Neurocomputing 73 (2009) 119–126

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Invariant pattern recognition using analog recurrent associative memories

Iulian B. Ciocoiu *

Faculty of Electronics and Telecommunications, Technical University of Iasi, Bd. Carol I, No. 11, Iasi 700506, Romania

ARTICLE INFO

Available online 6 August 2009

Keywords:
Associative memory
Invariance
Tangent distance
Stable equilibria

ABSTRACT

A novel invariant pattern recognition approach is proposed based on a special gradient-type recurrent analog associative memory. The system exhibits stable equilibrium points in predefined positions specified by feature vectors extracted from the training set, while invariance to geometrical transformations is inferred by using the tangent distance. Experimental results for handwritten character recognition and face recognition tasks indicate that the proposed approach may yield superior performances over classical solutions based on the Euclidean distance metric. Possible extensions towards modular and sequential pattern recognition are finally outlined.

1. Introduction

How to infer invariance to elementary transformations has represented one of the key issues of pattern recognition literature in the last decades. Depending on the task at hand, this problem adds to a list of other difficult aspects such as dealing with a limited number of training samples, partial occlusion, or illumination variability, to cite just a few. Robustness against affine geometric transformations such as translations, rotations, or scale changes is typically tackled by one of the following approaches:

(a) preprocessing algorithms aiming at extracting specific invariant features from the original patterns; and
(b) the effect of applying these transformations may be built into the definition of the distance metric itself [28]. Methods belonging to the first approach are typically simpler and examples include (higher order) autocorrelation functions [21], and spectral features of log-polar images [15]. The second class of methods is illustrated by the tangent distance (TD) [27], the Image Euclidean distance (IMED) [32], and the deformable templates method [31].

Standard classification procedures such as k-nearest neighbors or Bayesian rule have been extensively used in pattern recognition applications, and a huge corpus of literature has been devoted to analyzing their theoretical grounds, limitations, and practical performances. Nevertheless, more recent classification techniques have also emerged as potential alternative solutions. These were mainly introduced in the context of neural networks, and examples include Support Vector Machines (SVM), neural autoassociators [16], and associative memories [9]. The present contribution introduces a novel approach to invariant pattern recognition based on a combination of two elements presented above, namely a special type of (recurrent) associative memory and an invariant distance metric such as the tangent distance (TD).

Associative memories represent one the most interesting applications of artificial neural networks and many solutions have been reported in the literature. Basically, a set of patterns is stored by using a training database and a proper learning procedure. In the testing phase, the system should output correct results even if noisy, incomplete or distorted data are applied as input. Two strategies are mainly employed, depending upon the type of architecture that is used: (a) for *feedforward* networks (usually algebraic) functional dependencies between input and target data are approximated based on limited training information. If the network has *generalization* capabilities, it would successfully deal with previously unseen testing data; (b) for *recurrent* networks, desired memories are stored as stable states of dynamical systems. When certain conditions are met such systems are *globally stable* and the dynamics will evolve from any initial state towards one particular stable equilibrium and no other complex behavior can occur [14]. Such systems should satisfy the following requirements:

- no spurious memories (stable states which do not correspond to the desired ones) should exist;
- the number of desired equilibria should be arbitrarily large and the dimension of the corresponding basins of attraction should be controllable; and
- the addition/elimination of an equilibrium should be performed without redesigning the whole system.

* Tel.: +40 232 213737.
E-mail address: iciocoiu@etc.tuiasi.ro

0925-2312/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2009.02.024

Tangent distance has been introduced in optical character recognition (OCR) applications as an alternative to classical Euclidean distance, which is known to be very sensitive to geometrical transformations such as translations, rotations, or scale changes. This metric is defined as the minimum (squared) distance between the *manifolds* that are generated by the set of the transformed patterns. Since these manifolds generally do not have an analytic expression, a natural solution is to use a proper approximation. Simard [27] proposed as a valid approximation the tangent subspace obtained by adding to the current vector a linear combination of the tangent vectors corresponding to seven distinct transformations.

The next paragraph clarifies the key elements of both components of the proposed solution. Experimental results for a handwritten classification task using the USPS database are reported, and possible extensions towards modular and sequential pattern recognition are finally outlined.

2. Invariant associative memory design

Design details regarding both the associative memory and the distance metric involved are presented below. While the two components have been separately used previously, their combination is novel and may yield superior performances in invariant pattern recognition applications.

2.1. Recurrent associative memory

The main drawbacks of existing solutions for associative memory design are related to the presence of many spurious states and limited memory capacity. In order to alleviate these, we use a special gradient-type dynamic system defined according to

$$\frac{dx_i}{dt} = -\frac{\partial V(\mathbf{X})}{\partial x_i}, i = 1, \ldots, N, \tag{1}$$

where $\mathbf{X} = \{x_i\}$ defines the state-vector, N is the order of the system, and $V(\mathbf{X})$ is the associated Lyapunov function. A well-known result states that all isolated minima of $V(\mathbf{X})$ are asymptotically stable states of system (1) [14]. Function $V(\mathbf{X})$ will be chosen in order to satisfy the set of requirements indicated in the previous paragraph. Moreover, any desired memory pattern should be stored as a point in a multidimensional state space where the Lyapunov function $V(\mathbf{X})$ has a minimum. The key feature of our approach lies in the special way of constructing the function $V(\mathbf{X})$ *as a sum of individual functions exhibiting good space localization properties*, having deep minima at the desired locations and been practically constant in rest:

$$V(\mathbf{X}) = \sum_{m=1}^{M} w_m g_m(\mathbf{X}), \tag{2}$$

where M is the number of memories to be stored, w_m are scalar weights, and functions $g_m(\mathbf{X})$ are chosen as

