HAL
archives-ouvertes.fr

Implicit and Explicit Representations

Nicolas Rougier

▶ To cite this version:

Nicolas Rougier. Implicit and Explicit Representations. Neural Networks, Elsevier, 2009, 22 (2), pp.155-160. 10.1016/j.neunet.2009.01.00 . inria-00336167

HAL Id: inria-00336167
https://hal.inria.fr/inria-00336167

Submitted on 7 Jan 2009

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.

Implicit and Explicit Representations

Nicolas P. Rougier

Abstract

During the past decades, the symbol grounding problem, as it has been identified by [9], became a prominent problem in the cognitive science society. The idea that a symbol was much more than a mere meaningless token that can be processed through some algorithm, sheds a new light on higher brain functions such as language and cognition. We present in this article a computational framework that may help our understanding of the nature of grounded representations. Two models are briefly introduced that aim at emphasizing the difference we make between implicit and explicit representations.

Key words: Computational Neuroscience, Representation, Symbol, Embodied cognition

Email address: Nicolas.Rougier@loria.fr (Nicolas P. Rougier).
URL: http://www.loria.fr/~rougier (Nicolas P. Rougier).

1 Introduction

One of the central notions in language is the notion of a symbol that may be naively described as some shared knowledge between entities that serve the purpose of representing and exchanging information. More precisely, Saussurian semiotic defines a *sign* as a deterministic functional regularity of a system where a signifier stands for a signified. When there exists a causal relationship between the signifier and the signified (e.g. smoke and fire), the sign is called an index. This notion of semiotic index is deeply rooted in most animal and human behavior that learn to associate for example, the odor or the sight of a predator to some imminent danger. In this case, the odor (the signifier) is a precursor of the predator (the signified) and elicits a fleeing behavior. More generally, if A is always followed by B, then A is said to be a precursor of B and having A is equivalent to having B (but you do not necessarily need A to get B). This constitutes the base of Pavlovian conditioning where a dog learns that the bell ring is an index of some incoming food. Based on an existing index A–B, it is possible to make the dog learn a second arbitrary index A'–B' (most of time, the conditioned answer B' is equal to the unconditioned answer B but they may differ in some paradigms).

Symbols are similarly defined as a functional regularity where a signifier stands for a signified but this function is grounded on an arbitrary conventional rule established by some entity. One difficulty is first constituted by this arbitrary rule that needs to be shared among the different entities engaged in communication. If symbols are not shared among entities, communication based exclusively on those symbols is not possible at all. The second but greater difficulty lies in the nature of the signified that also needs to be shared among the two entities independently and prior to any symbolic relationship. For example, if you decide to name a given object a *glass* and decide to share this symbol with someone else, you need some formal ways to indicate precisely what you mean by a *glass*. One way of achieving such a description of the *glass* entity is to use language itself, i.e. to use other already shared symbols, in order to describe some unique property of the glass that makes it a glass. However, you cannot pretend to do such for any given symbols because you would enter a circular graph where symbols are recursively defined. A simple example of such circular graphs are dictionaries that need to define words using words. They are naturally and deeply recursive. For instance, if you take a closer look at the definition of *light*, you would find yourself redirected to the definition of *sun* that would redirect you to the definition of *light* again. This has been stated quite clearly using the well known example of the Chinese dictionary from which you cannot pretend to learn Chinese if you don't possess proper entry points. Consequently, there is a need to *open* the graph at some points and to root is somewhere outside of it. One striking example

of such an open graph where the root points are particularly well identified are mathematics. The foundations work by [28] explicitly identified in their 3 volumes book entitled *Principia Mathematica* a reduced set of axioms that aim to serve as a basis for the derivation of all mathematical truth using inference rules. In Russell's word, "all pure mathematics follows from purely logical premises and uses only concepts definable in logical terms". Even if this statement has been proved to be wrong later by Kurt Gödel, this yields nonetheless an interesting framework for the understanding of what axioms are.

Such axioms implicitly exist in language but are very far from being well identified. They found their origin in the idea that since people share a common perceptive and motor apparatus, they are hence able to develop some common representation such as *color*, *pain*, *hunger*, etc that can be named without further explanation. The challenge for an artificial system is then to be able to develop such representations that are grounded into a physical reality in a straightforward way. This is precisely what has been explained by Harnard in [9] and named the symbol grounding problem that is (quoted from the original article) *"How can the semantic interpretation of a formal symbol system be made intrinsic to the system, rather than just parasitic on the meanings in our head"*. What it does mean indeed is that you cannot pretend to acquire primary axioms out of nowhere, they should found some support into the physical reality.

