Neural Process Lett (2008) 28:189–207
DOI 10.1007/s11063-008-9089-6

A New Associative Model with Dynamical Synapses

Roberto A. Vázquez Espinoza de los Monteros ·
Juan Humberto Sossa Azuela

Published online: 2 November 2008

Abstract The brain is not a huge fixed neural network, but a dynamic, changing neural network that continuously adapts to meet the demands of communication and computational needs. In classical neural networks approaches, particularly associative memory models, synapses are only adjusted during the training phase. After this phase, synapses are no longer adjusted. In this paper we describe a new dynamical model where synapses of the associative memory could be adjusted even after the training phase as a response to an input stimulus. We provide some propositions that guarantee perfect and robust recall of the fundamental set of associations. In addition, we describe the behavior of the proposed associative model under noisy versions of the patterns. At last, we present some experiments aimed to show the accuracy of the proposed model.

Keywords Associative memories · Dynamical synapses · Pattern recognition

1 Introduction

Humans possess several capabilities such as: learning, recognition, memorization and recalling that make them unique. In the last 60 years, scientists and researchers of different communities have tried to implement these capabilities into a computer. Along these years, several approaches have emerged, one common example are neural networks [1–3]. Since the rebirth of neural networks, several models inspired in neurobiological process have been proposed. Such models are often dedicated and incorporate some existing clustering or classification algorithms. Among these models, perhaps the most popular is the feed-forward multilayer perceptron trained with the back-propagation algorithm [4].

R. A. Vázquez Espinoza de los Monteros (✉) · J. H. Sossa Azuela
Centro de Investigación en Computación – IPN, Av. Juan de Dios Bátiz s/n, Esquina con Miguel Othón de Mendizábal, Colonia Nueva Industrial Vallejo, C. P. 07700, México DF, Mexico
e-mail: ravem@ipn.mx

J. H. Sossa Azuela
e-mail: hsossa@cic.ipn.mx

Other classical neural models are associative memories. Several associative models have emerged in the last years. For example, Steinbuch [5] suggested an associative memory based on a switching matrix. Anderson [6] presented a simple neural network generating an interactive memory. Kohonen [7] replaces the lernmatrix of Steinbuch by a correlation matrix, and Nakano [8] replaces the lernmatrix of Steinbuch by an autocorrelation matrix. The output in these models is produced in a single feedfoward computation where correlation or Hebbian learning is used to synthesize the synaptic weight matrix. Another type of associative memory was proposed by Hopfield [9]. This model is different from other previous models in that it computes its output recursively in time until the system becomes stable. Hopfield's model works only with binary and bi-polar patterns. Although this model has several constraints that limit its applicability in complex problems, perhaps this is one of the most popular associative memories which through the years has been object of research for several authors. For example, in [10] the authors explore how the topology affects performance in Hopfield associative memory using regular lattice, random, small-world and scale free structures. In [11] the author extends the original model to a complex-valued multi-state in order to storing and recalling gray-level patterns.

Other schemes of associative memories based on recurrent neural networks, see for example [12,13], and other techniques [14–18] have been proposed.

Recently, morphological, median and fuzzy associative memories were proposed; refer for example to [19–23]. Although the performance obtained with these models is pretty acceptable, they also have to satisfy several constrains which limit their applicability in complex problems. In general, these constraints are related to the type and quantity of noise supported by these models.

However, the brain is not a huge fixed neural network as had been previously thought, but a dynamic, changing neural network that adapts continuously to meet the demands of communication and computational needs. In most of these associative models, synapses are only adjusted during the training phase. After this phase, synapses are no longer adjusted.

Neurobiologists have identified key mechanisms by which the intricate "protein machines" that govern the strength of connections among neurons build and remodel themselves to adjust those connections. Such remodeling of the connections, called synapses, is central to establish brain pathways during learning and memory [21]. Nowadays, scientists focus its research on new mechanisms based on current study of biological neural networks. Modern brain theory no longer uses the binary model of the neuron, but instead it uses continuous-time model that either represent the variation in average firing rate of the neuron or actually capture the time course of membrane potentials [25], see for example [26]. In this direction, several works have been proposed, see for example [27]; most of these works are focused to emulate and understand some process in human brain during learning and recognition and not their capabilities to solve complex problems. In [28] the authors examined the role of dynamic synapses in the stochastic Hopfield-like network behavior. In [29] the author analyses the effect of synaptic depression on the stability of patterns stored in neural networks with low activity level and later in [30] it is proposed an associative neural network with chaotic neuron model and synaptic depression.

The main goal of this paper is to apply some process of human brain during learning and recognition to develop new associative models with fewer restrictions than classic models already mention. In this paper we describe a new dynamical model where synapses of the associative memory could be adjusted even after the training phase as a response to an input stimulus. This dynamic approach is inspired in some ideas described in [24,31,32]. We define a set of synapses that determine the behavior of the model. This set of synapses will be modified in response to an input stimulus based on an up-date rule. We provide a set of propositions that guarantee the recall of the fundamental set of associations. In addition, we describe the behaviour of the associative model proposed under noisy versions of the patterns. At last, we present some experiments aimed to show the accuracy of the proposed model.

2 The Associative Model

Associative memories have been deeply researched in the last years. They can be seen as a particular type of neural network specially designed to recall output patterns in terms of input patterns that can appear distorted by some kind of noise. Some of these associative memories have several constraints that limit their applicability for solving problems from real life. A common application of an associative memory is as a filter, refer for example to [19–21]. In this case, the associative memory is fed with a pattern, probably altered by noise; at the output the original image (without noise) should be obtained. However, to achieve the best performance, input patterns must satisfy some conditions.

The model proposed in [19] is based on **max** and **min** operators; although authors proof its robustness with noisy patterns, the model has several constraints. For example input patterns can only be contaminated by additive or subtractive noise, but not both. In [22], the authors proposed a new model based on **median** operator. In this model, patterns can only be contaminated by mixed noise (both kinds of noises). These constraints limit their applicability in complex problems.

The proposed model is not an iterative model as Hopfield's model [9]. Our model emerges as an improvement of the model proposed in [22], which is not an iterative model, and the results presented in [33]. This model is one-shot trained and also is capable to storing and recalling real-valued pattern and true-color patterns. The main difference of this model against classical models is that while in classical models the synapses of the associative memories are adjusted only during training phase, in our model the synapses are adjusted also after training phase as a respond to an input stimulus.

Definition 1 Let $\mathbf{x} \in \mathbf{R}^n$ and $\mathbf{y} \in \mathbf{R}^m$ an input and output pattern, respectively. An association between input pattern $\mathbf{x}$ and output pattern $\mathbf{y}$ is denoted as $\left\{\mathbf{x}^k, \mathbf{y}^k \,|k = 1, \ldots, p\right\}$, where k is the corresponding association and p the number of associations.

Definition 2 Let operator $A(x, y) = y - x$.

Definition 3 Associative memory $\mathbf{W}$ is represented by a matrix whose components w_{ij} can be seen as the synapses of the neural network.

$$\mathbf{W} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{mn} \end{bmatrix}_{m \times n}$$

where $w_{ij} = A\left(x_i, y_j\right) = y_i - x_j$

If $\mathbf{x}^k = \mathbf{y}^k \forall k = 1, \ldots, p$ then $\mathbf{W}$ is auto-associative, otherwise it is hetero-associative. A distorted version of a pattern $\mathbf{x}$ to be recalled will be denoted as $\tilde{\mathbf{x}}$. If an associative memory $\mathbf{W}$ is fed with a distorted version of $\mathbf{x}^k$ and the output obtained is exactly $\mathbf{y}^k$, we say that recalling is robust.

2.1 Building the Associative Memory

As we have already mentioned, the brain is a dynamic, changing neural network that adapts continuously to meet the demands of communication and computational needs [34]. This fact suggests that some connections of the brain could change in response to some input stimuli.

Humans, in general, do not have problems in recognizing patterns even if these are altered by noise. Several parts of the brain interact together in the process of learning and recalling a pattern. For example, when we read a word the information enters the eye and the word is transformed into electrical impulses. Then electrical signals are passed through the brain to the *visual cortex*. After that, specific information about the patterns passes on the other areas of the *cortex*. From here information passes through the *arcuate fasiculus*, a path that connects a large network of interacting brain areas; paths of this pathway connect language areas with other areas involving in cognition, association and meaning, for the details refer to [35,36].

