Cogn Comput (2014) 6:101–112
DOI 10.1007/s12559-013-9212-5

A Multimodal Connectionist Architecture for Unsupervised Grounding of Spatial Language

Michal Vavrečka · Igor Farkaš

Received: 30 August 2012 / Accepted: 12 March 2013 / Published online: 22 March 2013

Abstract We propose a bio-inspired unsupervised connectionist architecture and apply it to grounding the spatial phrases. The two-layer architecture combines by concatenation the information from the visual and the phonological inputs. In the first layer, the visual pathway employs separate 'what' and 'where' subsystems that represent the identity and spatial relations of two objects in 2D space, respectively. The bitmap images are presented to an artificial retina and the phonologically encoded five-word sentences describing the image serve as the phonological input. The visual scene is hence represented by several self-organizing maps (SOMs) and the phonological description is processed by the Recursive SOM that learns to topographically represent the spatial phrases, represented as five-word sentences (e.g., 'blue ball above red cup'). Primary representations from the first-layer modules are unambiguously integrated in a multimodal second-layer module, implemented by the SOM or the 'neural gas' algorithms. The system learns to bind proper lexical and visual features without any prior knowledge. The simulations reveal that separate processing and representation of the spatial location and the object shape significantly improve the performance of the model. We provide quantitative experimental results comparing three models in terms of their accuracy.

Keywords Unsupervised learning · Self-organizing map · Symbol grounding · Spatial phrases · Multimodal representations

M. Vavrečka (✉)
Department of Cybernetics, Czech Technical University, Karlovo náměstí 13, Prague, Czech Republic
e-mail: vavrecka@fel.cvut.cz

I. Farkaš
Department of Applied Informatics, Comenius University, Mlynská dolina, 84248 Bratislava, Slovakia

Introduction

The question of how to acquire, represent and use knowledge in the learning agent is fundamental in artificial intelligence and cognitive science research. Within the modern perspective, fueled by growing empirical evidence, we are looking for a system that, through interaction with the environment, learns the internal representations. These should store the constant attributes and regularities of the environment, giving rise to forming concepts, which become connected to the symbolic level (language). This approach to the representation of meaning differs from the classical symbolic (designer) approach based on formal principles [28, 33], which are subject to the symbol grounding problem (Harnad 1990).

Harnad (1990) proposed a hybrid architecture based on discrimination and identification, where the former process is considered a subsymbolic (non-arbitrary) representation of perceptual inputs, while the latter assigns (non-arbitrary) concepts to (arbitrary) symbols. Harnad used neural networks for the subsymbolic representations and the classical architecture for symbol operations. In the overview of grounding architectures, Taddeo and Floridi [41] introduced the zero semantical commitment condition as a criterion for valid solution to the symbol grounding problem, completely avoiding the designer approach. This criterion, however, appears unsatisfiable not only in artificial, but in living systems as well [46].

In the past two decades, there was a number of different approaches and models of the symbol grounding (e.g., [1, 4, 7, 11, 23, 37, 40, 42]). These models typically ground linguistic symbols by linking them with agent's sensorimotor behavior, or with objects and their features. Other approaches, instead, focus on the social symbol grounding where the symbols become grounded by simulating the cultural evolution in a population of agents (e.g., [38, 52]).

With respect to learning paradigms, we can distinguish two types of connectionist models that link subsymbolic (conceptual) knowledge with (linguistic) symbols. The supervised approach is based on error correction learning in which input patterns are linked with symbolic targets (labels). The models listed above typically have this feature. Both inputs and outputs are assumed to be provided by the environment, and the error information is used to find the desired mapping between them.

