuro-processor, are then described and discussed in Section 3. Experimental results and a comparison of the presented method with conventional ones have been reported in that section. Finally, the conclusion highlights the obtained results.

2. RCE Neural Network and it's IBM ZISC-036 Hardware Implementation

2.1. *From RCE Model to IBM ZISC-036 Architecture and Operation Modes*

The RCE (Restricted Coulomb Energy—Radial basis Functions) like neural models belong to the class of "evolutionary" learning strategy based ANN. That means that the neural network's structure is completed during the learning process. Generally, such kind of ANNs comprises three layers: an input layer, a hidden layer and an output layer. It is the hidden layer, which is modified during the learning phase. Each node of the hidden layer is a processing unit which computes the distance between the input layer and the prototype stored within each node (thanks connections between the input and the hidden layer). The output layer is used to give the categories, which correspond to the categorization of the input data. Connections between hidden and output layers are dynamically established during the learning phase. So, as presented in [2, 3], RCE's algorithm (for classification) consists of an output mapping of an n-dimensional space by prototypes, which are characterized by their own categories and influence fields [7]. The goal of the influence field is to activate or not activate the neighboring neuron.

The goal of the learning phase is to partition the input space by prototype where each prototype is associated with a category and an influence field, a part of the input space around the prototype where generalization is possible. In the RCE algorithm, the influence field defines the subspace limited by a hyper-spherical region in which the neuron is active. The influence field evolution for a given neuron is computed during the learning phase; when a prototype is memorized, the influence fields, called also regions of influence (ROI), of neighboring neurons are adjusted to avoid the problem of conflict between neurons. The neural network's response is obtained from relation R1 where C_j represents a "category", $I = [i_0\ i_1 \ldots i_n]^T$is the input vector, $P^j = [p_0^j\ p_1^j \ldots p_n^j]^T$ represents the "prototype" memorized (learned) thanks to connections of the neuron j to the hidden layer, and $\lambda_j$the influence field associated to neuron j (or Region Of Influence—ROI).

$$C_j = 1 \quad IF\ dist(I, P^j) < \lambda_j \tag{R1}$$

The classification task consists of computation the distances between the input data and the stored prototypes, and then comparing those results with the associated influence fields (learned prototypes). Thus, the presented vector could belong to zero, one or several categories. Two strategies could be used to construct the feature space, which maps a set of appropriated (related to the problem to be solved) categories: ROI (Regions Of Influence) algorithm and k-nearest neighbors (KNN) algorithm. In the ROI learning mode, a given class will be represented (in the feature space) by a set of prototypes (points in the feature space) and their influence fields. In the KNN learning mode, after the above-mention distance computation a matching

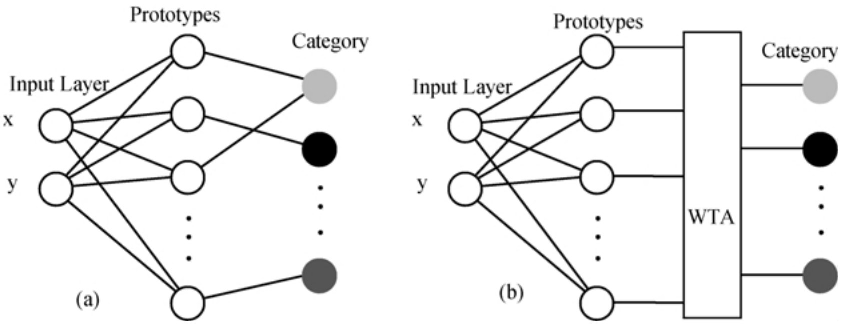


Figure 1. Block diagrams of neural network topology for the case of ROI (a) and KNN (b) learning modes.

operation, based on a "Winner Takes All" (WTA) like strategy, is performed to give preference to the shortest distance. Figure 1 gives bloc diagrams of neural network topology for ROI and KNN learning modes. Figure 2 shows the learning mechanism (neuron connections process during the learning phase) and associated feature space mapping in the case of the ROI based learning mode.

The IBM ZISC-036 is a parallel neural processor based on the RCE and KNN algorithms. Each chip is capable of performing up to 250000 recognitions per second. Thanks to the integration of an incremental learning algorithm, this circuit is very easy to program in order to develop applications; a very few number of functions (about ten functions) are necessary to control it. Each ZISC-036 like neuron implements two kinds of distance metrics called L1 and LSUP respectively. Relations R2 and R3 define the above-mentioned distance metrics were P represents the memorized prototype and V is the input pattern. The first one (L1) corresponds to a polyhedral volume influence field and the second (LSUP) to a hyper-cubical influence field.

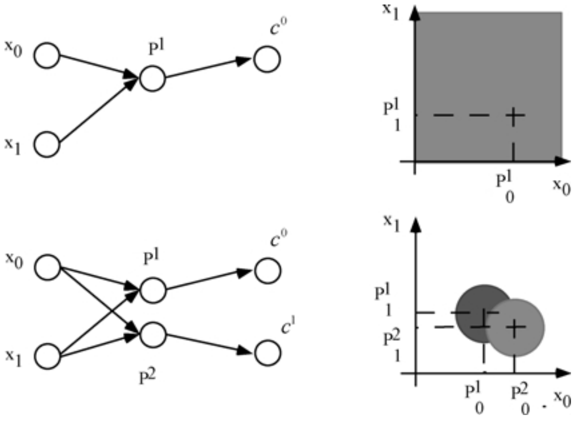


Figure 2. Example of neurons connections and feature space mapping in a 2-D space.

$$L1: dist = \sum_{i=0}^{n} |V_i - P_i| \quad \text{and} \qquad \text{(R2)}$$

$$LSUP: dist = \max_{i=0\ldots n} |V_i - P_i| \qquad \text{(R3)}$$

ZISC-036 is composed of 36 neurons. This chip is fully cascadable which allows the use of as many neurons as the user needs (a PCI card has been developed with a capacity of 684 neurons). A neuron is an element, which is able to:

- memorize a prototype (64 components coded on 8 bits), the associated category (14 bits), an influence field (14 bits) and a context (7 bits),
- compute the distance, based on the selected norm (norm L1 given by relation R2 or LSUP given by relation R3) between its memorized prototype and the input vector (the distance is coded on fourteen bits),
- compare the computed distance with the influence fields,
- communicate with other neurons (in order to find the minimum distance, category, etc.),
- adjust its influence field (during learning phase).

Beside the two possible distance computation modes, ZISC implements the two above-mentioned different learning modes leading to two different operation modes for the neuro-processor. Figure 3 shows an example of a 2-D feature space mapping using the ZISC-036 ROI and KNN learning modes respectively.

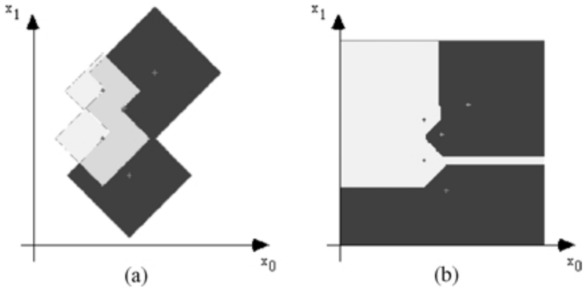


Figure 3. Example of an input mapping in a 2-D space: ROI (a) and 1-NN (b) using norm L1 in the case of the ZISC-036.

Figure 4(a) and (b) give the ZISC-036 neuron's structure and neural chip's bloc diagrams respectively, illustrating the connection, and accessing of the neurons. A 16 bit data bus handles input vectors as well as other data transfers (such as category and distance), and chip controls. Within the chip, controlled access to various data in the network is performed through a 6-bit address bus. Controlling the ZISC-036 is, by definition, accessing its registers, and requires an address definition via the address bus, and data transfer via the data bus. The inter-ZISC communication bus which is used to connect several devices within the same network, and the decision bus which carries classification information allow the use of the ZISC in a 'stand alone' mode. All neurons communicate via the 'inter-neuron communication bus.' An efficient protocol allows a true parallel operation of all neurons of the network even during the learning process. The inter-neuron communication bus is internally redriven to allow the connection of several ZISC modules without impact on performance. Because ZISC is a coprocessor device, it must be controlled by a master (state machine or controller). This can be done by a standard I/O bus. The I/O bus of ZISC036 has been designed to allow a wide variety of attachments from simple state machine interface to standard micro-controllers or buses.