Based upon the above example we have defined in our model p interacting areas, one per association we would like the memory to learn. Each interacting area will allocate useful information for recalling phase. Also we have integrated the capability to adjust synapses in response to an input stimulus.

As we could appreciate from the previous example, before an input pattern is learned or processed by the brain, it is hypothesized that it is transformed and codified by the brain. According to this idea, input patterns used to train the associative model are firstly coded by using the following procedure recently introduced in [37]:

Procedure 1 Transform the fundamental set of associations into coded pattern versions and de-coding patterns:

Input: FS Fundamental set of associations:

{1. Make $d = const$ and make $\left(\bar{\mathbf{x}}^1, \bar{\mathbf{y}}^1\right) = \left(\mathbf{x}^1, \mathbf{y}^1\right)$

2. For the remaining couples do {

For $k = 2$ to p{

For $i = 1$ to n{

$$\bar{x}_i^k = \bar{x}_i^{k-1} + d; \quad \hat{x}_i^k = \bar{x}_i^k - x_i^k; \bar{y}_i^k = \bar{y}_i^{k-1} + d; \quad \hat{y}_i^k = \bar{y}_i^k - y_i^k$$

}}} Output: Set of coded and de-coding patterns.

This procedure allows computing a coded version from input and output patterns. These patterns are called *coded patterns* and are denoted by $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ respectively. In addition, this procedure computes other kind of patterns useful to decode the *coded patterns*. These patterns are called *decoding patterns* and are denoted by $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ respectively. As was shown in [37], *coded patterns* are equidistant among them in a ratio of d; this radio also determines the noise supported by our model. This fact allows us to use any association to train the associative memory $\mathbf{W}$. The reader can easily corroborate that any association can be used to compute the synapses of $\mathbf{W}$ without modifying the results. Ratio d can be any value belonging to real numbers domain; however, a better accuracy is obtained if we set d to the minimum absolute difference among simplified input patterns.

Coded and de-coding patterns are allocated in different interacting areas of the model. In other words, each interacting area k will allocate the coded version and de-coding pattern of each k association. In addition, a simplified version of $\mathbf{x}^k$ denoted by s_k is allocated in the corresponding k interacting area. Simplified versions are obtained as:

$$s_k = s\left(\mathbf{x}^k\right) = \mathbf{mid}\ \mathbf{x}^k \tag{1}$$

where **mid** operator is defined as $\mathbf{mid}\ \mathbf{x} = x_{(n+1)/2}$.
mid operator has some interesting properties and advantages against **median**, **max** and **min** operators. **mid** operator does not require of any ordering algorithm to find the median, maximum or minimum values. Some of these properties are described in the Annex.

Once computed the *codified patterns*, the *de-codifying patterns* and s_k we can build the associative memory as in Definition 3. In short, building of the associative memory can be performed in three stages as:

1. Transform the fundamental set of association into coded and de-coding patterns by means of previously described Procedure 1. Set d as the minimum absolute distance among whole set of associations.
2. Compute simplified versions of input patterns by using Eq. 1.
3. Build $\mathbf{W}$ in terms of codified patterns by using definition 3.

2.2 Modifying Synapses of the Associative Model

When the brain is stimulated by an input pattern, some regions of the brain (interacting areas) are stimulated and synapses belonging to those regions are modified. The next step in this approach is to determine which interacting area of this model is the most excited area by the input pattern. The most excited interacting area is called *active region* and could be estimated in different manners. In order to determine the *active region* we will use the simplified versions s_k allocated in each k interacting area and the simplified version of the input pattern. The *active region* could be estimated as follows:

$$ar = r\left(\mathbf{x}\right) = \arg\left(\min_{i=1}^{p} \left|s\left(\mathbf{x}\right) - s_i\right|\right) \tag{2}$$

As we had already mentioned, synapses could change in response to an input stimulus; but which synapses should be modified? For example, a head injury might cause a brain lesion killing hundred of neurons; this entails some synapses to reconnect with others neurons. This reconnection or modification of the synapses might cause that information allocated on brain will be preserved or will be lost, the reader could find more details concerning to this topic in [38,39].

This fact suggests there are synapses that can be drastically modified and they do not alter the behavior of the associative memory. In the contrary, there are synapses that can only be slightly modified to do not alter the behavior of the associative memory; we call this set of synapses *the kernel* of the associative memory and it is denoted by $\mathbf{K_W}$. In the model we find two types of synapses: synapses that can be modified and do not alter the behavior of the associative memory; and synapses belonging to the kernel of the associative memory. These last synapses play an important role in recalling patterns altered by some kind of noise.

Definition 4 Let $\mathbf{K_W} \in \mathbf{R}^n$ the kernel of an associative memory $\mathbf{W}$. A component of vector $\mathbf{K_W}$ is defined as: $kw_i = \mathbf{mid}\left(w_{ij}\right), j = 1, \ldots, m$

According to the original idea of our proposal, synapses that belong to $\mathbf{K_W}$ are modified as a response to an input stimulus. Input patterns stimulate some *active regions*, interact with these regions and then, according to those interactions, the corresponding synapses are modified. Synapses belonging to $\mathbf{K_W}$ are modified according to the stimulus generated by the input pattern. The related adjusting factor is denoted by Δw and can be computed as:

$$\Delta w = \Delta(\mathbf{x}) = s\left(\bar{\mathbf{x}}^{ar}\right) - s\left(\widehat{\mathbf{x}}\right) \tag{3}$$

where ar is the index of the *active region* and $\widehat{\mathbf{x}} = \mathbf{x} + \hat{\mathbf{x}}^{ar}$.

Finally, synapses belonging to $\mathbf{K_W}$ are modified as:

$$\mathbf{K_W} = \mathbf{K_W} \oplus (\Delta w - \Delta w_{old}) \tag{4}$$

where operator $\oplus$ is defined as $\mathbf{x} \oplus c = x_i + c \ \forall i = 1, \ldots, m$ and c is a scalar.

As you can appreciate, modification of $\mathbf{K_W}$ in Eq. 4 depends on the previous value of Δw denoted by Δw_{old} obtained with the previous input pattern. Once trained the **AM**, when it is used by first time, the value of Δw_{old} is set to zero.

2.3 Recalling a Pattern Using the Proposed Model

Once synapses of the associative memory have been modified in response to an input pattern, every component of vector $\bar{\mathbf{y}}$ can be recalled by using its corresponding input vector $\bar{\mathbf{x}}$ as:

$$\bar{y}_i = \mathbf{mid}\left(w_{ij} + \bar{x}_j\right), \quad j = 1, \ldots, n \tag{5}$$

In short, pattern $\bar{\mathbf{y}}$ can be recalled by using its corresponding key vector $\bar{\mathbf{x}}$ or $\tilde{\mathbf{x}}$ in six stages as follows:

1. Obtain index of the active region ar by using Eq. 2.
2. Transform $\mathbf{x}^k$ using de-codifying pattern $\hat{\mathbf{x}}^{ar}$ by applying the following transformation: $\widehat{\mathbf{x}}^k = \mathbf{x}^k + \hat{\mathbf{x}}^{ar}$.
3. Compute adjusting factor $\Delta w = \Delta\left(\widehat{\mathbf{x}}\right)$ by means of Eq. 3.
4. Modify synapses of associative memory $\mathbf{W}$ that belong to $\mathbf{K_W}$ by using Eq. 4.
5. Recall pattern $\widehat{\mathbf{y}}^k$ by using Eq. 5.
6. Obtain $\mathbf{y}^k$ by transforming $\widehat{\mathbf{y}}^k$ using de-codifying pattern $\hat{\mathbf{y}}^{ar}$ by applying transformation: $\mathbf{y}^k = \widehat{\mathbf{y}}^k - \hat{\mathbf{y}}^{ar}$.

Perfect and robust recall is guaranteed if the problem satisfies any of the next propositions:

Proposition 1 *Let* $\left\{\left(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k\right) | k = 1, \ldots, p\right\}$, $\bar{\mathbf{x}}^k \in \mathbf{R}^n$, $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel. Every component of vector* $\mathbf{K_W}$ *is given as* $kw_i = \bar{y}_i - \mathbf{mid}(\bar{\mathbf{x}})$.

