



Competition in high dimensional spaces using a sparse approximation of neural fields

Jean-Charles Quinton, Bernard Girau, Mathieu Lefort

► To cite this version:

Jean-Charles Quinton, Bernard Girau, Mathieu Lefort. Competition in high dimensional spaces using a sparse approximation of neural fields. Carlos Hernández, Ricardo Sanz, Jaime Gómez-Ramírez, Leslie S. Smith, Amir Hussain, Antonio Chella, Igor Aleksander. From Brains to Systems: Brain-Inspired Cognitive Systems 2010, Springer, 2011, Advances in Experimental Medicine and Biology, 978-1461401636. inria-00568924

HAL Id: inria-00568924

<https://inria.hal.science/inria-00568924>

Submitted on 24 Feb 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Competition in high dimensional spaces using a sparse approximation of neural fields

Jean-Charles Quinton, Bernard Girau and Mathieu Lefort

Abstract The Continuum Neural Field Theory implements competition within topologically organized neural networks with lateral inhibitory connections. However, due to the polynomial complexity of matrix-based implementations, updating dense representations of the activity becomes computationally intractable when an adaptive resolution or an arbitrary number of input dimensions is required. This paper proposes an alternative to self-organizing maps with a sparse implementation based on Gaussian mixture models, promoting a trade-off in redundancy for higher computational efficiency and alleviating constraints on the underlying substrate. This version reproduces the emergent attentional properties of the original equations, by directly applying them within a continuous approximation of a high dimensional neural field. The model is compatible with preprocessed sensory flows but can also be interfaced with artificial systems. This is particularly important for sensorimotor systems, where decisions and motor actions must be taken and updated in real-time. Preliminary tests are performed on a reactive color tracking application, using spatially distributed color features.

1 Introduction

Most biological systems need to differentiate their environment through interaction in order to undertake adapted behaviors. Adaptation here means that this selection or regulation of behavior must be normative for the living agent to maintain its organization [5, 3]. At any time, the agent must therefore choose from a set of potentialities the ones that will not be detrimental to its survival. From a more continuous perspective, it must influence the dynamics of its coupling with the environment as to bias bifurcations and sustain viable conditions for its ongoing activity. With the

Jean-Charles Quinton, e-mail: quintonj@loria.fr,
INRIA/LORIA Laboratory, Campus Scientifique
B.P. 239, 54506 Vandoeuvre-lès-Nancy Cedex, France

decentralized approach adopted in this paper, the many processes from which the behavior emerges must therefore compete through reciprocal excitations and inhibitions. Whether in an overt way with actions guiding the agent towards meaningful situations and stimuli, or in a covert manner with attention focusing on specific perceptual features, decisions must be taken as to direct the sensorimotor flow.

1.1 Competition in neural fields

The model presented in this paper is derived from research done in the field of computational neuroscience. Considering competition within the brain and interactions between assemblies of neurons, we adopt a distributed approach to cognition and focus on models of the cerebral cortex at a mesoscopic scale. We further commit to a biologically plausible though debated decomposition of the cortex sheet into cortical maps, themselves made of cortical columns [8]. Such considerations are grounded on the laminar structure of the cortex conserved throughout species evolution, on the correlated activities found across the cortical layers [22] and on the apparent topological organization at different scales deduced from cytoarchitectural and functional differences [7, 16].

The generic nature and topology of the cortical circuitry are reflected in dynamic neural field models (DNF) where the evolution of the membrane potential of neurons is described by differential equations. We here focus on a particular kind of DNF implementing dynamic competition, namely the Continuum Neural Field Theory (CNFT) [2]. The lateral connectivity pattern in the CNFT follows a difference of Gaussians profile (DoG), most often further constrained to a Mexican hat profile with local excitation and large-scale inhibition. Under adequate conditions, such neural networks are able to maintain so-called bubbles of activity [33, 2, 29]. These bubbles in fact correspond to spatiotemporally coherent patches of activity that emerge on the neural field, in response to external stimulation. The CNFT leads to robust attentional properties, bubbles tracking and focusing on stimuli despite the presence of noise or distracters [28]. Additionally, the CNFT allows non-linear bifurcations to occur when similar stimuli are presented and has the ability to switch between targets if the bottom-up stimulation disappears or if top-down modulation biases the field activity. To put it briefly, the CNFT implements all the desired properties we may expect from a dynamical competition model.

However, while mathematical analysis was limited to a maximum of two dimensions [29] and most simulations used a discretized version of the continuous equations [28, 10], there is no theoretical barrier that prevents applying the CNFT equation to high dimensional continuous inputs. The idea of the current implementation is therefore to alleviate the constraints imposed by the 2D structure of the physical substrate and simultaneously reduce the computational complexity of the algorithms as to scale the model up. For this purpose, we approximate the overall field activity by Gaussian mixture models instead of considering individual units.

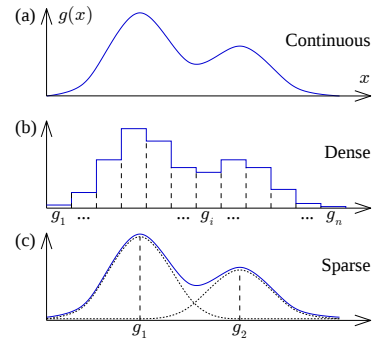
1.2 Sparse modeling

Initially, DNFs were continuous models of the neuronal activity over a manifold, therefore disconnected from any particular implementation. Although matrix-like regular meshes of the manifold are highly compliant with modern computer architectures, they do not basically allow the manipulation of unbounded and continuous coordinates in a variable or multi-resolution perspective (see Fig. 1). Spiking neuron based models, such as the Leaky Integrate-and-Fire (LIF)-CNFT model presented in [10], would largely benefit from the use of sparse matrices [30], but this optimization would only address the computational problem.

Determining an adequate resolution a priori is indeed a concern, as on one hand computing on a coarse resolution discrete neural map introduces artifacts and strong discontinuities, and on the other hand complexity is a polynomial function of the resolution. A biologically inspired solution resides in self-organizing maps (SOM) [19] or its extensions for continuous learning such as Growing-SOM (GSOM) [1] or Dynamic-SOM [27]. These permit cortical magnification phenomena to take place and introduce a variable resolution across an otherwise fixed topology. Nevertheless, discontinuities still occur and other problems arise when projecting a high dimensional space onto a 2D manifold [21]. Whereas discontinuities are indeed found in the cortical organization (for instance in the visual cortex [16]), the model presented here adopts a more artificial yet direct approach by implementing the CNFT in a high dimensional space. It thereby avoids the need for asymmetric lateral connectivity required to establish a high dimension topology on the 2D cortical sheet (such asymmetries are found between nearby hypercolumns in the case of the primary visual cortex [6]).