Two kinds of registers hold information in ZISC-036 architecture: global registers and neuron registers. Global registers hold information for the device or for the full network (when several devices are cascaded). There are four global registers implemented in ZISC-036: a 16-bits Control & Status Register (CSR), a 8-bits Global Context Register (GCR), a 14-bits Min. Influence Field register (MIF) and a 14-bits Max. Influence Field register (MAF). Neuron registers hold local data for each neuron. Each neuron includes five neuron registers: Neuron Weight Register (NWR), which is a 64-by-8 bytes register, a 8-bits Neuron Context Register (NCR), Category register (CAT), Distance register (DIST) and Neuron Actual Influence Field register

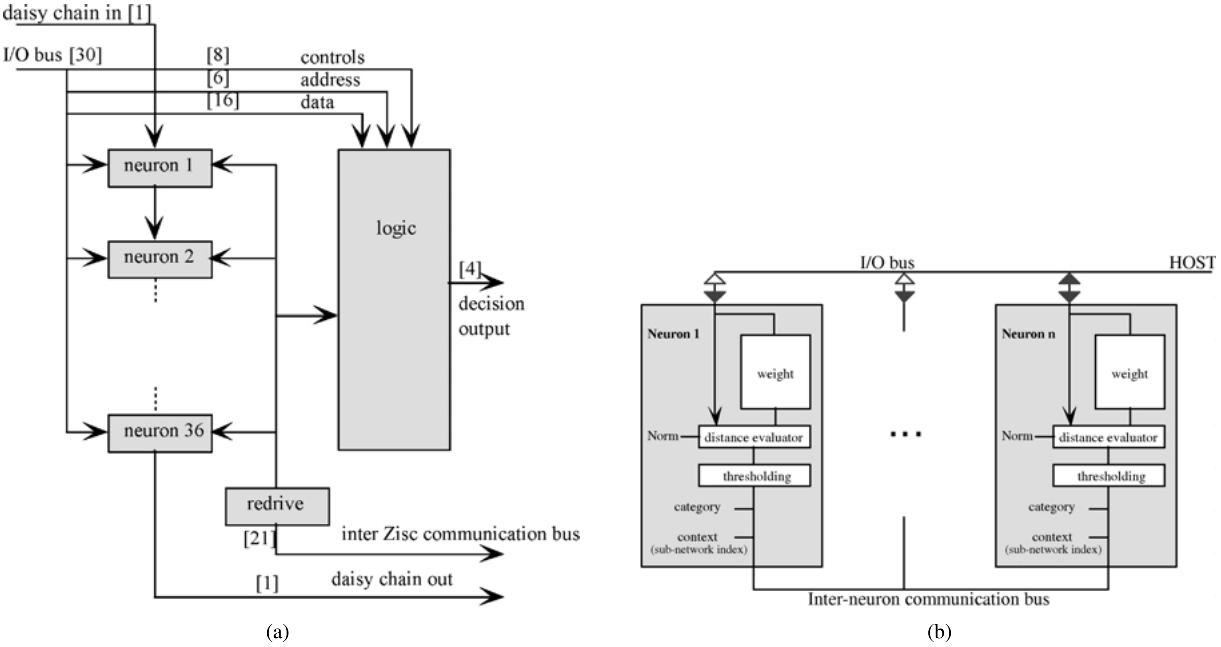


Figure 4. (a) IBM ZISC-036 neuron bloc diagram; (b) IBM ZISC-036 chip bloc diagram.

(NAIF). The last three registers are both 14-bites registers. Association of a context to neurons is an interesting concept, which allows the network to be divided in several subsets of neurons. Global Context Register (GCR) and Neuron Context Register (NCR) hold information relative to such subdivision at network and neuron levels respectively. Up to 127 contexts can be defined. When a neuron is committed, only the neurons having the same context as the GCR are activated. When GCR is set to zero, all neurons are activated whatever their specific context. The NCR register's value could be used to select sub-networks of neurons, which contribute to solve the same problem. The 7-th bit of both GCR and NCR registers corresponds to the norm (distance metrics) witch should be used [7–10].

2.2. *Main Parameters Analysis*

This subsection is dedicated to analysis of some main parameters in relation with learning and operation (relaxation) modes of the presented neural model and the hardware implementation described in the previous subsection. The necessity of analyzing influence of such parameters is related to the fact that due to hardware implementation constraints, the ZISC-036 implements an approximation of the original RCE neural model. So, mathematical proprieties (theorems), based on hyper-spherical nature of the influence field in original RCE, are not valid in the case of the ZISC-036 [14–17]. Thus before designing industrial applications using ZISC-036, it is of major importance to know the influence of a number of key parameters on neuro-processor's behavior and on the result's quality.

Before presenting different parameters analysis let us introduce the comparison criteria [15]. Such criteria can be used to evaluate the result's quality. Concerning the neural network based computation area, the most of used relations for result qualification is based on learning and generalization errors evaluation. In our case, the problem which we are concerned by deals with neural networks based image processing, and so, results qualification parameters are related to resemblance between images (output and desired ones). Relations R4 and R5 give two criteria relations that we used to evaluate obtained results quality. In relation R4, $R_{i,j}$ and $O_{i,j}$ represent pixels values of the reference (desired) image and output image, respectively. In relation R5, $\varphi_1(\cdot)$ and $\varphi_2(\cdot)$ represent some functions (threshold functions, or combinations, etc.).

$$\varepsilon = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}|R_{i,j} - O_{i,j}|}{n \times m} \tag{R4}$$

$$\varepsilon = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}\varphi_1(R_{i,j} - O_{i,j})}{\sum_{i=1}^{n}\sum_{j=1}^{m}\varphi_2(R_{i,j} - O_{i,j})} \tag{R5}$$

$$\text{with } \varphi_1(x) = \begin{cases} |x| & \text{if } |x| > \theta \\ 0 & \text{if } |x| \le \theta \end{cases} \quad \text{and}$$

$$\varphi_2(x) = \begin{cases} 1 & \text{if } |x| > \theta \\ 0 & \text{if } |x| \le \theta \end{cases}$$

2.2.1. Distance Computation. The choice of the distance calculation (choice of the used norm) is one of the main parameters in the case of the RCE-KNN like neural models (and derived approaches). The most usual function used to evaluate the distance between two patterns is the relation expressed by R6, known as Minkowski function:

$$dist = \sqrt[n]{\sum_i \left|V_i - P_i^j\right|^n} \tag{R6}$$

where V_i represents the i-th component of the input vector and P_i^j the i-th component of the j-th memorized pattern (learned pattern). As particular case of the Minkowski function (distance), one can recognize the Manhattan distance ($n = 1$, implemented as L1 norm on ZISC-036) and the Euclidean distance ($n = 2$). One can remark that:

$$\sum_i |x_i - y_i| \le \left(\sum_i (x_i - y_i)^2\right)^{\frac{1}{2}} \le \max_i |x_i - y_i| \tag{R7}$$

It is also possible to compute the distance between patterns thanks to a linear or none linear combination of presented relations. Of course, in that case, the computation could not be performed by ZISC-036 neural processor, which implements the Manhattan and LSUP norms. Figure 5 compares the obtained representation using the above-presented distances. The Fig 5(a) shows the case of the Manhattan's distance. The Fig. 5(b) gives the form of the influence region for the case of the Euclidean distance. The Fig. 5(c), represents the case of the LSUP norm and the Fig. 5(d) shows the influence region obtained combining Manhattan's and LSUP distances.

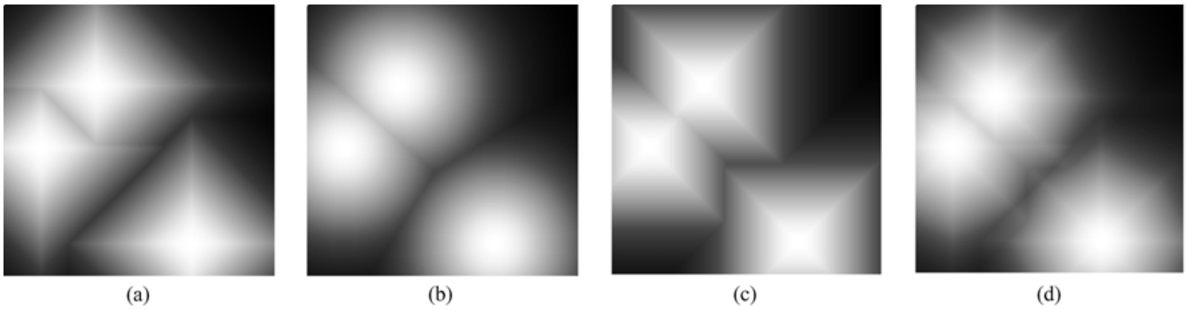


Figure 5. Examples of region of influence representations obtained from: Manhattan (a), Euclidean (b), LSUP (c) and, a linear combination of Manhattan and LSUP.

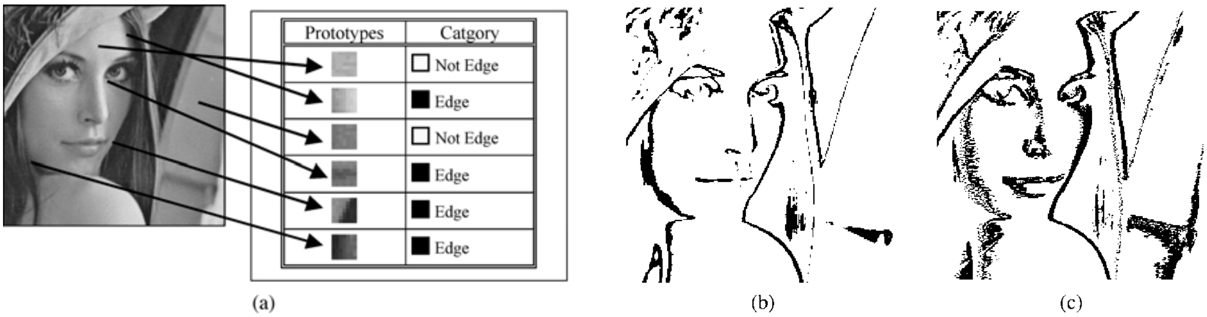


Figure 6. (a) Principle of the ZISC-036 based learning process in the case of edge detection application; (b) Result of the edge detection with Manhattan distance; (c) Result of the edge detection with LSUP distance.

