Nonlinear Dyn (2019) 95:1999–2017
https://doi.org/10.1007/s11071-018-4673-4

ORIGINAL PAPER

A CNN-based neuromorphic model for classification and decision control

Paolo Arena · Marco Calí · Luca Patané · Agnese Portera · Angelo G. Spinosa

Received: 4 August 2017 / Accepted: 21 November 2018 / Published online: 4 December 2018

Abstract In this paper, an insect brain-inspired computational structure was developed. The peculiarity of the core processing layer is the local connectivity among the spiking neurons, which allows for a representation under the cellular nonlinear network paradigm. Moreover, the processing layer works as a liquid state network with fixed internal connections and trainable output weights. Learning was accomplished by adopting a simple supervised, batch approach based on the calculation of the Moore–Penrose matrix. The architecture, taking inspiration from a specific neuropile of the insect brain, the mushroom bodies, is evaluated and compared with other standard and bio-inspired solutions present in the literature, referring to three different scenarios.

Keywords Cellular neural networks · Insect brain · *Drosophila melanogaster* · Neural gas · Mushroom bodies · Classification · Decision-making

P. Arena · M. Calí · L. Patané (✉) · A. Portera · A. G. Spinosa
DIEEI, University of Catania, Viale A. Doria 6, 95100 Catania, Italy
e-mail: lpatane@dieei.unict.it

P. Arena
National Institute of Biostructures and Biosystems (INBB), Viale delle Medaglie d'Oro 305, 00136 Rome, Italy

1 Introduction

Any living being is asked to make decisions on the particular behaviour to be implemented in front of specific environmental stimuli. Good and bad experiences guide the action selection strategy, and this capability is fundamental for survival. Neuromorphic models able to address such capabilities need, on the other side, the availability of suitable circuit paradigms and structures to be efficiently implemented, for example, in autonomous machines asked to perform such important tasks as object recognition and decision-making in structured and unstructured environments. Circuit implementation, in turn, recalls the need to look for simple architectures that should be, at the same time, efficient and performing. For such reason, recently the insect world has been addressed as the one joining these two aspects: insects possess simple brains that, at the same time, are able to show impressive capabilities which are till nowadays unattainable in autonomous machines. In previous works, the authors developed an insect-inspired architecture to model a relevant part of the multitude of interesting behaviours shown by insects and in particular by *Drosophila melanogaster* taken into consideration as a model organism [5,6,8,12]. Following the paradigm of neural reuse [3], different neural capabilities have been ascribed to one of the relevant neuropiles within the insect brain: the mushroom bodies (MBs). These are a multimodal centre, mainly responsible for olfactory

learning [25], and play also a primary role in a variety of different behaviours related to visual tasks (e.g. background foreground separation), adaptive termination of behaviours, mechanosensory processing and others [14]. The inspiration behind the present work derived from the evidence that a decision-making task is based on the preliminary capability to discriminate and categorise, which recalls the task of classification. Indeed, insects are used to learn and discriminate between good and bad experiences, upon which all the experimental learning set-up are based.

In insects, MBs are part of a complex network that involves other neural circuitries such as the antennal lobes, the projection neurons, the lateral horn, the MBs extrinsic neurons, and others. Previous models, present in the literature, have been developed focusing on the dynamic properties of these structures. One of the first models proposed in relation to olfactory learning was introduced in [28]. Here, an associative learning-based structure was introduced, where the combination of different inputs stimulates a reduced number of KCs whose activity was then associated with a rewarding or a punishing event. Further architectures have been subsequently designed as in [43] where spatio-temporal inputs can be codified into the neural structure through spatial patterns. This sparse activity can be associated with classes using reinforcement learning in a structure similar to a support vector machine [31,32].

Despite the glomerular architecture of the MB Kenyon cells [47], axo-axonal connections have been discovered in the insect MB neurons [36]; this opens the way to consider neural processing within MB neurons as modulated by local interactions which shape the neural spatial-temporal dynamics. The presence of local connections in the neural lattice allows a formalisation of the network under the paradigm of the cellular nonlinear networks (CNNs). This characteristic is extremely useful in view of the implementation of the architecture in hardware for real-time applications. CNNs have been deeply studied in the last decades as paradigms for the generation of complex, brain-like dynamics. To enforce the biological relevance, spiking cells have been taken into account in the structure presented in this work, where a CNN-based architecture, containing a computational core based on locally connected spiking neurons, is exploited as a classifier to support decision-making strategies for potential applications in different fields, among which is the autonomous robotic systems. The model developed takes inspiration from the Drosophila MBs. A lot of experiments are currently performed in insects to assess their capabilities in complex classification tasks which involve either simple linear (associative) learning, or positive/negative patterning, which are examples of nonlinearly separable classification tasks.

The proposed work aims to develop a bio-inspired CNN-like neural network essentially devoted to classification tasks. The approach adopted is inspired by the MB architecture and, on the other hand, exploits basic machine learning concepts about supervised and offline (i.e. batch) learning algorithms to develop a network for which both the training and testing steps are easy to assess. More in detail, following the paradigm of liquid state networks [39], the CNN lattice works as a reservoir system, which receives input signals and provides a set of outward synaptic current. These are collected and processed through a set of output weights which are trained to provide the network output. The approach adopted is typically batch: only output weights undergo learning and are determined only after collecting data within a fixed time-dependent observation window. In this way, the training phase does not need iterations. Neural dynamics is strictly dependent on the neuron types used for modelling the internal lattice. The choice of adopting Class I excitable Izhikevich neurons is twofold: from the one hand these neurons constitute an excellent compromise among complexity, biological plausibility and computational costs; on the other hand, Class I excitability allows to encode the strength of input signals in terms of firing rates; thus, the network operates essentially in terms of frequency contents. The richness of internal nonlinear dynamics generated within the CNN lattice is read out and exploited via a massive connection to linear output neurons. This creates a linear weighted sum of the highly nonlinear dynamics generated by the different input stimuli.

The structure is first applied to a traditional nonlinearly separable classification problem (the *Iris* database) at the primary aim to assess the capabilities of the system in solving such class of problems and to experimentally show how this neuromorphic structure, designed to mimic parts of the insect brain, can be efficient also with traditional classification problems. For these reasons, the results obtained are compared with solutions in the state of the art adopting both standard and bio-inspired architectures. To further assess the capabilities of the approach proposed, we have employed other two well-known datasets for additional performance bench-

mark: the diagnostic version of the *Breast Cancer Wisconsin* dataset and the navigation data from the *Wall-Following Robot Navigation* dataset. In particular, the former has been utilised to evaluate the performance of the system with bigger datasets made up of several features and patterns, while the latter may be meant as a basic example of feature-to-action classification, i.e. decision-making process, by which a system can assign a specific motor operation to a certain set of input data coming from sensors.

The paper is organised as follows: Section 2 introduces the use of cellular neural/nonlinear networks (CNNs) for the work proposed in this manuscript, Sect. 3 provides the biological foundations our work is inspired from, Sect. 4 is focused on the description of the computational model employed for classification, Sect. 5 reports the results obtained after carrying out several simulations over different datasets, Sect. 6 is a further explanation of the work done and reported in this manuscript. Comments and comparisons are drawn from the outcomes obtained for the sake of completeness and Sect. 7 concludes the paper.

2 CNNs for neuromorphic modelling

CNNs have been widely recognised as a paradigm for modelling complex dynamics. Moreover, the possibility of addressing hardware implementation closes the loop to real-time working in several scientific and practical scenarios. In its former definition, CNNs were conceived by Chua as arrays of identical, nonlinear and simple computing elements, whose main characteristics were local interaction [23]. Such structures have also been deeply studied from the perspective of their dynamical behaviours, and several important results on their stability, also in front of several kinds of delays, have been introduced [59]. Even if CNNs were primarily applied to real-time image processing, several other application fields have been discovered, ranging from chaotic circuits modelling to nonlinear spatial temporal dynamics, from biological phenomena to robot locomotion control. In particular regarding such aspect, the authors were involved in modelling locomotion control in legged robots, introducing a new type of slow–fast dynamical circuit possessing a neural-like dynamics and able to reproduce a variety of locomotion patterns for multi-legged machines [13]. At the aim of endowing bio-inspired moving robots with perceptual capabilities, attention has been posed on the insect brain, since such structure possesses a not-so-complex brain architecture, being, at the same time, able to show impressive learning, memory and perceptual capabilities. To build an insect brain computational model, a deep scanning of the state of the art as well as focalised neurogenetic experiments has been performed, giving the possibility to investigate the details of the insect neural structures in charge for such behaviours as decision-making and sequence learning. In particular, the attention has been focalised on the mushroom bodies. The main architectural aspects which are to be outlined here are the specific organisation of the axon connectivity, which recalls for a honeycomb-like structure, where axons make local links to neighbours. From here, a really high resemblance to the CNN core paradigm naturally emerges, as predicted by Leon Chua in the 1980s, who hypothesised the main role of CNNs as generalised brain model architectures. In the literature, CNNs have already been used to model part of the insect brain devoted to early-stage olfactory classification [2], where small-world connectivity in CNN structures has been employed to classify olfactory signals, at the aim of building an electronic nose. The small-world connectivity has been included by adding random, long-range connections to the regular CNN grid. This has also shown to increase performance in some applications [41]. In this paper, we have made the opposite choice: while maintaining local connectivity, we have pruned connections in the trainable read-out maps, experimentally showing that the CNN performance has not been degraded, also in front of traditional classification tasks considered as benchmarks. A CNN lattice containing spiking neurons has been considered, randomly connected to an input layer. The structure of the input layer to the CNN has been simplified to focalise attention to the processing stage within the CNN.