Instead of being bound to matrices, researchers have also explored variations of the basic square grid mesh of the neural maps, starting with other Bravais lattices.

Fig. 1 Approximation of a 1D continuous function by dense and sparse representations. The *dense* representation (b) approximates the continuous activity (a) by using a fixed resolution mesh with n elements taking real values. Their number does not depend on the complexity of the curve. On the contrary, the *sparse* representation (c) gives an estimate of the activity as the sum of a reduced but varying number of components (here Gaussian), that each needs several parameters to be defined: $g_k = (\mathbf{x}_k, I_k)$.



Moreover, because neural maps may not be studied in isolation but as forming high level networks, alternative shapes such as disk-like maps may be considered to increase the symmetry of the system [21]. Furthermore, to study the robustness of the network dynamics in the presence of artificial lesions, or simply because biological systems are inherently variable at all levels, units should be allowed to take arbitrary locations on the maps. As the neural map topology is the key to distance based interactions, the sparse model presented in this paper can also simulate the dynamics of some Gas-nets models [17].

Perception systems may also benefit from only taking into account a few units at any time, for instance those above a given threshold with rate-coding or simply those firing with spike-coding models (even when sticking to time-driven models). Neglecting the influence of weakly activated units in the current context would drastically reduce the amount of computations to perform.

Interfacing such competition mechanisms with artificial sensorimotor systems gives another argument for trying to use compact representations. Robots performing complex behaviors often use a closed loop to control their actions and get feedback from their environment, and most sensory inputs and motor commands take a simple real value (joint angles for example). As cortical maps use population coding, these single values need to be projected through diffusion processes and receptive fields. Computations then occur on these dense representations, but in the end, an inverse conversion is nevertheless required to synthesize the maps activity into exact commands to be sent to effectors. Abstracting from the substrate and directly approximating the field activity avoids such conversions, by considering the global influence of inputs signals on the activity dynamics.

Using matrical representations, the CNFT differential equation gets easily translated into standard operations. Alas, the required convolutions result in a polynomial complexity for updating the neural fields, which further prevents a direct high dimensional implementation. Even with optimization techniques such as using a singular value decomposition (SVD) of the kernel, the polynomial power still increases with the dimensionality [20]. This is a consequence of the exponential growth in the number of connections with neighboring units, illustrating the "curse of dimensionality" [4]. Finally, when implementing such competition algorithms on parallel hardware like field-programmable gate arrays (FPGA), the benefits gained from simultaneously updating several units are attenuated by the routing constraints and additional circuitry required by the increased volume of the convolution kernel [31].

2 Model

This paper introduces a novel implementation of the CNFT, focusing on the increased expressiveness the model may convey for high dimensional data. It will be briefly described and analyzed in this section, as its properties derive from mathematical considerations.

In this paper, we will adopt and extend the notations introduced by Amari [2]. At each time step, a focus neural field activity u is updated by the CNFT, integrating stimulations from an input neural field s . The field is represented by a manifold M in bijection with $[-0.5, 0.5]^d$, where $d \in [1, +\infty[$ is the finite dimension of the manifold. Periodic boundary conditions are used to avoid numerical issues and dissymmetry near the edges, thus introducing a toric topology. The membrane potential at the position vector $\mathbf{x}$ and time t on this field is defined by $u(\mathbf{x}, t)$ and is maintained in $[0, 1]$. Similar notations are used for the input stimulation $s(\mathbf{x}, t)$. The dynamics of the membrane potential is then described by:

$$\tau \frac{\partial u(\mathbf{x}, t)}{\partial t} = -u(\mathbf{x}, t) + c(\mathbf{x}, t) + s(\mathbf{x}, t) + h \quad (1)$$

where h is the resting potential and c the potential over an intermediate neural field, artificially introduced to simplify the explanations and defined by:

$$c(\mathbf{x}, t) = \int_{\mathbf{x}' \in M} w(\mathbf{x}, \mathbf{x}') u(\mathbf{x}', t) d\mathbf{x}' \quad (2)$$

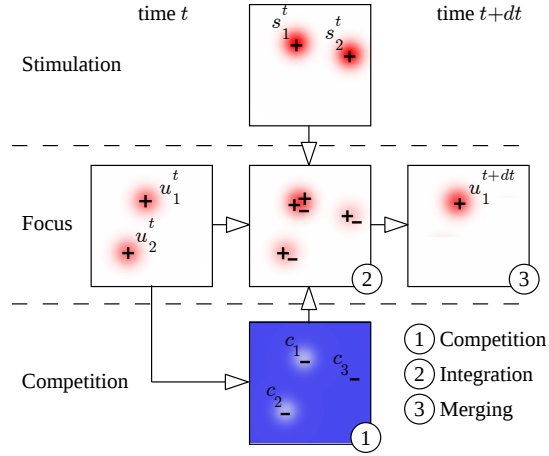
where $w(\mathbf{x}, \mathbf{x}')$ is the lateral connection weight function satisfying Eq. 3.

$$w(\mathbf{x}, \mathbf{x}') = Ae^{-\frac{|\mathbf{x}-\mathbf{x}'|^2}{a^2}} - Be^{-\frac{|\mathbf{x}-\mathbf{x}'|^2}{b^2}} \quad (3)$$

The realization that motivated the rough approximation of the neural field activity by a reduced number of Gaussian components comes from the observation of the CNFT dynamics. For experimental conditions where competition is effective (and it has to be for interactive systems), the focus map rapidly converges to a –potentially empty– set of distant bubbles. When global inhibition is considered ($b > |M| > a$ and $A > B$), general convergence results and experimental studies have shown that a single bubble may emerge and track the associated stimulus, which is the case in the experiments presented in this paper [29].

The second mathematical result used is that arbitrary signals can be approximated by a sum of Gaussians, their number being positively correlated with the precision required [15]. Even though we do not focus on optimization procedures in this publication, efficient decomposition algorithms have been proposed when input signals are not suitable for direct manipulation [11]. Receptive fields have moreover been found to be approximately separable into a sum of amplitude modulated Gaussian components, for instance in the visual system [26, 32]. Chaotic dynamics may be argued to be the norm in natural cognitive systems at the microscopic scale, but population coding averages the variability of single neuron spike trains [18] and receptive fields further smooth out the activity.