Proof See the Annex.

Proposition 2 *Let* $\left\{\left(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k\right) | k = 1, \ldots, p\right\}$, $\bar{\mathbf{x}}^k \in \mathbf{R}^n$, $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel. Every component of vector* $\bar{\mathbf{y}}$ *can be perfectly recalled by using its corresponding key vector* $\bar{\mathbf{x}}$ *if* $\bar{y}_i = kw_i + \mathbf{mid}(\bar{\mathbf{x}})$.

Proof See the Annex.

Proposition 3 *Let* $\{(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k) \mid k = 1, \ldots, p\}$, $\bar{\mathbf{x}}^k \in \mathbf{R}^n$, $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel,* Δw *an updated value of* $\mathbf{K_W}$ *and* $\tilde{\mathbf{x}}$ *a distorted version of* $\bar{\mathbf{x}}$. *Every component of vector* $\bar{\mathbf{y}}$ *can be perfectly recalled by using the distorted version of* $\bar{\mathbf{x}}$ *if* $kw_i = kw_i + \Delta w$ *and* $\Delta w = \mathbf{mid}(\bar{\mathbf{x}}) - \mathbf{mid}(\tilde{\mathbf{x}})$.

Proof See the Annex.

Proposition 4 *Let* $\{(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k) \mid k = 1, \ldots, p\}$, $\bar{\mathbf{x}}^k \in \mathbf{R}^n$, $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel,* Δw *an updated value of* $\mathbf{K_W}$ *and* $\tilde{\mathbf{x}}$ *a distorted version of* $\bar{\mathbf{x}}$. *Every component of vector* $\bar{\mathbf{y}}$ *can be perfectly recalled by using the distorted version of* $\bar{\mathbf{x}}$ *if* $\mathbf{mid}(\tilde{\mathbf{x}}) = \mathbf{mid}(\bar{\mathbf{x}}) - \Delta w$.

Proof See the Annex.

Proposition 5 *Let* $\{(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k) \mid k = 1, \ldots, p\}$, $\bar{\mathbf{x}}^k \in \mathbf{R}^n$, $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel,* Δw *an updated value of* $\mathbf{K_W}$ *and* $\tilde{\mathbf{x}}$ *a distorted version of* $\bar{\mathbf{x}}$. *Every component of vector* $\bar{\mathbf{y}}$ *can be perfectly recalled by using the distorted version of* $\bar{\mathbf{x}}$ *if* $|\Delta w| < \frac{d}{2}$.

Proof See the Annex.

Some important to remark is that if d is bigger than the maximum absolute difference among simplified input patterns, then the active region will not be determined accurately. For example, we have the simplified versions of three input patterns $s(\mathbf{x}^1) = 0.2$, $s(\mathbf{x}^2) = 0.5$ and $s(\mathbf{x}^3) = 1.2$. If d set to the maximum absolute difference among simplified input patterns, $d = 1.0$; this implies that after applied procedure 1 each simplified version allocated in each interacting area will be $s_1 = 0.2$, $s_2 = 1.2$ and $s_3 = 2.2$. Suppose that we have a simplified noisy version of $\mathbf{x}^2$, $s(\tilde{\mathbf{x}}^2) = 0.6$ which implies that active region will be the interacting area 2; however, if we applied Eq. 2, we obtain that active region is interacting area 1. Now suppose that we have a simplified noisy version of $\mathbf{x}^3$, $s(\tilde{\mathbf{x}}^3) = 1.3$ which implies that active region will be the interacting area 3; however, by applying Eq. 2, we obtain that active region is interacting area 2. Although the amount of noise added to the patters was small (0.1), in both case, the active region was not computed correctly because d does not satisfied proposition 5.

Finally, we can say that when d satisfied proposition 5, the noise supported by the associative model will be low if d is small and the noise supported by the model will be high if d is big.

3 Numerical Examples

Suppose we want to first memorize and then recall the following general fundamental set using the proposed model:

$$\mathbf{x}^1 = \begin{pmatrix} 0.1 \\ 0.0 \\ 0.2 \end{pmatrix}, \mathbf{y}^1 = \begin{pmatrix} 1.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}; \mathbf{x}^2 = \begin{pmatrix} 0.6 \\ 0.4 \\ 0.3 \end{pmatrix}, \mathbf{y}^2 = \begin{pmatrix} 0.0 \\ 1.0 \\ 0.0 \\ 0.0 \end{pmatrix}; \mathbf{x}^3 = \begin{pmatrix} 0.8 \\ 0.7 \\ 0.9 \end{pmatrix},$$

$$\mathbf{y}^3 = \begin{pmatrix} 0.0 \\ 0.0 \\ 1.0 \\ 0.0 \end{pmatrix}; \mathbf{x}^4 = \begin{pmatrix} 1.1 \\ 1.2 \\ 1.4 \end{pmatrix} \mathbf{y}^4 = \begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 1.0 \end{pmatrix}$$

Building the associative memory. By applying procedure 1 and computing d as described in Sect. 2 we have that $d = 0.4$ and

$$\bar{\mathbf{x}}^1 = \begin{pmatrix} 0.1 \\ 0.0 \\ 0.2 \end{pmatrix}, \hat{\mathbf{x}}^1 = \begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}, \bar{\mathbf{y}}^1 = \begin{pmatrix} 1.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}, \hat{\mathbf{y}}^1 = \begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}; \bar{\mathbf{x}}^2 = \begin{pmatrix} 0.5 \\ 0.4 \\ 0.6 \end{pmatrix},$$

$$\hat{\mathbf{x}}^2 = \begin{pmatrix} -0.1 \\ 0.0 \\ 0.3 \end{pmatrix}, \bar{\mathbf{y}}^2 = \begin{pmatrix} 1.4 \\ 0.4 \\ 0.4 \\ 0.4 \end{pmatrix}, \hat{\mathbf{y}}^2 = \begin{pmatrix} 1.4 \\ -0.6 \\ 0.4 \\ 0.4 \end{pmatrix};$$

$$\bar{\mathbf{x}}^3 = \begin{pmatrix} 0.9 \\ 0.8 \\ 1.0 \end{pmatrix}, \hat{\mathbf{x}}^3 = \begin{pmatrix} 0.1 \\ 0.1 \\ 0.1 \end{pmatrix}, \bar{\mathbf{y}}^3 = \begin{pmatrix} 1.8 \\ 0.8 \\ 0.8 \\ 0.8 \end{pmatrix}, \hat{\mathbf{y}}^3 = \begin{pmatrix} 1.8 \\ 0.8 \\ -0.2 \\ 0.8 \end{pmatrix}; \bar{\mathbf{x}}^3 = \begin{pmatrix} 1.3 \\ 1.2 \\ 1.4 \end{pmatrix},$$

$$\hat{\mathbf{x}}^3 = \begin{pmatrix} 0.2 \\ 0.0 \\ 0.0 \end{pmatrix}, \bar{\mathbf{y}}^4 = \begin{pmatrix} 2.2 \\ 1.2 \\ 1.2 \\ 1.2 \end{pmatrix}, \hat{\mathbf{y}}^3 = \begin{pmatrix} 2.2 \\ 1.2 \\ 1.2 \\ 0.2 \end{pmatrix}.$$

Once obtained the coded patterns, the de-coding patterns associative memory $\mathbf{W}$ is build using any couple of $\left(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k\right)$. For example, by using Definition 3 and $\left(\bar{\mathbf{x}}^2, \bar{\mathbf{y}}^2\right)$ we have that

$$\mathbf{W} = \begin{bmatrix} 0.9 & 1.0 & 0.8 \\ -0.1 & 0.0 & -0.2 \\ -0.1 & 0.0 & -0.2 \\ -0.1 & 0.0 & -0.2 \end{bmatrix}.$$

At last, simplified pattern given by Eq. 1 is $s_1 = 0.0$, $s_2 = 0.4$, $s_3 = 0.7$ and $s_4 = 1.2$.

Example 1 Suppose we want to recall pattern: $\tilde{\mathbf{x}} = \begin{pmatrix} 0.4 & 0.5 & 0.7 \end{pmatrix}^T$.