3 Biological background

Multimodal learning and memory are primarily important for insect survival. Based on key experiments involving wild-type and mutant flies, several models have been proposed in the literature: analysing the flow of information between the different neural centres, it is possible to identify the classification and learning mechanisms involved in the particular process studied.

Referring to *D. melanogaster*, the main neural structures responsible for generalised learning and memory functions are the mushroom bodies (MBs), for which a rather detailed anatomical and functional model has been derived.

The neurons constituting the input stage for MB processing are the so-called Kenyon Cells (KCs) which contribute to form the Calyx region. From here, KCs run in parallel through the peduncle, until they reach the lobes, which can be distinguished anatomically and functionally into α, β and γ lobes. While α and β lobes are primarily involved in olfactory processing, γ cells are believed to serve to different kinds of sensory modalities. In fact, although there is a prominent olfactory input from the AL into the Calyces, signals coming from other modalities are being discovered to play a role. Very recently, MB involvement into visual processing has been definitely assessed [54].

In particular, a subset of KC, the so-called γ_d neurons, has been electro-physiologically characterised to spike in the presence of visual stimuli and not in response to odours. Similarly to olfactory projection neurons (PNs), visual PNs have been found to arborise within the optic lobe and project to the γ_d KCs. These new findings unequivocally identify the MBs as a unique place in the insect brain for multimodal learning and memory. γ_d neuron axons run in parallel to the other KCs, so we hypothesise that they share the same connection topology and the same way of information processing. In particular, a detailed study of the KC lobes in the locust reveals a honeycomb-like structure, where KC axons share local axo-axonal connections [36]. Moreover, the lobe system in the locust has been found to be cyclically inhibited (with a period of around 50 ms) by the lateral horn (LH) [43]. On the other hand, several LH neurons responding to both visual and olfactory stimuli have been experimentally identified in [30], providing direct evidence for multimodal responses in the LH. Here we are hypothesising that the same, generalised, inhibitory effect of the LH influences also the dynamics of the γ_d neurons. The KC axons run in parallel through the MB lobes, where they are connected to a newly identified set of MB output neurons, whose response is associated with punishment/rewarding events through the activation of dopaminergic/octopaminergic neurons linked to other parts of the brain. This structure is at the basis of a generalised learning and memory model, firstly assessed for olfactory modality [16] and now ready to be extended to vision.

Even if the whole MB neuropile contains a much larger number of neurons and connections than our developed model, nevertheless the recently discovered visual input to the γ_d neurons, their specificity in processing visual stimuli and their reduced number (around 75 in the adult fly [54]) give us the possibility to model a fairly focalised part of the MBs using a reasonable circuit size.

4 MB-inspired computational model for classification

A scheme of the developed MB-inspired computational model is reported in Fig. 1. The architecture consists of three distinct layers: the first one is an input layer that acquires information in terms of general features of the relevant objects present in the scene. We are also assuming here that input neurons have no mutual connections, representing independent stimuli. The activity of the input neurons is projected, through random connections with probability fixed to 25%, to a CNN structure representing the KC layer [46]. This model, with respect to the previous implementations, which reproduced a much wider model for different purposes [4,6], contains local synaptic connections arranged in a regular lattice. Mimicking the inhibitory effect of the LH, the network, upon presentation of an input stimulus, is allowed to evolve for a given time window, after which the CNN structure dynamics is inhibited (reproducing the LH effect) and the system is able to process the new sensory stimuli.

The sparse spiking activity within the CNN lattice is collected by output neurons possessing a linear activation function. In particular, each CNN neuron is connected, through trainable synapses, to the output neurons creating a read-out map of the CNN dynamics. In the case of classification, a number of output neurons equal to the number of classes are used. To handle with a supervised classification task, during the learning phase, a target signal is provided to each output neuron to learn the suitable weights of the read-out map that can best match the internal dynamics of the network with the target signal. The mathematical formalisation of the neural structure and learning mechanisms are reported in the following sections.

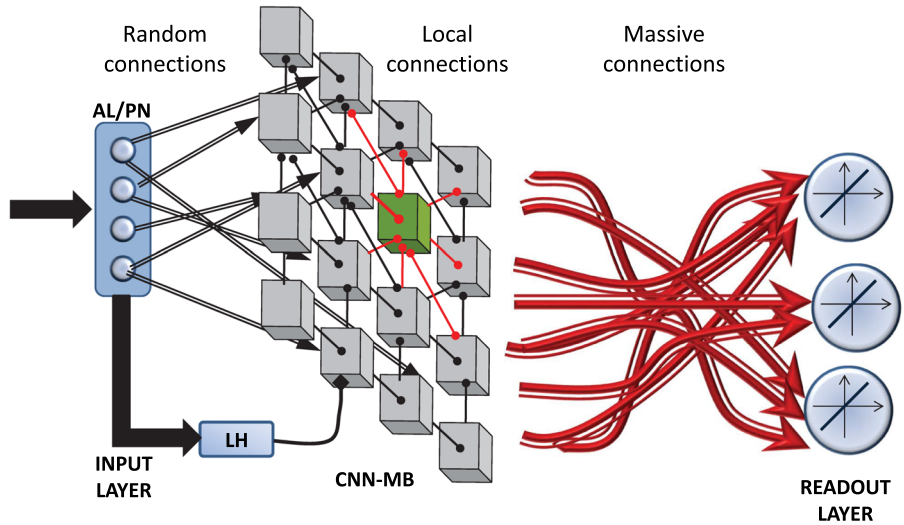

Fig. 1 Block scheme of the MB-inspired developed architecture devoted to classification. The external input is processed by the block AL/PN that randomly excite the CNN-MB made of locally connected spiking neurons; the lateral horn (LH) suppresses the lattice activity after a given time window. The spiking activity is exploited for classification using multiple read-out maps at the output layer that are trained to distinguish the different presented classes

4.1 The CNN-MB structure

The CNN-MB model proposed in this work is the core of a multi-layer architecture used here for classification. The CNN cells are nonlinear models of identical spiking neurons [8]. Considering that memory-relevant MB neurons are excitatory (cholinergic) [20], the CNNs connections here consisted of a percentage of 75% excitatory and 25% inhibitory neurons. Connectivity among the CNN cells implements a regular lattice with radius one, by means of fixed synaptic weights (w) randomly distributed between -0.5 and 0.5. The weights connecting the input neurons to the CNN are fixed to 1 and subject to the 25% connection probability outlined above. Finally, the output weights, representing the read-out map, are the only parameters subject to learning.

The CNN cell here proposed includes both the spiking neuron dynamics and the synaptic activity which transforms the generated spikes into an input current for the connected post-synaptic cells. The synaptic weights are embedded into the feedback template values (template A). The neuron model used in this architecture is the Izhikevich Class I spiking neuron proposed by [33]. This is well known in the literature and offers many advantages from the computational point of view. The following equations represent the dynamics of a single CNN neuron (i.e. cell i, j) in the lattice.

$$\begin{cases} \dot{x}_1 = -x_1 + f_1(x_1, x_2) \\ \dot{x}_2 = -x_2 + f_2(x_1, x_2) \end{cases} \tag{1}$$

where

$$\begin{aligned} f_1(x_1, x_2) &= \left(0.04x_1^2 + 6x_1 + 154 - x_2\right)\left(1 - H\left(x_1 - x_1^{\text{th}}\right)\right) \\ &\quad + x_1 H\left(x_1 - x_1^{\text{th}}\right) + \left(x_1 - x_1^{ref}\right)\delta\left(x_1 - x_1^{\text{th}}\right) \\ f_2(x_1, x_2) &= (-0.002x_1 + 0.98x_2)\left(1 - H\left(x_1 - x_1^{\text{th}}\right)\right) \\ &\quad + x_2 H\left(x_1 - x_1^{\text{th}}\right) + x_2^{\text{ref}}\delta\left(x_1 - x_1^{th}\right) \end{aligned} \tag{2}$$

$H(\cdot)$ is the Heaviside function and $\delta(\cdot)$ is the Dirac impulse; $x_1^{\text{th}} = 30\,\text{mV}$ is the threshold upon which a spike is generated; $x_1^{\text{ref}} = -55\,\text{mV}$ and $x_2^{\text{ref}} = 6\,\text{mV}$ are resetting values for the variables after the spike generation.