$$g_m(\mathbf{X}) = 1 - e^{-d_p^p(\mathbf{X},\mathbf{X}_m)/2\sigma_m^2}, \tag{3}$$

where $d_p(\mathbf{X},\mathbf{X}_m)$ is the distance induced by the L_p measure defined on the N-dimensional vector space. As a consequence, the equations governing the dynamics of the system become

$$\frac{dx_i}{dt} = -\frac{x_i}{\sigma^2}\sum_{m=1}^{M} w_m e^{-\sum_{j=1}^{N}(x_j - x_j^m)^2/2\sigma^2} + \frac{1}{\sigma^2}\sum_{m=1}^{M} w_m x_i^m e^{-\sum_{j=1}^{N}(x_j - x_j^m)^2/2\sigma^2},$$

$$i = 1, \ldots, N. \tag{4}$$

In Fig. 1 we present an example of the function $V(\mathbf{X})$ for a system with $N = 2$ and $M = 4$ stable equilibrium points: $(-1, -1)$; $(-1, 1)$; $(1, -1)$; $(1, 1)$. We used a common value $\sigma_m = 0.25$, and the weights vector was $\{w_m\} = \{1, 1, 1, 1\}$.

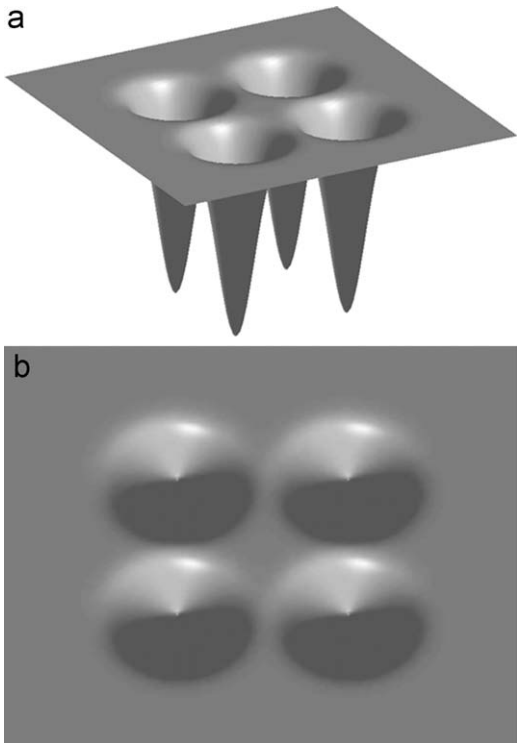

Fig. 1. (a) Example of a Lyapunov function as in Eq. (2) ($M = 4$, $N = 2$); (b) upside view.

Remarks. (a) the idea was first presented in [11] to solve a toy-problem classification task. Later on it was used for soft decision decoding of block codes [5] and face recognition [8]. Storing (a limited number of) high-dimensional grey-level patterns has been reported in [24];

(b) when using Gaussian-type space selective functions, Eq. (2) defines a special Radial Basis Functions (RBF) expansion [13] of the Lyapunov function $V(\mathbf{X})$, where the desired equilibrium points act as the centers.

The proposed design procedure has a number of important advantages, including:

- a clear correspondence between the set of memories to be stored and the equations governing the system dynamics,
- a transparent interpretation of the effect of the parameters (centers, weights, width) on the time and state-space evolution,
- guaranteed convergence based on Lyapunov stability theory, and
- implementation advantages in terms of limited number of interconnections.

The operating mode of the recurrent associative memory acting as a classifier is quite straightforward: when a test pattern is presented to the system it is applied as an *initial condition* to the (neural) dynamical system, which will eventually settle down to one of the stable equilibrium points, hopefully to one obtained from a training pattern of the correct class. According to the positions of the training images, complex basins of attractions are developed around the equilibrium points, which may include

besides the available test images many others, e.g. ones corresponding to occluded, distorted or noisy versions of the training set. In this respect, it is worth mentioning that proper choice of the individual σ_m parameters offers an additional handle for shaping those basins of attraction. Moreover, the proposed neural classifier exhibits implicit modularity, in that storing additional memories does not influence the positions of the previously stored ones and, more importantly, the dynamics of the system and thus the final solution is influenced only by a small fraction of the existing stable equilibria (ideally, only by a single stable point whose basin of attraction the test vector falls into).

If the proposed approach is to be superior to a standard nearest neighbor classifier, it is worth discussing why test vectors closer (in terms of Euclidean distance) to a specific training pattern may still fall into the basin of attraction of another one. The answer is closely related to the shape of the energy landscape corresponding to $V(\mathbf{X})$, which is strongly influenced by the width parameters σ_m. We have two choices:

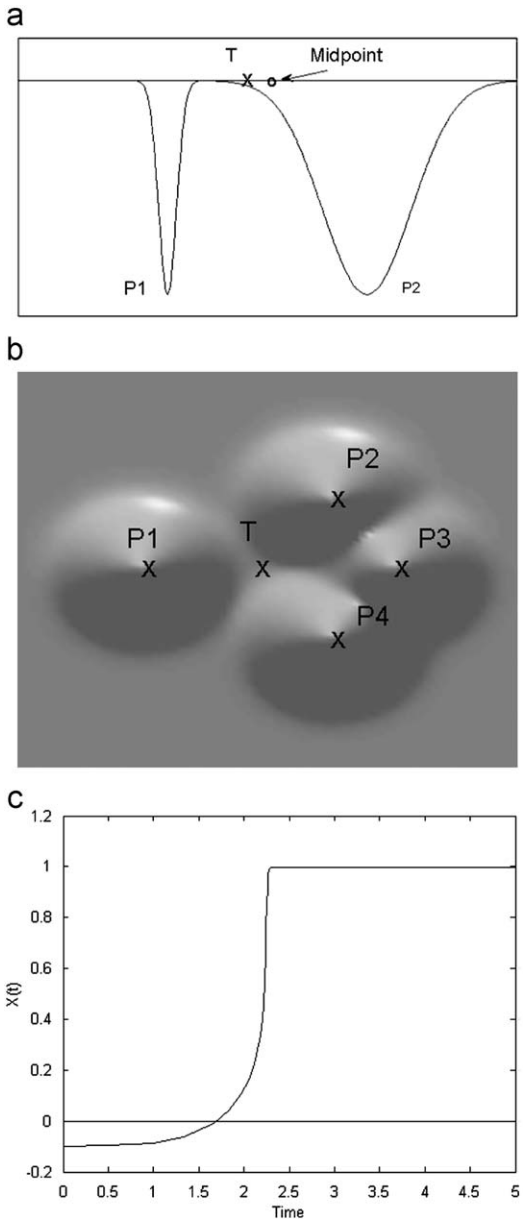

Fig. 2. (a) Effect of the width parameter and attractors distribution on the Lyapunov function (a) unequal widths; (b) equal width, $V(\mathbf{X})$ landscape; and (c) time evolution during convergence.

(a) Using distinct values for those parameters, as in Fig. 2a: pattern T falls into the basin of attraction of pattern P2 due to its wider width, although T is closer to P1. This may be justified for example by considering the formulation of the Lyapunov function as a Gaussian mixture model (GMM), whose parameters are obtained by using Expectation–Maximization (EM) algorithm [2]. Anyway, besides the well-known non-uniqueness of the solution, the procedure does not guarantee that the resulting centers correspond to the desired ones: if the resulting centers are too close to each other, they could merge into another, spurious one. A biologically motivated effect known as *priming* (widely used for constructing psychological models of human cognition [33]) may also suggest using different σ_m. Basically, the effect means faster reaching of a specific attractor if it has been visited recently. Priming could be achieved by increasing the probability of reaching a specific recently visited pattern by enlarging the basins of attraction around it. This is implemented by increasing the value of the width parameter σ_m (and, if necessary, of the w_m weight).

(b) Using a common σ_m value: Another biological model known as the *gang effect* [33] may explain the efficiency of this choice. It refers to the mutual influence between attractors, generated by their spatial distribution. The example in Fig. 2b and c shows that although point T is closer to point P1, it still evolves towards P3. The specific coordinates are as follows: P1(−1,0), P2(0.5, 0.7), P3(1,0), P4(−0.5,0.7), T(−0.1,0). The distances between T and points P1–P4 are {0.9, 0.922, 1.1, 0.992}, respectively.

2.2. Tangent distance

Simple distance measures like the Euclidean distance are very sensitive to affine transformations like scaling, translation, rotation, shearing or axis deformation. In 1993, Simard et al. proposed an invariant distance measure called *tangent distance* (TD), which proved to be especially effective in OCR tasks [27]. In order to introduce it, we must first notice that when an image is affected by a transformation $t(\mathbf{x},\boldsymbol{\alpha})$ which depends on L parameters $\boldsymbol{\alpha}\in\Re^L$ (e.g., the scaling factor and rotation angle), the set of all transformed patterns is *a manifold* of at most dimension L in pattern space. The distance between two patterns can now be defined as the minimum (squared) distance between their respective manifolds, being truly invariant with respect to the L regarded transformations. Computation of this *manifold distance* is a hard non-linear optimization problem and the manifolds concerned generally do not have an analytic expression, hence a natural solution is to use an approximation of the manifold. Simard proposed as a valid approximation the tangent subspace obtained by adding to the current vector $\mathbf{x}$ a linear combination of the vectors that span the tangent subspace and are the partial derivatives of $t(\mathbf{x},\boldsymbol{\alpha})$ with respect to $\boldsymbol{\alpha}$ ($\mathbf{x}\in\Re^{N\times 1}, \mathbf{T}\in\Re^{N\times L}$):

$$t(\mathbf{x},\boldsymbol{\alpha})\approx\mathbf{x}+\sum_{l=1}^{L}\alpha_l\frac{\partial t(\mathbf{x},\boldsymbol{\alpha})}{\partial\alpha_l}=\mathbf{x}+\sum_{l=1}^{L}\alpha_l x_l=\mathbf{x}+\mathbf{T}\cdot\boldsymbol{\alpha}. \tag{5}$$

The *double-sided* tangent distance between two vectors $\mathbf{x}$ and $\mathbf{y}$ is defined as follows:

$$TD_{DS}(x,y)=\min_{\boldsymbol{\alpha}_x,\boldsymbol{\alpha}_y\in R^L}\{\|(\mathbf{x}+\sum_{l=1}^{L}\alpha_{x_l}x_l)-(\mathbf{y}+\sum_{l=1}^{L}\alpha_{y_l}y_l)\|^2\}. \tag{6}$$

Due to computational burden, the *single-sided* tangent distance may be computed with less effort (in this case *TD* is not a proper distance, since it is not symmetric):

$$TD_{SS}(x,y)=\min_{\boldsymbol{\alpha}_x\in R^L}\{\|\mathbf{x}+\sum_{l=1}^{L}\alpha_{x_l}x_l-\mathbf{y}\|^2\}. \tag{7}$$

Simard proposed seven transformations to be used for OCR applications, six accounting for affine variations of the image, and one that models a line thickness deformation. The expressions of the corresponding derivatives are presented in Table 1 [18], where we consider a general affine transformation of an image grid as

$$\begin{pmatrix} i' \\ j' \end{pmatrix} = \begin{pmatrix} 1+\alpha_1 & \alpha_2 \\ \alpha_3 & 1+\alpha_4 \end{pmatrix} \begin{pmatrix} i \\ j \end{pmatrix} + \begin{pmatrix} \alpha_5 \\ \alpha_6 \end{pmatrix}. \tag{8}$$