On the other hand, the unsupervised approach treats both perceptual stimuli and symbols equally as inputs, to be associated (typically) by Hebbian-like learning. This implies a different way of incorporating the symbolic (lexical) level. The target signal (here the lexical level) only functions as an additional input rather than being the source for error-based learning. The unsupervised models are typically based on self-organizing maps (SOM) that organize (high-dimensional) input vectors according to their similarities [19]. For instance, the DevLex model [20] also consists of two self-organizing networks, one for lexical symbols (phonological representations) and the other for conceptual (semantic) representations, that are bidirectionally connected. They can activate each other, but there is no additional layer for multimodal representations, as opposed to the model proposed here, and some other models of grounding (e.g., [4, 36]).

Dorffner et al. [1] have proposed unsupervised binding between two primary (symbolic and conceptual) layers mediated by the central linking layer. The linking layer (which could be seen as the bimodal layer) interconnects the two primary layers via its localist units that link both representations (i.e., one unit connects one word–concept pair of primary representations). First, one set of links (weights to the linking layer) is trained using a competitive mechanism exploiting the winner-take-all approach. Then, the winner's weights toward the other layer are updated according to the outstar rule [12]. Similarly, to DevLex, these mappings were aimed at simulating word comprehension (the form to meaning) and word production (meaning to the form).

Among the unsupervised approaches, there emerged an alternative to link both the perceptual and symbolic information (treated as an input) with multimodal representations at the output. The example of this architecture is the unsupervised feature-based model that was used to account for early category formation in young infants [9]. Interestingly, this approach postulates the unsupervisory role of linguistic labels that can effect categorization during the acquisition process, which has also been supported by experimental evidence.

The idea of unsupervised binding of two modalities (as inputs) has also been applied in recent generative probabilistic models such as the deep belief net (DBN). The DBN was successfully trained to classify the isolated hand-written digits, so the visual inputs were linked with categorical labels [14]. The linking was established via the training on image–label pairs, using the higher (bimodal) layer that learned the joint distribution of input pairs.

Our model is similar to that of Gliozzi et al. [9] by treating the information from two modalities as input. It differs from it, however, by higher complexity and the task. Our model was designed for grounding the spatial phrases rather than object names (typical for early language learning). We test our model in the area of spatial cognition, similarly to Regier [34], who created a supervised neural network model consisting of several modules to ground the spatial phrases. Regier's model was able to ground both static spatial relations (e.g., left, right) and dynamic relations (e.g., around, through). However, in Regier's model, the symbolic representational level was considered prior and fixed. On the contrary, we focus on unsupervised learning of spatial relations of two objects in 2D space, by linking perceptual information and linguistic description, where neither level is considered prior and fixed. The neural architecture we propose satisfies the requirement that the artificial system (agent) should learn its own functions and representations [53].

In this paper, we describe the 'experimental trajectory' of our work whose aim was to design a bio-inspired model at a reasonable level of abstraction. We converged to a model that processes visual input separately using 'what' and 'where' pathways, which is also a feature of biological systems [45]. The motivation for our model was to experimentally test whether it is possible (without errors) to bind location, color and shape of two objects (Visual Feature-Binding) without any prior knowledge and without external information. The model also proposes a solution to the (unsupervised) symbol grounding that can be considered as a temporal synchrony [6]. In this process, the sequences of symbols (words), describing the spatial layout of two objects and their identity, processed in the phonological layer are grounded (bound) to proper features from the visual subsystem (shape, color and location). Our model exploits the simplification, being the fixed sentence structure that facilitates the thematic role assignment in the model(s).

The benefit of modularity in the model (including that for separation of 'what' and 'where' information) was already emphasized in earlier works. For instance, Jacobs et al. [15] proposed a supervised approach to designing a modular system, composed of competing expert networks and the gating network, that could simultaneously learn two different tasks. They applied their model to the learning of the 'where' and 'what' information (using simple bitmap images) and pointed to the advantages of this modular feed-forward architecture compared to the standard multi-layer perceptron. In our models, the units also compete for inputs,

albeit using the principles of self-organization. Unlike Jacobs et al., the modules are assumed to be given in our models (the competition does not occur at the level of modules, but rather the level of individual units).