When the single cell is introduced within the CNN lattice, the formulation of the whole neural network under the CNN paradigm reads [23]:

$$\dot{\mathbf{x}}_{ij} = -\mathbf{x}_{ij} + \mathbf{f}_{ij}(\mathbf{x}_{ij}) + A * \mathbf{y}_{ij} + \sum_{k=1}^{N} B_k * \mathbf{u}_{k,ij} \tag{3}$$

where

$$A = \begin{bmatrix} A_{11} & 0 \\ 0 & 0 \end{bmatrix};$$

$$A_{11} = \begin{bmatrix} w_{i,j;i-1,j-1} & w_{i,j;i,j-1} & w_{i,j;i+1,j-1} \\ w_{i,j;i-1,j} & 0 & w_{i,j;i+1,j} \\ w_{i,j;i-1,j+1} & w_{i,j;i,j+1} & w_{i,j;i+1,j+1} \end{bmatrix}; \tag{4}$$

$$B_k = \begin{bmatrix} B_{k,11} & 0 \\ 0 & 0 \end{bmatrix}; \quad B_{k,11} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & p_{k,ij} & 0 \\ 0 & 0 & 0 \end{bmatrix};$$

$$\mathbf{y}_{ij} = \begin{bmatrix} y_{1,ij}(t) \\ 0 \end{bmatrix}$$

$$y_{1,ij}(t) = \sum_{l \in S_{ij}} 10 \frac{t_l}{\tau} e^{\left[1 - \frac{(t-t_l)}{\tau}\right]} H(t - t_l) \tag{5}$$

Here $u_{k,ij}$ represents the contribution of the exogenous input current k to the cell (ij); N is the number of different inputs; $y_{1,ij}(t)$ is the cell output which represents the synaptic response at time t to all the set S_{ij} of spike events, each one occurring at time t_l for the cell (ij); $\tau = 20$ ms is the synaptic time constant. The output of the second state variable of the cell is not used. The term $p_{k,ij} = \{0; 1\}$ in the template $B_{k,11}$ is a Boolean value depending on the connection probability between the input and the lattice. It can be modelled in terms of a salt-and-pepper noise on the central element of the input template, $p_{k,11} = H(rand - 0.75)$, where $rand$ is a number extracted from a uniform random distribution in the interval [0 1]; therefore, the connection probability is set to 25%. The equations define the dynamics of a 2-layer, N-inputs nonlinear CNN. In fact, we have, in this case, N independent input channels to each single cell, which influence the cell dynamics according to a probabilistic rule defined by $p_{k,ij}$. The number of outputs matches the class number, and each output neuron is fully connected to the CNN cell outputs, whereas the activation function $O_m(t)$ for the mth output neuron is a simple linear summation of its weighted inputs over the simulation time of the CNN-MB network:

$$O_m(t) = \sum_{i=1}^{N_r} \sum_{j=1}^{N_c} y_{1,ij}(t) w_{ij}^{\text{out},m} \tag{6}$$

$y_{1,ij}(t)$ being the output of the CNN-MB cell $C(ij)$. N_r and N_c represent the lattice dimension (row and column numbers, respectively), and $m = [1 \ldots c]$, where c corresponds to the number of the different classes.

The synaptic outputs $w_{ij}^{\text{out},m}$ from the CNN-MB to the output neurons are trainable. Most biologically inspired learning algorithms are not truly supervised: learning in those cases is guided by reward/punishment mechanisms. Since our purpose is to use this network as a supervised classifier, the following formulation employs the Moore–Penrose pseudo-inverse algorithm [19], which provides better results over other incremental learning mechanisms. Let P_l and P_t be the number of learning and test patterns, respectively. Every time an input pattern is presented, the network executes t_s simulation steps, before the presentation of the next pattern. Once all the learning patterns have been presented, the $P_l \cdot t_s$ data are collected for each CNN neuron C_{ij}. All these vectors are arranged column-wise to build the matrix Z, which possesses $P_l \cdot t_s$ rows and $N_r \cdot N_c$ columns. Z thus contains the samples of activity of each neuron of the lattice in the considered time window $[1 \cdots t_s]$ for all the learning patterns.

Optimal weight values, for the c different read-out maps, are then obtained through the relation: $W_{m,\text{opt}} = Z^+ T_m$, where T_m is the vector of the desired output values for the mth output neuron and Z^+ is the pseudo-inverse matrix of Z.

The different steps of the learning procedure can be therefore summarised as follows:

(a) provide to the network the part of the dataset chosen for the learning phase and collect in the columns of the matrix Z the samples resulting from the evolution of the CNN over the assigned time window;
(b) apply the pseudo-inverse algorithm to compute the weights W_{opt} for each set of synapses between the CNN-MB and each output neuron, having preliminary defined a proper target T_m for each specific output neuron;
(c) using W_{opt}, perform a testing phase with the remaining part of the dataset.

4.2 Classification performance: R_K coefficient

To evaluate the performance using a standard index and to make comparisons with other models, the Gorodkin's R_K correlation coefficient for discrete multivariate data has been exploited [29]. The goal of this index is to evaluate the prediction made with each class creating a confusion matrix Q. $q_{k,k}$ contains the number of correctly predicted instances of class k in the main diagonal, and off-diagonal elements in Q contain the number of falsely predicted instances (e.g. $q_{i,k}$ contains the number of instances predicted to belong to class k, but actually belonging to class i).

The K-category correlation coefficient is computed as in Eq. 7:

$$R_K = \frac{\sum_{klh} \{q_{k,k} q_{l,h} - q_{k,l} q_{h,k}\}}{\sqrt{\sum_k (\sum_l q_{k,l})(\sum_{l',k' \neq k} q_{k',l'})} \sqrt{\sum_k (\sum_l q_{l,k})(\sum_{l',k' \neq k} q_{l',k'})}} \tag{7}$$

where the subscripts l and h represent the rows and columns of the Q matrix. Compared with other fre-

quently used performance measures like the success ratio, R_K is more sensitive to small performance differences when overall performance is already high and thus better suited for benchmarking. In addition, R_K is corrected for the bias introduced by skewed class proportions.

There are multiple ways for evaluating the goodness of a classifier choosing a metrics that can highlight specific aspects. Nevertheless, R_K is a good compromise among discriminancy, consistency and coherent behaviours with varying number of classes, unbalanced datasets and randomisation [34].

4.3 Input coding

A very important aspect taken into account while using a dataset is the data input coding into the neural system. A normalisation is needed to allow the network to discriminate the differences between entries: contrarily to [51], we preferred to avoid complex pre-filtering techniques to outline the classification capabilities of the structure. As known, the number of features within a given dataset matches the size of the input layer. The input neurons are identical, but their input currents are related to the entry of the dataset. In particular, the feature values of the dataset are introduced as a bias to the default current value that in our case is 35 μA, in the range of ± 50% for each neuron of the input layer. In detail, for each of the four features $k = 1, \ldots, 4$ in the dataset, the minimum min_k and maximum value max_k are computed and all the datasets are normalised obtaining:

$$f_k = \frac{f_{\text{original}} - \min_k}{\max_k - \min_k} - 0.5 \tag{8}$$

Therefore, the effect of the input, u_k, as reported in Eq. (3) corresponds to:

$$u_k = 35(f_k + 1) \tag{9}$$

It is easy to deduce that for those patterns producing a strong input current the resulting activity will be more pronounced. This justifies the usage of Class I excitable neurons, because discrimination is due to the capability shown by the network of eliciting a widespread variety of frequency contents. The normalisation procedure is designed so as to obtain a suitable averaged spiking activity for the neurons in the hidden layer.

4.4 Output coding

Target signals have been chosen considering the network dynamics. Unlike traditional neural network supervised classifiers, here the output network activity, starting from zero initial conditions, is a weighted combination, through exponential functions (Eq. 5), of the neuron spiking activity in a given time window. Since a classifier should respond with an enhanced activity only for the output neuron m corresponding to the class of the current network input, instead of using a constant target function as commonly chosen in other classifiers, we adopted different signals. These have to enhance, within the processing time window, the output response only for the output neuron corresponding to a specific class and to inhibit all the other outputs. A suitable signal able to solve this task consists of an exponential function. In particular, if an enhanced response is desired (i.e. for the class corresponding to the currently injected input), it has to reach the maximum value in the time window of 80 ms (corresponding to 1000 integration steps with an integration step $dt = 0.08$ ms): this is simply obtained using a short time constant $\tau_t = 8$ ms. On the contrary, the target signals are confined to low values using a larger time constant $\tau_t = 800$ ms. The target function is generated accordingly:

$$T(t) = 1 - e^{\frac{-t}{\tau_t}} \tag{10}$$

During the testing phase, all the patterns not provided in learning have been fed to the network and, without performing weight updates, the winning neuron in the output layer is evaluated using a winner-takes-all approach evaluated on the average value of the output signals. If the learning phase has been suitably performed, the neuron with the highest output value will correspond with the one associated with the right class. As introduced above, the calculation of the pseudo-inverse matrix could be time-consuming. A simple method to improve the computational efficiency consists of enhancing the difference between target signals in order to emphasise the discrimination among the winning class and all the others (e.g. in our set-up, we adopted a 1–100 factor for the time constant).

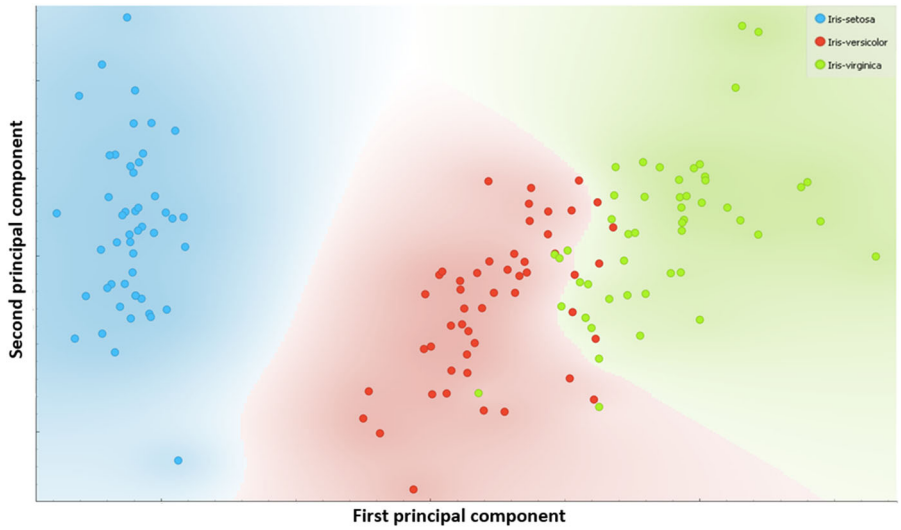

Fig. 2 First two components of the PCA for the Iris dataset

5 Simulation results

In this section, three different datasets, publicly available in the literature, have been reported and employed to evaluate the performance of the proposed bio-inspired structure. Each dataset has been characterised by particular analyses that have been performed to outline peculiar characteristics of the structure introduced. Comparisons with other approaches have also been reported. For all the examples, the network introduced has shown performance which places on the average among the other approaches. This is in line with the aim of showing that a structure inspired to the insect brain can also cope with classical problems, not being aimed to act as the best classifier.