Since computing the tangent distance for large training databases is computationally intensive, several alternatives to the classical approach have been proposed:

- Instead of computing the set of tangents for every training data vector, we may use a clustering procedure (e.g., k-means algorithm) in order to obtain a set of *centroids* that are representative for large subsets of the training data [12]. When a test vector is to be classified, the tangent distance is now computed against the (limited number of) centroid vectors, and not the original training database (the method is called *tangent centroid*).
- We may speculate the linear nature of tangent plane computation and apply the procedure not on the original (high-dimensional) images, but on compressed version of those, obtained after performing a linear subspace projection, e.g. based on the well-known Principal Components Analysis (PCA) method [20]. More specifically, denoting by $\mathbf{P} \in \Re^{D\times N}$ the projection matrix, we may write: $\mathbf{P}\cdot(\mathbf{x}+\mathbf{T}\cdot\boldsymbol{\alpha}) = \mathbf{P}\cdot\mathbf{x} + (\mathbf{P}\cdot\mathbf{T})\cdot\boldsymbol{\alpha} = \tilde{\mathbf{x}} + \tilde{\mathbf{T}}\cdot\boldsymbol{\alpha}$, hence the tangent vectors grouped in matrix $\mathbf{T}$ are transformed using the same subspace projection procedure as the original images (and the tangent distance is further computed in the compressed space of dimension D).

We may combine the ideas presented above in order to infer invariance to the recurrent associative memory. In this respect, we will replace the Euclidean distance in Eq. (3) by the tangent distance. RBF functions using TD have been previously used in the context of invariant SVM's yielding TD kernels or more general distance substitution kernels [10]. The main advantage of the associative memory approach is that it avoids *explicit computation* of the distances between the test patterns and the prototype ones, that could be computationally expensive especially in cases of large databases storing high-dimensional prototype patterns.

One limitation of the current approach is related to the fact that computing the tangent vectors as indicated in Table 1 is optimized for the particular case of OCR applications. For other data classes, for example face images, this approach may yield unrealistic effects as indicated in the following paragraph, hence other choices should be considered. Limited convergence speed towards the equilibrium points for distant test patterns is also a disadvantage, although it could be compensated by introducing an additional scaling coefficient in the right term of Eq. (1). Moreover, since a test pattern placed far away from a memory pattern does not contribute significantly to the energy function $V(\mathbf{X})$ defined in Eq. (2), we may keep only the summation terms that exceed a certain threshold magnitude. The convergence speed and computational cost of the Euclidean-type recurrent associative memory have been analyzed in [24].

Proper choice of the individual σ_m parameters has a critical influence on the shape of the energy function and corresponding basins of attraction around equilibrium points. Although a learning algorithm could provide optimized performances, we set their values according to a heuristic rule, namely as a fraction of the distance between a memory vector $\mathbf{X}_m$ and its closest neighbor (neighbors originating from the training images of the same person are excluded). As a consequence, distinct σ_m values yield basins of attraction having unequal widths, combining the two approaches that were discussed at the end of Section 2.1 that may explain why test vectors closer to a training pattern could still fall into the basin of another one.

3. Experimental results

3.1. Optical character recognition

We have performed extensive computer experiments using the United States Postal Service (USPS) database [30]. It comprises 7291 training images, and 2007 test images. Each image consists of 16×16 pixels of grayscale values ranging from 0 to 255. The tangent vectors were computed using MATLAB, starting from a publicly available C implementation [17]. Examples of tangent vectors are presented in Fig. 3.

Table 1
Derivative expressions for affine transformations.

Operation	Parameters α_l, $l=1,\ldots,7$	Grid	Derivative
Horizontal translation	$\alpha_l=0$, $l=1,2,3,4,6$	$i'=i+\alpha_5$	$x_1(i,j)=\lim_{\alpha_5\to0}\frac{x(i+\alpha_5,j)-x(i,j)}{\alpha_5}$
Vertical translation	$\alpha_l=0$, $l=1,2,3,4,5$	$j'=j$ $i'=i$	$x_2(i,j)=\lim_{\alpha_6\to0}\frac{x(i,j+\alpha_6)-x(i,j)}{\alpha_6}$
Rotation	$\alpha_l=0$, $l=1,4,5,6$ $\alpha_2=-\alpha_3$	$j'=j+\alpha_6$ $i'=i+\alpha_2 j$ $j'=j-\alpha_2 i$	$x_3(i,j)=jx_1(i,j)-ix_2(i,j)$
Scaling	$\alpha_l=0$, $l=2,3,5,6$ $\alpha_1=\alpha_4$	$i'=i+\alpha_1 i$ $j'=j+\alpha_4 j$	$x_4(i,j)=ix_1(i,j)+jx_2(i,j)$
Axis deformation	$\alpha_l=0$, l $=1,4,5,6$ $\alpha_2=\alpha_3$	$i'=i+\alpha_2 j$ $j'=j+\alpha_3 i$	$x_5(i,j)=jx_1(i,j)+ix_2(i,j)$
Diagonal deformation	$\alpha_l=0$, l $=2,3,5,6$ $\alpha_1=-\alpha_4$	$i'=i+\alpha_4 i$ $j'=j-\alpha_4 j$	$x_4(i,j)=ix_1(i,j)-jx_2(i,j)$
Thickness	–	–	$x_7(i,j)=x_1(i,j)^2+x_2(i,j)^2$

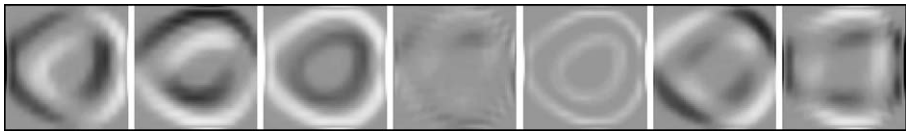

Fig. 3. Examples of tangent vectors.