In this sense, traditional Artificial Intelligence (a.k.a. Symbolic A.I.) completely escaped the problem by stating that human intelligence was equivalent/reducible to a mere symbol manipulation problem. It did not address at all the problem of what these symbols are supposed to mean and rather attempted at using a set of symbols to achieve high-level thinking algorithms. The quest for intelligence was finally a quest for finding clever deduction algorithms that could support reasoning capacities and problem solving. In the other hand, the numerical nature of artificial neural networks (ANN) made them good and natural candidates in early A.I. to try to tackle the problem of anchoring a symbol into some physical reality. But dealing with a numerical model is not necessarily a sufficient condition as we would like to explain in the rest of this article.

2 Symbol Grounding Problem

In order to properly address the symbol grounding problem, there is a critical need to avoid to explicitly embed any symbol anywhere in the model, a priori or a posteriori. If this elementary precaution has not been taken, there is a risk of facing a situation where one cannot decide whether *emergent* symbols

are not simply deductive of symbols primarily embedded within the system. This is quite equivalent to axioms in mathematics from where you can derive most of the existing theorems. This does not mean that such models are useless in the quest for cognition and language. Axioms in mathematics do not make mathematics and thus, there is still an urging need to know how new representations may be built on top of the most grounded ones and what are the mechanisms behind.

However, to not embed any symbol in a model may be harder than it appears at a first glance. One natural and classical way to try achieve this is to use numerical models such as for example artificial neural networks (ANN). ANN may be roughly described as a set of interconnected groups of artificial units that uses a specific model for information processing based on parallel and distributed computations. In most cases, artificial neural networks are adaptive systems since they change their inner structure based on external or internal information such a reward, cost or error signal. On these bases, there exists virtually a myriad of models that distinguish themselves either by the elementary computational model, by the architecture or by the adaptive algorithm that supports learning. For example, the multi-layer perceptron [18, 23, 4] is a set of feed-forward layers where units from a layer possess directed connections to the subsequent layer. The elementary computational model is most generally based on a simple sigmoid function of the weighted sum of the input and the back-propagation algorithm ensures that the universal approximation theorem applies such that a multi-layer perceptron with one hidden layer may approximate any function from [0,1] to [0,1]. One strong property of such models is that they are able to virtually process any numerical inputs. However, most derivatives of those early artificial neural networks do not really try to capture any biological reality anymore and if they were initially inspired by brain and neural studies, they lost this inspiration in favor of efficiency and performance. If we now turn ourselves towards computational neuroscience, we may find again some biologically plausible neural networks whose goals are to explicitly try to understand cognition and propose some functional mechanisms in this direction. Models in this domain are generally of a high precision and can actually account for biological or psychological data to some extent. Once again, in this domain, we face many different models that implement many sorts of computational paradigms using different sets of constraints ([15, 7]). But being biologically plausible does not guarantee anything concerning the emergence of symbols. If those models can legitimately pretend to be more realistic than their classical counterparts, they may also completely escape the problem by considering only symbolic inputs and concentrate their efforts in understanding some higher mechanisms (see for example [24, 29, 13, 14])

Based on the computational paradigms introduced in [15], we have been exploring such a model (see [21, 20] for details) where the model has been shown

to develop self-organized abstract rule-like representations that are used for cross-task generalization. The model itself is designed around a set of tasks that aim at manipulating attributes such as color, shape, size, etc. This model is supposed to learn and choose adequately the relevant dimension at any time based only on a yes/no signal. This is strictly equivalent to the Wisconsin card sorting task (WCST) where subjects are asked to sort cards along some criteria chosen by the experimenter but not communicated to the subject. After some learning, the model is able to develop representations for dimensions and to use them adequately to solve the different tasks. One important property of these representations is that they are shared among the different tasks. This means that a representation learned in the context of a specific task may be re-used in the context of a novel task. These representations have been shown to develop through experience on this basic set of sensory-motor tasks via synaptic learning mechanisms through a broad range of experience across multiple tasks. However, if this model describes a biologically-based alternative to abstract symbol processing models, it fails at explaining the very nature of those representations. More precisely, explicit symbols have been introduced at the level of the input (i.e. one unit for red, one unit for big, etc.) and thus it came as no surprise that the so called self-developed representations are tightly linked to the symbolic nature of those inputs. The *color* representation can develop itself only because red, blue, yellow and green are statistically relevant in the same context during some time and can be explicitly manipulated. Consequently, we decided to go down one level in our modeling hypotheses in order to try to get rid of explicit symbols once and for all.