5.1 Case study I: Processing of the *Iris* dataset

One of the most used datasets in the literature for computing classification and machine learning performance indices is the *Iris Flower Dataset* [26]. This has also been used in assessing the classification capabilities in different spiking neural networks, in particular to assess the robustness of classification in front of perturbations of input signals [57]. This multivariate dataset was proposed by Ronald Fisher in 1936 to quantitatively investigate the morphology variation in flowers. In detail, it consists of 50 samples for each of the three species of Iris flowers: Iris Setosa, Iris Virginica and Iris Versicolor. Four features have been measured for each sample: the length and the width of the sepals and petals, in centimetres. In the proposed simulations, the dataset has been split in a subset of 40 entries, chosen randomly for each species, used during the learning phase and the remaining part (10 entries for each species) used for the test phase to evaluate the performance index. The complete dataset, elaborated using a PCA and depicted in Fig. 2, contains three main clusters, one for each species of flower. It can be observed that for the last two classes, the clusters are very close because many features overlap; therefore, a discrimination between the two species is not easy to be performed with a linear classifier. The main goal is therefore to test the classification capability of the network with this real dataset and extract some parameters for comparison with other classification approaches used as benchmarks.

As already written before, firstly the network has been tested using the Iris dataset and the R_K index has been considered for benchmarking. The CNN cell number in the lattice has been fixed to 8×8; the typical neural activity of the network when stimulated with an entry of the dataset is reported in Fig. 3.

After training, the simulation of the proposed architecture provided a value of $R_K = 0.89$ that is comparable with the $R_K = 0.87$ obtained by the network proposed in [51] and with the result obtained using a standard neural gas algorithm used to find the optimal data representations based on feature vectors [40].

To reduce the computational load, the simulation time window has been reduced from 80 to 32 ms (i.e. from 1000 to 400 steps) without any degradation of

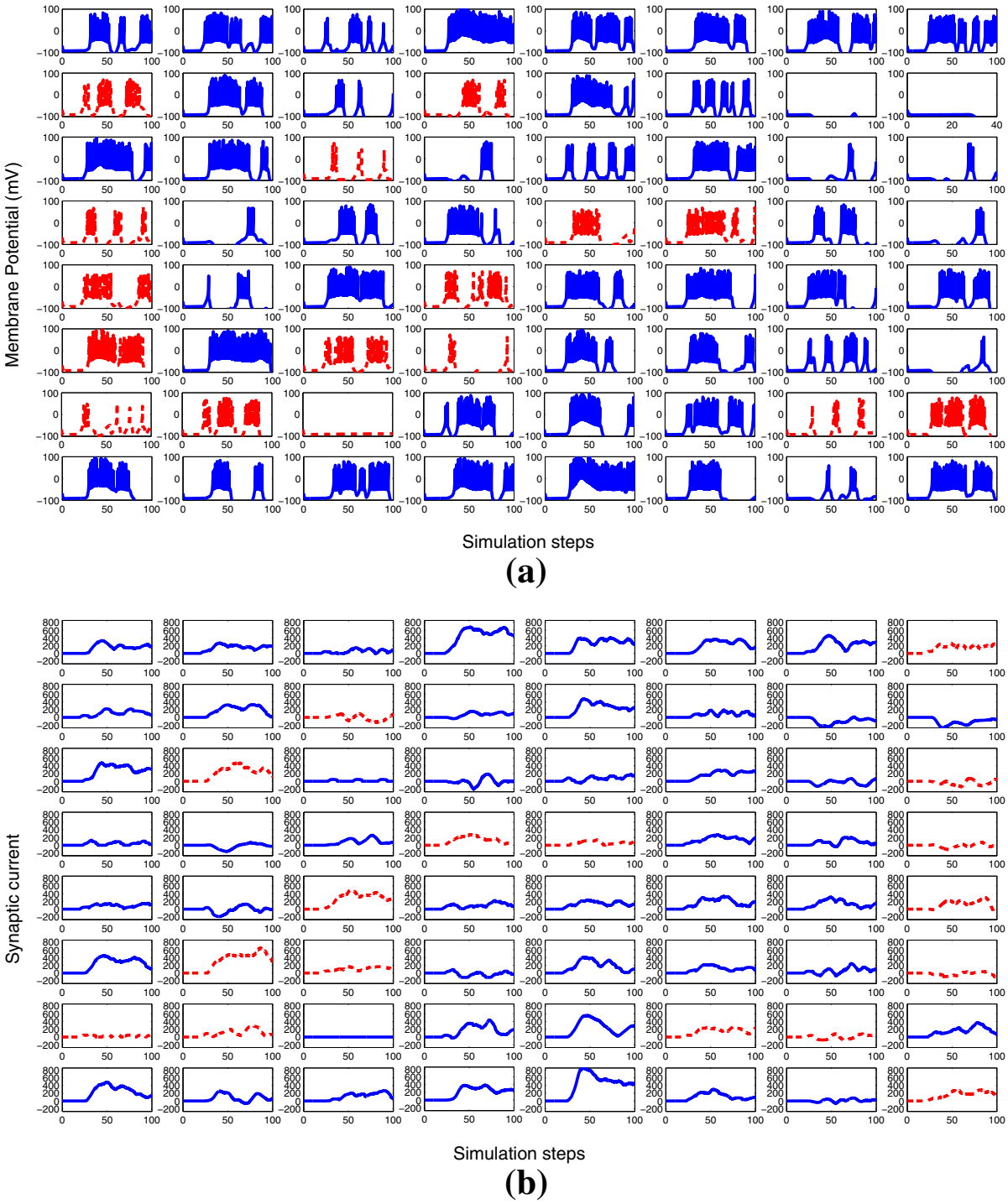

Fig. 3 **a** Membrane potential and **b** synaptic response of the 8 × 8 neurons constituting the CNN lattice; the inhibitory neurons are reported with dashed red lines

the performance in terms of the R_K index. In Fig. 4, the responses of the three output neurons are shown for three different input entries. It is evident that each neuron has a larger response mainly for the entries belonging to that particular class represented by that neuron. The patterns of *Iris Setosa* are better discriminated because this one elicits a strong response only in the first neuron, whereas the patterns of *Iris Versicolor* and *Virginica* sometimes share the response of the last two output neurons.

Taking into account the number of parameters involved in the network, the 8×8 structure proposed has a number of fixed weights connecting the input to the lattice layer equal to $25\%(4 \times 64) \simeq 64$. The number of local connections within the neurons in the lattice is $64 \times 8 = 512$, whereas the number of trainable weights is: $64 \times 3 = 192$. The latter involves the major computational cost, but is far lower than the number of trainable weights requested from the spiking network in [57], which is more than 8000.

A striking aspect we have had to deal with has been the effect of different network sizes on the output performance. Starting from tiny networks, we have evaluated the goodness of classification in terms of R_k index. Firstly, a simple plot describing how the quality of classification has changed with respect to several network sizes, both with and without pruning, is reported in Fig. 5. Pruning consists of removing those trained weights whose contribution is negligible. More specifically, after the pseudo-inverse calculation the mean of the trained weights, except for the outliers, has been computed, and all the weights possessing a value less than a given percentage of the mean are set to zero. The response of the output neurons to the entries of the dataset is then computed using this new set of weights. By simply visually inspecting Fig. 5, it derives that larger networks have produced better results.

Similarly, Table 1 summarises these results quantitatively, from which it is clear that increasing the dimension of the layer leads to a performance improvement, and there is only a slight difference between either with or without pruning. The effects of this algorithm are reported in Table 2, where the first and the last columns report the fixed percentage of the mean weight and the amount of removed synapses, respectively. The neural gas algorithm of course maintains the same trend described before, but reaches the maximum value $R_K = 1$ with less units.

Some further comparisons with respect to other structures drawn from the literature have been made and reported here. Among the various results we have obtained, the best ones have produced a success rate equal to 93.33% of rightly classified patterns. After looking for additional works on the Iris dataset, the final success rates drawn from them are listed in Table 3.

The results have confirmed that the discussed bio-inspired network has been able to reach performance comparable to the existing machine learning architectures available in the literature.

5.2 Case study II: Processing of the *Breast Cancer Wisconsin* dataset

Literature offers several versions of the *Breast Cancer Wisconsin* dataset [17,56]. However, in this work we have utilised the diagnostic one which comprises 569 patterns. As it can be clearly noticed, the dimensionality of such a dataset is bigger, even though the number of output classes is reduced to 2. This dataset has been realised by extracting the aforementioned features from a digitised image of a fine needle aspirate of a breast mass. These features can be used to predict whether some characteristics of the cell nuclei present in the image may lead to the presence of a cancer; then, the possible outcomes may be either *malignant* or *benign* only. Similarly to the previous case study, we carried out a preliminary PCA to show the first two components derived from the dataset. The results are shown in Fig. 6.

This additional dataset has been employed to evaluate the network performance when the dataset is particularly sizeable. To do this, five different learning phases have been performed using the same coding strategy already discussed. The average success rate obtained during the test phase has been equal to 96.84%, which is strongly comparable to those reported in the literature. For the sake of completeness, we have listed some of these results in Table 4, whose details are reported in the referred papers.