Table 2
Classification error rates for USPS database (%).

Original data+L2 distance	Original data+*TD* distance	Centroids+L2			Centroids+*TD*		
		No. of centroids			No. of centroids		
		8	12	16	8	12	16
5.63	3.54	8.97	8.57	7.92	6.5	6.7	5.8

No. of centroids	Dimension of projection subspace			
	20	30	40	50
	L2+PCA-compressed centroids			
8	10.2	10.1	9.1	9.2
12	9.6	8.1	8.2	8.5
16	9.3	8.3	8	7.6
	TD+PCA-compressed centroids			
8	8.4	7.7	5.4	5.7
12	8.2	6.9	6.5	6.4
16	8.2	6.9	5.4	5.7
	Associative memory+L2+PCA-compressed centroids			
8	10.1	9.2	8.9	9.3
12	9.6	8.5	8.4	8.5
16	9.4	8.5	7.9	7.4
	Associative memory+TD+PCA-compressed centroids			
8	8.2	7.5	7	6.8
12	8.1	7	6.3	6
16	7.2	6.6	5.5	5.7

Experiments were using the following setups (single-sided TD was used throughout all tests): (a) original images+Euclidean/TD distances; (b) centroids+Euclidean/TD distances; (c) PCA-compressed centroids+Euclidean/TD distances; (d) associative memory+PCA-compressed centroids+Euclidean/TD distances. We report results for 8, 12, and 16 centroids always selected using k-means clustering algorithm, over 20 independent trials (considering more centroids could result in clusters having too few points). The dimension of the projection subspace varies between 20 and 50, capturing more than 90% of the energy of the original images. Classification error rates are given in Table 2.

Available experimental results indicate that the autoassociative memory typically behaves better than the nearest neighbor alternative. In all experimental setups using tangent distance yields superior performances than the Euclidean choice. While not outperforming the nearest neighbor approach, the autoassociative memory may still offer the correct class label in cases such as those presented in Fig. 4, where the nearest neighbor rule fails to. Nevertheless, the reciprocal is also valid, namely nearest neighbor decision may still be the correct one, as indicated in Fig. 5 (classification results given in Figs. 4 and 5 were obtained in the same experiment).

The dependence of the performances on the dimension of the PCA subspace is not always clear, although higher dimensions seem to be favored. Interesting enough, some of the PCA-based results improve over the non-compressed versions for both Euclidean and TD distances.

The USPS database has been extensively used for comparing the performances of various pattern recognition approaches. Ref. [18] lists more than 40 distinct solutions, with recognition rates varying from 16% to $<2\%$. Best results indicated in Table 2 place our invariant associative memory design in the medium range of performance, although many of the superior methods use significantly more training data or even augment it with virtual samples.

3.2. Face recognition

The success of TD-oriented pattern recognition applications critically depends on how the tangent vectors are computed. We may choose between several distinct alternatives [18]: (a) using finite differences; (b) convolve with a smooth kernel function then differentiate. Typical choices include Gaussian smoothing or the Sobel operator (that combines differentiation with a smoothing kernel); (c) smooth the image then use finite differences. While the second approach proved successful for OCR applications [18,27], it is not directly applicable for other classes of images, since the input may not be smooth enough to reliably compute the local tangent vectors. For example, in case of face images we may obtain completely unrealistic effects, as illustrated in Fig. 6. As a consequence, we followed the first approach as in [25], and applied controlled affine transformations in order to generate virtual samples to be further used for computing finite

differences to approximate the tangent vectors. The images given in Fig. 6 exemplify performing (in-plane) rotation and clearly indicate that classical smoothing accounts only for a limited transformation range, while finite differences may cover a broader range. We have considered distinct tangent vectors for left and right rotation.

The performances of the invariant associative memory have been tested on the Olivetti database. It comprises 10 distinct images of 40 persons, and includes variations in pose, light conditions, and expression. Each image has 112×92 pixels. For the Olivetti database we used a training set of five images per person, randomly selected from the available 10, and the rest for the testing phase. We also preprocessed the original images by performing a multiresolution decomposition based on the Discrete Wavelet Transform (DWT) and kept only the low-frequency components (as 32×32 images) for further classification. The resulting images are known as *waveletfaces* [3]. Besides dimensionality reduction, this procedure is also known to yield face expression invariance. Moreover, the smoothing implied by the low-pass filtering may improve the quality of the tangent vectors. After performing DWT, several distinct feature extraction procedures were used: (a) standard PCA (eigenfaces) using Euclidean (L2) measure; (b) PCA using TD measure; (c) PCA+associative memory+L2; (d) PCA+associative memory+TD. Recognition performances (averaged over 10 distinct trials) are given in Table 3, and indicate that the proposed approach matches some of the top solutions reported in the literature. Much similar to the OCR application, tangent distance also yields superior performances when compared to the Euclidean choice, and both improve when used in conjunction with the associative memory framework.

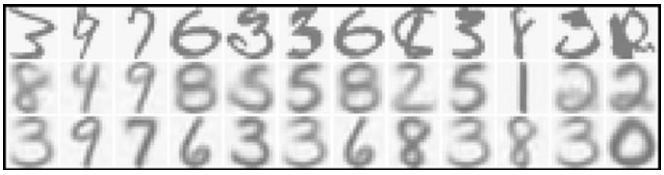

Fig. 4. Associative memory may yield correct class label when nearest neighbor fails to: first row—test images; second row—centroids indicating the corresponding class label given by nearest neighbor rule+TD; third row—class label given by associative memory+TD.

Fig. 5. Nearest neighbor may yield correct class label when associative memory fails to: first row—test images; second row—centroids indicating the corresponding class label given by nearest neighbor rule+TD; third row—class label given by associative memory+TD.