The Fig. 6 (a)–(c) compares results relative to an edge detection application performed on ZISC-036. Of course, such application doesn't represent significant difficulty from the point of view of processing complexity and has not been presented here for that purpose. The goal is here to show, through a simple application, how a ZISC-036 based neural approach proceed to handle the problem (how the ZISC-036 learns the task and how the choice of the distance computation mode acts on obtained results). Presented in [11] the Fig. 6(a), shows the original gray level image and the learning principle (association of memorized prototypes to categories). Each prototype in this application is a 7 × 7 image. Two categories have been used: "Edge" and "Not Edge". The category "Edge" is a black pixel and the category "Not Edge" is represented by a white pixel. Figure 6(b) and (c) reproduce results corresponding to the edge detection with Manhattan's and LSUP distances, respectively [10]. On the side of the use of the L1 distance or LSUP norm, heuristic studies point out that the efficiency of the L1 norm is pertinent in the case of the pattern recognition (image and signal processing applications). The same studies conducted to a preference for the use of the LSUP norm in the case of supervision related applications (where some global parameters minimum or maximum magnitudes should be respected).

2.2.2. Influence of the Precision. A number of studies on effects of data precision (number of bits encoding the information) and effects on synaptic weights precision in the case of usual neural models (Back-Propagation, Multi-Layer Perceptron, Self Organizing Map, etc.) pointed out the influence of the precision on neural network's convergence performances. Effects related to the precision (data or synaptic weights) concern both learning and relaxation (operation) phases [9]. For usual neural models, these effects affect learning and generalization capabilities at both local and global levels. If w_{ij}^{t}, γ and ξ represent synaptic weight at instant t, learning control parameter and encoding precision, respectively, then the synaptic weights updating could be expressed by R8.

$$w_{ij}^{t+1} = w_{ij}^{t} + \gamma f\left(w_{ij}^{t}, \xi\right) \qquad \text{(R8)}$$

If α_{ij}^{t} represents the quantified value of the synaptic weights and If α_{Max} is the upper limit of the

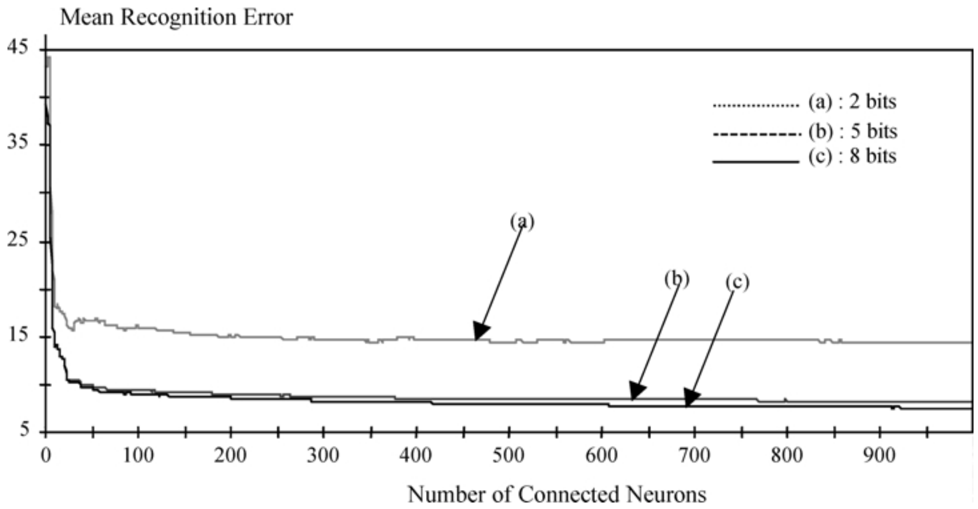


Figure 7. Learning mean error as function of number of connected neurons in the hidden layer for 2 bits precision (a), for 5 bits precision (b), and for 8 bits precision (c).

quantification dynamic ($\alpha_{ij}^{t} \in [0, \alpha_{\text{Max}}]$), then the condition $\alpha_{ij}^{t+1} = \alpha_{ij}^{t} + \gamma \text{Max}(f(w_{ij}^{t}, \xi)) = \alpha_{\text{Max}}$ will lead to a neural network evolution saturation.

In the case of RCE-KNN based neural networks (evolutionary learning), the precision on data does not affects local performances. However, it could affect global generalization precision [10]. As the ZISC-036 implements RCE-RBF like neural model, we have simulated the behavior of the ZISC-036 based neural implementation in the case of the learning procedure of the network for different data encoding precision (2 bits, 5 bits and 8 bits). Figure 7 represents the learning mean error as function of number of connected neurons (in hidden layer). One can remark that if a 2 bits precision seems be insufficient (obtained learning error is about 17%), 5 bits and 8 bits encoding precision lead to similar performances. So, a 5 bits precision could be considered as an acceptable lower limit. As in the case of the ZISC-036 data are encoded on 8 bits, we consider that effects related to data precision will not affect learning or generalization performances in the case of applications presented in this paper.

2.2.3. Operation Mode and Learning Criteria.

As it has been mentioned in Section 2.1, ZISC-036 implements two different operational modes: ROI based on region of influence adjustment and "KNN" mode, operating according to the WTA (Winner Takes All) principle. The operation mode is closely related to the computational and decision strategy: each operation mode implements a different computational and decision strategy. That will result on a different internal representation of a same learned pattern.

On the other hand, the used learning criteria will also lead to a different internal representation of the learned pattern. We call "learning criteria" the strategy (manner, order of presented patterns to be learned) which will be used to learn a set of patterns representing a given object (image, signal, information, etc.). The patterns presented as learning inputs (which should be learned by the neural network) are selected according to learning criteria. The learning criteria could be based on a randomly selection of presented patterns or be based on some deterministic rules. Generally, the used learning criteria relations are based on local difference evaluation between patterns parameters. Relations R9 gives an example of used learning criteria. In this relation, V_l^k represents the l-th component of the input vector V^k, P_l^j represents the l-th component of the j-th memorized prototype, C^k represents the category value associated to the input vector V^k, C^j is the category value associated to the memorized prototype P^j and, α and β are real coefficients adjusted empirically.

$$Learn\_Crit(V^k) = \alpha \sum_{l} \left| V_l^k - P_l^j \right| + \beta \left| C^k - C^j \right| \tag{R9}$$

We have learned the reference image of the Fig. 6 (using ZISC-036 neuro-processor), using the two

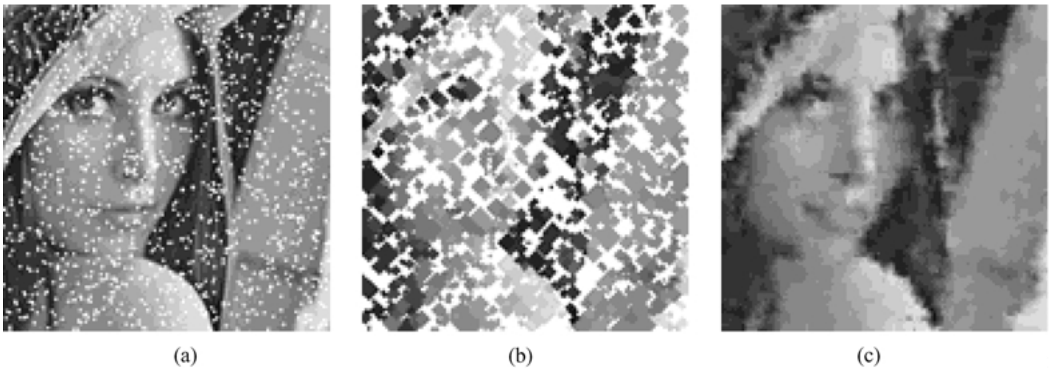


Figure 8. Selected learning patterns (a) according to the Crit_N°1 and internal representation of the learned image for ROI (b) and KNN (c) ZISC-036 operation modes.

ZISC-036 operation modes (ROI and KNN). For each operation mode, three different relations relative to learning criteria have been studied. The first learning criteria (Crit_N°1) is based on a randomly selection of sub image pattern. The second one (Crit_N°2) is based on a selection of the memorized prototype according to the distance between the selected sub image and the closest learned prototype. Finally, the last one (Crit_N°3) is based on the difference between the category value associated to the last (previous) learned pattern and the category value associated to the input pattern to be learned by the network. Figures 8–10 reproduces the obtained results. In each figure, the image (a) represents the reference image and locations of sub images that have been selected (as learning patterns) according to the used learning criteria. The second image of each figure represents the internal representation of the learned image when the operation mode is the ROI mode. Finally, the last image of each figure shows internal representation of the learned image in the case of the KNN operation mode.

The first remark could be made analyzing the above presented results is related to the choice of the operation mode. As one can remark from presented figures, the learning using ROI operation mode (Figs. 8(b), 9(b) and 10(b)) leads to some discontinuous internal representation (the representation of the image to be learned on the neural network's level) of the learned image. Even if for the case of Crit_N°3, the result seems to be a little better, in the case of Crit_N°1 and Crit_N°3 learning, such feature space mapping discontinuity will lead to undefined regions in the feature space resulting on ambiguous neural network's output. The best results, from the point of view of internal representation continuity, are obtained in the case of the use of the KNN operation mode. That is why later we will use ZISC-036 with this operation mode.