5.3 Case study III: Processing of the *Wall-Following Robot Navigation* dataset

This famous dataset [27] comprises 5456 recordings from a navigating SCITOS G5 robot which was

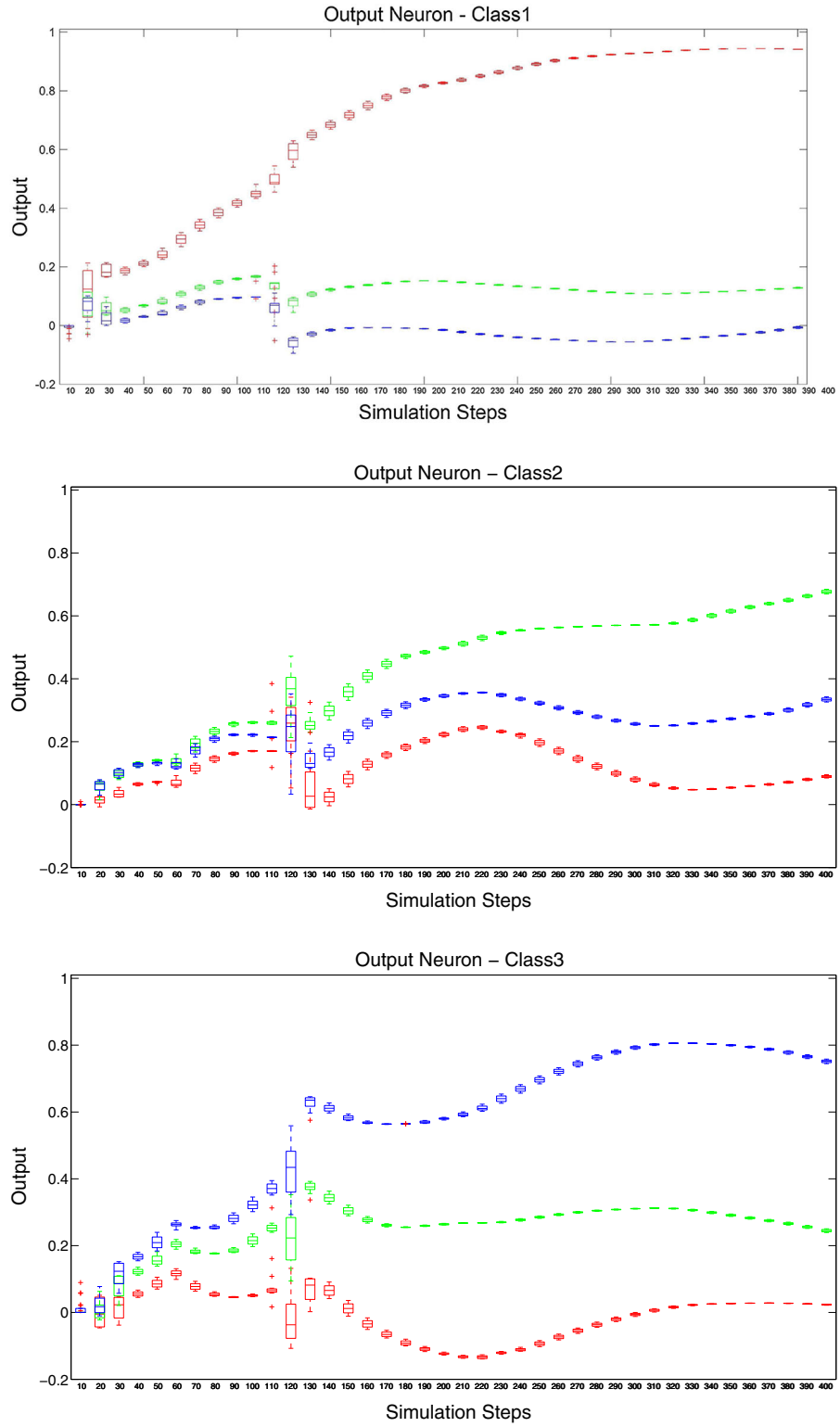

Fig. 4 Representation of the response of the three output neurons when the CNN-MB layer is stimulated by the entries of the Iris Dataset after the learning phase. The signals related to the *Iris Setosa*, the *Iris Versicolor* and the *Iris Virginica* are reported in red, green and blue, respectively. (Color figure online)

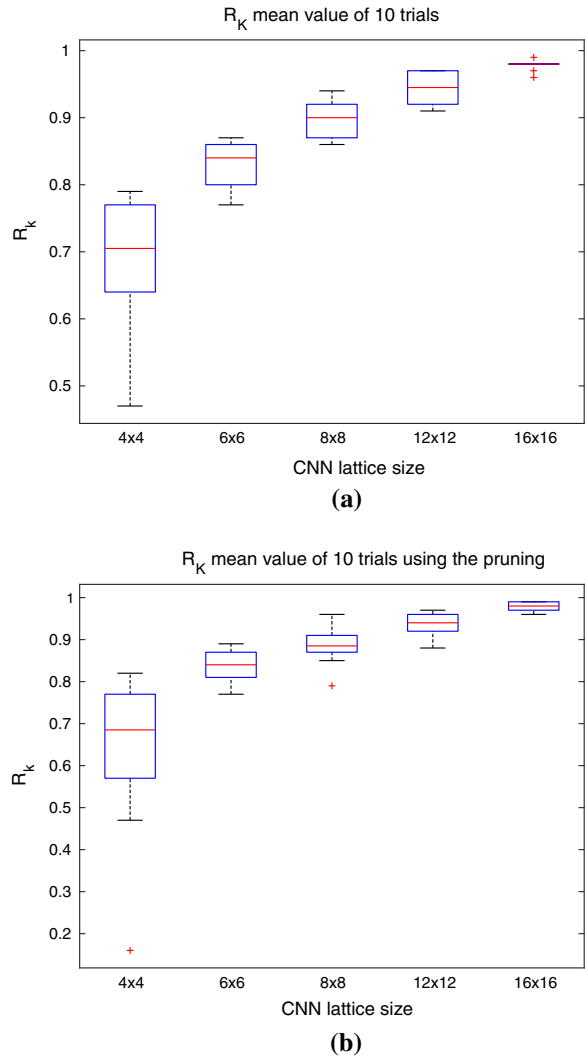

Fig. 5 R_k value for different network configurations without (**a**) and with (**b**) the application of the pruning strategy for the output weights. Within each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, the whiskers extend to the most extreme data points the algorithm considers to be not outliers, and the outliers are depicted individually as '+'

Table 1 The R_K index of the network and comparison

	CNN-MB	With pruning	Neural gas
4×4	0.683	0.632	0.92
6×6	0.830	0.837	0.92
8×8	0.899	0.886	0.97
12×12	0.943	0.939	1
16×16	0.978	0.979	0.99

Table 2 Effect of the pruning mechanisms in terms of the performance index (R_k) in a lattice with 8×8 neurons

Pruning threshold (%)	R_k	Deleted Syn (%)
5	0.89	2.0
10	0.85	5.2
30	0.48	14.4
50	0.26	21.5

For each defined pruning threshold, the performance index is indicated together with the percentage of removed synapses

endowed with 24 ultrasound sensors arranged circularly around its "waist". By falling back on them, the robot ought to have been able to select a proper steering direction. Even in this case, we can exploit different forms of the dataset, depending on how many features we want to use. In fact, there are three available versions that are made up of 2, 4 and 24 features, respectively. In our tests, we have adopted the 4-feature version; the results drawn from it have been compared to

Table 3 Classification results for the Iris dataset drawn from various additional works

	Algorithm	Result (%)
[35]	Self-aware MCS	96.25
[22]	Gaussian-based clustering	96.67
[18]	Real-time spiking neural network	83.4
[50]	STDP-based spiking neural network	83
–	Our method	93.33

other studies available in the literature. The 4-feature version, which is a simplified one of the more general 24-feature version, takes into account the front, left, right and back distances, which consist, respectively, of the minimum sensor readings among those within 60 degree arcs located at the front, left, right and back parts of the robot. According to these recordings, each pattern is associated with one of 4 possible output classes, which are *move-forward*, *slight-right-turn*, *sharp-right-turn* and *slight-left-turn*. Two main characteristics of the dataset are its nonlinear separability and unbalanced classes distribution, which may make the problem of classifying it quite hard to accomplish. Again, the first two components from an exploratory PCA analysis are depicted in Fig. 7.

In this case study, changing the network size has played a relevant as well as in the analyses about *Iris*. To make it clearer, let us consider both the 8×8 and the 16×16 networks undergoing the classification of the *Wall-Following Robot Navigation* dataset. We have observed a deep, drastic change in terms of success rate from the former size to the latter, as well described in Fig. 8. We can look at the improvement by checking the confusion matrices produced by the classifier (Fig. 9). The results we have obtained can be interpreted by taking into consideration the unbalanced nature of the dataset, which favour the classes corresponding to the top, leftmost boxes of each confusion matrix. Additionally, Table 5 shows some results drawn from the literature which have been reported for the sake of comparison. As it can be clearly pointed out, our classification results have been largely comparable to other classical approaches.

6 Discussion

The manuscript reports a structure modelled upon a key part of the insect brain devoted to learning and memory tasks. The new bio-inspired architecture has been proved to be a general-purpose learning/processing core for concurrent association of different families of stimuli implementing a clear example of the neural reuse biological paradigm. The general structure has already been presented in [9]. The key aspects outlined in this updated architecture are local connectivity, which further enhances the potential applications in autonomous machines requiring on-board hardware. The proposed approach, on the other hand, is well placed inside an active research field on bio-inspired solutions for learning systems.