1. By using Eq. 2, active region is:

$$\begin{aligned} ar &= \arg\left(\min_{i=1}^{p} |s(\tilde{\mathbf{x}}) - s_i|\right) = \arg(\min(|0.5-0.0|, |0.5-0.4|, |0.5-0.7|, |0.5-1.2|)) \\ &= \arg(\min(0.5, 0.1, 0.2, 0.7)) = 2. \end{aligned}$$

2. Transform $\tilde{\mathbf{x}}$ using $\hat{\mathbf{x}}^{ar}$

$$\widehat{\mathbf{x}} = \tilde{\mathbf{x}} + \hat{\mathbf{x}}^2 = \begin{pmatrix} 0.4 & 0.5 & 0.7 \end{pmatrix}^T + \begin{pmatrix} -0.1 & 0.0 & 0.3 \end{pmatrix}^T = \begin{pmatrix} 0.3 & 0.5 & 1.0 \end{pmatrix}^T.$$

3. By using Eq. 3, adjusting factor is equal to:

$$\Delta w = s\left(\bar{\mathbf{x}}^2\right) - s(\tilde{\mathbf{x}}) = 0.4 - 0.5 = -0.1.$$

4. By using Eq. 4, the new values for synapses belonging to the kernel are

$$\mathbf{K_W} = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}^T \oplus (-0.1 - 0.0) = \begin{bmatrix} 0.9 & -0.1 & -0.1 & -0.1 \end{bmatrix}^T$$

Notice how Δw_{old} in this case is equal to zero. Once adjusted the kernel, we have that

$$\mathbf{W} = \begin{bmatrix} 0.9 & 0.9 & 0.8 \\ -0.1 & -0.1 & -0.2 \\ -0.1 & -0.1 & -0.2 \\ -0.1 & -0.1 & -0.2 \end{bmatrix}.$$

5. By using Eq. 5, operate the associative memory as

$$\widehat{\mathbf{y}} = \mathbf{mid}\left(w_{ij} + \widehat{x}_j\right) = \begin{bmatrix} \mathbf{mid}\left(0.9+0.3 \quad 0.9+0.5 \quad 0.8+1.0\right) \\ \mathbf{mid}\left(-0.1+0.3 \quad -0.1+0.5 \quad -0.2+1.0\right) \\ \mathbf{mid}\left(-0.1+0.3 \quad -0.1+0.5 \quad -0.2+1.0\right) \\ \mathbf{mid}\left(-0.1+0.3 \quad -0.1+0.5 \quad -0.2+1.0\right) \end{bmatrix} = \begin{bmatrix} 1.4 \\ 0.4 \\ 0.4 \\ 0.4 \end{bmatrix}$$

Transform $\widehat{\mathbf{y}}$ by using $\hat{\mathbf{y}}^{ar}$. Finally,

$$\begin{aligned} \mathbf{y} &= \widehat{\mathbf{y}} - \hat{\mathbf{y}}^2 = \left(1.4 \quad 0.4 \quad 0.4 \quad 0.4\right)^T - \left(1.4 \quad -0.6 \quad 0.4 \quad 0.4\right)^T \\ &= \left(0.0 \quad 1.0 \quad 0.0 \quad 0.0\right)^T \end{aligned}$$

Example 2 Now suppose we want to recall the distorted pattern: $\tilde{\mathbf{x}} = \left(0.2 \quad 0.3 \quad 0.8\right)^T$.

1. By using Eq. 2, active region is:

$$\begin{aligned} ar &= \arg\left(\min_{i=1}^{p} |s(\tilde{\mathbf{x}}) - s_i|\right) = \arg\left(\min\left(|0.3-0.0|, |0.3-0.4|, |0.3-0.7|, |0.3-1.2|\right)\right) \\ &= \arg\left(\min\left(0.3, 0.1, 0.4, 0.9\right)\right) = 2. \end{aligned}$$

2. Transform $\tilde{\mathbf{x}}$ using $\hat{\mathbf{x}}^{ar}$

$$\widehat{\mathbf{x}} = \tilde{\mathbf{x}} + \hat{\mathbf{x}}^2 = \left(0.2 \quad 0.3 \quad 0.8\right)^T + \left(-0.1 \quad 0.0 \quad 0.3\right)^T = \left(0.1 \quad 0.3 \quad 1.1\right)^T$$

3. By using Eq. 3, adjust factor is equal to:

$$\Delta w = s\left(\bar{\mathbf{x}}^2\right) - s\left(\tilde{\mathbf{x}}\right) = 0.4 - 0.3 = 0.1.$$

4. By using Eq. 4, the new values for synapses belonging to the kernel is

$$\mathbf{K_W} = \left[0.9 \quad -0.1 \quad -0.1 \quad -0.1\right]^T \oplus (0.1 - (-0.1)) = \left[1.1 \quad 0.1 \quad 0.1 \quad 0.1\right]^T$$

Notice how Δw_{old} take the value of Δw previously calculated in example 1. Then $\mathbf{W}$ is

$$\mathbf{W} = \begin{bmatrix} 0.9 & 1.1 & 0.8 \\ -0.1 & 0.1 & -0.2 \\ -0.1 & 0.1 & -0.2 \\ -0.1 & 0.1 & -0.2 \end{bmatrix}.$$

Notice how the values of the **AM** again have changed.

5. By using Eq. 5, operate the associative memory as

$$\widehat{\mathbf{y}} = \mathbf{mid}\left(w_{ij} + \widehat{x}_j\right) = \begin{bmatrix} \mathbf{mid}\left(0.9+0.1 \quad 1.1+0.3 \quad 0.8+1.1\right) \\ \mathbf{mid}\left(-0.1+0.1 \quad 0.1+0.3 \quad -0.2+1.1\right) \\ \mathbf{mid}\left(-0.1+0.1 \quad 0.1+0.3 \quad -0.2+1.1\right) \\ \mathbf{mid}\left(-0.1+0.1 \quad 0.1+0.3 \quad -0.2+1.1\right) \end{bmatrix} = \begin{bmatrix} 1.4 \\ 0.4 \\ 0.4 \\ 0.4 \end{bmatrix}$$

Fig. 1 Set of images A composed by 40 different images of flowers and animals

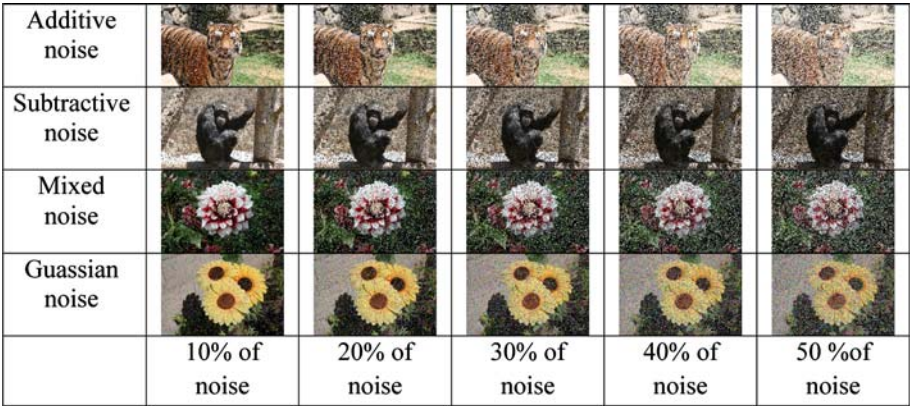

Fig. 2 Sets of images B1, B2, B3 and B4. Each set is composed by 600 images

6. Transform $\widehat{\mathbf{y}}$ using $\hat{\mathbf{y}}^{ar}$. Finally,

$$\mathbf{y} = \widehat{\mathbf{y}} - \hat{\mathbf{y}}^{2} = \begin{pmatrix}1.4 & 0.4 & 0.4 & 0.4\end{pmatrix}^{T} - \begin{pmatrix}1.4 & -0.6 & 0.4 & 0.4\end{pmatrix}^{T}$$
$$= \begin{pmatrix}0.0 & 1.0 & 0.0 & 0.0\end{pmatrix}^{T}$$

4 Behaviour of the Proposed Associative Model Under Noisy Patterns

To corroborate the behaviour and accuracy of the proposed model we have performed several experiments divided into two cases. In the first case we have verified the behaviour and accuracy of our model with patterns altered with additive, subtractive and mixed noise. Different sets of images used in the first kind of experiments are shown in Figs. 1 and 2. In the second kind of experiments we verified the behaviour and accuracy of our model with patterns distorted by image transformations such as rotations, scale changes and object orientations. Different sets of images used in the second kind of experiments were taken from [40] and some images are shown in Figs. 3a and b. For both kinds of experiments, images were firstly transformed into raw vectors. After that, performance of the proposed model was tested.