3 A Tentative Modeling Framework

As we explained in the previous section, designing a predictive model about emergence using *regular* artificial neural networks does not guarantee you anything against modeling artifacts. If you do not design your model into a strongly constrained and well defined modeling framework, you're taking the risk of having some a posteriori interpretation of properties that have been primarily embedded within the model. Based on our own experience in modeling, we think that it is possible to avoid most modeling artifacts by using a set of simple conditions that need to be enforced anytime and anywhere in the model:

- Distributed
- Asynchronous
- Numerical

These conditions are not really something new and have been used more or less explicitly in a number of works ([8, 25, 16, 12]). We would like to make them explicit and to stress each one of them in order to explain why it is important to enforce them if we want to go further in our understanding of cognition.

3.1 Distributed

Distributed property is one of the most advertised properties of neural networks while it is actually hardly the case in a number of models. This distributed property must indeed be considered at two different levels: computation and representation. We may have computational paradigms using distributed representations and unified computations and we may of course also have the counterpart of unified representations with distributed computations. Fundamentally, a distributed representation is one in which the underlying information it represents cannot be accurately extracted from a single unit. One would say that not only it cannot be accurately extracted from a single unit, but it cannot be extracted at all from a single unit. While this may sound nicer, it is not always the case since it depends on the very nature of the information that needs to be encoded. If this information is partitionable in some way, then you can respectively assign the representation of n sub-parts of the information to n distinct units and as a consequence, you need to have all the units to get the whole picture. The other way around is to consider that each unit holds a degraded representation of the whole information and only the combination of those representations allows to build accurately the original information. This yields the advantage of graceful degradation in face of dysfunctional or missing units (see for example [11] where a color variable is continuously encoded using a distributed representation). As an illustration, let us consider the simple case of encoding a variable X in the interval [0,1]. The simplest representation is to have a single unit whose activity varies continuously between 0 and 1 and thus directly represents X. A distributed representation of this same information may be done using n units whose activities represents how far is X from a given value characteristic of each unit. If these characteristic values are evenly spread on the [0,1] segment, then we can rebuild the original value with fair precision (without having the exact value of X, only if n tends toward the infinite can we get an exact representation of the original information). The advantage for such representation is that any unit holds anytime a degraded representation of the information that can be used effectively.

This has direct consequences on the nature of computations that may be performed. Let us consider again the illustration introduced previously and let

us suppose we would like now to discriminate cases where the X variable is above or below a given threshold. If there is a single unit representing X, the computation that can be performed to make the decision can be reduced to a single comparison problem where we only need to know if the source value is above or below the threshold. We could use as many *decision* units as we want to compute the decision without fundamentally changing the underlying algorithm since the decision source is unique. Any extra unit would be actually redundant with the first one. In the case of a distributed representation however, there are several decisions sources available that may be used distinctively within one or more "decision" units. This leads to an extended set of decision functions where you can, for example, explicitly decide to give higher precision for a given range of the source value.

3.2 Asynchrony

Most computational paradigms linked to artificial neural networks or cellular automata use implicitly what is called synchronous evaluation of activity. This means that information at time $t + \Delta t$ is evaluated exclusively on information available at time t. The usual way of performing such a synchronization is to explicitly implement a temporary buffer at the unit level where activity computed at time $t + \Delta t$ is stored. Once all units have evaluated their activity at time t+1, the "public" activity is replaced by the content of the buffer (there exists of course other ways of doing this synchronous evaluation but the idea remains the same not to mix information between time t and time t+1). To perform such a synchronization, there is thus a need for a global signal that basically tell units that evaluation is over and they can replace their previous activity with the newly computed one. At the computational level, this synchronization is rather expensive and is mostly justified by the difficulty of handling asynchronous models. For example, cellular automata have been extensively studied during the past decades for the synchronous case and many theorems has been proved in this context. However, some recent works on asynchronous cellular automata showed that the behavior of the models and associated properties may be of a radical different nature depending on the level of synchrony of the model (you can asynchronously evaluate only a subpart of all the available automata, see [5]). At a more behavioral level, we want to avoid to any kind of global synchronization since it could first, interfere with the interpretation of results in the end (as we will explain in next section), but more importantly, it would mean that there exists some synchronization *supervisor* that would be able to synchronize every units.

