New Improved Algorithm for the Training of a Morphological Associative Memory

Ricardo Barrón, Humberto Sossa and Benjamín Cruz
Centro de Investigación en Computación-IPN
Av. Juan de Dios Bátiz esquina con Miguel Othón de Mendizábal
Mexico City, 07738. MEXICO
E-mails: rbarron@cic.ipn.mx, hsossa@cic.ipn.mx, benjamincruz@sagitario.cic.ipn.mx.

(Paper received on September 01, 2006, accepted on September 27, 2006)

Abstract. We describe in this work a new algorithm for the training of an associative memory based on the so-called multi-layered morphological perceptron with maximal support neighborhoods. We describe how to improve the original algorithm by adding new steps at the training phase. We have performed several experiments with real patterns where we show the superiority of the new proposal. We also give formal conditions for correct classification.

1. Introduction

Neural networks are a computational alternative to solve problems where it is difficult to find or there is no an algorithmic solution. Based on the functioning of the human nervous system, lots of researchers have proposed different neural processing models. The study of the internal structure of the neural cells has revealed that all cells have the same simple structure, independently of their size and shape (Fig. 1).

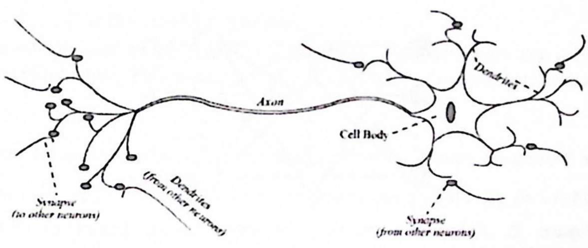


Figure 1. Structure of a biological neuron.

Information from this cell voyages through the signals that neurons send to other neurons through their dendrites. It is believed that the cellular body adds up the received signals; when enough inputs are available a discharge is produced. Initially, this discharge occurs in cellular body, it propagates between the axon until the synapses that sends a new signal to the other neurons.

Special Issue in Neural Networks and Associative Memories
Research in Computing Science 21, 2006, pp. 49-59

One of the drawbacks of this proposal to build the supports is that if two of these supports are too close, the radius of their neighborhoods reduces drastically.

In [7] it is proposed another method to increase the neighborhoods of the supports by means of the kernels' method. According to [7], with this the range of permissible noise is increased. However, this method makes expensive the computational cost and besides it imposes restrictions over the patterns very difficult to get.

In the following section we describe a new way to train this kind of associative memories that will allow, as we will see, to obtain wider support neighborhoods.

3. Proposed training algorithm

If we suppose that a pattern is represented in terms of n describing features, then at each coordinated axis we can compute the variation obtained per feature by ordering all components. By computing the average of variability per pattern it is possible to define a threshold. For practical purposes, this threshold allows to consider if two patterns can be considered to be the same from the point of view of one of their components. This way we avoid having very tiny supports with respect to the coordinated axis. This way the drawback of the algorithm described in last section is surpassed.

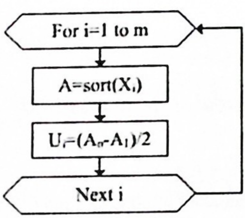


Figure 3. Flowchart to get average variation threshold.

The algorithm to find the average variation threshold by axis is observed in Fig. 3. In this case, X is a matrix of $n \times m$. At j-th column it is j-th pattern, while at i-th line are the n features of the i-th axis of each pattern. In U_i it is the average variation threshold for the i-th axis. The key point for the training of the multi- layered morphological perceptron consists on constructing the supports for each pattern. In general, these supports must be disjoint two by two to avoid that two patterns are assigned to the same output.

In some cases we have patterns that are too close: the neighborhoods of both patterns present occlusions. In this particular case we follow the original algorithm only for the component where occlusion occurs. Evidently if all patterns are too close the proposal is the same as the original algorithm.

With these arrangements we have disjoints supports two by two. This is fulfilled in the proposed algorithm thanks to the following:

Proposition 2. *Let Ω_i and Ω_j two arbitrary supports corresponding to different patterns in* $\mathbf{R}^n$*, built according to average variation threshold, then it holds that* $\Omega_i \cap \Omega_j = \varnothing$.