Fig. 2 Algorithmic decomposition of the sparse implementation. ① A competition field C is produced by propagating activity from $\{u'_1, u'_2\}$ to $\{u'_1, u'_2, s'_1, s'_2\}$. ② Components from the focus field $\{u'_1, u'_2\}$, input field $\{s'_1, s'_2\}$ and competition field $\{c_1, c_2, c_3\}$ are integrated. ③ Close components are merged, and resulting components with negative intensity removed ($s'_2 \cup c_3$ and $u'_2 \cup c_2$). Only one component remains ($u'_1 + dt = u'_1 \cup s'_1 \cup c_1$), reflecting the convergence towards a single bubble of activity.



2.1 Gaussian mixture based three-step procedure

From now on, neural fields will be denoted as U (focus field produced by the CNFT), C (intermediate competition field) and S (input stimulation field), respectively associated with u , c and s of Eq.1. A generic field G (either U , C or S) will be defined as a mixture of components $\{g_k\}$ (respectively $\{u_k\}$, $\{c_k\}$ and $\{s_k\}$). Determined by the set of parameters $(\mathbf{x}_k, I_k)$, g_k denotes a Gaussian function of amplitude I_k centered on $\mathbf{x}_k$. Let $g_k(\mathbf{x})$ be the activity propagated by the Gaussian component g_k at the point $\mathbf{x}$ satisfying:

$$g_k(\mathbf{x}) = I_k * e^{-\frac{|\mathbf{x}_k - \mathbf{x}|^2}{\sigma^2}} \quad (4)$$

For the focus field U , σ is fixed to a value between the excitatory and inhibitory standard deviations a and b of Eq. 3 as to synthesize each stereotyped bubble by a single component. The potential at any point of a field G can then be computed as follows:

$$g(\mathbf{x}, t) = \sum_k g_k^t(\mathbf{x}) \quad (5)$$

As we will now exclusively manipulate Gaussian components and not directly the activity at any given point on the field, Eq. 1 must be translated into a three step procedure (see Fig. 2). First, the **competition step** consists in generating the necessary components of C , i.e. wherever the focus components $\{u_k\}$ would have an effect whether on each others or on the stimulations $\{s_k\}$ through the lateral connectivity. This is some sort of sparse convolution, and therefore for each $g_k = (\mathbf{x}_k, I_k) \in U \cup S$ considered, an inhibitory component c_k of parameters $(\mathbf{x}_k, I_k^c)$ is produced. The value of I_k^c is determined by:

$$I_k^c = \frac{1}{n} \sum_{i=1}^n w(\mathbf{x}_k, \mathbf{x}_i) I_i \quad (6)$$

where n is the number of components on the focus field. The lateral competition weight function w has already been introduced in Eq. 3.

Secondly, the focus field U^t generated at the last timestep, inner field C resulting from the lateral competition and current input field S^t must be combined and integrated over time to reproduce the dynamics of Eq. 1. This **integration step** to produce the new focus fields U^{t+dt} is described by the following equation:

$$U^{t+dt} = U^t \cup \frac{dt}{\tau} [(-U^t \cup C \cup S^t) + h] \quad (7)$$

where the $\cup$ operator applied to fields actually corresponds to the union of the component sets, which is equivalent to adding the contributions of the various components as reflected by Eq. 5. The scalar multiplication (by $\frac{dt}{\tau}$) and addition (of h) are directly applied to the intensity of the Gaussian components.

Finally, a **merging step** is performed to avoid combinatorial explosion. All operations indeed increase the number of components until now, either because of the constant flow of new input stimulations at arbitrary locations, or because of the pair-wise competition components generated. Close components not only need to be merged to account for the reinforcement of bubbles iteration after iteration, but also to easily detect and eliminate negative activity components where inhibition is dominant. This guarantees that a bounded number of components with positive activity will remain on the focus field, the positivity of the field being already discussed in reference implementations [28].

Several merging algorithms have been proposed for Gaussian components in other contexts, but they often require an exact knowledge of the underlying distribution of point-like stimulations that is missing here [34]. A simple Euclidian distance based criterion is thus used, since the iterative nature of the CNFT can compensate for instantaneous small errors. The threshold for the merging is chosen to match the excitatory standard deviation a as to facilitate the emergence of stereotyped bubbles of activity (see Alg.1).

$$\begin{aligned} \mathbf{x}_{new} &= \frac{I_i}{I_i + I_j} \mathbf{x}_i + \frac{I_j}{I_i + I_j} \mathbf{x}_j \\ I_{new} &= I_i + I_j - I_i \times I_j \times \frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{\alpha^2} \end{aligned} \quad (8)$$

Eq. 8 provides the exact computations performed to determine the new parameters as a function of the two components to be merged. α is a constant higher than a , ensuring a smooth transition between the aligned and separated cases. For more

Algorithm 1 Merging algorithm for the Gaussian components on the focus field

<pre> 1. %Find close Gaussian pairs 2. $P \leftarrow \emptyset$ 3. for all $(u_i, u_j) \in U^2$ 4. if $\mathbf{x}_i - \mathbf{x}_j < a$ 5. $P \leftarrow P \cup (u_i, u_j)$ 6. end 7. end </pre>	<pre> 8. %Iterate the merging on pairs 9. while $P \neq \emptyset$ 10. $u_{new} \leftarrow \text{merge}(u_i, u_j)$ 11. $P \leftarrow P \setminus \{\text{pairs with } u_i \text{ or } u_j\}$ 12. $U \leftarrow U \cup u_{new} \setminus \{u_i, u_j\}$ 13. for all $u_i \in U$ 14. if $\mathbf{x}_i - \mathbf{x}_{new} < a$ 15. $P \leftarrow P \cup (u_i, u_{new})$ 16. end 17. end 18. end </pre>
---	--

In practice P is kept sorted as to easily select and always merge the closest components in the algorithm when choosing (u_i, u_j)

details on the different steps or to better understand the subtleties of the transformed equations, please refer to [25].

2.2 Complexity and convergence analysis

Contrary to the matrix implementation, the complexity of the sparse implementation no more fundamentally depends on the number of dimensions, hence its performance for the direct manipulation of high dimensional input spaces. Let d be the number of dimensions and n the resolution along each dimension (supposed to be the same for all dimensions for simplification purpose).