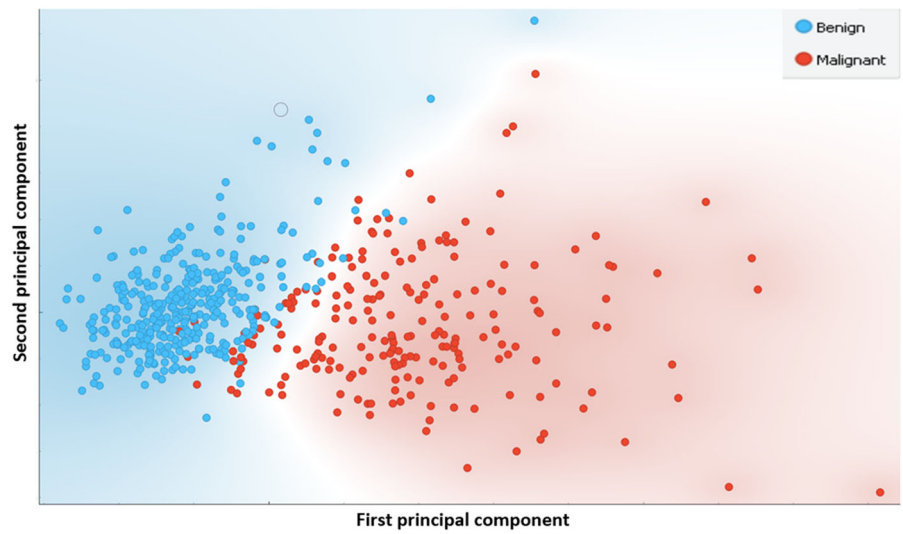

Fig. 6 First two components of the PCA for the diagnostic *Breast Cancer Wisconsin* dataset

Table 4 Classification results for the *Breast Cancer Wisconsin* dataset drawn from various additional works

	Algorithm	Result (%)
[1]	SVM	96.09
	GRU-SVM	93.75
	Linear regression	96.09
	Softmax regression	97.66
	MLP	99.04
	L1-NN	93.57
	L2-NN	94.74
[48]	Naive Bayes	92.97
	MLP	96.66
	J48	93.15
	SMO	97.72
	IBK	95.96
	SMO + Naive Bayes	97.54
	SMO + MLP	97.72
	SMO + J48	94.90
	SMO + IBK	97.72
	SMO + Naive Bayes + IBK	97.36
	SMO + MLP + IBK	97.16
	SMO + J48 + IBK	97.36
	SMO + Naive Bayes + IBK + MLP	97.54
	SMO + MLP + IBK + J48	97.01
[58]	Radial Basis Function Network	90.18
	Generalised Regression Neural Network	94.64
	Feed Forward Neural Network	91.07
[49]	Naive Bayes	71.68
	kNN	72.38
	SMO	69.58
	Bagging bootstrap	68.88
	Decision tree	73.43
	C4.5	75.52
	ADTree	73.78
	REPTree	70.63
	DNTB	71.68

6.1 Comparisons

One of the most recent works on this topic, and the one used as a benchmark in our paper, addresses the clustering capabilities of an MB-inspired network, proposing a spiking neural network for the classification of multivariate data [51]. In this model, data are converted into spike trains using a set of virtual receptors, each one representing a vector whose coordinates are chosen according to a preliminary clustering phase performed via the neural gas algorithm. Their output is processed by lateral inhibition and drives a winner-take-all circuit that supports supervised learning. The network is strongly bio-inspired and complete from a computational perspective. The original idea of authors was to create a significant biologically grounded classifier and compare it with a naive Bayes classifier. The authors implemented an online optimisation method for classification once the network has been trained. Structurally, the network is composed of three layers, which are functionally equivalent to the layers of our network. The main difference lied in the learning algorithm, since the authors adopted online methods: whenever a class is correctly predicted some connections are created. Even if the network can work on multiple datasets, it is not conceived as a multi-task system and the number of variables to be handled by designers is surprisingly higher than ours. This surely makes the complexity of the entire model more pronounced. In our approach, the CNN-based MB model has been tested to face with the same problem without needing any clustering technique to pre-process the input data. Moreover, it is envisaged that the use of the neural gas algorithm largely helps the successive clustering performance of the overall approach and further increases the number of parameters involved. In this paper, the neural gas algorithm has been chosen for the primary reason of comparing our results with those ones presented in [52].

Classical nonspiking distributed structures have also been adopted to model classification mechanisms in biological structures [52]. The proposed network was a self-organised map (SOM), whose organisation corresponds to the arrangement of glomeruli in the antennal lobe of the fruit fly. A winner-takes-all strategy was adopted whenever two glomeruli responded similarly to a set of inputs. As happens in our model, classification performance is affected by the SOM size: changing the dimensionality of the SOM, the results do not change significantly. Furthermore, a supervised classifier was presented in [21] for cellular images. The main novelty was the introduction of what they defined the 2D difference of Gaussians, whose qualitative behaviour is similar to that found in ganglion cells when they are stimulated by local changes in terms of luminance contrast intensities. Despite the good results, the model is not a proper neural network, but a chain of

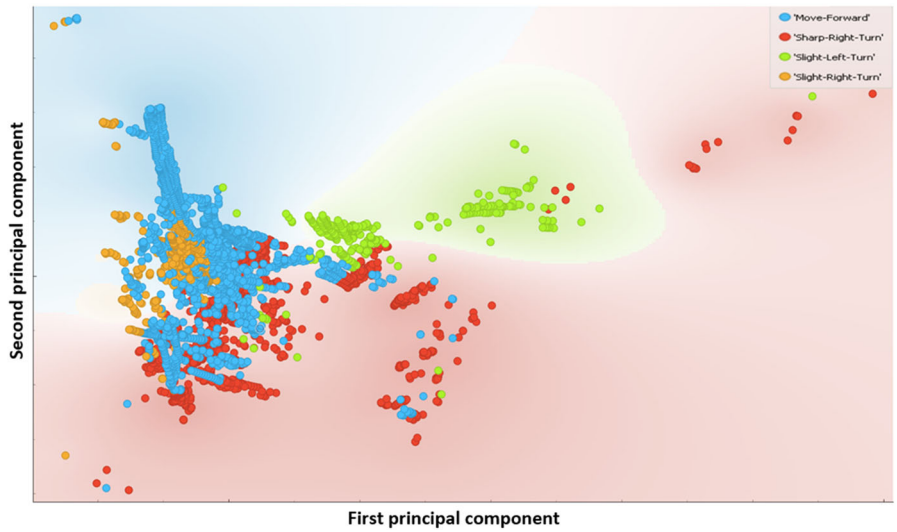

Fig. 7 First two components of the PCA for the *Wall-Following Robot Navigation* dataset

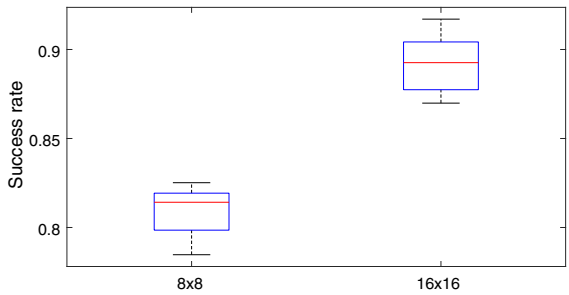

Fig. 8 Analysis of the network size for the *Wall-Following Robot Navigation* dataset classification session

functional blocks with bio-inspired capabilities. The two approaches discussed above, though retaining a high degree of biological inspiration, faced with the classification task with particular emphasis to sensory modelling and processing. Our approach was instead mostly focused on modelling a high-level insect brain centre, obtaining a general-purpose dynamical mapping network able to solve several tasks, including classification, concurrently.

6.2 On the use of offline optimisation algorithms

As a note on the capability of our bio-inspired classification method, under the hypothesis that all the non-linear complexity is faced with by the internal liquid lattice, and therefore only a linear read-out mapping is needed, the Moore–Penrose matrix provides the optimal solution. Obtaining such solution depends on the network design as well as on the quality of input data. The read-out weight matrix learning is performed in batch, after all the data have been collected. Batch methods are surely cheaper and faster, but they have also some drawbacks. First, the method we used works fine for relatively small networks, where the number of neurons in the lattice is limited. The reason why the Moore–Penrose matrix-based approach could be not suitable for bigger networks is due to the size dependency of the algorithm for computing the pseudo-inverse matrix. To extend the proposed approach to large-scale problems with continuous updates in the dataset, it is important to consider different solutions that take into account incremental learning methods. Most of the available solutions are based on the least mean square (LMS) approach, and a possible application for the proposed architecture was presented in [9], where the superiority of batch over recursive LMS was appreciated. Furthermore, it has to be noticed that either in Echo State Networks or Liquid State machines, read-out weight training is equivalent to a linear regression problem. In this case, LMS outperforms algorithms like gradient descent, where noise and initial conditions can produce problems of trapping into local minima, which need additional computations to be faced with. Additionally, it has been shown how pseudo-inverse-based algorithms ought to be preferred to backpropagation-based ones when there is no need to tune hyperparameters or if convergence issues may affect the outcoming performance [55]. It has been also observed that gradient descent is not particularly sug-