4.1 First Kind of Experiments

Experiment 1 Auto-associative case (memory acting as a filter): In this experiment we firstly trained the associative memory with the set of images A. In this case, each image was

Fig. 3 **a** Set of image C is composed by 100 classes of objects, 72 images per class. **b** Set of image D is composed by 100 classes of objects, 20 images per class

associated with itself (40 associations). This implies that $\mathbf{x}^k = \mathbf{y}^k$. Once trained the associative memory, we proceed to test the accuracy of the proposal. Firstly we verified if the AM was able to recall the fundamental set of associations using set of image A, and then we verified if the AM could recall the images from noisy versions of them by using the set of images B1, B2, B3 and B4.

Set of images A is composed by 40 different images of flowers and animals. Set of image B1 is composed by 40 classes of flowers and animals, 15 images per class. For each class the pixels of each image were altered by additive noise (we begun with 5% of the pixels of the image until 75% of the total of pixels in steps of 5), this set is composed by 600 images. Set of images B2 is composed by 40 classes of flowers and animals, 15 images per class. For each class the pixels of each image were altered with subtractive noise (we begun with 5% of the pixels of the image until 75% of the total of pixels in steps of 5), this set is composed by 600 images. Set of images B3 is composed by 40 classes of flowers and animals, 15 images per class. For each class the pixels of each image were altered by mixed noise (we begun with 5% of the pixels of the image until 75% of the total of pixels in steps of 5), this set is composed by 600 images. Set of images B4 is composed by 40 classes of flowers and animals, 15 images per class. For each class the pixels of each image were altered by Gaussian noise (we begun with 5% of the pixels of the image until 75% of the total of pixels in steps of 5), this set is composed by 600 images.

In the first case, the fundamental set of associations was correctly recalled. For the second case, the AM recalled the corresponding image, even when the 75% of the pixels where modified. In summary, the accuracy of the proposal was, in both cases, of 100%. This result supports the applicability of the proposal for recalling images from noisy versions of them.

Experiment 2 Hetero-associative case: In this experiment we firstly trained the associative memory with the set of images A. In this case each image was associated with a word that best describes the content of the image i.e. we associated the raw vector of the image with a vector of ASCII values that form the associated word. For example, the image of the tiger was associated with the word "tiger". Once trained the associative memory with the 40 associations, we used the sets of images B1, B2, B3 and B4 expecting recall the corresponding describing word that best describes the content of the input image. Instead of recalling an image, the trained associative memory was used to recall the describing word. The accuracy of the proposal in this case was also of 100%. This result supports the applicability of the proposal for categorizing images in terms of their content; refer for example to [41].

For both experiments 1 and 2, the accuracy of 100% was obtained because the noise affecting the simplified version of the vector-image never surpassed $\mathbf{mid}\,(\bar{\mathbf{x}}) - \mathbf{mid}\,(\tilde{\mathbf{x}}) < d/2$. Finally, the results obtained with the proposal in these two experiments were compared with the results provided by the associative memories based on the **min**, **max** and **median** operators. In short, the associative memory based on **min** and **max** operators presented a high accuracy only with sets B1 and B2, with B3 and B4 the accuracy drastically decreased. The associative memory based on **median** operator only presented a high accuracy with set B3. Our proposal presented a high accuracy with B1, B2, B3 and B4.

4.2 Second Kind of Experiments

Experiment 3 Auto-associative case (invariant image recall): In this experiment we firstly trained the associative memory only with the first image of each class of set of images C (100 associations). Once trained the associative memory as proposed, we proceed to test its performance. Firstly we verified the accuracy of the AM under rotations using set of images C, and then we verified the accuracy of the proposal for different orientations using set of images D.

Set of images C was generated by using the dataset COIL-100. Set of images C is composed by 100 classes of objects, 72 images per class. For each class the object appears rotated from 5° to 360° (in steps of 5), this set is composed by 7200 images. Set of images D was generated by taking only the first 20 images of each object from dataset COIL-100.

For the first case, the AM recalled the corresponding image, even when the image was rotated. Due to **mid** operator is invariant to rotations the accuracy of the AM for this transformation was of 100%.

In the second case, the accuracy of the proposal diminished to 25%, but considering that the AM was trained with 100 associations the results are acceptable. If we reduce the number of associations the accuracy of the proposal increases. For example, when the AM was trained with 15 associations the accuracy increased to 55%. In general, low accuracy is obtained when the noise affecting the simplified version of the vector-image is $\mathbf{mid}\,(\bar{\mathbf{x}}) - \mathbf{mid}\,(\tilde{\mathbf{x}}) > d/2$.

At last, the results obtained with the proposal were compared with the results provided by the associative memories based on the **min**, **max** and **median** operators. In short, the accuracy of the associative memory based on **mix** and **max** operators was less than 10%, with sets C and D. For the case of median associative memory, the accuracy was also low. Our proposal presented a high accuracy with set C, and a low accuracy with set D.

Although the percentage of recalling is low, it is one of the first results reported in literature for recalling images under image orientations by means of an associative memory. It is worth mentioning, that to our knowledge nobody in this field had before reported results of this type. Authors only report results when images or patterns are distorted by additive, subtractive or both noises, but not when images suffer affine transformations.

Important to say is that the model was not compared against Hopfield-based models or recurrent-based models because those model cannot be used to store real-color patterns.

5 Object Recognition Using Dynamic Memories

In this section, the proposal is tested with real patterns. We used the objects shown in Fig. 4.

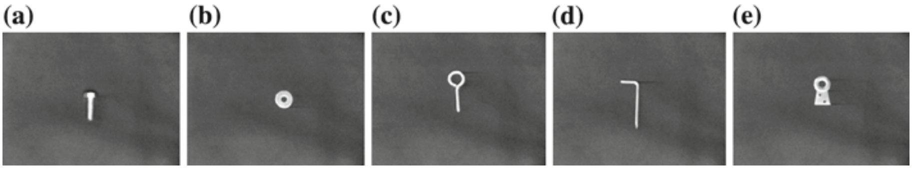

Fig. 4 The five objects used in the experiment. **a** A bolt. **b** A washer. **c** An eyebolt. **d** A hook. **e** A dovetail

5.1 Construction of the Associative Memory

Objects are not directly recognized by their images. Instead of this, we preferred to do it indirectly by invariant descriptions of each of them. Ten images of each object in different positions, rotations were captured. To each image a standard thresholder [42] was applied to get its binary version. Small spurious regions were eliminated from each image by means of a size filter. To each of the twenty images of each object the first three Hu geometric invariants, to translations, rotations and scale changes were computed [43]. Using the set of ten images of each object the average vector of each object was computed. These vectors will be $\mathbf{x}^k$ and the corresponding $\mathbf{y}^k$ will have a "1" that represents the index of the class of each object.

$$\mathbf{x}^1 = \begin{pmatrix} 0.445 \\ 0.163 \\ 0.005 \end{pmatrix} \mathbf{y}^1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \mathbf{x}^2 = \begin{pmatrix} 0.189 \\ 5.79E-5 \\ 4.09E-6 \end{pmatrix} \mathbf{y}^2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \mathbf{x}^3 = \begin{pmatrix} 0.708 \\ 0.291 \\ 0.182 \end{pmatrix} \mathbf{y}^3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$\mathbf{x}^4 = \begin{pmatrix} 1.435 \\ 1.587 \\ 0.873 \end{pmatrix} \mathbf{y}^4 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \mathbf{x}^5 = \begin{pmatrix} 0.249 \\ 0.019 \\ 2.41E-5 \end{pmatrix} \mathbf{y}^5 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

After applying the training phase to the couples of patterns described in Sect. 2.1, we obtained that:

$$\mathbf{M} = \begin{bmatrix} 0.554 & 0.836 & 0.994 \\ -0.445 & -0.163 & -0.005 \\ -0.445 & -0.163 & -0.005 \\ -0.445 & -0.163 & -0.005 \\ -0.445 & -0.163 & -0.005 \end{bmatrix}$$

5.2 Object Recognition Using Recalling Phase

In order to obtain the identity of the objects appearing in an image we follow the next steps:

1. Apply a standard thresholder [42] to get its binary version.
2. Apply a connected component-labeling algorithm [44] to get all possible connected binary components
3. Eliminate small spurious regions by means of a standard size filter.
4. For each object compute the first three Hu geometrics invariant, to translation, rotations and scale changes [43].