The rest of the paper is organized as follows. In section "The Models", we introduce the architecture(s) of our models in a greater detail. Section "Results" presents results from four series of simulations. Section "Discussion" contains the discussion about our final model and its relation to other models. Section "Conclusion" concludes the paper.

The Models

In our model, the representation process takes advantage of the unimodal layers of units. The visual layers represent spatial location, shape and color of objects and the phonological layer represents sentences. The multimodal level integrates the outputs of these unimodal layers. In contrast to the classical approaches that postulate the abstract symbolic level as fixed and prior (defined by the designer), our model offers possibility to learn and modify the phonological layer, visual layer and, consequently, the multimodal level. The schema of the system is depicted in Fig. 1.

In the simulations, we compare different versions of the visual subsystem, analyzing the distinction between 'what' and 'where' pathways. The results help us to decide whether this simplification is important for enhancing the overall model performance. The visual system of our model is therefore tested in three different configurations (see Fig. 1; Table 1): a single SOM that learns to capture both 'what' and in Model 3 information (Model 1), two separate SOMs for 'what' and 'where' information (Model 2), and two separate SOMs with reduced 'where' representations (Model 3).

In the last simulation, we compare two different types of multimodal integration. Inspired by the biological evidence about topographic organization of sensory and motor brain areas, we assume that primary unimodal layers are topographically organized. Although examples of this organizing principle exist in higher areas as well [22], it remains an empirical question whether topographically organized responses are a general principle of the brain at higher levels of organization. In the multimodal layer, we hence compare the SOM and 'neural gas' (NG; [25]) algorithms as representatives of topographic and non-topographic approaches, respectively. Both algorithms are unsupervised, based on the competition among units, but NG uses a flexible neighborhood function, as opposed to the fixed neighborhood in SOM (that enforces topography). The goal was to experimentally investigate the effect of the neighborhood function in the multimodal layer. We used the modified SOM Toolbox [50] for all simulations.