3.3 Numerical

What is actually computed by a unit is of the utter most importance since it directly impacts model behavior. But here again, we have to be very careful about **what** is computed by a unit and **how** it is actually computed. The philosophy of artificial neural networks is to consider a unit to be a very simple processor that is able to handle some inputs in order to compute some output. The difficulty that immediately arise is to define what we call *simple* in order to avoid to have too smart units ([3]). We propose to endow units with a numerical property that requires for a unit to compute its activity unconditionally (no *if* within the computation algorithm) and in a straight forward way. This clearly opposes symbolic computation as it can be found, for example in rule based or derivative neural networks models. More specifically, it means that a unit is not able to "decide" to perform this or that evaluation based on a set of premises regarding inputs or any other kind of information, not even self information. Evaluation is unconditional and must be performed each time it is required. This evaluation may result in no change at all in the unit activity but this must be clearly integrated at the level of the equation that govern its behavior. Moreover, the activity of a unit must be a continuous (possibly bound) variable. If it is supposed to hold discrete levels of activation, this must be integrated within the equation. This is indeed a strongly constrained context that prevents, for example, to implement classic cellular automata such as the game of life. Apart of its synchronous nature, this game depends also and explicitly on the comparison of the number of neighbors against some game constants. This numerical constraint is hence very strong and implicitly rejects a number of models out of the framework. This is the price to pay to avoid having units that are finally smarter than the model they're embedded in.

4 Emergence

Using this modeling framework, we have been exploring the concept of emergence through a very simple model of visual attention using the Continuum Neural Field Theory (CNFT, see appendix). The proposed model is composed of two maps (*input* and *focus*), each of them being of size $n \times n$ units. These units are governed by the discretized version of the equation 3 given in appendix, resulting in a single numerical output value bound between 0 and 1 for any unit. The equation is numerically solved using forward Euler method with asynchronous update (see [19] for details). Units are organized into maps and for any unit, every link originating from the same map is considered to be a lateral links (i.e. it is part of the lateral connection weight kernel) while every other link are considered to be either feed-forward or feed-back and are

part of the input (I) in equation 3. The model perceived the external world via a primitive low-resolution camera that is only able to detect a specific color intensity. The world itself is made of colorful objects (fruits in our case) lying on a table and the resulting input is a set of blob-like activities feeding the model. At each time step, the *input* map is clamped with the perception coming from the camera and this activity fed the $focus$ map using a one to one connectivity. An object is consequently represented in the *input* map as a set of several active units. This model is actually enforcing the three properties we introduced previously:

- Distributed: the position of an object is perceived as a set of several active units.
- Numerical: the state of a unit is entirely described by a single real value (potential) that is updated at each time step from inputs.
- Asynchronous: all computations are asynchronous.

4.1 Emergence of Attention

As explained in [22], the model (see fig. 1) is able to focus on a specific blob and to remain focused on this blob in spite of noise, movement, distractors or saliency. This has been accurately measured in the context of different experimental setups by explicitly decoding the activity in the focus map. Furthermore, it is quite easy to actually see the point of attention using proper visualization tools. One can see the formation of the blob of activity within the focus map that represents the focused stimulus. Since we are able to decode the information provided by the activity in the focus map, the existence of this blob of activity is not really questionable.

The fact is that the blob of activity within the focus map exists only for an external observer that know how to decode the information. Said differently, the blob is held in the eye of the beholder. A focus map is a set of units whose activations, *if interpreted correctly*, may be linked to the assumed focused stimuli. But such interpretation does not exist within the model since it is made of a single map. There is no such thing as a homunculus or central supervisor who would be able to interpret anything. This is what makes the challenge of not embedding any symbol a priori or a posteriori in a model quite counter-intuitive and difficult. Our natural anthropomorphic attitude towards the model makes us project ourselves in the model and decide that the blob of activity does represent a point of attention while it does not.