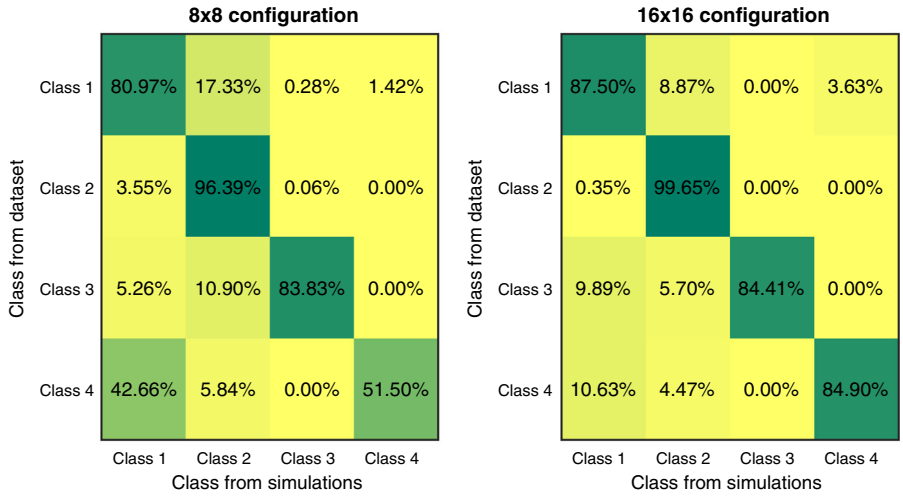

Fig. 9 Example of performance change from a 8×8 network to a 16×16 network after classifying the *Wall-Following Robot Navigation* dataset

Table 5 Classification results for the *Wall-Following Robot Navigation* dataset drawn from various additional works

	Algorithm	Result (%)
[24]	MLP	81.32
[42]	Associative model	96.38
[44]	Elman network	83.12
–	Our method	89.12

gested whenever a large-scale problem must be solved, while its stochastic variant cannot provide high precision [45]. When applying reservoir architectures to large-scale classification problems, an important issue is to reduce the computational time needed for each iteration, for instance limiting the time evolution of the neuron dynamics used for each input pattern, finding a good compromise between computational time and classification performance.

Other learning paradigms for spiking networks have been proposed, inspired to the classical Hebbian rule, like spike timing-dependent plasticity and, more recently, input timing-dependent plasticity (ITDP) [38, 53]. This involves a synaptic plasticity induced by correlations between two pre-synaptic pathways (instead of pre- and post-synaptic paths, like STDP), and it has been found in some cortex/hippocampal regions. Our approach avoids the use of learning within the CNN spiking lattice, in order to exploit the rich nonlinear dynamics generated therein by fixed and local connections and realise simply trainable linear read-out maps which act as linear projection, on to the output feature, of the multidimensional space generated within the lattice. With this solution, we avoided further trainable recurrent weights within the CNN lattice. Our training is simply a one-step setting of the weights. This avoids iterations, at the expenses of requiring all the learning dataset available. Another important aspect that should be highlighted is the generality of our approach: the proposed system was not specifically designed to work as a classifier: classification is only one of the plethora of different tasks this network is able to face with (accordingly to neural reuse theory), as previously introduced.

A final remark concerns the pruning process: although it has been applied to the read-out maps, it may be applied also for internal connections among neurons belonging to the lattice. In fact, it would be interesting to analyse the contribution of a given connectivity scheme to the overall performance metrics, discriminating minimal connectomes from more dense organisations. This challenging study could also reveal some important information about the reasons why some natural highly ordered structures, like the honeycomb-like axo-axonal Kenyon Cells connections in locusts, appear [37].

7 Conclusions

In this work, we propose a spiking neural structure inspired by the insect MBs that is able to perform classification and decision-making tasks, obtaining results comparable with other standard methods, in front of traditional benchmarking tests. The key aspect of the pro-

posed architecture is the presence of local connectivity between neurons which allows a direct formalisation under the CNN paradigm, easy to be implemented in a hardware framework. The dynamics generated within the lattice is used to combine the information coming from a sensing layer, to produce a neural activity which is collected from a series of read-out maps working in parallel. These, converging to the output neurons, extract the needed information to provide the network answer.

The whole insect MBs neuropile contains a larger number of neurons and connections than the developed computational model. It is, in many cases, still beyond the actual neurobiological knowledge to identify which neuron in the MBs is responsible for what. However, from the up-to-date neurobiological knowledge, the known information about topological connections among the different areas involved in the considered processes (e.g. the Antennal lobes and the Lateral Horn) and their functional role was taken into account to develop a basic model.

The performance of the system was first evaluated on the classical *Iris* database: different network configurations have been tested to select the most suitable number of neurons. Here the classical mean square-based algorithm has been adopted for generating the optimal weight values. Additionally, the network has been requested to classify two additional datasets, the diagnostic version of the *Breast Cancer Wisconsin* dataset and the *Wall-Following Robot Navigation* dataset. In particular, the latter has some striking implications for decision-making processes. In every case, our system has been able to manifest performance comparable to other previous studies.

The network here proposed has been made up of a reduced number of neurons. In spite of this, they have been able to show distinct learning skills and behavioural responses as in the biological case of the insect model organism. The research aim is in fact finalised to identify the core of a learning structure, that can be scaled-up in terms of number of neurons and connections for potential applications in complex decision-making tasks in particular on robotic platforms. The structure used here can be integrated within a more complex neuro-computing architecture, inspired by the insect brain and already reported in the literature, able to show other complex tasks and behaviours, like attention, expectation, motor and sequence learning [8,15]. For instance, the different output neurons can be independently trained for different purposes: to classify inputs as well as to provide time-varying signals to control relevant parameters of the locomotion system in case of motor learning [7]. In the field of bio-inspired robotics, CNNs were already exploited by the authors to implement locomotion controllers [10] and perception-oriented machines [11]. The application of CNN-based architectures for classification and decision-making further extends the potentiality of such structures for real-time control of complex robotic systems.

Funding This study was funded by MIUR Project CLARA—Cloud platform for LAndslide Risk Assessment (Grant Number SNC_00451).

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Agarap, A.F.M.: On breast cancer detection: an application of machine learning algorithms on the Wisconsin diagnostic dataset. In: Proceedings of the 2nd International Conference on Machine Learning and Soft Computing, ICMLSC '18, pp. 5–9. ACM, New York (2018). https://doi.org/10.1145/3184066.3184080
2. Aihan, T., Yalcin, M.E.: An application of small-world cellular neural networks on odor classification. Int. J. Bifurc. Chaos **22**(01), 1250013 (2012). https://doi.org/10.1142/S0218127412500137
3. Anderson, M.: Neural reuse: a fundamental organizational principle of the brain. Behav. Brain Sci. **33**(4), 245–266 (2010)
4. Arena, E., Arena, P., Strauss, R., Patané, L.: Motor-skill learning in an insect inspired neuro-computational control system. Front. Neurorobot. **11**, 12 (2017). https://doi.org/10.3389/fnbot.2017.00012
5. Arena, P., Berg, C., Patané, L., Strauss, R., Termini, P.S.: Aninsect brain computational model inspired by *Drosophila melanogaster*: architecture description. In: The 2010 International Joint Conference on Neural Networks (IJCNN), pp. 1–7 (2010). https://doi.org/10.1109/IJCNN.2010.5596974
6. Arena, P., Caccamo, S., Patané, L., Strauss, R.: A computational model for motor learning in insects. In: IJCNN, Dallas, TX, pp. 1349–1356 (2013)
7. Arena, P., Caccamo, S., Patané, L., Strauss, R.: A computational model for motor learning in insects. In: International Joint Conference on Neural Networks (IJCNN), Dallas, TX, Aug 4–9, pp. 1349–1356 (2013)
8. Arena, P., Calí, M., Patané, L., Portera, A., Strauss, R.: Modeling the insect mushroom bodies: application to sequence learning. Neural Netw. **67**, 37–53 (2015)
9. Arena, P., Calí, M., Patané, L., Portera, A., Strauss, R.: A fly-inspired mushroom bodies model for sensory-motor control