The second remark is related to the choice of learning strategy. Even if a randomly selection criterion

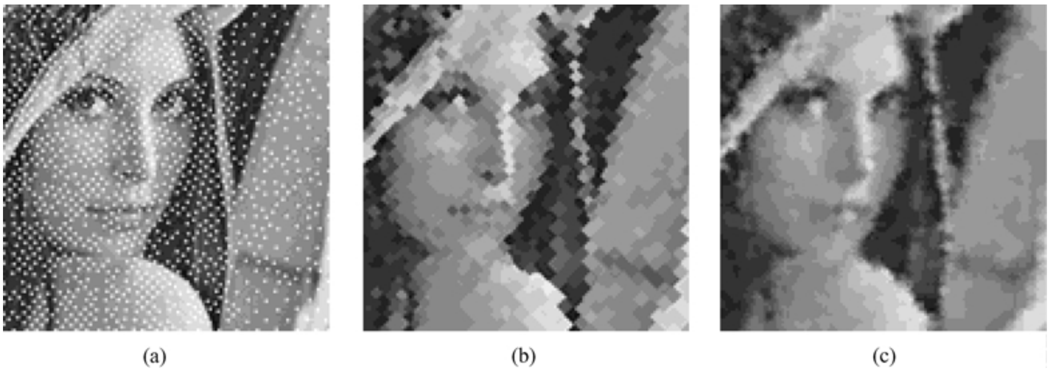


Figure 9. Selected learning patterns (a) according to the Crit_N°2 and internal representation of the learned image for ROI (b) and KNN (c) ZISC-036 operation modes.

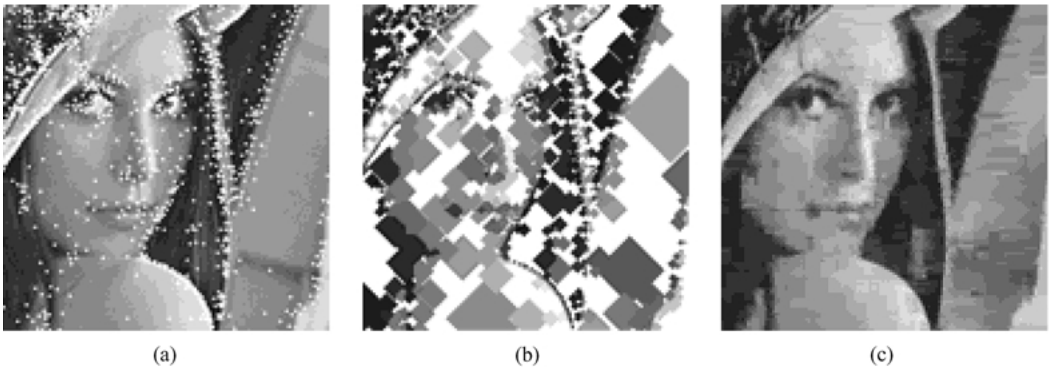


Figure 10. Selected learning patterns (a) according to the Crit_N°3 and internal representation of the learned image for ROI (b) and KNN (c) ZISC-036 operation modes.

(Crit_N°1) seems to be the most easy to implement, this learning strategy don't lead to the best internal representation. For the two other criteria relations the choice is more complicated and could be depended on a set of supplementary objective or subjective parameters. The problem of pertinent parameter consideration is still an open problem and there does not exist yet analytical methods unifying the pertinent parameter choice. Generally, such parameters are determined empirically. To illustrate this purpose let us consider a results qualification according to relations R4 and R5, presented previously. If on the basis of these results qualification on error computation between the neural net's output and a desired reference image is not good, the choice of the error evaluation criteria could lead to contradictory conclusions. Figures 11 and 12 show learning quality evaluation on the basis of relations R4 and R5 respectively. Each figure compares the mean learning error in the case of the three learning criteria relations that have been used previously (Crit_N°1, Crit_N°2 and Crit_N°3). One can remark that the error evaluation based on relation R4 leads to consider the Crit_N°2 as the best strategy. However, the use of the relation R5 conducts to consider the Crit_N°3 as the best learning patterns selection criteria.

The analyses of different above-considered parameters show, on the one hand, that the ZISC-036 has enough precision. On the other hand, they show the importance of the learning strategy choice as well as the choice of an evaluation criteria (error evaluation).

3. Application to Real-World and Real Complexity Industrial Problems

Taking into account the previous analysis, the present section is dedicated to the industrial solutions on the

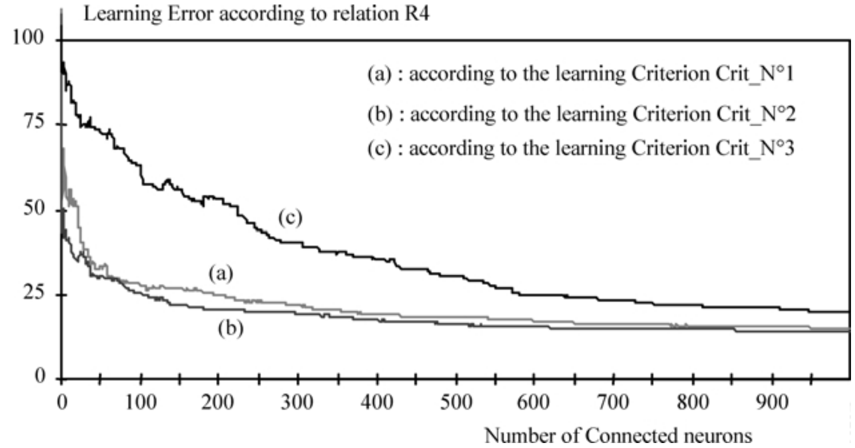


Figure 11. Learning error using relation R4 obtained for Crit_N°1 (a), Crit_N°2 (b), and Crit_N°3 (c).

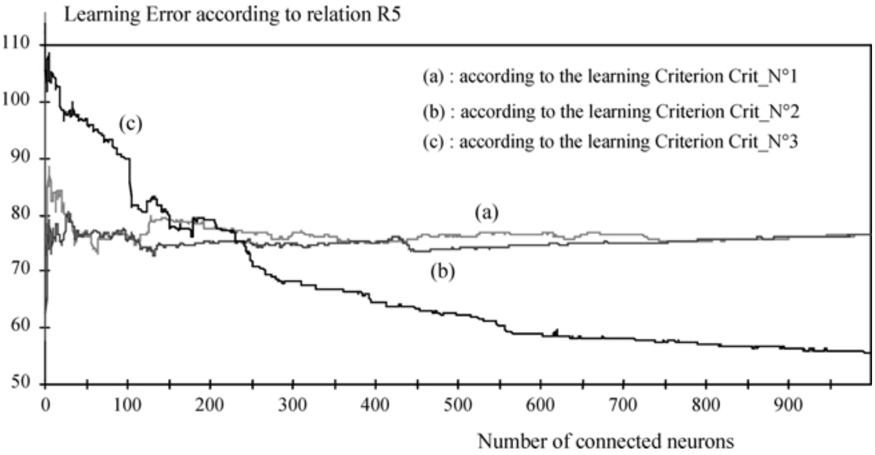


Figure 12. Learning error using relation R5 obtained for Crit_N° 1 (a), Crit_N°2 (b), and Crit_N°3 (c).

basis of the ZISC-036. Two problems have been considered. The first one is related to the noise reduction, contrast enhancement and colors restoration in images. This problem concerns the movie industry and especially, the old movies restoration and digitalization. The second problem reported in this section is related to VLSI manufacturing industry and concerns the probe mark inspection (PMI), which is an essential step in VLSI circuits production. It is important to note that the actual PMI procedures are based on conventional statistical analysis techniques and are performed manually.

3.1. Image Restoration and Coloration

3.1.1. Noise Reduction and Image Enhancement. Noise reduction of an image's represents an item of great interest for image analysis and understanding. Several solutions exist to accomplish this, such as median filter, low-pass and high-pass filters, etc., [12]. The main drawback of the aforementioned methods lies in their restrictive efficiency. Each kind of filter is efficient for only a precise kind of noise (gaussian noise, impulse noise, etc.), or for a precise kind of image. Artificial Neural Networks present the important property of adaptability, because the same network can be trained to remove some precise noise or a mixture of several kinds of noise presented in [3]. Another advantage of using ANN, and especially hardware implemented ANN, lies in the parallelism of the computational task, which allows real time processing. The following examples show how ZISC-036 can be used for noise reduction and image enhancement. After a learning phase on a non-corrupted image (image 13(a)), this system can be used to clean different kinds of images.

The presented principle is based on an image's physics phenomena which states that when looking at an image through a small window, there exist several kinds of shapes that no one can ever see due to their proximity and high gradient. The number of existing shapes that can be seen with the human eye is limited. ZISC-036 is used to learn as many shapes as possible that could exist in an image, and then to replace inconsistent points by the middle value of the closest memorized example. The learning phase consists of memorizing small blocks of an image (as an example 5 × 5) and associating to each the value of the middle pixel as a category. These blocks must be chosen in such a way that they represent the maximum number of possible configurations in an image. To determine them, the proposed solution consists of computing the distances between all the blocks and keeping only the most different.