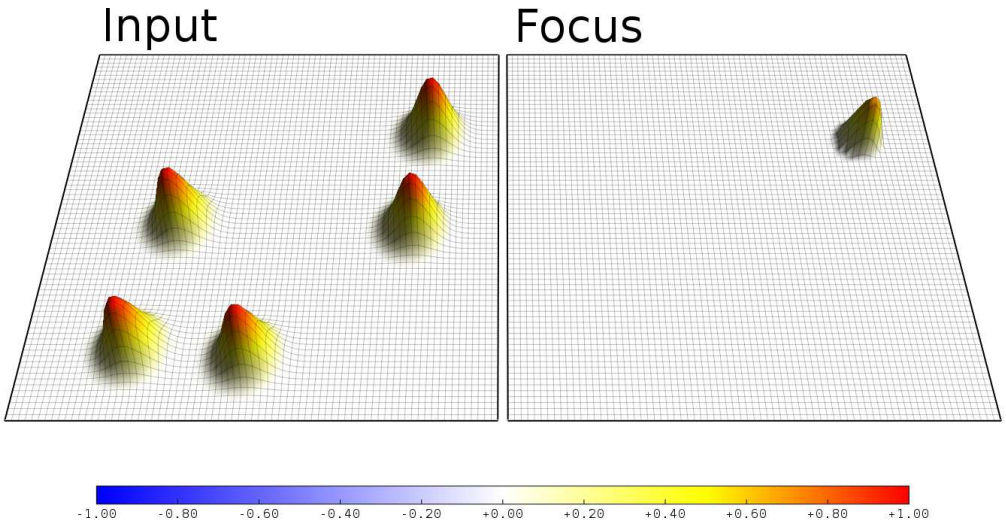

Fig. 1. The model of attention is made of two maps (input and focus), each of them being of size $n \times n$ units ($n = 40$ in experiments). Input map (on the left) corresponds to an entry that is feeding the focus map (on the right) which represents a cortical layer whose units possess localized receptive fields on the surface of the input. In other words, each unit x_{ij} of map focus receives its input from the input map which corresponds to a localized receptive field, being more or less broad depending on some constants. The input map does not have any lateral interaction nor feedback while each unit in the focus map is laterally connected using a difference of Gaussian. Activation of units from the focus map are governed by equation introduced by [1] and extended by [26] which considers a neural continuum governed by a simple differential equation. This architecture implements the most rudimentary form of attention that allows the model to focus on one static or moving stimulus without being disturbed by noise, movement or distractors.

4.2 Grounding the representation

One way to avoid having any interpretation of what is really happening within the model is to ground this model into the physical reality. As we explained in the previous subsection, the emerging bubble of activity may be interpreted as the focused stimuli. One way to achieve such interpretation is to consider that each unit of the focus map codes for an elementary displacement that is relative to the position of the unit within the map. Let us consider unit at position (i, j) with activity a_{ij} in the focus map (which is of size $n \times n$). We can consider this unit to code for the elementary movement $a_{ij}.\frac{i}{n}$ along the x axis and $a_{ij}.\frac{j}{n}$ along the y axis. To get the overall movement (x_c, y_c) from the focus map, we can compute it as follows:

$$(x_c, y_c) = \left(\frac{\sum_{i,j} \frac{i}{n} a_{ij}}{\sum_{i,j} a_{ij}} - 0.5, \frac{\sum_{i,j} \frac{j}{n} a_{ij}}{\sum_{i,j} a_{ij}} - 0.5 \right) \quad (1)$$

Furthermore and since the CNFT equation ensures us that there is only one bubble of activity within the focus map at any time, we can simplify the equation by considering the mean activity A of the focus map which is more or less constant:

$$(\hat{x}_c, \hat{y}_c) = \left(\frac{1}{A} \sum_{i,j} \frac{i}{n} a_{ij} - 0.5, \frac{1}{A} \sum_{i,j} \frac{j}{n} a_{ij} - 0.5 \right) \tag{2}$$

If we now consider again the pan tilt camera that served to acquire the original image, we can now control its pan tilt motors by using the previous definition of $(\hat{x}_c, \hat{y}_c)$ as the desired motor position along pan and tilt. This requires the abstraction of the actual motors (with only one global command for pan and one global for tilt) into $n \times n$ elementary motor commands. Each unit at position (i, j) can then be linked to corresponding pan motor i and tilt motor j (for example). This would result in the camera to automatically center on the actual focused stimuli.