Proof. Let x^i and x^j the corresponding patterns to neighbors supports Ω_i and Ω_j, respectively, then we have two cases:

a. If it holds that $x_k^i - x_k^j < U_k, \forall_{k=1,\dots,n}$, with U_k is the average variation. If this holds, we apply the original algorithm and by proposition 1 it holds that $\Omega_i = \Omega_j$.

b. If $\exists_{k=1,\dots,n}$ such that $x_k^i - x_k^j > U_k$ would imply that at $k-th$ coordinate, the support does not coincide and on the axis they are disjoint, thus they are disjoint in $\mathbf{R}^n$. ■

One advantage of this enhancement is that neighborhoods allowing more noise to be added to the patterns will be enlarged.

4. Numerical Examples

Example No. 1. Let the following set of key patterns in $\mathbf{R}^n$:

$$\left\{x^1 = \begin{bmatrix} 1 \\ -4 \end{bmatrix}, x^2 = \begin{bmatrix} 1 \\ -5 \end{bmatrix}, x^3 = \begin{bmatrix} -3 \\ 5 \end{bmatrix}, x^4 = \begin{bmatrix} -4 \\ -4 \end{bmatrix}, x^5 = \begin{bmatrix} 4 \\ 2 \end{bmatrix}, x^6 = \begin{bmatrix} 5 \\ 3 \end{bmatrix}\right\} \tag{4}$$

a) Solution obtained by means of algorithm proposed in [7]:

By applying the algorithm proposed in [7], according to equation (2) $\alpha = 0.5$. Fig. 4 shows the neighborhoods obtained when using this value of α.

b) Solution obtained by means of algorithm proposed in this paper (first variant):

When using the algorithm to get the maximal support neighborhoods, by applying the flowchart shown in Fig. 3, we get the threshold value U as:

$$U = \begin{bmatrix} 1.5 \\ 1.6 \end{bmatrix}. \tag{5}$$

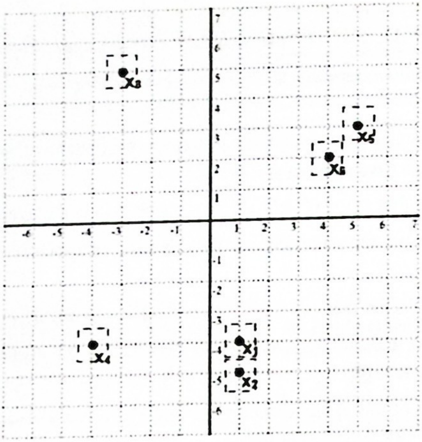


Figure 4. Neighborhoods obtained when algorithm proposed in [7] is used.

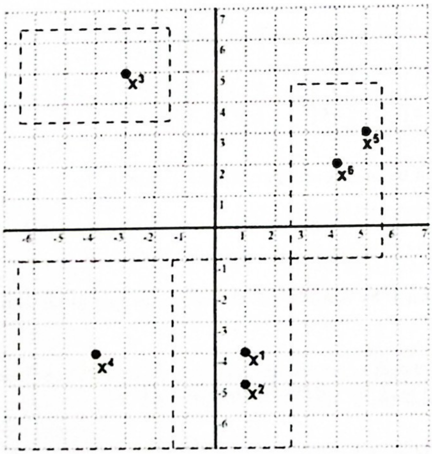


Figure 5. Neighborhoods obtained when using first variant of the proposed algorithm.

The neighborhoods obtained by using this threshold value are shown in Fig. 5. In this case we do not take into account the possible occlusions that can occur. We use this approach in order to compare with the results with the new approach. The differences between the proposal and original one can be immediately appreciated. In this second case the range on noise for each pattern, as can be appreciated, is bigger.

Although, in some cases when we want recognize original patterns (patterns that are not affected by noise) we get mistakes during classification because two or more patterns are assigned at same support neighborhood.

c) Solution obtained by means of algorithm proposed in this paper (second variant):

We apply the same procedure used by the first variant, but in this case the neighborhoods present occlusions, we apply the original algorithm. The threshold is the same given by equation (5). We use the value of $\alpha = 0.5$.

The neighborhoods obtained are shown in Fig. 6. We can observe that the neighborhoods do not present occlusions and the supports for noise are bigger.