The convolution of a map of size n^d with a kernel of identical size involved in the matrix based computations determines the overall complexity in $O(n^{2d})$ using Landau notation. If we suppose that the weights are fixed, a higher-order singular value decomposition (HOSVD) of the kernel can be done at initialization [20], and the high-dimensional convolution is reduced to d linear convolutions to perform with the singular vectors, thus dropping the complexity to $O(n^{d+1})$, independently of the input dynamics or parameters of the model. By additionally transposing the computations in the frequency domain using a fast Fourier transform (FFT), the circular convolutions are further reduced to pointwise multiplications, but the complexity remains a monotonic function of n and d .

On the contrary, the number of components remaining at the end of the Gaussian mixture based procedure highly depends on the coherence of the stimulations. If the CNFT cannot find any structure in the input, for instance in presence of pure noise, the lateral inhibition will not be sufficient to eliminate low activity components. In this case, and with n being this time equal to $1/a$ so as to correspond to non-overlapping Gaussians, a maximum of $O(n^d)$ elements might appear on the focus field, as additional components would be merged in virtue of the threshold in Alg. 2.1. If we suppose that the number of input components is similarly bounded, we obtain a worst-case complexity of $O(n^{2d})$ for the competition step, which is dom-

inant with infinite asymptotics. Even though n in the sparse implementation can take much lower values than its equivalent for dense matrices, this remains prohibitive as the atomic operations considered are also much heavier.

However, the cost of updating the focus field components can drastically decrease when considering realistic inputs for which competition is indeed useful and effective. At the other end of the complexity spectrum, a unique static and stationary stimulus will generate a component that will get maximally reinforced for each timestep and will rapidly inhibit all other components on the field (see Fig.3). The minimal complexity is thus linear in the number of stimulation components, that might be very low for artificial systems.

In practice, the computational cost of the three-step procedure is of course variable and initially chaotic, but as soon as convergence towards a stimulus occurs, it induces a non linear transition in the number of components, allowing only a few distant elements to maintain their activity over time. This should not come as a surprise, as this observed behavior of the CNFT was the main reason for developing a Gaussian mixture based implementation.

To consider more realistic scenarios compared to the idealistic case previously analyzed, relaxed constraints on the spatiotemporal characteristics of the stimuli lead to local instabilities on the focus field. Whereas perfectly aligned components will simply see their activity summed, the linear weighting applied on the location vectors for the input and focus components in Eq. 7 (respectively $\frac{dt}{\tau}$ and $1 - \frac{dt}{\tau}$) generally leads to a drift of the bubble in direction of the stimulus movement (see Fig.3). For a low value of dt relatively to the stimulus speed, successive components should remain close enough the merging and consequent drift to occur. Although

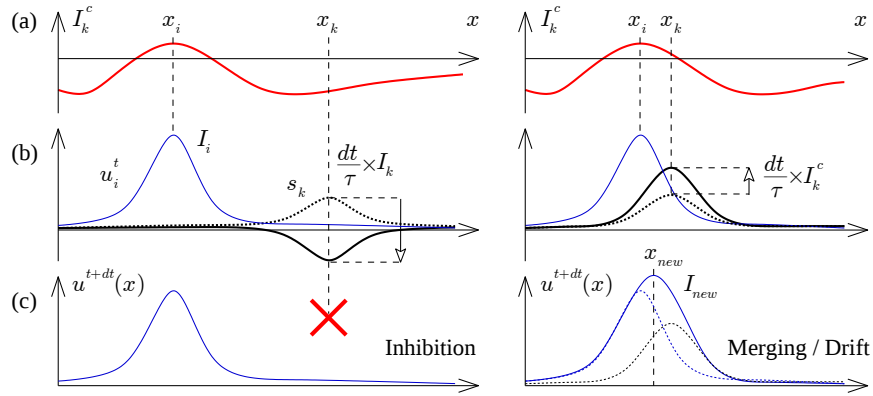


Fig. 3 Inhibition of weak distant components (*left*) and merging with close components (*right*). The convolution of the focus components (*blue curve*) with the CNFT weight function (*red*) inhibits or reinforces the input component (*plain black*) based on distance. This leads either to the removal of the input component from the focus field, or its merging with the other components, thus resulting in a slight drift of the focus bubble.

the dynamics is purely reactive and the focus is thus necessarily lagging behind the stimulation, it hence robustly tracks the coherent set of stimulations it has focused upon as long as their movement is not too fast relatively to the integration constant τ .

The next section compiles other previously obtained experimental results and introduces a dedicated toy application for high dimensional tracking.

3 Evaluation

In order to evaluate the competition mechanism realized by all versions of the CNFT, artificial input dynamics are provided to the system. These scenarios allow to test the ability to select and track what the designer considers as valid stimulations (generally those which initially have the highest spatiotemporal consistency). Various focus field characteristics are automatically computed and integrated over time, including a tracking error as the distance between the focus bubbles and the associated input stimuli. As we here only consider global inhibition and therefore the emergence of a unique bubble, a single locus synthesizing the entire focus field activity is used whether for undertaking actions or for statistical analysis purpose. The barycenter $\mathbf{c}$ of the remaining components after the update procedure (those that could not be merged together) can be rapidly computed by applying Eq. 9. Computing this kind of center of mass gives a good indication of the ability of CNFT models to converge on a single localized bubble.

$$\mathbf{c} = \frac{\sum_k I_k \times \mathbf{x}_k}{\sum_k I_k} \quad (9)$$

The tracking capacities of the Gaussian mixture based model have already been compared to previous implementations in 2D [25]. As optimal parameters for the CNFT equation depend on the task to perform, on the dynamics of the inputs and on the exact numerical model implemented, genetic algorithms have been used to find the optimal parameters in each case and compare the implementations on a fair basis [23]. The model has been shown to reproduce the following main emergent properties, which will be extended to high dimensional inputs:

1. Tracking of a spatiotemporally coherent moving stimulus whatever its trajectory under a maximal speed fixed by the CNFT equation parameters
2. Non linear bifurcation when distant but similar stimuli are presented, leading to the selection of one stimulus (phase transition in the distributed dynamics)
3. Robust tracking despite the presence of noise (up to 100% additive Gaussian white noise to signal ratio) and distracters (as long as they do not overlap with the tracked stimulus, in which case the CNFT follows the most stable inputs)

3.1 3D tracking application

Inputs can be considered as an arbitrary set of feature vectors and the sparse version of the CNFT can be said to detect and track coherent clusters in the feature space. Information theorists often describe the processing within the primary visual cortex to act as filters extracting information from the optical flow. Even when based on bio-inspired computations, filters used in computer vision are often combined using artificial techniques (segmentation for instance in [14]), which could be replaced by distributed competition algorithms. Although the sparse implementation is made to scale up to many dimensions, we will limit ourselves to three as to facilitate the visualization of the maps.