Table 1 Percentage of classification for the test set

Object	Bolt (%)	Washer (%)	Eyebolt (%)	Hook (%)	Dovetail (%)
Bolt	100	0	0	0	0
Washer	0	100	0	0	0
Eyebolt	0	0	100	0	0
Hook	0	0	0	100	0
Dovetail	0	0	0	0	100
Percent of classification	100	100	100	100	100

5. Feed the associative memory with patterns obtained from step 4 and apply recalling phase described in Sect. 2.3.

Fifty images (ten for each object), and different from those used to train the associative memory were used to measure the efficiency of the proposal. Table 1 summarizes the recalling results. As you can appreciate the efficiency obtained was of 100% for this set of images. Compared with the result showed in [22] (in average 95%) the proposal was better.

6 Conclusions and Ongoing Research

In this paper we have proposed a new associative model with dynamical synapses. Due to the brain is a dynamic, changing neural network that adapts continuously to meet the demands of communication and computational needs, synapses of our model change in response to an input stimulus.

The major trouble with this approach is to determine which synapses should be modified in response to a stimulus and which of these synapses have to be modified. The properties of the associative memory using **mid** operator were analyzed. There were also determined the principal synapses (kernel) of the associative memory. One of the results obtained was if the synapses belonging to K_{M} change, the associative memory will not be able to recall a pattern, this can be seen as loss of memory.

We have introduced the concept of *active region* and have integrated the capability to adjust synapses according to an input stimulus. Training our model involves three main stages: information codification, simplified version computation and building of the memory. We have described which synapses and how these synapses are modified. We have also provided some propositions that guarantee the recall of the fundamental set of patterns using noisy patterns.

The accuracy of the propose model was tested using different sets of complex images and the result obtained supports the robustness of the proposals. We studied the behaviour of the model when object in the images suffer affine transformations and the results were highly encouraged (possibly applied to 3D object recognition and face recognition; some steps in this direction can be found in [45–47]). It is wealthy mention that to our knowledge nobody in this field had before reported results of this type.

On the other hand, there are other advanced in associative memories with continuous adaptation. Refer for example to [27]. However, we cannot compare the results obtained in this research against the results obtained in [27] because:

1. We are not looking to reproduce the behavior of some process of brain,
2. We are looking to up-perform the accuracy of classical associative models based on some process of brain, and

3. In [27] the authors only report how their model reproduces several features of experimentally observed local UP states, as well as oscillatory behavior on the gamma and theta time scales observed in the cerebral cortex.

From the experiments we can conclude that the proposed model can be efficiently used to:

(1) Find the class of an object (the model can be used in object classifications when they are described in terms of invariant describing feature vectors).
(2) Restore images when these appear distorted with additive, subtractive or mixed noise. The proposed model is sensible to noise due to image transformations.
(3) Associate describing words to images. This suggests that the proposed model can be used for generic object recognition, see for example [48,49].

Nowadays we are studying who increases the robustness of the proposal for complex problems such as face recognition and 3D object recognition.

Annex

Some important properties of **mid** operator are:

Property 1 $\mathbf{mid}\,(x) = x$

Property 2 $\mathbf{mid}\,(\mathbf{x}+\mathbf{y}) = \mathbf{mid}\,(\mathbf{x}) + \mathbf{mid}\,(\mathbf{y})$

Property 3 $\mathbf{mid}\,(\mathbf{x}-\mathbf{y}) = \mathbf{mid}\,(\mathbf{x}) - \mathbf{mid}\,(\mathbf{y})$

Proposition 1 *Let* $\left\{\left(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k\right)|k = 1, \ldots, p\right\}$*,* $\bar{\mathbf{x}}^k \in \mathbf{R}^n$*,* $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel. Every component of vector* $\mathbf{K_W}$*is given as* $kw_i = \bar{y}_i - \mathbf{mid}\,(\bar{\mathbf{x}})$

Proof Starting from Definition 4:

$$
\begin{aligned}
kw_i &= \mathop{\mathbf{mid}}_{i=1}^{n}\left(w_{ij}\right),\ j = 1, \ldots, m \text{ by applying Definition 3}\\
kw_i &= \mathbf{mid}\left(\bar{y}_i - \bar{x}_j\right),\ j = 1, \ldots, m \text{ by applying Property 3}\\
kw_i &= \mathbf{mid}\,(\bar{y}_i) - \mathbf{mid}\left(\bar{x}_j\right),\ j = 1, \ldots, m \text{ by applying Property 1}\\
kw_i &= \bar{y}_i - \mathbf{mid}\left(\bar{x}_j\right),\ j = 1, \ldots, m \text{ by reducing common terms. Q.E.D.}
\end{aligned}
$$

Proposition 2 *Let* $\left\{\left(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k\right)|k = 1, \ldots, p\right\}$*,* $\bar{\mathbf{x}}^k \in \mathbf{R}^n$*,* $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel. Every component of vector* $\mathbf{K_W}$*is given as* $kw_i = \bar{y}_i - \mathbf{mid}\,(\bar{\mathbf{x}})$

Proof Starting from Definition 4:

$$
\begin{aligned}
kw_i &= \mathop{\mathbf{mid}}_{i=1}^{n}\left(m_{ij}\right),\ j = 1, \ldots, m \text{ by applying Definition 3}\\
kw_i &= \mathbf{mid}\left(\bar{y}_i - \bar{x}_j\right),\ j = 1, \ldots, m \text{ by applying Property 3}\\
kw_i &= \mathbf{mid}\,(\bar{y}_i) - \mathbf{mid}\left(\bar{x}_j\right),\ j = 1, \ldots, m \text{ by applying Property 1}\\
kw_i &= \bar{y}_i - \mathbf{mid}\left(\bar{x}_j\right),\ j = 1, \ldots, m \text{ by reducing common terms}\\
kw_i &= \bar{y}_i - \mathbf{mid}\,(\bar{\mathbf{x}}) \text{ Q.E.D.}
\end{aligned}
$$

Proposition 3 *Let* $\left\{\left(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k\right) | k = 1, \ldots, p\right\}$, $\bar{\mathbf{x}}^k \in \mathbf{R}^n$, $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel. Every component of vector* $\bar{\mathbf{y}}$ *can be perfectly recalled by using its corresponding key vector* $\bar{\mathbf{x}}$ *if* $\bar{y}_i = kw_i + \mathbf{mid}(\bar{\mathbf{x}})$

Proof $\bar{y}_i = kw_i + \mathbf{mid}(\bar{\mathbf{x}})$ by applying Proposition 1

$$\begin{aligned}\bar{y}_i &= \bar{y}_i - \mathbf{mid}(\bar{\mathbf{x}}) + \mathbf{mid}(\bar{\mathbf{x}}) \text{ by reducing common terms}\\ \bar{y}_i &= \bar{y}_i \text{ Q.E.D.}\end{aligned}$$

Proposition 4 *Let* $\left\{\left(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k\right) | k = 1, \ldots, p\right\}$, $\bar{\mathbf{x}}^k \in \mathbf{R}^n$, $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel,* Δw *an updated value of* $\mathbf{K_W}$ *and* $\tilde{\mathbf{x}}$ *a distorted version of* $\bar{\mathbf{x}}$. *Every component of vector* $\bar{\mathbf{y}}$ *can be perfectly recalled by using the distorted version of* $\bar{\mathbf{x}}$ *if* $kw_i = kw_i + \Delta w$ *and* $\Delta w = \mathbf{mid}(\bar{\mathbf{x}}) - \mathbf{mid}(\tilde{\mathbf{x}})$.