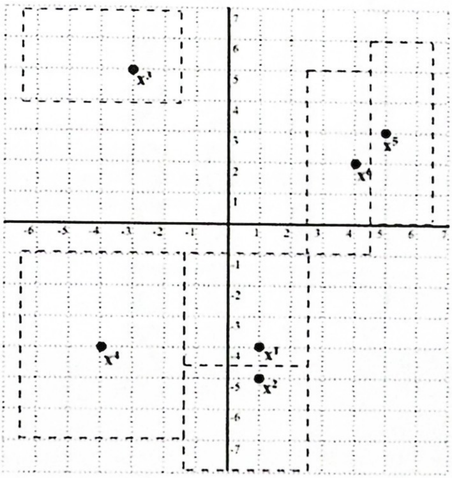
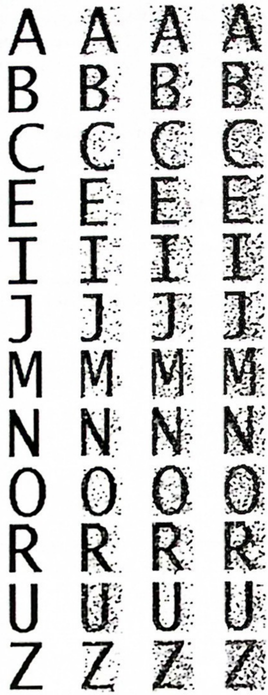


Figure 6. Neighborhoods obtained when using second variant of the proposed algorithm.

5. Experiments

To verify the efficiency of the proposed two variants in this work with respect to the classical algorithm described in [7] used to build neighborhood supports, we have done the following simple test. It consisted on recognizing the images shown in Fig. 7. These images are binary of 37×31 elements. These patterns were perturbed with noise from 5% to 15% in steps of 5%.



(a) 0% (b) 5% (c) 10% (d) 15%

Figure 7. Images used to test the proposal. (a) Original images. (b), (c),(d) Altered images with 5%, 10% and 15% of noise, respectively.

The input key patterns where formed by describing each image by means of known Hu invariants ([10]. [11]). Fig. 8 shows graphically the results obtained when using the algorithm proposed in [7]. and the algorithm proposed in this paper to the images shown in Fig. 7(a). In this case, as can be appreciated in all cases, the original proposal presents the best performance. The first proposed variant of this paper do not provide does not provide good results.

Figures 9, 10 and 11 show the recognition results obtained when applying the original algorithm and the two variants described in this paper to the distorted versions of the patterns. From these figures, we can immediately appreciate that in practically any of the cases the original algorithm fails to correctly classify the desired patterns. The new show the best performance.

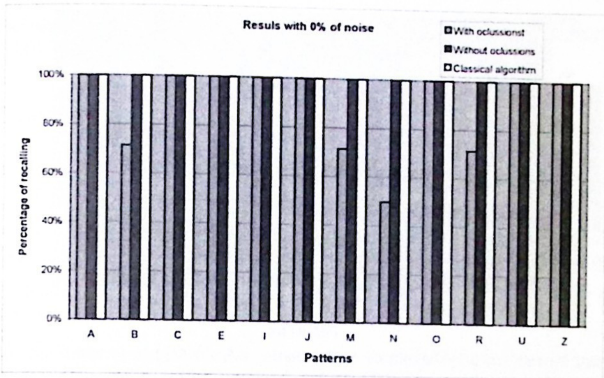


Figure 8. Results when recalling the images of Figure 7(a).

From all of these graphics we can observe that when adding noise to the patterns, even small quantities of it, the original algorithm proposed in [7] fails to recall practically all patterns, while the proposed two variants, although with modest percentage, they arrive to recall several of the desired patterns.

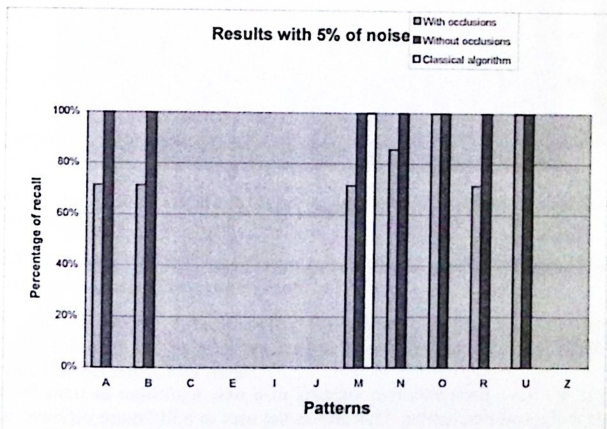


Figure 9. Results when recalling the images of Figure 7(b).