3.1.1 Input structure and dynamics

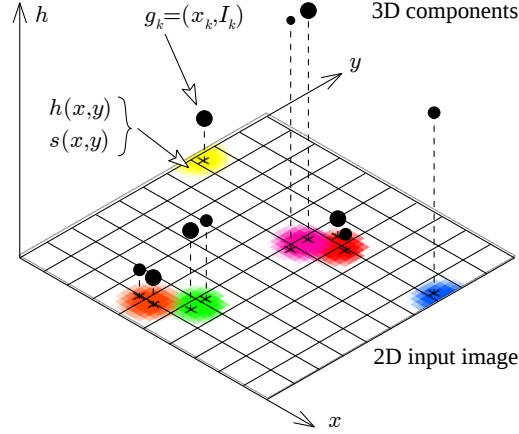
To show the generality of the approach, color is used to generate a third dimension (see Fig. 4). Using a flow of 3D points representing a 3D scene would of course have been possible in a computer generated simulation, but both living and artificial systems never get instantaneous access to such information through their sensors. Methods have been developed to reconstruct the full structure environment, but they generally are neither bio-inspired nor efficient on unconstrained inputs. When projecting a 3D environment on a 2D sensor, occlusions and many different kind of ambiguities can indeed occur, reflecting a loss in the informational content.

The environment is here represented as a set of layers, each composed of simple colored objects. These objects can move, change color, see their transparency level or shape transform, with occlusions additionally occurring between layers. These layers are then merged into a single colored dense map, where an HSV decomposition of the colors is performed to obtain a toric hue dimension and for saturation to directly act as a meaningful intensity component, contrary to what can be done with RGB. The CIELAB coding scheme provides a more human-inspired and uniform representation of color [12], but would have made the point less clear by lengthening the descriptions and making the preprocessing more complex. While the color saturation s acts as the stimulation intensity, the hue h is thus added as a third toric dimension to the 2D position (x, y) of the original CNFT.

To test the behavior of the sparse CNFT within a 3D feature space, a set of input scenarios is introduced. Although the results presented in this paper instantiate bell-shaped stimuli in the spatial domain as defined by Eq. 10, other shapes and color gradients have been tested and lead to qualitatively similar results. When not specified, the hue of the stimuli should not make a difference in the dynamics and can therefore be set to any value. The following scenarios are defined:

- A) 2 bell-shaped distant stimuli s_1 and s_2 are introduced at time $t = 0$. Their intensity are governed by $I_1 = 0.4$ and $I_2(t) = 0.5 + 0.5 \cos(\pi \times (t/5))$.

Fig. 4 Sparsification of a 2D color image into a set of Gaussian components. Once the objects have been projected on a 2D plane, the image is segmented and synthesized by a set of components (*black dots*) in the 3D feature space defined by the dimensions (x, y, h) . The size of the 3D dots reflects the intensity of the components (saturation). The background activity and the possible thresholding of low activity components are not illustrated on this figure.



- B) 1 bell-shaped stimulus of standard deviation 0.1 and intensity 1.0 follows a circular trajectory of radius 0.2 around the point $(0, 0)$ at 10 deg/s from $t = 0$. From $t = 1$, 5 distracters of the exact same shape are added and take new random hue and positions on the field every 1 s.
- C) 1 moving bell-shaped stimulus (same as in B). At $t = 1$, Gaussian noise of amplitude 0.5 is added at each point of the field, with a random hue in $[0, 1]$.
- D) 1 moving bell-shaped stimulus (same as in B) with additional full range oscillation on the hue dimension, with a period of 10 seconds.
- E) 1 moving bell-shaped stimulus (same as in B) with a hue of 0.5 (cyan) is present from the beginning. At $t = 1$, a second bell-shaped stimulus of the same intensity is introduced. However, it has a hue of 0.0 (red) and moves at 1 deg/s on the same trajectory and in the same direction.

Input stimuli centered on (x_s, y_s) with a hue of h_s and maximal saturation of s_s are defined in Eq.10. Although the hue should also vary for the three dimensions to take perfectly symmetric roles, this does not fundamentally change the dynamics as the bubble would converge on the average color, but leads to weird graphical representations and makes it more complex to compare with the 2D versions.

$$h(x, y) = h_s \quad s(x, y) = s_s * e^{-\frac{(x-x_s)^2 + (y-y_s)^2}{\sigma^2}} \quad (10)$$

Whereas the distracters are placed on top of the tracked stimulus to generate maximum perturbations and occlusions in scenario B, the second red stimulus for scenario E is placed in a deeper layer as to not totally occlude the tracked cyan stimulus. When several stimuli overlap or noise is added, alpha blending is used to combine the color components, where the transparency alpha value equals saturation (see Fig. 5).

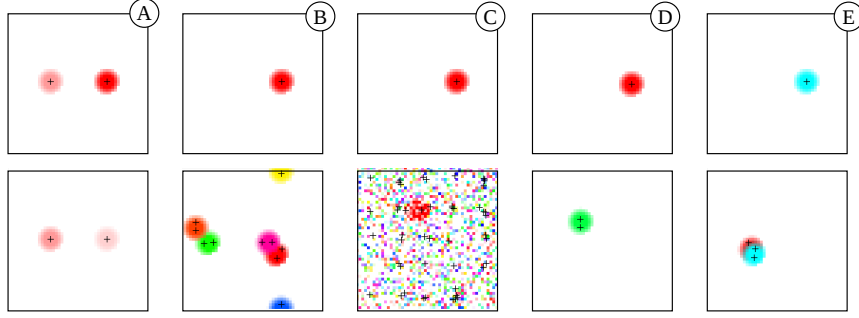


Fig. 5 Illustration of scenarios A to E (refer to the main text for their description). Each scenario is represented by one column of 2 input colored images. The first snapshot always corresponds to $t = 0$, where the algorithm gets some time to converge on the stimulus (except for scenario A where 2 stimuli are directly presented). The second snapshot is associated to a later time, chosen to be representative of the input dynamics.