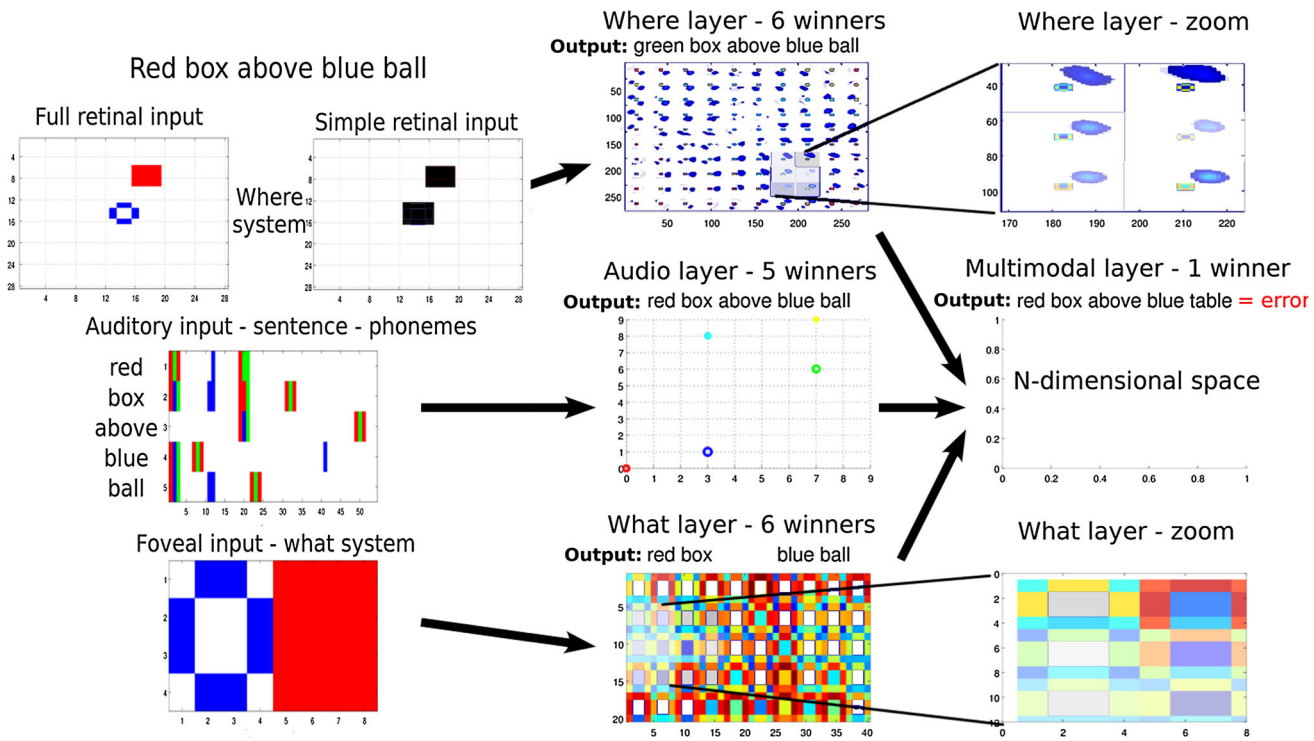

Fig. 1 Multimodal connectionist architecture for grounding spatial phrases. The phonological layer represents sentences and the visual layers represent spatial location, shape and color of objects. The multimodal level integrates the outputs of these unimodal layers

Table 1 Summary of the visual features of the 3 models used in experiments. Each model uses the same phonological subsystem (RecSOM) and can be combined with the SOM or the NG module in the multimodal layer

Model	Visual input	Visual system
1	full	single SOM
2	'where'- full	'where' SOM
	'what' -2 objects' features	'what' SOM
3	'where' -blobs	'where' SOM
	'what' -2 objects' features	'what' SOM

Visual Input

The visual scenes consist of the trajector and the base objects in different spatial configurations. The base position is fixed in the center of the scene (the center of the retina) and the trajector position is located in one (or at the boundary between two) of the spatial quadrants relative to the base. The positions along the main semiaxes are linguistically referred to as up, down, left and right, but perceptually, the trajector position is fuzzy and random. The scene size (artificial retina) contains 28 × 28 pixels and both objects consist of 4 × 4 pixels (Fig. 2a). The color of each pixel is encoded by the activity level, scaled to values between 0 and 1 (0 = white, 0.33 = red, 0.66 = green, 1 = blue). Inputs to the visual SOM (Model 1) or just 'where' subsystem (Model 2 and 3) are the 784-dimensional vectors. Each dimension represents the color information for Model 1 and 2, or monochrome activity (0 = white, 1 = black) for Model 3. Inputs to the 'what' subsystem (Model 2 and 3) are the 32-dimensional vectors. The 'what' system incorporates a simple attentional mechanism and represents the foveal input of two consequently observed objects. Two visual fields (each with 4 × 4 receptors) simultaneously project visual information about the trajector and the base in a fixed position to the unimodal 'what' system. This subsystem represents color and shape of pairs of objects (trajector and base) irrespective of their spatial position.

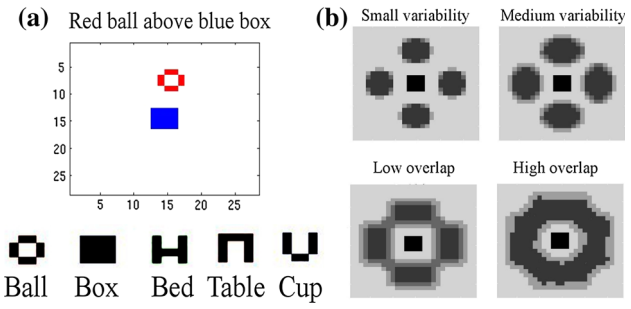

Fig. 2 **a** Example of a visual input scene and the monochrome visual 'vocabulary,' **b** Superimposed visual inputs with varying levels of spatial fuzziness