However, we could also consider a completely different scheme of connection where for example a unit at position (i, j) is linked to pan motor $n - i$ and tilt motor $n - j$. The resulting behavior would then be the camera to move at the exact opposite of the "focused" stimulus. In fact there is a large amount of connectivity patterns that lead to as much different behaviors. This underlines the fact that the a priori interpretation of the bubble of activity cannot be made without the embodiment of the model into the physical reality. This is very similar to actual motor-neurons that act on muscle fibers in a stereotypical way, grounding their action into the reality.

4.3 Using representation

The question that remains is to determine how this supposed focus of attention can be moved to another location, especially when the currently attended place has no behavioral relevance. A solution used in the bottom-up visual attention model by [10] is to locally inhibit the attended location to allow the system to switch to another salient location. This has been done in reference to the inhibition of return (IOR) phenomenon proposed by [17] who showed that previously attended locations have decreased processing abilities, as if they were partially inhibited. The drawback of this mechanism is that the switch of attention is automatic (each location is attended a fixed amount of time depending on the neural dynamics) and that nothing ensures that each potentially interesting location will be attended. For example, if the inhibition is too short, the focus of attention can switch back and forth between the two most salient locations only. The purpose of the extended model (see fig. 2) was

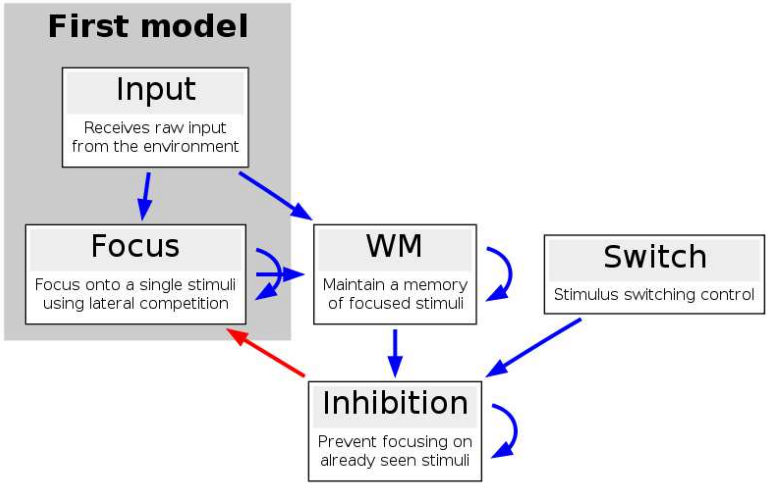

Fig. 2. The model is a direct extension of the attention model introduced in the previous section. The raw camera input is received within the *input* map and the *focus* maps implement the visual attention on a single stimulus as explained in the previous section. To ensure a consistent attention-switching mechanism, a dynamic working memory map (*WM*) has been added to the model in order to keep track of already attended stimuli. The inhibition map (*inhibition*) ensures an inhibitory action that prevents the model from focusing onto an already focused stimuli. All the maps share a common topology (with different input and output) and all units are governed by the same equation with slightly different parameters. A detailed description of the model as well as parameters can be found in [27].

to deal with these issues: building a system that can explore all the salient locations in a scene without exploring twice the same place and that is able to stop switching whenever a satisfying target is found. Thus there is a need for two different mechanisms: a mechanism able to change the focus of attention when required; a mechanism ensuring that a previously visited location can not be chosen again.

Without entering details of the model (see [27, 6], we showed that the model exhibits a purely sequential behavior (the sequential scanning of the salient locations) in the real (but simplified) world despite noise, target positions and movements, etc. This sequential behavior is a direct consequence of the inner dynamics of the units and of the architecture of the system (the links between the units). While this architecture is not meant to model precisely the actual structure of the brain (even if the overall architecture is inspired from biological data), it nonetheless shows that a unique homogeneous substrate (several maps using the same dynamic) can be involved in a given function without ever being explicitly specialized to a given sub-problem of the task. If we now turn back to the original question that initially motivated the model, the answer is yes: the point of attention is somehow *virtual* for the model but it can be exploited accurately without the need for a central supervisor or an homunculus. All computations remain distributed, numerical and asynchronous.