After having learned examples, it is possible to use this database to enhance unlearned noisy images. All the blocks of the noisy image are compared one after another, with the memorized examples. For each block of the noisy image, the block's central pixel is replaced with the category value of the closest example from the database.

After having learned about one thousand sub images from the image 13(a), a noise reduction process using the neural network (ZISC-036) have been performed on a noisy version of an unlearned image (image 13(b)). Figure 13(c) shows the noisy version of the unlearned

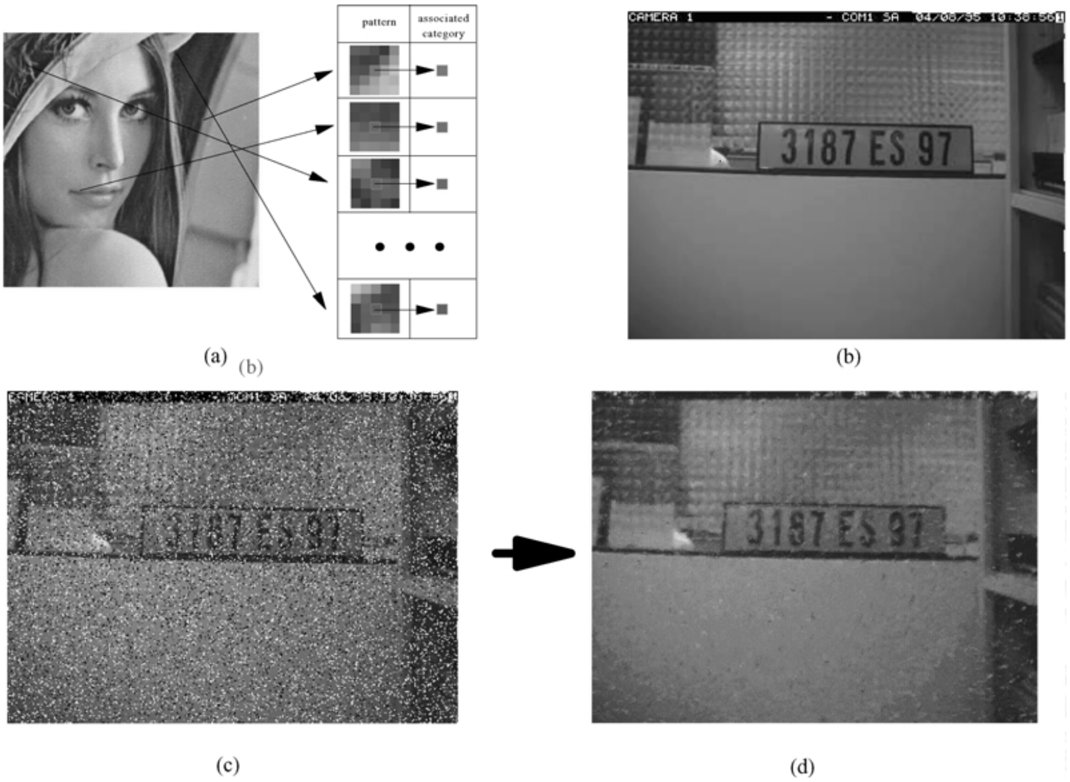


Figure 13. Results relative to noise reduction. Image used for learning phase and category association strategy (a), unlearned image (reference image) used for generalization phase (b), the noisy version the reference unlearned image (c), and the result given by the ZISC-036 (d).

reference image. The result of noise elimination is given by image 13(d). The best results are generally obtained when the learning phase is done on one or several images with characteristics (mean gray level, gradient, etc.) close to those on which the recognition is performed. When the learning phase is done images, which are very different from the image on which the restoration is performed (for example differing mean gray level), results may to be not as suitable as expected. That's because the closest memorized examples are evaluated for their gray level and not for their shapes. That's why a representative learning database (including a sufficient panel of different samples) should firstly be constructed.

The image enhancement principles are the same as those described above. The main difference lies in the pixel value associated to each memorized example. Previously, the learned input for the neural network consisted of blocks from the original image to be used to enhance a new image. The learned input of the neural network is a noisy form of the original image to be used for the enhancement of a corrupted image. For example, consider the two images presented in Fig. 14. Image 14(a) was obtained from image 14(b) by reducing the coding dynamic and fuzzying the pixel value (each pixel is the mean of itself and its 8 neighbors). For each memorized example (a block of 5 × 5) from image 14(a), the middle pixel of the corresponding block from the image 14(b) is used as the "corrected pixel value" and is memorized as the associated category. After having learned about one thousand five hundred examples from Fig. 14, the ZISC-036 based system is able to enhance the image presented in Fig. 15(a) to obtain the result presented by Fig. 15(b).

The learning algorithm used here incorporates a threshold and a learning criteria (Lean_Crit (V)). The learning criteria is the rule given by relation R9 introduced in Section 2 of this paper. The learning strategy is based of following technique: a random example from the learning base is chosen and the learning criteria for that example is calculated. If the value of the learning criteria function is greater than the threshold, then a neuron is connected, and an index (the number of neurons used) is increased. However if the learning

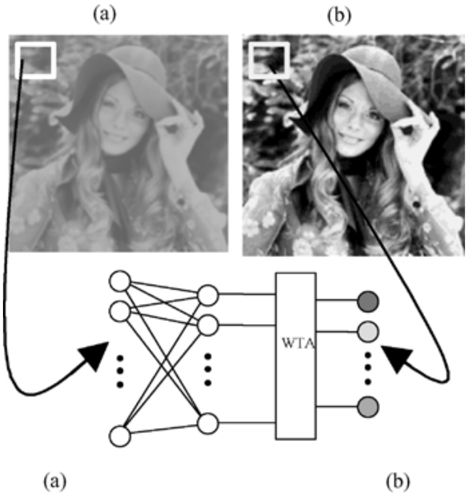


Figure 14. Example learning prototypes associating the degraded image (a) to a reference image (b).

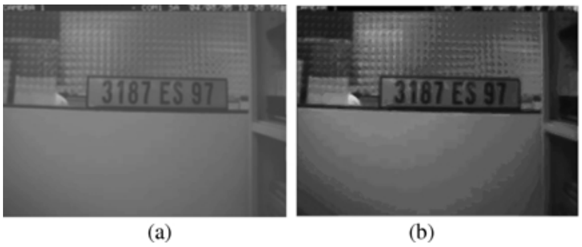


Figure 15. Example of image enhancement on using previous learning. Degraded unlearned image (a) and obtained enhancement using the ZISC-036.

criteria function's value is less than the threshold, no neuron is connected. After each iteration, the aforementioned threshold is decreased. Once the index reaches the desired level the learning phase is terminated. As it has been mentioned in Section 2, the choice of the coefficients α and β is linked to the application and not previously defined. Numerous simulations were performed to adjust the optimal values.

In order to smooth the resulting image, it is necessary to use a combination of the closest neighbors. Several solutions for combining the obtained values exist: combination of the distance of the blocks and the corresponding pixel value, the mean distance, etc. Relation R10 defines the function used, which give good results.

$$v = \frac{\frac{1}{0.01d_1^2+1} v_1 + \frac{1}{0.01d_2^2+1} v_2}{\frac{1}{0.01d_1^2+1} + \frac{1}{0.01d_2^2+1}} \qquad \text{(R10)}$$

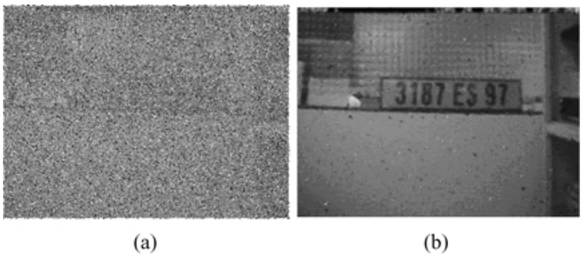


Figure 16. Example of image enhancement of a noisy unlearned image (a) and the ZISC-036 results (b).

where v is the estimation of the pixel value for the input block, v_1 is the pixel value associated with the closest memorized example, v_2 is the pixel value associated with the second closest memorized example, and d_1 and d_2 are the distances between the input block and the first and the second closest memorized examples respectively.

The response time obtained with our evaluation system (not optimized) using a PCI card is less than 0.6 s. By optimizing this algorithm and the associated hardware, it seems possible to divide the response time by a factor 10. In order to show the efficiency and robustness of the presented system, it is possible to add random impulse noise to the already corrupted image 15(a) (shown by image 16(a)) without modifying the data base (memorized examples) described above. The resulting image enhancement is seen in Fig. 16(b).