through sequence and subsequence learning. Int. J. Neural Syst. **26**(6), 1650035 (2016)
10. Arena, P., Castorina, S.M., Frasca, L.F., Ruta, M.: A CNN-based chip for robot locomotion control. ISCAS **3**, 510–513 (2003)
11. Arena, P., Crucitti, P., Fortuna, L., Frasca, M., Lombardo, D., Patané, L.: Turing patterns in RD-CNNs for the emergence of perceptual states in roving robots. Bifurc. Chaos **17**(1), 107–127 (2007)
12. Arena, P., Fiore, S.D., Patané, L., Pollino, M., Ventura, C.: Insect inspired unsupervised learning for tactic and phobic behavior enhancement in a hybrid robot. In: The 2010 International Joint Conference on Neural Networks (IJCNN), pp. 1–8 (2010). https://doi.org/10.1109/IJCNN.2010.5596542
13. Arena, P., Fortuna, L., Branciforte, M.: Reaction–diffusion CNN algorithms to generate and control artificial locomotion. IEEE Trans. Circuits Syst. I **46**(2), 253–260 (1999)
14. Arena, P., Patané, L.: Spatial Temporal Patterns for Action-Oriented Perception in Roving Robots II: An Insect Brain Computational Model. Cognitive Systems Monographs, vol. 21. Springer, Berlin (2014)
15. Arena, P., Patané, L., Strauss, R.: The insect mushroom bodies: a paradigm of neural reuse. In: ECAL, pp. 765–772. MIT Press, Taormina (2013)
16. Aso, Y., et al.: The neuronal architecture of the mushroom body provides a logic for associative learning. eLife **3**, e04577 (2014). https://doi.org/10.7554/eLife.04577
17. Baek, W., Ignizio, J.P.: Pattern classification via linear programming. Comput. Ind. Eng. **25**(1), 393–396 (1993). https://doi.org/10.1016/0360-8352(93)90304-G
18. Bako, L.: Real-time classification of datasets with hardware embedded neuromorphic neural networks. Brief. Bioinform. **11**(3), 348–363 (2010)
19. Barata, J.C.A., Hussein, M.S.: The Moore–Penrose Pseudoinverse. A Tutorial Review of the Theory. John Hopkins University Press, Baltimore (2013)
20. Barnstedt, O., David, O., Felsenberg, J., Brain, R., Moszynski, J., Talbot, C., Perrat, P., Waddell, S.: Memory-relevant mushroom body output synapses are cholinergic. Neuron **89**(6), 1237–1247 (2017). https://doi.org/10.1016/j.neuron.2016.02.015
21. Bel haj ali, W., Piro, P., Giampaglia, D., Pourcher, T., Bar laud, M.: Biological cells classification using bio-inspired descriptor in a boosting k-NN framework. In: 2012 25th IEEE International Symposium on Computer-Based Med ical Systems (CBMS), pp. 1–6 (2012). https://doi.org/10.1109/CBMS.2012.6266359
22. Chang, H., Astolfi, A.: Gaussian based classification with application to the Iris data set. IFAC Proc. **44**(1), 14271–14276 (2011)
23. Chua, L.O., Roska, T.: The CNN paradigm. IEEE Trans. Circuits Syst. I Fundam. Theory Appl. **40**(3), 147–156 (1993). https://doi.org/10.1109/81.222795
24. Dash, T., Sahu, S.R., Nayak, T., Mishra, G.: Neural network approach to control wall-following robot navigation. In: 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, pp. 1072–1076 (2014). https://doi.org/10.1109/ICACCCT.2014.7019262
25. Davis, R., Han, K.: Neuroanatomy: mushrooming mushroom bodies. Curr. Biol. **6**, 146–148 (1996)
26. Fisher, R.A.: The use of multiple measurement in taxonomic problems. Ann. Eugen. **7**(2), 179–188 (1936). https://doi.org/10.1111/j.1469-1809.1936.tb02137.x
27. Freire, A.L., Barreto, G.A., Veloso, M., Varela, A.T.: Short-term memory mechanisms in neural network learning of robot navigation tasks: a case study. In: 2009 6th Latin American Robotics Symposium (LARS 2009), pp. 1–6 (2009). https://doi.org/10.1109/LARS.2009.5418323
28. Gerber, B., Tanimoto, H., Heisenberg, M.: An engram found? Evaluating the evidence from fruit flies. Curr. Opin. Neurobiol. **14**, 737–744 (2004)
29. Gorodkin, J.: Comparing two k-category assignments by a k-category correlation coefficient. Comput. Biol. Chem. **28**(5–6), 367–374 (2004). https://doi.org/10.1016/j.compbiolchem.2004.09.006
30. Gupta, N., Stopfer, M.: Functional analysis of a higher olfactory center, the lateral horn. J. Neurosci. **32**(24), 8138–8148 (2012). https://doi.org/10.1523/JNEUROSCI.1066-12.2012
31. Huerta, R., Nowotny, T., Garcia-Sanchez, M., Abarbanel, H., Rabinovich, M.: Learning classification in the olfactory system of insects. Neural Comput. **16**(8), 1601–40 (2004)
32. Huerta, R., Vembu, S., Amigó, J.M., Nowotny, T., Elkan, C.: Inhibition in multiclass classification. Neural Comput. **24**(9), 2473–2507 (2012)
33. Izhikevich, E.M.: Which model to use for cortical spiking neurons? IEEE Trans. Neural Netw. **15**(5), 1063–1070 (2004)
34. Jurman, G., Riccadonna, S., Furlanello, C.: A comparison of MCC and CEN error measures in multi-class prediction. PloS ONE **7**(8), e41882 (2012). https://doi.org/10.1371/journal.pone.0041882
35. Kholerdi, H.A., TaheriNejad, N., Jantsch, A.: Enhancement of classification of small data sets using self-awareness—an Iris flower case-study. In: 2018 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1–5 (2018)
36. Leitch, B., Laurent, G.: Gabaergic synapses in the antennal lobe and mushroom body of the locust olfactory system. J. Comp. Neurol. **372**(4), 487–514 (1996). https://doi.org/10.1002/(SICI)1096-9861(19960902)372:4<487::AID-CNE1>3.0.CO;2-0
37. Leitch, B., Laurent, G.: Gabaergic synapses in the antennal lobe and mushroom body of the locust olfactory system. J. Comp. Neurol. **372**(4), 487–514 (1996)
38. Leroy, F., Brann, D.H., Meira, T., Siegelbaum, S.A.: Input-timing-dependent plasticity in the hippocampal CA2 region and its potential role in social memory. Neuron **95**(5), 1089–1102.e5 (2017). https://doi.org/10.1016/j.neuron.2017.07.036
39. Maass, W., Markram, H.: On the computational power of circuits of spiking neurons. J. Comput. Syst. Sci. **69**(4), 593–616 (2004). https://doi.org/10.1016/j.jcss.2004.04.001
40. Martinez, T., Schuten, K.: A neural-gas network learns topologies. Artif. Neural Netw. **1**, 397–402 (1991)
41. Matsumoto, K., Mori, H., Uehara, M.: Fault tolerance in small world cellular neural networks for image processing. In: 21st International Conference on Advanced Information Networking and Applications Workshops, 2007, AINAW '07, vol. 1, pp. 835–839 (2007). https://doi.org/10.1109/AINAW.2007.183

42. Navarro, R., Acevedo, E., Acevedo, A., Martínez, F.: Associative model for solving the wall-following problem. In: Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F., Olvera López, J.A., Boyer, K.L. (eds.) Pattern Recognition, pp. 176–186. Springer, Berlin (2012)
43. Nowotny, T., Rabinovich, M., Huerta, R., Abarbanel, H.: Decoding temporal information through slow lateral excitation in the olfactory system of insects. J. Comput. Neurosci. **15**, 271–281 (2003)
44. Ooi, S.Y., Tan, S.C., Cheah, W.P.: Experimental Study of Elman Network in Temporal Classification, pp. 245–254. Springer, Singapore (2017)
45. Papamakarios, G.: Comparison of Modern Stochastic Optimization Algorithms. University of Edinburgh, Edinburgh (2014)
46. Perez-Orive, J., Mazor, O., Turner, G., Cassenaer, S., Wilson, R., Laurent, G.: Oscillations and sparsening of odor representations in the mushroom body. Science **297**, 359–365 (2002)
47. Sachse, S., Galizia, C.: Role of inhibition for temporal and spatial odor representation in olfactory output neurons: a calcium imaging study. J. Neurophysiol. **87**, 1106–1117 (2002)
48. Salama, G.I., Abdelhalim, M.B., Zeid, M.A.: Experimental comparison of classifiers for breast cancer diagnosis. In: 2012 Seventh International Conference on Computer Engineering Systems (ICCES), pp. 180–185 (2012). https://doi.org/10.1109/ICCES.2012.6408508
49. Sathya, S., Joshi, S., Padmavathi, S.: Classification of breast cancer dataset by different classification algorithms. In: 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 1–4 (2017). https://doi.org/10.1109/ICACCS.2017.8014573
50. Sboev, A., Vlasov, D., Rybka, R., Serenko, A.: Solving a classification task by spiking neurons with STDP and temporal coding. Procedia Comput. Sci. **123**, 494–500 (2018)
51. Schmuker, M., Pfeil, T., Nawrot, M.P.: A neuromorphic network for generic multivariate data classification. Proc. Natl. Acad. Sci. **111**(6), 2081–2086 (2014). https://doi.org/10.1073/pnas.1303053111
52. Schmuker, M., Schneider, G.: Processing and classification of chemical data inspired by insect olfaction. Proc. Natl. Acad. Sci. **104**(51), 20285–20289 (2007). https://doi.org/10.1073/pnas.0705683104
53. Shim, Y., Philippides, A., Staras, K., Husbands, P.: Unsupervised learning in an ensemble of spiking neural networks mediated by ITDP. PLOS Comput. Biol. **12**(10), 1–41 (2016). https://doi.org/10.1371/journal.pcbi.1005137
54. Vogt, K., Aso, Y., Hige, T., Knapek, S., Ichinose, T., Friedrich, A.B., Turner, G.C., Rubin, G.M., Tanimoto, H.: Direct neural pathways convey distinct visual information to Drosophila mushroom bodies. eLife **5**, e14009 (2016). https://doi.org/10.7554/eLife.14009
55. Wang, J., Guo, P., Xin, X.: Review of pseudoinverse learning algorithm for multilayer neural networks and applications. In: Huang, T., Lv, J., Sun, C., Tuzikov, A.V. (eds.) Advances in Neural Networks—ISNN 2018, pp. 99–106. Springer, Cham (2018)
56. Wolberg, W.H., Mangasarian, O.L.: Multisurface method of pattern separation for medical diagnosis applied to breast cytology. Proc. Natl. Acad. Sci. **87**(23), 9193–9196 (1990)
57. Yang, J., Zhang, P., Liu, Y.: Robustness of classification ability of spiking neural networks. Nonlinear Dyn. **82**(1), 723–730 (2015). https://doi.org/10.1007/s11071-015-2190-2
58. Yavuz, E., Eyupoglu, C., Sanver, U., Yazici, R.: An ensemble of neural networks for breast cancer diagnosis. In: 2017 International Conference on Computer Science and Engineering (UBMK), pp. 538–543 (2017). https://doi.org/10.1109/UBMK.2017.8093456
59. Zhou, L.: Global asymptotic stability of cellular neural networks with proportional delays. Nonlinear Dyn. **77**(1), 41–47 (2014). https://doi.org/10.1007/s11071-014-1271-y