5 Conclusion

The two models that have been briefly introduced illustrate a situation where information is manipulated and used by a model without ever possessing an explicit knowledge of the underlying nature of the information. Considering a primitive body made of a pan-tilt camera situated in a simplified world (made of fruits lying on a table), the panel of internal representations that may arise from the interaction with such a world is of course quite limited. Nonetheless, as we explained, the first model has been proved to be able to focus on a single *object* (an object being defined as a blob in this simplified world) and we could externally interpret this point of attention as a representation for a concept like "this" or "object". If we consider now the underlying mechanisms supporting the model, we found that on one hand there does not exist such explicit interpretation (since the model does not possess the proper structure to do so) while on the other hand the second model showed us that there is no such need of an explicit information to carry out a visual search behavior. Using an equivalent constrained modeling framework, the model introduced in [2] goes even further and allows a control architecture to behave like an homeostatic machine that triggers a systematic imitation behavior thanks to an ambiguity in the perception.

Before even considering communication between some entities (either natural or artificial) using a set of shared symbols, an implicit prerequisite is that these two entities shared some common "concepts" that can be transcribed later into some formal symbols. But, as we illustrated using the proposed constrained modeling framework, this may be proved to be insufficient since those concepts may be largely implicit and not directly addressable by the entity. Thus, the question remains whether this implicit knowledge is sufficient to support some early communication with another similar entity. For example, considering the imitation game as proposed by [2], there is clearly a bias in favor of the human experimenter that teach the robot and know how to interpret robot behavior (is it lost at imitating me ? do I have to slow down ? etc.).

The fact that two entities share some common perception is probably a necessary condition but may be proved insufficient to establish any communication. The real challenge for artificial systems is not only be able to develop representations that are grounded into a physical reality, but also to be able to explicitly manipulate them in order to be able to enter some process that could lead to communication with another entity.

6 Appendix: Continuum Neural Field Theory

Using notations introduced by [1], a neural position is labelled by a vector $\mathbf{x}$. This represents a two-component quantity designing a position on a manifold M in bijection with $[-0.5, 0.5]^2$ and represents the membrane potential of a neuron at the point $\mathbf{x}$ at time t and is denoted by $u(\mathbf{x}, \mathbf{t})$. It is assumed that there is lateral connection weight function $w(\mathbf{x} - \mathbf{x}')$ which is in our case a difference of Gaussian function as a function of the distance $|\mathbf{x} - \mathbf{x}'|$. There exists also an afferent connection weight function $s(\mathbf{x}, \mathbf{y})$ from the position $\mathbf{y}$ in the manifold M' to the point $\mathbf{x}$ in M. The membrane potential $u(\mathbf{x}, t)$ satisfies the following equation (3):

$$\begin{aligned} \tau \frac{\partial u(\mathbf{x}, t)}{\partial t} = -u(\mathbf{x}, t) + \int_M w_M(\mathbf{x} - \mathbf{x}') f[u(\mathbf{x}', t)] d\mathbf{x}' \\ + \int_{M'} s(\mathbf{x}, \mathbf{y}) I(\mathbf{y}, t) d\mathbf{y} + h \end{aligned} \tag{3}$$

where f represents the mean firing rate as some function of the membrane potential u of the relevant cell, $I(\mathbf{y}, t)$ is the output from position $\mathbf{y}$ at time t in M' and h is the neuron threshold. w_M is given by the equation (4).

$$w_M(\mathbf{x} - \mathbf{x}') = A e^{\frac{|\mathbf{x}-\mathbf{x}'|^2}{a^2}} - B e^{\frac{|\mathbf{x}-\mathbf{x}'|^2}{b^2}} \text{ with } A, B, a, b \in \Re^{*+} \tag{4}$$

Furthermore, afferent connections are described by equation (5).

$$s(\mathbf{x}, \mathbf{y}) = C e^{\frac{|\mathbf{x}-\mathbf{y}|^2}{c^2}} \text{ with } C, c \in \Re^{*+} \tag{5}$$

References

[1] S. Amari. Dynamic of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics*, 27:77–88, 1977.

[2] P. Andry, P. Gaussier, and J. Nadel. Autonomous learning and reproduction of complex sequences: a multimodal architecture for bootstraping imitation games, 2005.

[3] M. Cook. It takes two neurons to ride a bicycle. In *Demonstration at NIPS'04*, 2004.