3.1.2. A Comparative Study. In order to evaluate more effectively the presented results, this subsection compares the results obtained using the above presented ZISC-036 based approaches with those obtained using the classical models of a histogram equalization followed by a median filter. To quantify the comparison mentioned before, we considered the case of the image enhancement of Fig. 16: so, the goal was to enhance the noisy image 16(a), using above-mentioned conventional image processing techniques. Firstly, a histogram equalization, consisting of adjusting the histogram of the original image's gray levels for the desired coding (in this case for eight bits), has been performed equalizing the image's gray level over the given range (result of such equalization is shown in Fig. 17(a)). Then, using the image obtained from the equalization, a conventional median filtering, consisting of placing in numerical order all of the pixel values of the current viewing box and replacing the block's central pixel with the median value [12], has

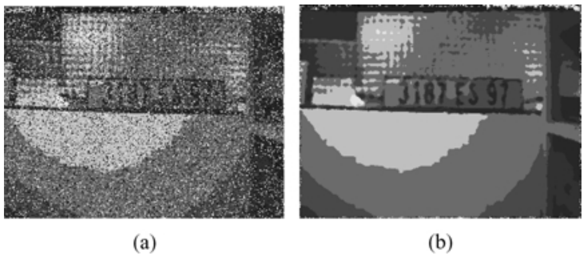


Figure 17. Example of image enhancement using histogram equalization (a) and followed by a median filter (b).

been realized over the image using a five by five block (results of the Fig. 17(b)). The median filter is used to sharpen the edges and decrease the blurring.

Graphs of the Fig. 18 represent a cross section of two images with the pixel index on the x axis and the corresponding gray level on the y axis. The Fig. 18(a) compares the gray level of the original image with that of the noisy image. Figure 18(c) and (d) show the results after the global equalization followed by the median filter (block of 7 × 1) respectively. Finally, the Fig. 18(b) shows the results obtained with the ZISC-036 with a block of 7 × 1.

It is also possible to calculate the mean error with respect to the image in order to evaluate the performance of these systems using the relation R11 where $R_{i,j}$ and $O_{i,j}$ correspond to the gray level of the points with coordinates (i, j) of the reference image and of the resulting image respectively and λ is a normalization parameter used to evaluate the error percentage. Table 1 compares the error obtained using the histogram equalization followed by median filter method with the ZISC-036 method on the images hat (Fig. 14 upon which random impulse noise has been added) and license plate (Fig. 18(b)) using optimal values for α and β as well as various block sizes.

$$\varepsilon = \lambda \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} |R_{i,j} - O_{i,j}|}{n \times m} \tag{R11}$$

3.1.3. Image Coloration. The image coloration technique operates also on the basis of same kind of principles, which have been described above. The difference here lies on the one hand, in the category associated to each memorized example, and on the other hand, in the nature of the learned input. The learned input for the neural network consisted of blocks from the original image, considered here as a pattern to be used to color a new image. The learned pattern is then used to determine the closest patterns in the new image, which will be colored identically as the learned pattern. Figure 19(a) shows the image used for learning database construction and the Fig. 19(b) gives the image, which has been colored. As in Fig. 18, graphs of the Fig. 20 represent a cross section of two

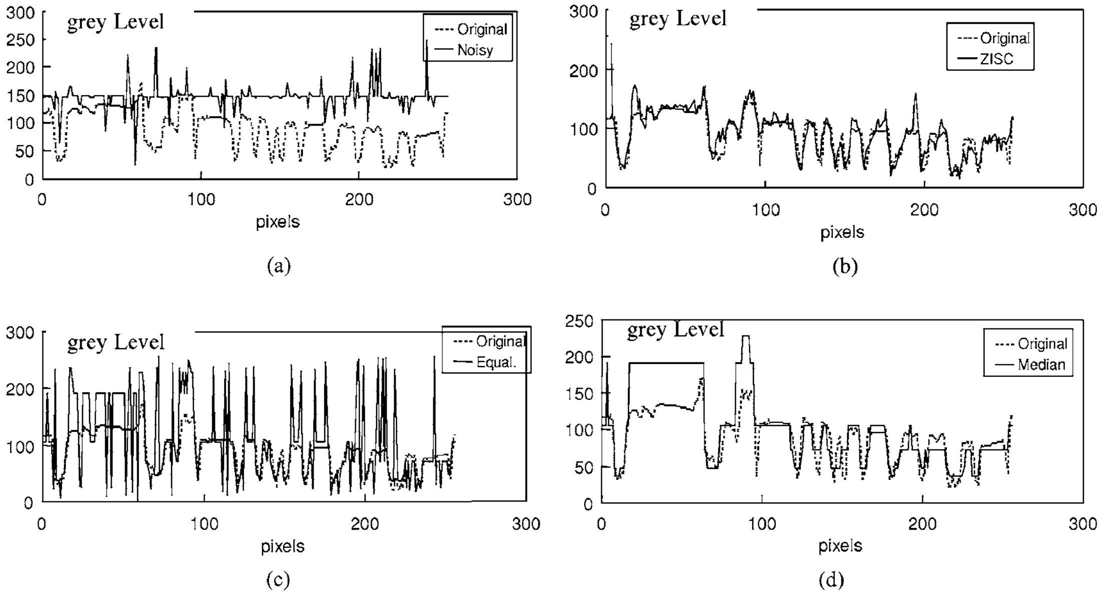


Figure 18. Cross section of the original and noisy images (a), ZISC-036 evaluation (b), histogram equalization (c) and median filter after the histogram equalization (d).

Table 1. Results: Mean error percentage with respect to the image.

	Hat (Fig. 14)			License plate (Fig. 15)		
	ZISC-036 $\alpha = 1/2\text{n}$, $\beta = 1/2$	Histogram equal.	Median filter	ZISC-036 $\alpha = 1/2\text{n}$, $\beta = 1/2$	Histogram equal.	Median filter
3 × 3	4.39		7.58	5.52	16.74	10.25
5 × 5	3.41	16.18	7.87	4.95		9.92
7 × 7	3.93		8.47	5.81		10.33
7 × 1	6.09		8.76	6.19		10.40
Noisy image	26.35			19.21		

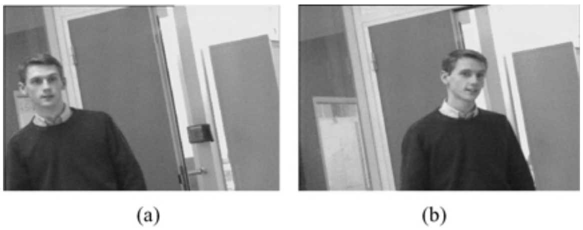


Figure 19. Experimental results relative to color images reconstruction. (a) Image used for the learning. (b) Image used for generalization test.

images with the pixel index on the x axis and the corresponding principle color component (red, green and blue components) on the y axis. The Fig. 20(a) compares the red component of the original image with that of the colored one, the Fig. 20(b) corresponds to the green component and the Fig. 20(c) evaluates the resemblance for the blue component.

Taking the above-presented subsections, it could now be remarked that all tasks necessary for the complete restoration and coloration process of a degraded movie (film) could be performed using the same neural based concept where only the learning databases change.

3.2. *Visual Probe Mark Inspection*

One of the main steps in VLSI circuits production is the testing step. This step verifies if the final product (VLSI circuit) operates correctly or not. The verification is performed thanks to a set of characteristic input signals (stimulus) and associated responses obtained from the

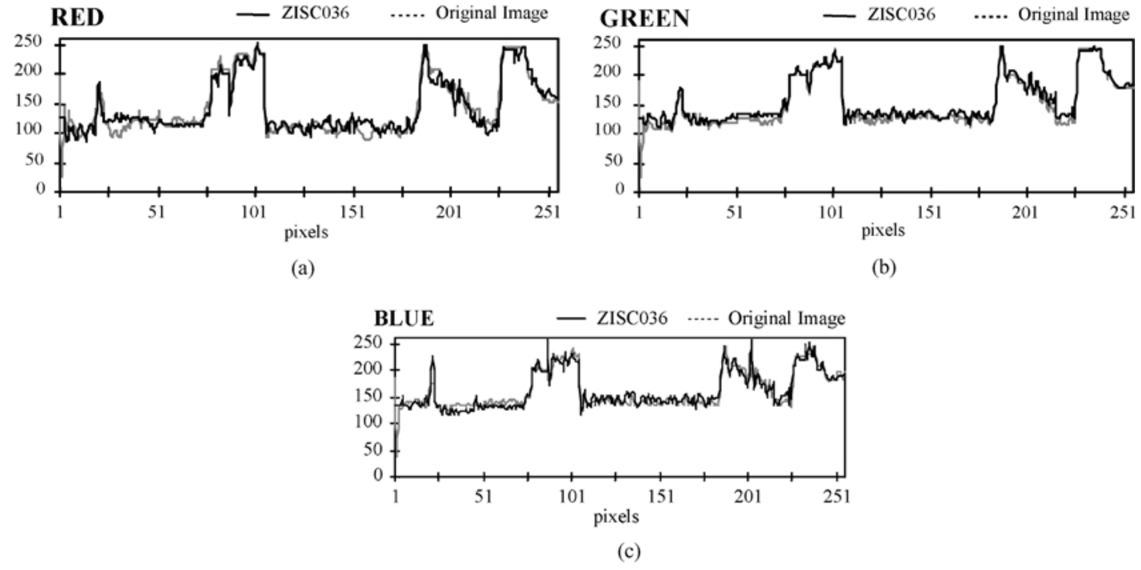


Figure 20. Comparison of the colored (reconstructed) image with the original image in generalization phase. Red (a), Green (b) and Blue (c).