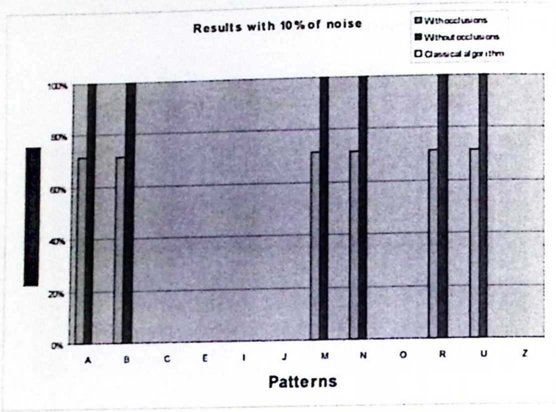


Figure 10. Results when recalling the images of Figure 7(c).

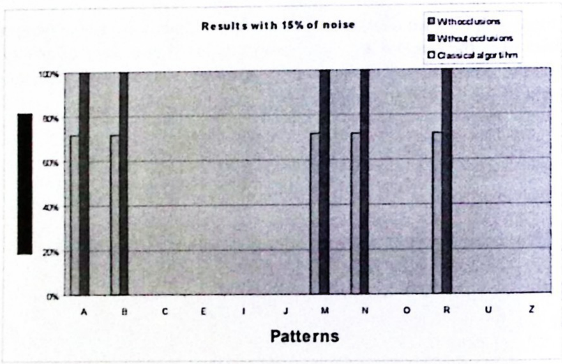


Figure 11. Results when recalling the images of Figure 7(d).

6. Conclusion

In this paper we have presented two variants of a new algorithm to train the multi-layered morphological perceptron. This allows the user to build more efficient support neighborhoods, since the point of view of pattern recall, from the set of key pattern patterns of the training set of an associative memory in both its auto-associative or hetero-associative way of operation. Experiments have been carried out that show the superiority of the proposed variants proposal with respect to the classical algorithm. Formal conditions for correct classification have also given.

Acknowledgements. This work was economically supported by CGPI-IPN under grants 20050156, 20060517 and CONACYT by means of grant 46805.

References

[1] C. Bishop (1995). Neural Networks for Pattern Recognition. Oxford University Press, Oxford, England.

[2] F. Rossenblatt (1958). The perceptron: a probabilistic model for information storage and organization in the brain, Psychological Review 65:386-408.

[3] F. Rossenblatt (1962). Principles of Neurodinamics: Perceptrons and the theory of brain mechanisms, Spartian Books, Washington, D.C.

[4] G. X. Ritter et al (1996). An introduction to morphological neural networks. In Proceedings of the 13th International Conference on Pattern Recognition, pp. 709-717. Vienna, Austria.

[5] G. X. Ritter et al. (1998). Morphological associative memories. IEEE Transactions on Neural Networks, 9:281-293.

[6] G. X. Ritter et al (1997). Morphological perceptrons. ISAS'97, Intelligent Systems and Semiotics, Gaithersburg, Maryland.

[7] G. X. Ritter (2004). A new auto-associative memory based on lattice algebra. Proc. 9th CIARP'04, La Havana, Cuba 148-155.

[8] M. Kishan, K. M. Chilukuri, and R. Sanjay (1997). Elements Of Artificial Neural Network, The MIT Press, Cambridge, Massachusetts, London, England.

[9] L. F. C. Pessoa et al. (1996) Morphological/Rank neural networks and their adaptative optimal image processing, Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, 6:3399-3402.

[10] M. K. Hu (1961). Pattern recognition by moment invariants, Proceedings of the IRE, 49:1428.

[11] M. K. Hu (1962). Visual pattern recognition by moment invariants, IRE Transactions on Information Theory, 179-187.

[12] P. Sussner (1998). Morphological Perceptron Learning. In Proceedings of the 1998 International Symposium on Intelligent Systems and Semiotics, pp. 477-482, Gaithersburg, Maryland.