[4] Y. Le Cun. Learning scheme for asymmetric threshold networks. In *Cognitiva 85*, Paris, 1985.

[5] N. Fates. Asynchronism induces second order phase transitions in elementary cellular automata. *Journal of Cellular Automata*, -, 2008.

[6] J. Fix, J. Vitay, and N.P. Rougier. A computational model of spatial memory anticipation during visual search. In *Anticipatory Behavior in Adaptive Learning Systems 2006*, 2006.

[7] W. Gerstner and W.M. Kistler. *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, Cambridge University Press, 2002.

[8] C. Giovannangeli and P. Gaussier. Interactive teaching for vision-based mobile robot: a sensory-motor approach. In *IEEE Transactions on Man, Systems and Cybernetics*, 2008. to appear.

[9] S. Harnard. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42:335–346, 1990.

[10] L. Itti. Visual attention. In M. A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*, pages 1196–1201. MIT Press, 2nd edition, 2003.

[11] J. S. Johnson, J.P. Spencer, and G. Schoner. Moving to higher ground: The dynamic field theory and the dynamics of visual cognition. *New Ideas in Psychology*, 2008. to appear.

[12] Olivier Ménard and Hervé Frezza-Buet. Model of multi-modal cortical processing: Coherent learning in self-organizing modules. *Neural Networks*, 18(5-6):646–655, 2005.

[13] P.R. Montague, P. Dayan, and T.J. Sejnowski. A framework for mesencephalic dopamine systems based on predictive Hebbian learning. *Journal of Neuroscience*, 16:1936–1947, 1996.

[14] R.C. O'Reilly, D. Noelle, T.S. Braver, and J.D. Cohen. Prefrontal cortex and dynamic categorization tasks: Representational organization and neuromodulatory control. *Cerebral Cortex*, 12:246–257, 2002.

[15] R.C. O'Reilly and Y.Munakata. *Computational Explorations in Cognitive Neuroscience: Understanding the Mind by Simulating the Brain*. MIT Press, Cambridge, MA, USA, 2000.

[16] R. Pfeifer and J. Bongard. *How the body shapes the way we think. A New View of Intelligence*. The MIT Press, 2006.

[17] M.I. Posner and Y. Cohen. Components of visual orienting. In H. Bouma and D. Bouwhuis, editors, *Attention and Performance*, volume X, pages 531–556. Erlbaum, 1984.

[18] F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 1958.

[19] N. Rougier and A. Hutt. Synchronous and asynchronous evaluation of dynamic neural fields. *Journal of Difference Equations and Applications*, 2008. submitted.

[20] N.P. Rougier, D.C. Noelle, T.S. Braver, J.D. Cohen, and R.C. O'Reilly. Prefrontal cortex and flexible cognitive control: Rules without symbols. *Proceedings of the National Academy of Science*, 102(20):7338–7343, 2005.

[21] N.P. Rougier and R.C. O'Reilly. A gated prefrontal cortex model of dynamic task switching. *Cognitive Science*, 2002.

[22] N.P. Rougier and J. Vitay. Emergence of attention within a neural population. *Neural Networks*, 19(5):573–581, 2006.

[23] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal rep-

resentations by error propagation. In D. E. Rumelhart, J. L. McClelland, et al., editors, *Parallel Distributed Processing: Volume 1: Foundations*, pages 318–362. MIT Press, Cambridge, 1987.

[24] S. J.P. Changeux S. Dehaene. A simple model of prefrontal cortex function in delayed-response tasks. *Journal of Cognitive Neuroscience*, pages 244–261, 1989.

[25] J.P. Spencer, V.R. Simmering, A.R. Schutte, and G. Schoner. *Insights from a dynamic field theory of spatial cognition*, chapter What does theoretical neuroscience have to offer the study of behavioral development ? Oxford University Press, 2008.

[26] J.G. Taylor. Neural bubble dynamics in two dimensions: foundations. *Biological Cybernetics*, 80:5167–5174, 1999.

[27] J. Vitay and N.P. Rougier. Using neural dynamics to switch attention. In *International Joint Conference in Neural Networks*, 2005.

[28] A.N. Whitehead and B. Russell. *Principia mathematica*. University Press, Cambridge, 1925.

[29] D. Zipser. Recurrent network model of the neural mechanism of short-term active memory. *Neural Computation*, 3:179–193, 1991.