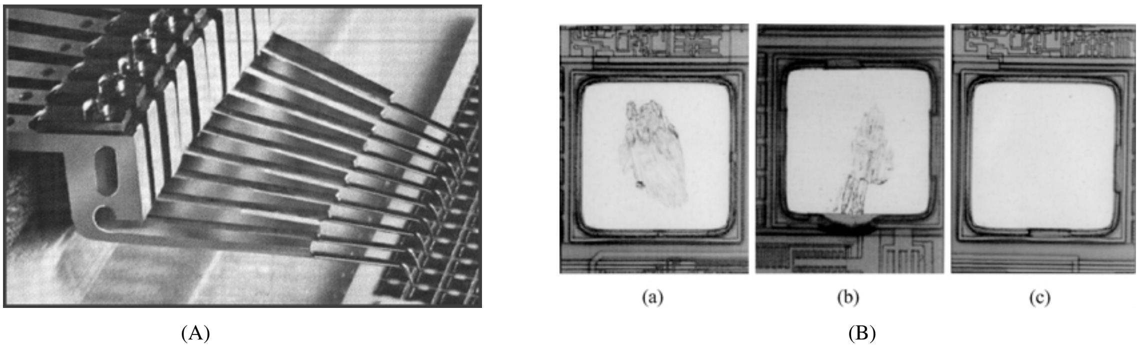


Figure 21. (A) Photograph giving an example of probes in industrial prober. (B) Example of probe impact: good (a), faulty (b) and absent (c).

circuit under test. A set of such stimulus signals and associated circuit's responses are called test vectors. Test vectors are delivered to the circuit and the circuit's responses to those inputs are catch through standard or test dedicated Input-Output pads (I/O pads) called also vias.

As in the testing step, the circuit is not yet packaged, the test task is performed by units, which are called probers including a set of probes performing the communication with the circuit. Figure 21(A) shows a picture of probes relative to such probers. The problem is related to the fact that the probes of the prober may damage the circuit under test. So, an additional step consists of inspecting the circuit's area to verify vias (I/O pads) status after circuit's testing: this operation is called developed Probe Mark Inspection (PMI). The Fig. 21(B) shows an example of faulty and correct vias.

Many prober constructors had already developed Probe Mark Inspection (PMI) software based on conventional pattern recognition algorithms with little success. The difficulty lies in the response time (real time execution with production speed constraints) and method reliability compromise. Even sophisticated hardware using DSPs and ASICs specialized in image processing have not performed sufficiently well to convince industrials to switch from human visual defects recognition to electronically automatic PMI. By now, prober constructors have finally abandoned automatic PMI in favor of another approach to the probe damage problem: checking the map of the probe and increasing precision in the mechanical movements which are quite expensive.

The use of hardware implementation of an ANN is interesting because it takes advantage of neural networks characteristics and mainly the small and fixed response time. Our PMI application, presented in [13], consists of software and a PC equipped with this neural board, a video acquisition board connected to a camera and a GPIB control board connected to a wafer prober system. Its goal is image analysis and prober control. Figure 22 represents the bloc diagram of the application.

The process of analyzing a probe mark can be described with the following steps:

- the PC commands the prober to move the chuck so that the via to inspect is precisely located under the camera.
- an image of the via is taken through the video acquisition board.
- the application, using the ZISC-036, then:
  - finds the via on the image.
  - check the integrity of the border (for damage) of via.
  - locates the impact in the via and estimates its surface for statistics.

The application then moves on to the next via.

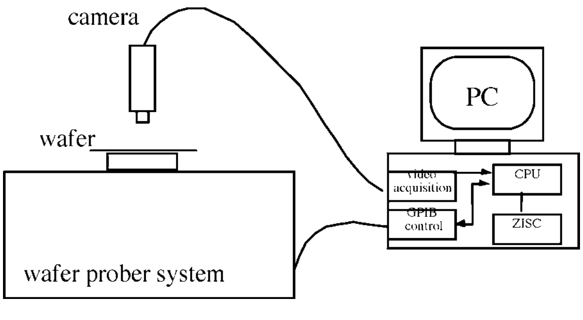


Figure 22. Structure of the presented application.

At the end of the process, the system shows a wafer map which presents the results and statistics on the probe quality and its alignment with the wafer. All the defects are memorized in a log file.

In summary, the detection and classification tasks of our PMI application are done in two steps:

- localizing the via on the acquired image.
- classifying the probe impact (good, bad or none) and estimate the size of this mark.

3.2.1. Centering the Vias. The method, which was retained, for centering the vias is based on profiles analysis [16]. Each extracted profile of the image (using a square shape, Fig. 23) are compared to a reference database in which each memorized profile is associated with its offset from the perfectly aligned profile (Fig. 24).

The used ISA card memorize all the examples and to do all the comparison in parallel. The used offset is a pseudo-linear extrapolation of the two offset given by the two nearest neighbor. This offset, for a profile, is also done using relation R12:

$$\delta = \frac{d_1 \cdot \mu_2 + d_2 \cdot \mu_1}{d_1 + d_2} \tag{R12}$$

where d_1 is the distance between the first nearest neighbor and the extracted profile, μ_1 is the associated offset and d_2 and μ_2 are the distance and the offset associated to the second nearest neighbor. In this kind of application, RCE's algorithm presents some drawbacks mainly due to the "hole" leaved during the space mapping out which explained why KNN has been chosen for this task. The mean of all responses gives a global offset (relation R13) for x and y axes:

$$\begin{aligned} \overline{\Delta_x} &= \frac{\sum_{i=1}^{m} \delta_i^l - \sum_{j=1}^{m} \delta_j^r}{2m} \quad \text{and} \\ \overline{\Delta_y} &= \frac{\sum_{i=1}^{m} \delta_i^u - \sum_{j=1}^{m} \delta_j^d}{2m} \end{aligned} \tag{R13}$$

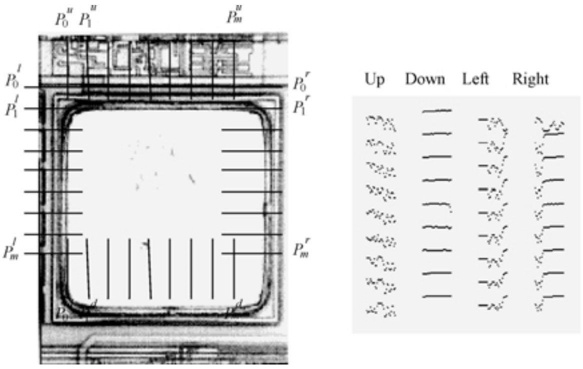
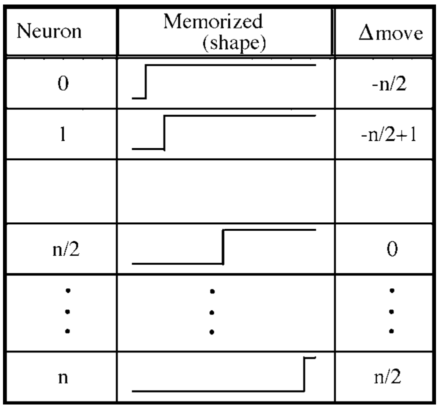


Figure 23. Profiles extraction.



Neuron	Memorized (shape)	Δmove
0		-n/2
1		-n/2+1
n/2		0
• • •	• • •	• • •
n		n/2

Figure 24. Database reference.

This step is repeated until $\overline{\Delta_x}$ and $\overline{\Delta_y}$ are null or until the square formed by the profiles oscillate. The problem of oscillation means that this square's size is not adapted. In this case, its size is adjusted (reduced or extended), using the range of oscillation, until $\overline{\Delta_x}$ and $\overline{\Delta_y}$ converge to zero. This phase of centering needs about 30 neurons and takes about 10 ms. A precise location needs, in general, 3 iterations [16].

3.2.2. Via Classification and Probe Mark Estimation. The principle of the classification is the same that centering one which is described previously. Each extracted profile of the image is compared to a database, which contains good and bad profiles (see Figs. 25 and 26). The reference database is done by the user at the beginning of the process or/and on-line when the system hesitates on a classification. The idea of hesitation is based on the distances which separates an extracted profile to a good one and to a bad one from the database. A via is declared bad if at least one of the extracted profile is nearest to a bad example of the data base than to a good one (without hesitation). At least a dozen of neurons are required in this phase to achieve 90 to 95% success. The classification of the border takes about 7 ms.

Another important point in the analysis of the image of a via is the size and localization estimation of

Profile	Category	Profile	Category
	Faulty		**OK**
	Faulty		**OK**
	Faulty		**OK**

Figure 25. Example of profiles to category association (learning phase).

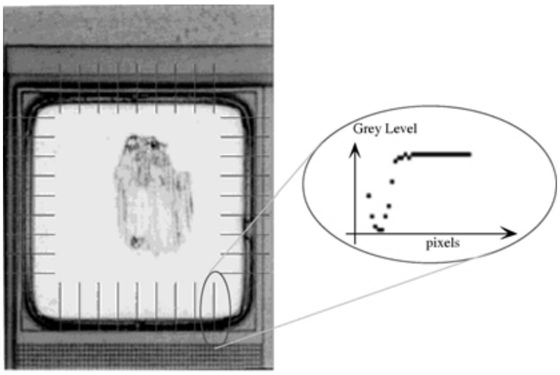


Figure 26. Example of profiles extraction after via centering process.