4. Generalization of the basic approach

The proposed technique could be generalized in several ways as follows:

(a) Considering modular approaches, by defining separate memory modules storing distinct types of information extracted from the original data. This information may include distinct patches placed in predefined or variable positions (e.g., given by interest point detectors), distinct resolution features as given by the Discrete Wavelet Transform (DWT), or even distinct binary layers obtained by decomposition of an image with 2^L grey levels into L binary patterns as in [9]. The final classification decision may be based on a simple majority voting scheme, or by employing more sophisticated training procedures. Combining modular local processing with the tangent distance has been considered in [20].

(b) We may store not only (stable) isolated equilibrium points, but also single frequency oscillatory patterns by describing system dynamics in polar coordinates instead of Cartesian ones [7]. Moreover, a particular choice of the amplitude and phases of the stored patterns offers the possibility of memorizing any periodic sequence, based on its Fourier expansion. The approach may be seen as a particular implementation of the "computing with attractors" paradigm.

(c) Considering sequential pattern recognition, for applications that require not only the necessity of storing separate pieces of information in a robust manner, but also the means by which the memory patterns can be sequentially visited, ideally in a predefined order. A possible solution related to the specific associative memory described above has been

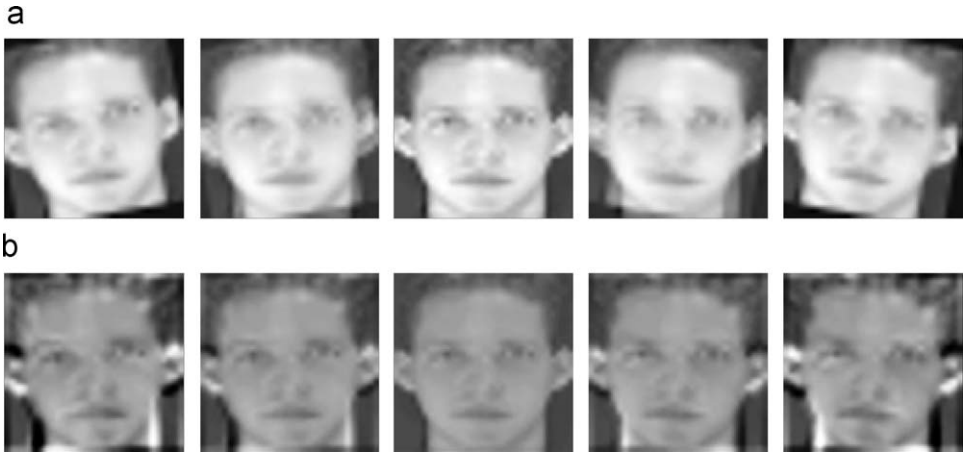

Fig. 6. Computing the tangent vectors: (a) original image I in the center; first and last pictures are rotated images with 10° to the left and to the right ($I+Tang^{left}$, $I+Tang^{right}$); intermediate images are obtained as $I+0.5*Tang^{left}$, $I+0.5*Tang^{right}$ and (b) images obtained using tangent vectors computed by combining smoothing and differentiation.

suggested in [6], based on the use of the tunneling effect proposed in [1] as a global optimization method. Basically, the idea relies on the violation of Lipschitz condition at an equilibrium point of a dynamic system, which enables state trajectories to approach an attractor or escape from a repeller in finite time. In fact, we modify Eq. (1) by introducing an additional energy term which transforms any particular equilibrium $\mathbf{X}^*$ in a *terminal repeller*:

$$\frac{dx_i}{dt} = -\{1 - \Theta[V(\mathbf{X}) - V^*]\}\frac{\partial V(\mathbf{X})}{\partial x_i} + \alpha\Theta[V(\mathbf{X}) - V^*]\frac{\partial}{\partial x_i}(\mathbf{X} - \mathbf{X}^*)^{4/3}, \quad (9)$$

where Θ is the Heaviside step function, V^* is the value of Lyapunov function from Eq. (3) at $\mathbf{X}^*$, and α is a scalar. We make the dynamics of the system *switch* between a tunneling phase and a gradient-descent phase as follows: we select one of the desired memory patterns as vector $\mathbf{X}^*$, and define the initial state of the system $\mathbf{X}_0$ by perturbing $\mathbf{X}^*$ with a small random amount. Since $\mathbf{X}^*$ corresponds to a local minimum of function $V(\mathbf{X})$ we have $V(\mathbf{X}_0) > V(\mathbf{X}^*)$ and, as a consequence, the system enters the tunneling phase: as long as $V(\mathbf{X})$ is higher than V^* the state of the system moves away from $\mathbf{X}^*$. When it reaches a point for which $V(\mathbf{X})$ is less than V^* the system enters the gradient descent phase and stabilizes to a different equilibrium point. In order to enable visiting other (lower $V(\mathbf{X})$ energy) memories, the last local minimum becomes $\mathbf{X}^*$, the new initial state is again a slightly perturbed version of it, and the process reenters the tunneling phase. Eventually, the system state rests on the lowest energy equilibrium. The above procedure is illustrated in Fig. 7. The dynamics of the system evolves as in the example given in Fig. 8, where the patterns to be stored are five images with dimensions 64×64 pixels and 256 grey levels. The weight vector was: $\{w_m\} = \{1,2,3,4,5\}$ in order to force visiting the stored memories in order from the left to the right pattern (the equilibrium points of system (9) have progressively lower energy from first to last pattern). In Fig. 8b we present typical evolution of the combined Lyapunov function of the system, that includes both function $V(\mathbf{X})$ as given in Eq. (2), and the

Table 3
Classification error rates for Olivetti database (%).