3.1.2 Input sparsification

To reduce the number of input components and test the robustness of the algorithm, a rough sparsification process is introduced. The feature space is partitioned in cubic blocks of dimensions $(\Delta x, \Delta y, \Delta h)$. The activity in a block k is then synthesized in a 3D center of mass $\mathbf{x}_k = (x_k, y_k, h_k)$, with mass being here associated with color saturation I_k . The intensity and vector components are combined in a stimulation component $s_k = (\mathbf{x}_k, I_k)$ and computed by applying the following equations:

$$\begin{aligned} I_k &= \int_{x,y}^{x+\Delta x, y+\Delta y} \bar{s}(x,y) . dx . dy & h_k &= \frac{1}{I_k} \int_{x,y}^{x+\Delta x, y+\Delta y} \bar{s}(x,y) h . dx . dy \\ x_k &= \frac{1}{I_k} \int_{x,y}^{x+\Delta x, y+\Delta y} \bar{s}(x,y) x . dx . dy & y_k &= \frac{1}{I_k} \int_{x,y}^{x+\Delta x, y+\Delta y} \bar{s}(x,y) y . dx . dy \end{aligned} \quad (11)$$

As the original colored image can be described by two functions $h(x,y)$ and $s(x,y)$, the mathematical nature of h is different from x and y , so that $\bar{s}$ must be introduced to only consider stimulations in the range $[h, h + \Delta h]$.

$$\bar{s}(x,y) = \begin{cases} s(x,y) & \text{if } h(x,y) \in [h, h + \Delta h] \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

The white background used in the toy application generates input components with a null saturation because of the HSV decomposition. These components have no effect on the CNFT dynamics since the propagated activity is proportional to the source intensity and will be eliminated immediately after the first merging step. A non mandatory threshold is thus introduced to remove them, as to limit the number

of components and make graphs easier to interpret (see Fig. 4).

Even though this preprocessing may seem to provide adequate inputs for the CNFT and ease the clustering/competition process, it suffers from major limitations relatively to the sparse implementation. Blocks form a partition and do not provide continuity between the stimulations, whereas receptive fields usually largely overlap. A shaded or localized object can therefore be discretized differently from one step to another, occupying a different number of blocks depending on its position or color. Similarly, the many components coding for close objects can expand or contract when the objects follow a linear trajectory, because of their boundaries moving inside a block. Indeed, although static blocks are used, the centers of mass take continuous coordinates in the feature space. For an illustration of these phenomenons, please refer to Fig. 6.

Such rough transformation of the inputs was intentionally chosen to show that the emergent properties of the CNFT equation combined with the Gaussian merging introduced in the three-step procedure compensate for spatiotemporal discontinuities. when considering artificial systems, this may be useful for processing real world video inputs, as encoding (specially compression algorithms) and low framerate can lead to artifacts and discontinuities.

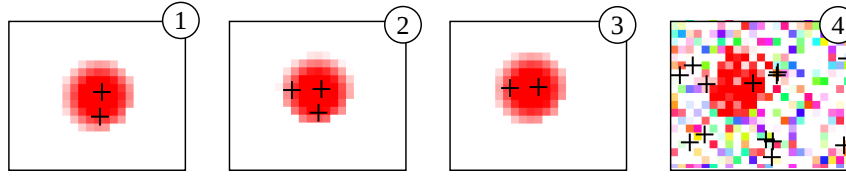


Fig. 6 Illustration of the input spatiotemporal discontinuities. The snapshots present the same partial 2D view of the input image with the input components produced by the sparsification process superimposed (*black crosses*). Although these correspond to successive timesteps, the number of extracted features varies because of the threshold introduced (one appears between frame 1 and 2, one disappears between 2 and 3). There also are strong spatial discontinuities, for instance when noise is introduced abruptly between frame 3 and 4.

3.1.3 Results

By neglecting the third hue dimension, the dynamics of the 3D sparse implementation can be compared to the dense 2D CNFT using the same parameters and input trajectories. The dense version is statistically more efficient at tracking stimuli in noisy environments as the spatial redundancy of the population coding reduces the impact of perturbations, whereas bubbles are coded by a single relatively fragile component in the sparse version. This effect could however be reduced by considering advanced merging algorithms and should be explored in further work.

Results include a comparison of three implementations: the dense matrix based 2D CNFT used as a reference [28], the 2D sparse implementation presented in [25] and the 3D sparse version presented in this paper. Error distances as a function of time are shown for typical runs with input scenarios *A* to *E* on Fig. 7.

Scenario *A* tests the ability of the system to rapidly decide between similar stimuli and to shift the focus to an unattended stimulus that becomes much more salient than the tracked stimulus. Local peaks on the error curves correspond to changes of target and are expected in this scenario only. Although the shifts in attention associated with a high error occur at slightly different times, the same kind of hysteresis appears for all three implementations. The shifts indeed do not correspond to the intersections between the intensity curves (represented not to scale by thin black lines on Fig. 7-A). They always happen later, with a distracter to target intensity ratio up to 2 (0.8/0.4 and 0.4/0.2). Because the stimuli only differ by their x position and intensity, the reproduction of the results indicates that the preprocessing and merging do not slow down or hinder the convergence.

Scenario *B* and *C* show that the 3D sparse implementation is more sensitive to noise and distracters. The stimuli and distracters cannot be differentiated by their intensity alone in the sparse and dense 2D versions and are thus only interpreted as modifying the shape and position of the tracked stimulus. The third color dimension and the sparsification process here make a difference as they allow occlusions between shapes, because of the 2D projection. The full occlusion of the target by 3 distracters occurring at 11 seconds on the graph leads to the disappearance of the target for about 1 second, leaving enough for the system to relax and focus on another target. However, due to the higher spatiotemporal continuity of the stimulus on the long-term (as distracters randomly take new positions every second), the CNFT finally focuses again on the stimulus at 15 seconds.

If the stimulus and distracters were directly provided to the system by removing the sparsification process, noise would affect the component coordinates as in the 2D sparse implementation. This possibility theoretically and practically displays the best results as it benefits from the color information without suffering from projection ambiguities. However, due to the artificiality of its performance, it is not presented in the results.

Scenario *D* simply extends the tracking results to a 3D stimulus trajectory, which means that it not only moves on the field but that its hue also changes. The bubble should thus emerge at a position close to the stimulus, but also take a similar color. Here again, the sparsification process introduces noise and discontinuities relatively to the other implementations, but the exploitation of the color information statistically improves the performance.