We trained all models with an increasing combinatorial complexity, starting with simple inputs with two colors, two object types and four spatial relations, up to more complex inputs consisting of three colors (red, green and blue), five object types (box, ball, table, cup and bed) and four spatial relations (above, below, left and right). The most complex scenario with two different objects in the scene amounts to 840 input configurations. The corresponding training set results in 42,000 examples (with 50 instances per input configuration). We also present stimuli with increasing fuzziness in the spatial location to investigate the relation between fuzziness and the error in the visual and multimodal layer. The fuzziness stands for variability of the trajector center with regard to the center of the spatial quadrant ranging from 2 to 8 pixels. The two conditions with the highest degree of fuzziness yield overlapping inputs (as seen in Fig. 2b).

Visual Subsystem

The sensory input of the visual subsystem is captured by an artificial retina that serves as an input to the primary visual layer. Visual layer consists of the SOM(s) that learn the nonlinear mapping of input vectors to output units in the topography-preserving manner (i.e., similar inputs are mapped to neighboring units in the map). The SOM performs standard computations in each iteration. After the presentation of a randomly chosen (rescaled) input vector $\mathbf{x}$, the output y_i of a unit i in the SOM is first computed as

$$y_i = 1 - \|[\mathbf{x}(t) - \mathbf{w}_i(t)\|$$

where $\|.\|$ denotes the Euclidean norm (it will also be used in forthcoming equations), and then, the k-WTA (winner-take-all) rule is applied. According to k-WTA, k most active units are proportionally kept active (with the activity of the best matching unit scaled to 1), and all other units are clamped to 0. In the models, we empirically found the optimal value $k = 6$. The motivation for this type of output representation rests in introducing some overlaps between similar patterns to facilitate generalization.

The output vectors of all unimodal modules are concatenated (including the phonological input) and serve as the input vector to the multimodal layer. For all visual maps, standard computations are performed to update weights. Then, the best matching unit (winner) c is calculated according to

$$c = \arg\min_i \{\|\mathbf{x}(t) - \mathbf{w}_i(t)\|\},$$

the weights in the winner's neighborhood are updated as

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + \mu h_{ci}(t)[\mathbf{x}(t) - \mathbf{w}_i(t)],$$

where μ is the learning rate and $h_{ci}(t)$ is the neighborhood kernel around the winner c, with the neighborhood radius

linearly shrinking over time. Let us now take a more detailed look at these layers and their inputs.

In Model 1, the single SOM was tested whether it could learn to differentiate various positions of two objects, as well as object types and their color. In Model 2, we used separate SOMs for spatial locations (abstraction of the 'where' system) and a separate SOM for color and shape of objects (abstraction of the 'what' system). Model 3 employs the same 'what' and 'where' systems as Model 2, but uses different inputs to the 'where' system consisting of two monochromatic boxes (rather than concrete object shapes in color) in the particular spatial position. The dimension of all visual layers was fixed for all models, namely $\dim(\mathbf{y}^{\text{what}}) = 25 \times 25$ neurons for the 'what' system and $\dim(\mathbf{y}^{\text{where}}) = 30 \times 30$ neurons for the 'where' system. The similar size of matrices were estimated from previous simulations [47], and they also stem from the number of combinations in the most complex scenario (840 combinations in the 'where' system and 210 in the 'what' system). All SOM maps have a hexagonal neighborhood function and the lattices have a toroid topology. The overview of the characteristics of the three models is summarized in Table 1.