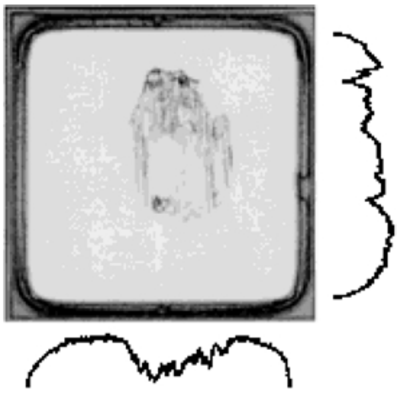


Figure 27. Profiles extraction (for size and localization of the probe mark).

the mark left by the probe during the test. These two parameters are used to estimate the wear and the adjustment of the probe. This estimation is also performed by using some profiles. But those ones are different, they are the horizontal and vertical projections (see Fig. 27).

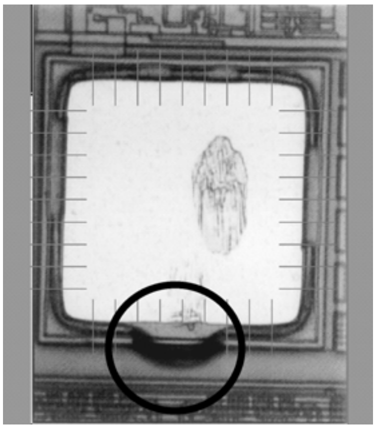


Figure 28. Experimental result showing a fault detection and its localization in the via.

Those profiles are compared with reference ones, memorized in the ZISC-036. All of the memorized profiles are associated with a position and a length. The use of the responses of the horizontal profile and of the vertical profile allows to estimates the size of the mark of the probe. Good results are obtained on the impact locating and surface estimating with 100 neurons. 10 ms are required for this task (see Fig. 28).

4. Conclusion

In this article, we have presented a robust, adaptive, and fast neural based image-processing concept. We have focused our purpose on neural based solutions for industrial real world and real complexity problems using the ZISC-036 neuro-processor, an IBM hardware processor which implements the Restricted Coulomb Energy algorithm (RCE) neural model. We have analyzed computational aspects concerning data precision, learning parameters and learning criteria related to the RCE neural model and it's ZISC-036 based implementation for ROI and KNN operating modes. We have developed, on the one hand, solutions for noise reduction, image enhancement and image coloration to process the old movies automatic restoration: an application concerning movie production industry. On the other hand, developed neural based solutions have been applied for visual probe mark inspection in VLSI production.

In the case of image restoration and coloration it has been shown that the same neural concept could perform different tasks as noise reduction, image enhancement and image coloration which are necessary to restore a degraded movie. Quantitative comparative studies show pertinence of such techniques. On the side of visual probe mark inspection in VLSI production experiments on different kinds of chips and on various probe defects have proven the efficiency of the neural approach to this kind of perception problem. Our prototype outperformed the best solutions offered by competitors by 30%: the best response time per via obtained using other wafer probers was about 600 ms and our neural based system analyzes one via every 400 ms, 300 of which were taken for the mechanical movements. Measures showed that the defect recognition neural module's execution time was negligible compared to the time spent for mechanical movements, as well as for the image acquisition (a ratio of 12 to 1 on any via). This application is presently being inserted on a high throughput production line.

For both of two above-mentioned industrial applications, obtained experimental results show that developed neural networks based concepts are well adapted to the posed problems and present the advantage of robustness. Furthermore, they are easy to implement, and can be improved by the number and choice of memorized examples. Finally, due to the ZISC-036 parallel processing capability, it is possible to use as many ZISC-036 processors as user needs without disturbing the response time.

References

1. L.M. Reyneri, "Weighted radial basis functions for improved pattern recognition and signal processing," in *Neural Processing Letters*, vol. 2, no. 3, pp 2–6, 1995.
2. G. Mercier, K. Madani, and S. Duchesne, "A specific algorithm for a restricted coulomb energy network used as a dynamical process identifier," in *RI_95, Academy Institute of Sciences*, St. Petersburg, Russia, May 1995.
3. Robert David, Erin Williams, Ghislain de Trémiolles, and Pascal Tannhof, "Noise reduction and image enhancement using a hardware implementation of artificial neural networks," VI-DYNN'98-Virtual Intelligence—Dynamic Neural Networks Stockholm-Sweden-June 22–26, 1998.
4. A. Eide, Th. Lindblad, C.S. Lindsey, M. Minerskjöld, G. Sekhviaidze, and G. Székely, "An Implementation of the zero instruction set computer (ZISC-036) on a PC/ISA-bus card," in *1994 WNN/FNN in Washington D.C.*, December 1994.
5. ZISC-036 Data Book, IBM Microelectronics, November 1994.
6. ZISC/ISA ACCELERATOR card for PC, User Manual, IBM France, February 1995.
7. ZISC (Zero Instruction Set Computer), http://www.fr.ibm.com/france/cdlab/zisc.htm, IBM France, January 1998.
8. ZISC ISA and PCI Cards for IBM Compatible PC User's Manual 1.2, http://www.fr.ibm.com/france/cdlab/zisc.htm, IBM France, May 1998.
9. ZISC User's Manual, http://www.fr.ibm.com/france/cdlab/zisc.htm, IBM France, May 1998.
10. G. De Tremiolles, Contribution à l'étude théorique des modèles neuromimétiques et à leur validation expérimentale: mise en œuvre d'applications industrielles ("Contribution to the theoretical study of neuromimetic models and to their experimental validation: use in industrial applications"), Ph.D. Report, University of PARIS XII–Val de Marne, March 1998.
11. Robert David, Erin Williams, Ghislain de Trémiolles, and Pascal Tannhof, "Description and practical uses of IBM ZISC-036," VI-DYNN'98 - Virtual Intelligence—Dynamic Neural Networks Stockholm, Sweden, June 22–26, 1998.
12. I. Rojas, O. Valuenzuela, and A. Prieto, "Statistical analysis of the main parameters in the definition of radial basis function networks," *IWANN'97*, Springer, Lanzarote, Canary Islands, Spain, June 4–6, 1997, pp. 882–891.
13. J.M. Moreno, J. Madrenas, S. san Anselmo, F. Castillo, and J. Cabestany, "Digital hardware implementation of ROI incremental algorithms," in *From natural to artificial neural computation*, LNCS vol. 930, Springer Verlag, 1995, pp. 761–770.
14. S. Lawrence, C. Lee Giles, A. Chung Tsoi, and A. Back, "Face recognition: A convolutional neural network approach," *IEEE Transactions on Neural Networks*, Special Issue on Neural Networks and Pattern Recognition, 1997.
15. R. Gonzalez and R. Woods, *Digital Image Processing*, Addison-Wesley: USA, 1993.
16. G. de Trémiolles, P. Tannhof, B. Plougonven, C. Demarigny, and K. Madani, "Visual probe mark inspection using hardware implementation of artificial neural networks," in *VLSI Production*, Lecture Notes in Computer Science, edited by Springer Verlag, Jun 1997.
17. S.P. Luttrel, "Optimal response functions in a network of discretely firing neurons," in *5th IEE Conference on Artificial Neural Networks*, Cambridge, 1997.

Kurosh Madani was born in 1959. Received his Ph.D. degree in Electrical Engineering and Computer Sciences from University PARIS XI (PARIS-SUD), Orsay, France, in 1990. From 1989 to 1990, he worked as assistant professor at Institut d'Electronique Fondamentale (Institute of Fundamental Electronics) of PARIS XI University, Orsay, France. In 1990, he joined Creteil-Senart Institute of Technology of PARIS XII—Val de Marne University, Lieusaint, France, where he worked from 1990 to 1998 as assistant professor. In 1995, he received the DHDR Doctorate degree (senior research doctorate degree) from University PARIS XII—Val de Marne.

Since 1998 he works as Chair Professor in Electrical Engineering of Senart Institute of Technology of PARIS XII—Val de Marne University. Since 1992 he is head of Intelligence in Instrumentation and Systems Laboratory of PARIS XII—Val de Marne University located at Senart Institute of Technology. He has worked on both digital and analog implementation of processor arrays for image processing by stochastic relaxation, Electro-Optical random numbers generation, and both analog and digital neural networks implementation. His current research interests include large neural network based structures behavior modeling and implementation, hybrid neural based information processing systems and their software and hardware implementations, design and implementation of real-time neuro-controllers and neural based fault detection and diagnostic.

G. de Trémiolles was born in 1970 and received his Ph.D. degree in Electrical Engineering and Computer Sciences from University PARIS XII, Créteil, France, in 1998. He worked as an R&D engineer in the Corbeil-Essonnes Component Development Laboratory of IBM on ZISC® (Zero Instruction Set Computer) neural network development and application when this article was written. His research interests include neural networks, image processing, pattern recognition, real-time control and both digital and analog hardware implementation of neural networks. Since 2000, he works as patent engineer for Intellectual Property Department of IBM.

Pascal Tannhof received a B.S. degree in fundamental physics, the M.S. degree and the Ph.D. degree in computer science from Paris VI University, France. He joined IBM in 1983; he has worked on various topics related to VLSI design. Since 1993, he has worked on the design and development of the IBM ZISC architecture. His research interests include parallel architectures, image analysis, pattern recognition and related applications. He has filed over twenty patents and has written technical papers related to VLSI design and parallel architectures.