More interestingly, scenario *E* shows the advantages of merging the information in a common space to track objects defined by their position and hue. Whereas the two other implementations diverge when the stimuli overlap, the 3D sparse version keeps track of the fastest stimulus against the natural stationarity of the CNFT bubbles. At the point where the fast blue tracked stimulus passes over the slow red

stimulus, the same tendency is found in the three curves. About three seconds before perfect superposition, the second stimulus attracts the bubble in its direction as it reaches the excitatory part of the CNFT kernel (at least for the x and y coordinates). This compensates for the inherent lag of the reactive equations relatively to the movement of the stimulus and results in a decrease of the error. Just after the crossing occurred, the 3D implementation still tracks the blue stimulus whereas the other versions mistake one stimulus for the other and preferentially focus on the stationary one.

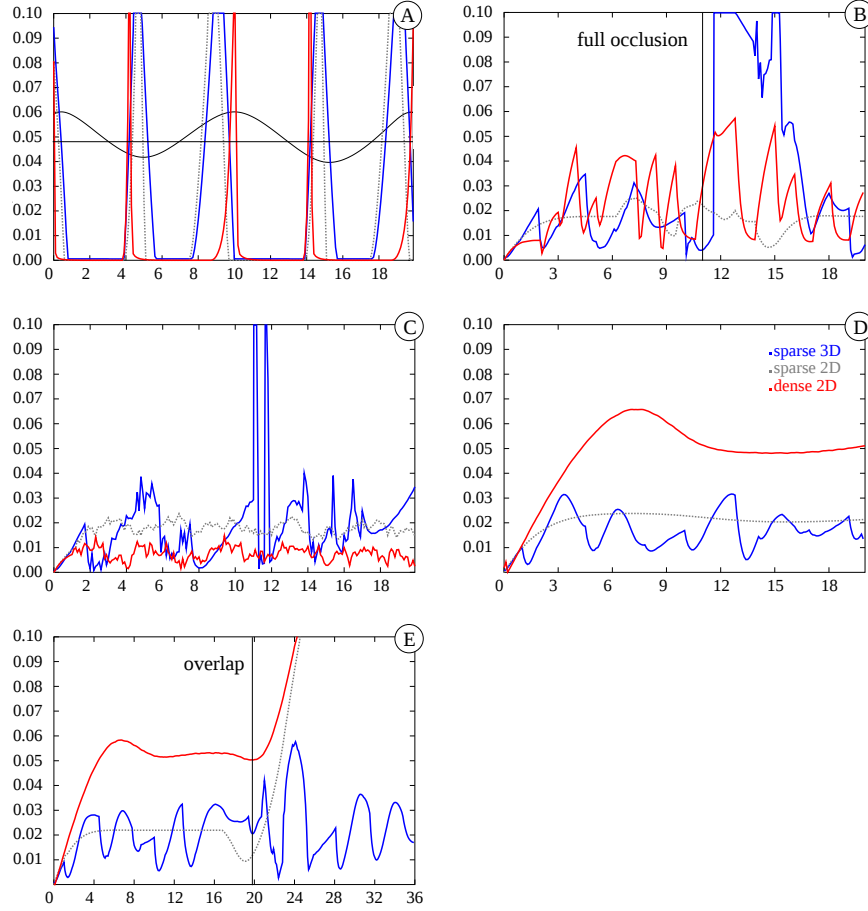


Fig. 7 Error distance as a function of time for scenarios A to E for three different implementations: 2D dense (plain red), 2D sparse (dashed), 3D sparse (plain blue). All error distances are clamped in $[0, 0.1]$ as divergence generally occurs when above (up to a distance of 0.4 on the manifold). The black lines for scenario A represent the 2 stimuli intensity as a function of time, while they represent peculiar events for scenario B and E.

4 Discussion

The previous sections presented a computationally efficient and scalable implementation of the CNFT, as well as its application to filtered inputs. The unconstrained number and the continuous coordinates of the components are both a strength and a weakness of the algorithm, as they may be hard to parallelize. However, preliminary tests have shown that reintroducing receptive fields and aligning the input components on a fixed grid improves the tracking performance compared to the rough sparsification presented earlier, by injecting continuity between nearby components. Although this means reintroducing a matrix-like field of components for the inputs, their number can be reduced as the focus field components would still take unconstrained coordinates and assimilate the stimulations based on a continuous distance. The minimal distance between two stimulations should not be more than the excitatory standard deviation a as to guarantee enough information is provided to the system for the merging step to be effective.

For more complex representational content, as expected in associative maps where information from various modalities converge and get combined, finding an adequate topology is an issue. Features and spatiotemporal relationships defining multimodal representations may indeed be highly dependent on the context and concept considered. This is highly incompatible with the kind of regular topology assumed by the projection of all features within a common space, and by the use of a single propagation function for components to interact. Nevertheless, replacing the Cartesian distance of Eq. 3 by a dissimilarity measure may at least partially solve the problem. For sensorimotor systems, local contingencies involving a limited number of dimensions (for instance the relationships between particular motor commands and associated proprioceptive feedback) can interact and compete by only considering their common dimensions [24].

Another issue raised by the toy application is that of the nature of dimensions. Although the saturation and hue can both be described as a function of the position on the field, the saturation is used as the components intensity (I_k), whereas the hue is translated into a spatial dimension of the feature space ($\mathbf{x}_k$). To put it differently, their nature in the input signal is quite similar, but they intervene in different parts of the CNFT equations. This asymmetry makes it possible to focus on salient elements, i.e. those with a high saturation. Symmetry could be restored by transforming the saturation into a forth spatial dimension (leading to slab boundary conditions, where the topology along all dimensions but one is toric). This facilitates a Bayesian interpretation of the CNFT: the focus field activity would be equivalent to a prior probability distribution over the 4D manifold of the stimulations characteristics, which is updated by integrating the information extracted from the input flow.

The improvement of the results demonstrated for the color tracking scenario simply means that the additional integration of information within a single feature space made the stimuli non ambiguous. If they were to remain ambiguous and that differentiating them was necessary to achieve the current goal of the agent, an anticipa-

tory model would be required. This is also the case when the tracking lag induced by the reactive dynamics of the CNFT equation is no more negligible and impedes the real-time interactions with the agent's environment. An anticipatory model is indeed required to make sense of the interactive dynamics over several timesteps, something the stationary reentrance of the focus activity in the original equation cannot account for. The activity should then be maximal where stimuli are expected to appear, and not where they were recognized during the previous timestep. In addition to its Bayesian interpretation, the CNFT can then be assimilated to an iterative basis function (IBF) network, with the feature space representing the high dimensional recurrent network where the inputs from various modalities are projected. With adequate internal connections to provide a forward model of the dynamics, such network can perform optimal sensorimotor integration in the spatiotemporal domain [13]. This also opens the door to sensorimotor and multimodal representations in high dimensional maps, instead of multiplying hierarchically organized 2D associative maps.