Phonological Input

Phonological input (English sentences) was encoded as high-dimensional patterns representing word forms using PatPho, a generic phonological pattern generator that fits every word (up to three syllables) onto a template according to its vowel–consonant structure [21]. PatPho uses the concept of a syllabic template: a word representation is formed by combinations of syllables in a metrical grid, and the slots in each grid are made up by bundles of features that correspond to consonants and vowels. Word representations can hence be compared according to their phonological similarities. In our case of 5-word sentences, each sentence consists of five 54-dimensional vectors with component values in the interval (0,1).

Phonological Subsystem

The phonological input is fed (one vector at a time) to the RecSOM [51], a recurrent SOM architecture, that uses a detailed representation of the context information (the whole output map activation) and has been demonstrated to be able to learn to represent much richer dynamical behavior [44], in comparison with other recurrent SOM models [13]. RecSOM learns to represent the input (words) in the temporal context (hence, capturing the sequential information). RecSOM output, in terms of the map activation, feeds to the multimodal layer, being integrated (by vector concatenation) with the visual pathway. Like SOM, RecSOM is trained by a competitive, Hebbian-like learning algorithm. As a property of the RecSOM, its units become the sequence detectors after training, topographically organized according to the suffix (the most recent words).

Formally, each neuron $i \in \{1, 2, \ldots, N\}$ in RecSOM has two associated weight vectors: $\mathbf{w}_i \in \mathcal{R}^n$ – linked with an n-dimensional input $\mathbf{s}(t)$ (in our case, the current word, with dimension $n = 54$) feeding the network at time t, and the weight vector $\mathbf{c}_i \in \mathcal{R}^N$ – linked with the context $\mathbf{y}(t-1) = [y_1(t-1), y_2(t-1), \ldots, y_N(t-1)]$ containing the unit activations $\mathbf{y}_i(t-1)$ from the previous time step. The output of a unit i at time t is $y_i(t) = \exp(-d_i(t))$, where

$$d_i(t) = \alpha \|\mathbf{s}(t) - \mathbf{w}_i\|^2 + \beta \|\mathbf{y}(t-1) - \mathbf{c}_i\|^2.$$

Here, $\alpha > 0$ and $\beta > 0$ are the model parameters that, respectively, influence the effect of the input and the context upon the neurons profile. Their suitable values are usually found heuristically (in our model, we use $\alpha = \beta = 0.1$). Both weight vectors are updated using the same form of a SOM learning rule

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + \gamma h_{ci}(\mathbf{s}(t) - \mathbf{w}_i(t)),$$
$$\mathbf{c}_i(t+1) = \mathbf{c}_i(t) + \gamma h_{ci}(\mathbf{y}(t-1) - \mathbf{c}_i(t)),$$

where $c = \arg\min_i\{d_i(t)\}$, is the winner index at time t, and $0 < \gamma < 1$ is the learning rate. (The winner can be equivalently defined as the unit c with the highest activation $y_c(t)$: $c = \arg\max_i\{y_i(t)\}$). The neighborhood function h_{ci} is a Gaussian (of width σ) on the distance $d(i, c)$ of units i and c in the map: $h_{ci} = \exp(-d(c,i)^2/\sigma^2)$. The neighborhood width σ linearly decreases in time to allow the formation of topographic representation of input sequences. After training, all RecSOM units become sensitive to particular sentences, ordered topographically according to sentence endings. The output vector is composed of five consecutive winners representing particular words in the sentence. The activations of winning units are slowly decayed in time (decreased by value 0.1 at each step) toward the end of a sentence. This function allows to represent the order of winners in the sequence, hence differentiating between similar phonetic features in a sentence (e.g., 'red ball above red table' or 'blue ball above red ball'). The size of RecSOM was set to $N = 20 \times 20$ neurons for all models based on results from previous simulations.