Proof Starting from result of Proposition 3

$$\begin{aligned}\bar{y}_i &= kw_i + \mathbf{mid}(\tilde{\mathbf{x}}) \text{ by substituting } kw_i \text{ with hypothesis}\\ \bar{y}_i &= kw_i + \Delta w + \mathbf{mid}(\tilde{\mathbf{x}}) \text{ by substituting } \Delta w \text{ with hypothesis}\\ \bar{y}_i &= kw_i + \mathbf{mid}(\bar{\mathbf{x}}) - \mathbf{mid}(\tilde{\mathbf{x}}) + \mathbf{mid}(\tilde{\mathbf{x}}) \text{ by applying Proposition 1}\\ \bar{y}_i &= \bar{y}_i - \mathbf{mid}(\bar{\mathbf{x}}) + \mathbf{mid}(\bar{\mathbf{x}}) - \mathbf{mid}(\tilde{\mathbf{x}}) + \mathbf{mid}(\tilde{\mathbf{x}}) \text{ by reducing common terms}\\ \bar{y}_i &= \bar{y}_i \text{ Q.E.D.}\end{aligned}$$

Proposition 5 *Let* $\left\{\left(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k\right) | k = 1, \ldots, p\right\}$, $\bar{\mathbf{x}}^k \in \mathbf{R}^n$, $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel,* Δw *an updated value of* $\mathbf{K_W}$ *and* $\tilde{\mathbf{x}}$ *a distorted version of* $\bar{\mathbf{x}}$. *Every component of vector* $\bar{\mathbf{y}}$ *can be perfectly recalled by using the distorted version of* $\bar{\mathbf{x}}$ *if* $\mathbf{mid}(\tilde{\mathbf{x}}) = \mathbf{mid}(\bar{\mathbf{x}}) - \Delta w$.

Proof Starting from result of Proposition 3

$$\begin{aligned}\bar{y}_i &= kw_i + \mathbf{mid}(\tilde{\mathbf{x}}) \text{ by substituting hypothesis}\\ \bar{y}_i &= kw_i + \mathbf{mid}(\bar{\mathbf{x}}) - \Delta w \text{ by substituting } kw_i \text{ using Proposition 4}\\ \bar{y}_i &= kw_i + \Delta w + \mathbf{mid}(\bar{\mathbf{x}}) - \Delta w \text{ by applying Proposition 1}\\ \bar{y}_i &= \bar{y}_i - \mathbf{mid}(\bar{\mathbf{x}}) + \Delta w + \mathbf{mid}(\bar{\mathbf{x}}) - \Delta w \text{by reducing common terms}\\ \bar{y}_i &= \bar{y}_i \text{ Q.E.D.}\end{aligned}$$

Proposition 6 *Let* $\left\{\left(\bar{\mathbf{x}}^k, \bar{\mathbf{y}}^k\right) | k = 1, \ldots, p\right\}$, $\bar{\mathbf{x}}^k \in \mathbf{R}^n$, $\bar{\mathbf{y}}^k \in \mathbf{R}^m$ *a fundamental set of associations of an associative memory* $\mathbf{W}$ *and* $\mathbf{K_W} \in \mathbf{R}^n$ *its kernel,* Δw *an updated value of* $\mathbf{K_W}$ *and* $\tilde{\mathbf{x}}$ *a distorted version of* $\bar{\mathbf{x}}$. *Every component of vector* $\bar{\mathbf{y}}$ *can be perfectly recalled by using the distorted version of* $\bar{\mathbf{x}}$ *if* $|\Delta w| < \frac{d}{2}$

Proof Let $\mathbf{x}^i = \mathbf{x}$, $\mathbf{x}^j = \mathbf{x} + d$ and $\tilde{\mathbf{x}} = \mathbf{x} - \Delta w$, if $|\Delta w| < \frac{d}{2}$ by expanding inequality then we have $-\frac{d}{2} < \Delta w$ and $\Delta w < \frac{d}{2}$

Case 1 $\mathbf{D}(\mathbf{x} - \Delta w, \mathbf{x} + d) < \mathbf{D}(\mathbf{x}, \mathbf{x} - \Delta w)$ by applying active region equation
$\mathbf{mid}(\mathbf{x} - \Delta w) - \mathbf{mid}(\mathbf{x} + d) < \mathbf{mid}(\mathbf{x}) - \mathbf{mid}(\mathbf{x} - \Delta w)$ by applying Properties 2 and 3
$\mathbf{mid}(\mathbf{x}) - \mathbf{mid}(\Delta w) - \mathbf{mid}(\mathbf{x}) - \mathbf{mid}(d) < \mathbf{mid}(\mathbf{x}) - \mathbf{mid}(\mathbf{x}) + \mathbf{mid}(\Delta w)$ by reducing terms
$-\mathbf{mid}(\Delta w) - \mathbf{mid}(d) < \mathbf{mid}(\Delta w)$ by applying Property 1
$-\Delta w - d < \Delta w$ by reducing terms
$-d < \Delta w + \Delta w$

$-d < 2\Delta w$
$-\frac{d}{2} < 2\Delta w$

Case 2 $\mathbf{D}(\mathbf{x}, \mathbf{x} + \Delta w) < \mathbf{D}(\mathbf{x} + \Delta w, \mathbf{x} + d)$ by applying active region equation
$\mathbf{mid}(\mathbf{x}) - \mathbf{mid}(\mathbf{x} + \Delta w) < \mathbf{mid}(\mathbf{x} + \Delta w) - \mathbf{mid}(\mathbf{x} + d)$ by applying Properties 2 and 3
$\mathbf{mid}(\mathbf{x}) - \mathbf{mid}(\mathbf{x}) - \mathbf{mid}(\Delta w) < \mathbf{mid}(\mathbf{x}) + \mathbf{mid}(\Delta w) - \mathbf{mid}(\mathbf{x}) - \mathbf{mid}(d)$ by reducing terms
$-\mathbf{mid}(\Delta w) < \mathbf{mid}(\Delta w) - \mathbf{mid}(d)$ by applying Property 1
$-\Delta w < \Delta w - d$ by reducing terms
$-\Delta w - \Delta w < -d$
$-2\Delta w < -d$
$\Delta w < \frac{d}{2}$

Acknowledgements The authors give thanks to CIC-IPN and COFAA-IPN for the economical support. Authors also thank SIP-IPN and CONACYT for their support under grants 20071438, 20082948 and 46805. We also greatly appreciate the reviewers' comments for the improvement of this paper.

References

1. McCulloch WS, Pitts WH (1943) A logical calculus of the ideas immanent in nervous activity. Bull Math Biophys 5:115–133. doi:10.1007/BF02478259
2. Hebb DO (1949) The organization of behavior. Wiley, New York
3. Rosenblatt F (1958) The perceptron: a probabilistic model for information storage & organization in the brain. Psychol Rev 65:386–408 Medline. doi:10.1037/h0042519
4. Rumelhart D, McClelland J (1986) Parallel distributed processing group. MIT Press, London
5. Steinbuch K (1961) Die Lernmatrix. Kybernetik 1(1):26–45. doi:10.1007/BF00293853
6. Anderson JA (1972) A simple neural network generating an interactive memory. Math Biosci 14:197–220. doi:10.1016/0025-5564(72)90075-2
7. Kohonen T (1972) Correlation matrix memories. IEEE Trans Comput 21(4):353–359
8. Nakano K (1972) Associatron: a model of associative memory. EEE Trans Syst, Man Cybern SMC-2(3):380–388
9. Hopfield JJ (1982) Neural networks and physical systems with emergent collective computational abilities. Proc Natl Acad Sci USA 79:2554–2558. doi:10.1073/pnas.79.8.2554
10. Lu J et al (2006) Topology influences performance in the associative memory neural network. Phys Lett A 354:335–343. doi:10.1016/j.physleta.2006.01.085
11. Lee D-L (2006) Improvements of complex-valued Hopfiel associative memory by using generalized projections rules. IEEE Trans Neural Netw 17(5):1341–1347. doi:10.1109/TNN.2006.878786
12. Casali D et al (2006) Associative memory design using a support vector machine. IEEE Trans Neural Netw 17(5):1165–1174. doi:10.1109/TNN.2006.877539
13. Tang H et al (2006) Dynamic analysis and analog associative memory of network with LT neurons. IEEE Trans Neural Netw 17(2):409–418. doi:10.1109/TNN.2005.863457
14. Charlier S et al (2006) NDRAM : Non linear dynamic recurrent associative memory for learning bipolar and nonbipolar correlated patterns. IEEE Trans Neural Netw 16(6):1393–1400. doi:10.1109/TNN.2005.852861
15. Zhu J, von der Malsburg C (2006) Associative memory of conectivety patterns. Neurocomputing 69:1305–1308. doi:10.1016/j.neucom.2005.12.097
16. Rehn M, Sommer FT (2006) Storing and restoring visual input with collavorative rank coding and associative memory. Neurocomputing 69:1219–1223. doi:10.1016/j.neucom.2005.12.080
17. Mu X et al (2007) A weighted voting model of associative memory. IEEE Trans Neural Netw 18(3):756–777. doi:10.1109/TNN.2007.891196
18. Wickramasinghe LK et al (2007) A novel episodic associative memory model for enhanced classification accuracy. Pattern Recognit Lett 28:1193–1202. doi:10.1016/j.patrec.2007.02.012
19. Ritter GX, Sussner P, Diaz de Leon JL (1998) Morphological associative memories. IEEE Trans Neural Netw 9(2):281–293. doi:10.1109/72.661123
20. Chung F-L, Lee T (1996) On fuzzy associative memory with multiple-rule storage capacity. IEEE Trans Fuzzy Syst 4(4):375–384. doi:10.1109/91.531778