To give a possible future application of the sparse implementation, local motion detectors (MT cells in the visual cortex) could be combined in a single space, where units would share the two spatial dimensions of the retina, one dimension for the direction of movement and a forth one for the velocity. Such a 4D space would be an alternative to using a large set of maps, each dedicated to a specific direction and speed [9]. By dynamically biasing the activity in a specific part of this space, the model would track extended objects following a determined trajectory. More generally, it would be possible to combine the bottom-up emergent properties of the CNFT with top-down modulations. A bias in activity in any part of the feature space can indeed drive the selection of stimuli with specific characteristics or further increase the robustness of the attentional focus.

References

1. D. Alahakoon, S. K. Halgamuge, and B. Srinivasan. Dynamic self-organizing maps with controlled growth for knowledge discovery. *IEEE Transactions on Neural Networks*, 11(3):601–614, 2000.
2. S.-I. Amari. Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics*, 27(2):77–87, 1977.
3. X. E. Barandiaran, E. Di Paolo, and M. Rohde. Defining agency: Individuality, normativity, asymmetry, and spatio-temporality in action. *Adaptive Behavior*, 17(5):367–386, 2009.
4. R. Bellman. *Adaptive control processes: a guided tour*. Princeton University Press, Princeton, N.J., 1961.
5. M. H. Bickhard and W. D. Christensen. Process dynamics of normative function. *Monist*, 85(1):3–28, 2002.
6. W. H. Bosking, Y. Zhang, B. Schofield, and D. Fitzpatrick. Orientation selectivity and the arrangement of horizontal connections in tree shrew striate cortex. *Journal of Neuroscience*, 17(6):2112–2127, 1997.
7. K. Brodmann. *Brodmann's 'Localisation in the Cerebral Cortex'*. Smith-Gordon, London, 1909/1994.
8. Y. Burnod. *An adaptive neural network: The cerebral cortex*. Masson, 1989.

9. C. Castellanos Sánchez and B. Girau. Dynamic pursuit with a bio-inspired neural model. In *Advanced Concepts for Intelligent Vision Systems - ACIVS 2005*, volume 3708 of *Lecture Notes in Computer Science*, pages 284–291, 2005.
10. S. Chevallier and P. Tarroux. Visual focus with spiking neurons. In *European Symposium on Artificial Networks - Advances in Computational Intelligence and Learning (ESANN'2008)*, number April, pages 23–25, 2008.
11. J. Childs, C.-C. Lu, and J. Potter. A fast, space-efficient algorithm for the approximation of images by an optimal sum of gaussians. In *Graphics Interface*, pages 153–162, 2000.
12. CIE (Commission Internationale d'Eclairage). Colorimetry, 3rd edition publication 15. Technical report, CIE Central Bureau, Vienna, 2004.
13. S. Denève, J.-R. Duhamel, and A. Pouget. Optimal sensorimotor integration in recurrent cortical networks: A neural implementation of kalman filters. *Journal of Neuroscience*, 27(21):5744–5756, 2007.
14. J. Díaz, E. Ros, S. Mota, G. Botella, A. Cañas, and S. Sabatini. Optical flow for cars overtaking monitor : the rear mirror blind spot problem. Technical report, Ecovision (European research project), 2003.
15. A. Goshtasby and W. D. O'Neill. Curve fitting by a sum of gaussians. *CVGIP. Graphical models and image processing*, 56(4):281–288, 1994.
16. D. Hubel and T. Wiesel. Receptive fields, binocular interaction, and functional architecture in the cat's visual cortex. *Journal of Physiology*, 160:106–54, 1962.
17. P. Husbands, T. Smith, N. Jakobi, and M. O'Shea. Better living through chemistry : Evolving gasnets for robot control. *Connection Science*, 10(3-4):185–210, 1998.
18. E. Kandel, J. Schwartz, and T. Jessell. *Principles of Neural Science*. McGraw-Hill, New York, 2000.
19. T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, Berlin, 1995.
20. L. D. Lathauwer, B. D. Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21(4):1253–1278, 2000.
21. O. Ménard and H. Frezza-Buet. Model of multi-modal cortical processing: coherent learning in self-organizing modules. *Neural Networks*, 18(5-6):646–55, 2005.
22. V. Mountcastle. Modality and topographic properties of single neurons of cat's somatic sensory cortex. *J Neurophysiol*, 20(4):408–434, 1957.
23. J.-C. Quinton. Exploring and optimizing dynamic neural fields parameters using genetic algorithms. In *Proceedings of IEEE World Congress on Computational Intelligence (IJCNN'2010) (Barcelona, Spain)*, 2010.
24. J.-C. Quinton and J.-C. Buisson. Multilevel anticipative interactions for goal oriented behaviors. In *Proceedings of EpiRob'08 - International Conference on Epigenetic Robotics (Brighton, UK)*, pages 103–110. Lund University Cognitive Studies, 2008.
25. J.-C. Quinton and B. Girau. A sparse implementation of dynamic competition in continuous neural fields. In *Brain Inspired Cognitive Systems (BICS'2010)*, Madrid, 2010.
26. R. Rodieck. Quantitative analysis of cat retinal ganglion cell response to visual stimuli. *Vision Research*, 5(11):583601, 1965.
27. N. Rougier and Y. Boniface. Dynamic self-organising map. *Neurocomputing*, 2010.
28. N. P. Rougier and J. Vitay. Emergence of attention within a neural population. *Neural Networks*, 19(5):573–581, 2006.
29. J. Taylor. Neural bubble dynamics in two dimensions: Foundations. *Biological Cybernetics*, 80:5167–5174, 1999.
30. R. Tewarson. *Sparse matrices*. Academic Press, 1973.
31. C. Torres Huitzil, B. Girau, and C. Castellanos Sánchez. On-chip Visual Perception of Motion: A Bio-inspired Connectionist Model on FPGA. *Neural Networks*, 18:557–565, 2005.
32. T. Wennekers. Separation of spatio-temporal receptive fields into sums of gaussian components. *Journal of Computational Neuroscience*, 16(1):27–38, 2004.
33. H. R. Wilson and J. D. Cowan. A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. *Kybernetik*, 13:55–80, 1973.
34. W. Xu, J. Duchateau, K. Demuyne, and I. Dologlou. A new approach to merging gaussian densities in large vocabulary continuous speech recognition. In *IEEE Benelux Signal Processing Symposium*, pages 231–234, 1998.