Method	Error rate (%)
Eigenfaces [29]	10
Pseudo-2D HMM [26]	5
Convolutional Neural Network [22]	3.8
Linear SVM [19]	3
Waveletface+L2 [3]	7.5
Discriminant Waveletface+L2 [3]	5.5
Discriminant Waveletface+NFL [3]	5
Waveletface+PCA+L2	6.5
Waveletface+PCA+TD	6
Waveletface+PCA+associative memory+L2	5.4
Waveletface+PCA+associative memory+TD	4.9

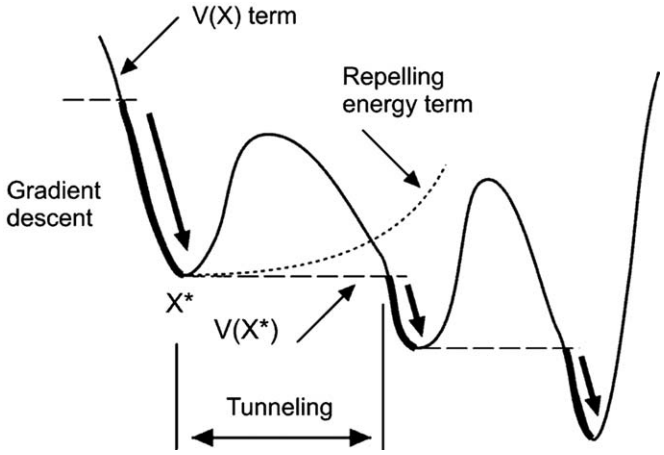

Fig. 7. Searching procedure: the system dynamics switches between a tunneling phase and a gradient descent phase.

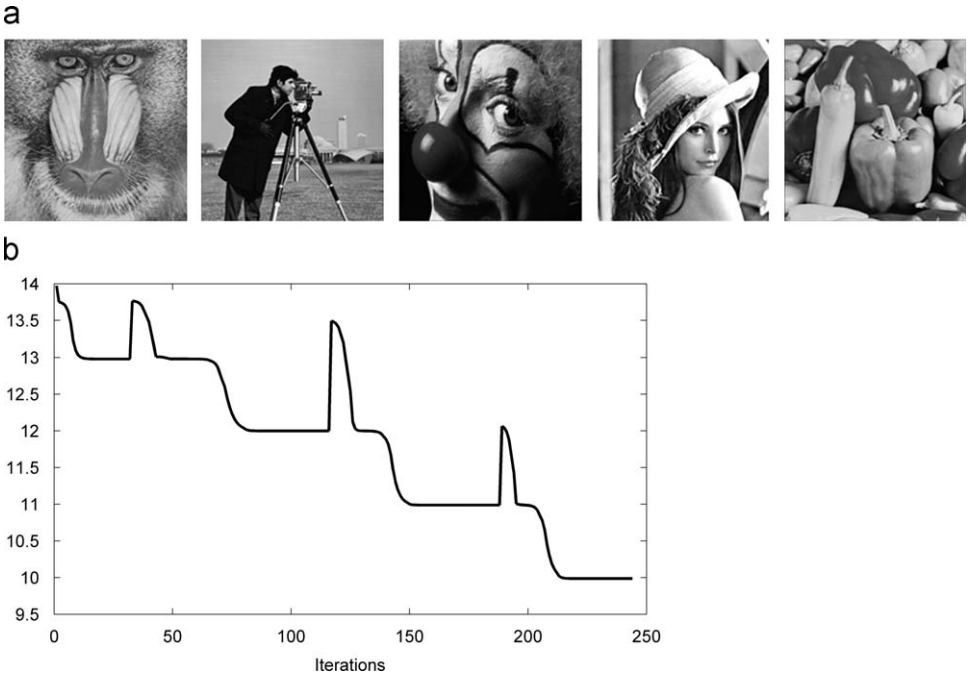

Fig. 8. Sequential search example: (a) memory patterns to be stored; (b) system dynamics evolution.

contribution of the repelling term. The system is initialized with a slightly perturbed version of the first image.

Remarks. (a) In a multidimensional space ($N>1$) a successful search is not always guaranteed. Typically, multiple runs starting from different initial conditions must be performed in order to reach the desired memory. We have successfully tried a (mostly empirical) solution to this problem, by choosing each vector of initial conditions on the direction defined by two consecutive memories to be visited, namely:

$$\mathbf{X}_n(0) = a\mathbf{X}_n^* + (1-a)\mathbf{X}_{n+1}^*, \tag{10}$$

where a is a positive constant, and $\mathbf{X}_n^*, \mathbf{X}_{n+1}^*$ are distinct stable equilibrium points of the gradient system. This is justified by the assumption of constant direction of repelling [1].

(b) This elegant global optimization method has also been used as a supervised learning algorithm for standard feedforward networks, as an enhancement to the well-known backpropagation family [4]. Moreover, sequential pattern recognition has also been demonstrated using non-monotonic neural networks [23].

5. Conclusions

Associative memories may represent an attractive alternative to the classical nearest neighbor classifier, especially in cases of large databases storing high-dimensional prototype patterns, since it avoids explicit computation of the distances between the test patterns and the prototype ones (and subsequent ordering step).

When taking into consideration implementation aspects, it is worth noting that key elements of the architecture (distance computing cells and Gaussian functions) have already been implemented in VLSI structures [34]. Anyway, the presence of any nonlinearities other than the space selective $g_m(\cdot)$ functions should be avoided since they could introduce additional equilibrium points that may degrade the performances of the solution.

The proposed system is a hardwired one, hence it needs no training phase. Further work will consider also learning procedures for adaptively adjusting the position and the shape of the basins of attraction around stored equilibria in order to cope with possible clustered, nonstationary or continuously varying memory distribution.