21. Wang ST, Lu H (2004) On new fuzzy morphological associative memories. IEEE Trans Fuzzy Syst 12(3):316–323. doi:10.1109/TFUZZ.2004.825977
22. Sossa H, Barron R, Vazquez RA (2004) New associative memories to recall real-valued patterns. In: Sanfeliu A, Martínez Trinidad JF, Carrasco-Ochoa JA (eds) Progress in pattern recognition, image analysis and applications, 9th Iberoamerican Congress on Pattern Recognition, CIARP 2004, Puebla, Mexico, October 26–29, 2004, Proceedings. Lecture Notes in Computer Science, N 3287, pp 195–202, Springer
23. Sussner P, Valle M (2006) Gray-scale morphological associative memories. IEEE Trans Neural Netw 17(3):559–570. doi:10.1109/TNN.2006.873280
24. Ehlers MD (2003) Activity level controls postsynaptic composition and signaling via the ubiquitin-proteasome system. Nat Neurosci 6(3):231–242. doi:10.1038/nn1013
25. Hecht-Nielse et al (2003) A theory of the thalamocortex. Computational models for neuroscience. Springer-Verlag, London, pp 85–124
26. Arbid MA (2003) The handbook of brain theory and neural networks. The MIT Press, London
27. Lundqvist M, Rehn M, Lansner A (2006) Attractor dynamics in a modular network model of the cerebral cortex. Neurocomputing 69:1155–1159. doi:10.1016/j.neucom.2005.12.065
28. Pantic L et al (2000) Associative memory with dynamic synapses. Neural Comput 14:2903–2923. doi:10.1162/089976602760805331
29. Bibitchkov D, Herrman JM, Giesel T (2000) Synaptic depression in associative memory networks. In Proceedings of IJCNN 2000, vol 5, pp 30–35
30. Wang Z, Fan H (2005) Memory retrieval in a neural network with chaotic neurons and dynamic synapses. In: Cabestany J, Prieto A, Sandoval Hernandez F (eds) IWANN 2005: Computational intelligence and bioinspired systems, 8th International work conference IWANN 2008, Barcelona, Spain, June 8–210, Proceedings. Lecture Notes in Computers Sciences, N 3512, pp 654–660. Springer, New York
31. Sejnowski TJ (1976) On the stochastic dynamics of neuronal interaction. Biol Cybern 22:203–211. doi:10.1007/BF00365086
32. Makeig S et al (2002) Dynamic brain sources of visual evoked responses. Science 295:690–694. doi:10.1126/science.1066168
33. Sossa H, Barron R, Vazquez RA (2007) Study of the influence of the noise in the values of a median associative memory. In: Beliczynski B, Dzielinski A, Iwanowski M, Ribeiro B (eds) Adaptive and natural computing algorithms. 8th International conference, ICANNGA 2007, Warsaw, Poland, April 11–14, 2007, Proceedings, Part II. Lecture Notes in Computer Science, N 4432, pp 55–62, Springer, New York
34. Laughlin SB, Sejnowski TJ (2003) Communication in neuronal networks. Science 301:1870–1874. doi:10.1126/science.1089662
35. Kutas M, Hillyard SA (1984) Brain potentials during reading reflect word expectancy and semantic association. Nature 307:161–163. doi:10.1038/307161a0
36. Price CJ (2000) The anatomy of language: contributions from functional neuroimaging. J Anat 197(3):335–359. doi:10.1046/j.1469-7580.2000.19730335.x
37. Sossa H, Barron R, Vazquez RA (2004) Transforming fundamental set of patterns to canonical form to improve pattern recall. In: Lemaître Ch, Reyes CA, González JA (eds) Advances in artificial intelligence - IBERAMIA 2004, 9th Ibero-American Conference on AI, Puebla, México, November 22–26, 2004, Proceedings. Lecture Notes in Artificial Intelligence, N 3315, pp 687–696, Springer
38. Reinvan I (1998) Amnestic disorders and their role in cognitive theory. Scand J Psychol 39(3):141–143. doi:10.1111/1467-9450.393068
39. Jovanova-Nesic KD, Jankovic BD (2005) The neuronal and immune memory systems as supervisors of neural plasticity and aging of the brain: from phenomenology to coding of information. Ann N Y Acad Sci 1057:279–295. doi:10.1196/annals.1356.022
40. Nene SA et al (1996) Columbia Object Image Library (COIL 100). Technical Report No. CUCS-006-96. Department of Computer Science, Columbia University
41. Vazquez RA, Sossa H (2006) Image categorization using associative memories. In: Martínez Trinidad JF, Carrasco-Ochoa JA, Kittler J (eds) Progress in pattern recognition, image analysis and applications, 11th Iberoamerican Congress in Pattern Recognition, CIARP 2006, Cancun, Mexico, November 14–17, 2006, Proceedings. Lecture Notes in Computer Science, N 4225, pp 549–558. Springer, New York
42. Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans SMC 9(1):62–66
43. Hu MK (1962) Visual pattern recognition by moment invariants. IRE Trans Inf Theory 8:179–187
44. Gonzalez RC, Woods RE (2002) Digital image processing. 2nd edn. Prentice Hall
45. Vazquez RA, Sossa H, Garro BA (2007) 3D Object recognition based on low frequencies response and random feature selections. In: MICAI 2007: Advances in Artificial Intelligence, 6th Mexican International Conference on Artificial Intelligence, Aguascalientes, Mexico, November 5–9, 2007, Proceedings. Lecture Notes in Artificial Intelligence, N 4827, pp 694–704, Springer, New York

46. Vazquez RA, Sossa H, Garro BA (2007) Face recognition using some aspects of the infant vision system and associative memories. In: Rueda L, Mery D, Kittler J (eds) Progress in Pattern Recognition, Image Analysis and Applications, 12th Iberoamerican Congress in Pattern Recognition, CIARP 2007, Valparaiso, Chile, November 13–16, 2007, Proceedings. Lecture Notes in Computer Science, N 4756, pp 437–446, Springer, New York
47. Vazquez RA, Sossa H, Garro BA (2007). Low frequency responses and random feature selection applied to face recognition. In: Kamel M, Campilho A (eds) ICIAR 2007: Image Analysis and Recognition, International conference ICIAR 2007, Toronto, Canada, August 22–24, Proceedings. Lecture Notes in Computers Sciences, N 4633, pp 818–830, Springer, New York
48. Vazquez RA, Sossa H (2008) Voice translator based on associative memories. In: Sun F, Zhang J, Tan Y, Cao J, Yu W (eds) Advances in neural networks. ISNN 2008, Beijing, China, September 24–128, 2008, Proceedings. Lecture Notes in Computer Science, N 5264, Part II, pp 341–350, 830, Springer, New York
49. Vazquez RA, Sossa H (2008) Associative memories applied to pattern recognition. In: Kurkova V, Neruda R, Koutnik J (eds) Artificial neural networks. ICANN 2008, Prague, Czech Republic, September 3–19, 2008, Proceedings. Lecture Notes in Computer Science, N 5164, Part II, pp 111–120, Springer, New York