References

[1] J. Barhen, V. Protopopescu, D. Reister, TRUST: a deterministic algorithm for global optimization, Science 276 (1997) 1096–1097.
[2] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, New York, 1995.
[3] J.-T. Chien, C.-C. Wu, Discriminant waveletfaces and nearest feature classifiers for face recognition, IEEE Transactions on PAMI 24 (2002) 1644–1648.
[4] P.R. Chowdhury, Y.P. Singh, R.A. Chansarkar, Dynamic tunneling technique for efficient training of multilayer perceptrons, IEEE Transactions on Neural Networks 10 (1) (1999) 48–55.
[5] I.B. Ciocoiu, Analog decoding using a gradient-type neural network, IEEE Transactions on Neural Networks 7 (1996) 1034–1038.
[6] I.B. Ciocoiu, Memory search using tunneling effect, Electronics Letters 35 (1999) 820–821.
[7] I.B. Ciocoiu, Dynamic RBF networks, in: R.J. Howlett, L.C. Jain (Eds.), Radial Basis Function Networks 1: Recent Developments in Theory and Applications, Physica-Verlag, Heidelberg, 2001.
[8] I.B. Ciocoiu, Face recognition using recurrent high-order associative memories, in: Proceedings of the ESANN 2004, Bruges, 2004, pp. 567–572.
[9] G. Costantini, D. Casali, R. Perfetti, Neural associative memory storing gray-coded gray-scale images, IEEE Transactions on Neural Networks 14 (3) (2003) 703–707.
[10] B. Haasdonk, H. Burkhardt, Invariant kernels for pattern analysis and machine learning, Machine Learning 68 (2007) 35–61.
[11] J.Y. Han, M.R. Sayeh, J. Zhang, Convergence and limit points of neural networks and its application to pattern recognition, IEEE Transactions on Systems, Man, and Cybernetics 15 (1989) 1217–1222.
[12] T. Hastie, P. Simard, Metrics and models for handwritten character recognition, Statistical Science 13 (1998) 54–65.
[13] S. Haykin, Neural Networks—A Comprehensive Foundation, Prentice-Hall, Englewoods Cliffs, NJ, 1998.
[14] M.W. Hirsch, S. Smale, Differential Equations, Dynamical Systems, and Linear, Algebra, Academic, New York, 1974.
[15] K. Hotta, T. Kurita, T. Mishima, Scale invariant face recognition method using spectral features of log-polar image, in: Proceedings of the SPIE'99, 1999, pp. 33–43.
[16] N. Japkowicz, C. Myers, M. Gluck, A novelty detection approach to classification, in: Proceedings of the Fourteenth Joint Conference on Artificial Intelligence, 1995, pp. 518–523.
[17] D. Keysers, Tangent distance implementation, C code, RWTH Aachen, Germany, ⟨http://www-i6.informatik.rwth-aachen.de/~keysers/td/⟩.
[18] D. Keysers, Modeling of image variability for recognition, Ph.D. Thesis, Aachen, Germany, 2006.
[19] K.I. Kim, K. Jung, H.J. Kim, Face recognition using kernel principal component analysis, IEEE Signal Processing Letters 9 (2002) 40–42.
[20] T. Kölsch, D. Keysers, H. Ney, R. Paredes, Enhancements for local feature based image classification, in: Proceedings of the International Conference on Pattern Recognition, Cambridge, UK, 2004, pp. 248–251.
[21] T. Kurita, K. Hotta, T. Mishima, Scale and rotation invariant recognition method using high-order local autocorrelation features of log-polar image, in: Proceedings of the Asian Conference on Computer Vision, 1998, pp. 89–96.
[22] S. Lawrence, C.L. Giles, A.C. Tsoi, A.D. Back, Face recognition: a convolutional neural network approach, IEEE Transactions on Neural Networks 8 (1997) 98–113.
[23] M. Morita, Associative memory with nonmonotone dynamics, Neural Networks 6 (1993) 115–126.
[24] M.K. Muezzinoglu, J.M. Zurada, RBF-based neurodynamic nearest neighbor classification in real pattern space, Pattern Recognition 39 (2006) 747–760.
[25] A. Pozdnoukhov, S. Bengio, Tangent vector kernels for invariant image classification with SVMs, IDIAP-RR 03-75, 2003.
[26] F.S. Samaria, Face recognition using hidden Markov models, Ph.D. Thesis, University of Cambridge, Cambridge, UK, 1994.
[27] P.Y. Simard, Y.A. Le Cun, J.S. Denker, B. Victorri, Transformation invariance in pattern recognition—tangent distance and tangent propagation, International Journal of Imaging System and Technology 11 (2001) 181–194.
[28] T.F. Smith, J. Wood, Invariant pattern recognition: a review, Pattern Recognition 29 (1996) 1–17.
[29] M. Turk, A.P. Pentland, Eigenfaces for recognition, Journal of Cognitive Neuroscience 3 (1991) 71–86.
[30] USPS database, ⟨ftp://ftp.kyb.tuebingen.mpg.de/pub/bs/data⟩.
[31] A.L. Yuille, P.W. Hallinan, D.S. Cohen, Feature extraction from faces using deformable templates, International Journal of Computer Vision 8 (1992) 99–111.
[32] L. Wang, Y. Zhang, J. Feng, On the Euclidean distance of images, IEEE Transactions on PAMI 27 (2005) 1334–1339.
[33] R.S. Zemel, M.C. Mozer, Localist attractor networks, in: Proceedings of the NIPS 12, MIT Press, 1999.
[34] ZISC036, ⟨www.general-vision.com⟩.

Iulian B. Ciocoiu was born in Miercurea-Ciuc, Romania, in 1963. He received the B.S. degree in electronic engineering from Technical University of Iasi, Romania, in 1988, and the Ph.D. degree in electronic engineering from the same University, in 1996. He published more than 40 papers, two books (in Romanian), and two book chapters. His research interests include artificial neural networks, adaptive filtering, biometric applications.