Multimodal Layer

The multimodal layer is the core of the system, since it learns to identify unique categories and represent them. The main task for this layer is to process the output from the unimodal layers and to find and learn the categories by mapping different sources of information (visual and phonological) that refer to the same objects in the external

world. Input vectors $\mathbf{m}(t)$ for the multimodal layer are taken as concatenated unimodal activation vectors (the 'where' and 'what' components are not separated in Model 1) using the above-mentioned k-WTA mechanism, explained in "Visual Subsystem", i.e.,

$$\mathbf{m}(t) = [\mathbf{y}^{\text{where}}(t); \mathbf{y}^{\text{what}}(t); \mathbf{y}^{\text{phono}}(t)].$$

The multimodal module receives a 1,300-dimensional input in Model 1 and a 1925-dimensional input in Model 2 and 3. Unlike sparse localized output codes ($k = 6$) used at the unimodal layer (to facilitate generalization), the output representation in the multimodal layer with the WTA mechanism is chosen to be localist ($k = 1$) for better interpretation of results and the error calculation.

We tested two unsupervised algorithms in the multimodal layer, SOM and NG, that differ in the neighborhood function. The size of the multimodal layer was set to allow a distinct localist representation of all 840 object combinations in the most complex data set, so we used 841 neurons (arranged in a 29 × 29 grid in case of SOM).

For clarity, we explain the NG algorithm briefly here. NG shares a number of features with the SOM. In each iteration t, an input vector $\mathbf{m}(t)$ is randomly chosen from the training dataset. Subsequently, we compute $d_i(t) = \|\mathbf{m}(t) - \mathbf{z}_i\|$ for all units, and then, we sort the units according to their increasing distances d_i, using indices $l = 0, 1, \ldots, N-1$ (where $l(0)$ corresponds to the current winner's index). We then update all weight vectors $\mathbf{z}_i$ according to

$$\mathbf{z}_i(t+1) = \mathbf{z}_i(t) + \eta \exp(-l(i)/\lambda)(\mathbf{m}(t) - \mathbf{z}_i(t))$$

with η being the learning rate and λ the so-called neighborhood range. We used $\eta = 0.5$ and $\lambda = n/2$ where n is the number of neurons. Both parameters are reduced with increasing t. It is known that after sufficiently many adaptation steps, the feature vectors cover the data space with minimum representation error [25]. Mathematically, the adaptation step of the NG can be interpreted as the gradient descent on a cost function.

Quantification of the Model Accuracy

To quantify the model accuracy, we designed the following procedure for computing the classification error. After the model has been trained, we again make a single sweep through the training set, in order to label all neurons, reflecting their responsiveness to each of the five input features (base color, base shape, spatial location, trajector color, trajector and shape). We attach five counter arrays $c_f^{(i)}(j)$ to each neuron, initialized to zeros, each consisting of $n(f_j)$ slots, corresponding to the number of different (possible) values of the feature f_j (depending on the task complexity), i.e., $j = 1, 2, \ldots, n(f_j)$. For each training input pattern, we find the winner (as in the SOM algorithm) whose five counter values are increased by one (i.e., for each current feature value). After sweeping through the training set, we assign unique feature labels to all neurons by applying the 'maximum response principle,' according to which each neuron becomes a representative of only the most frequent value of the given feature (for which that neuron became the winner most often), i.e., $f_{j^*}^{(i)} = \arg\max_j \{c_f^{(i)}(j)\}$.

Then, we can measure the model accuracy, as the percentage of correctly classified test inputs. The feature of the testing pattern is considered to be correctly classified, if it matches the winner's representative feature. The calculation of the classification error rate is first made for each feature separately and then also for the whole scene-sentence input (overall error) that requires that all features in the testing input be correctly classified.

In the case of the sequential RecSOM, in addition to the classification error, we also compute the confusion error. It occurs if the same neuron wins more than once during a sentence, most typically in case of multiple occurrences of the same word in a sentence, e.g., in 'red box above red ball,' or 'red ball below blue ball.' So whenever the same winner occurs twice, we increase the error counter by one. The confusion error stands for the percentage of examples with the same winner and it helps to detect erroneous cases not revealed by the classification error.

Results

We present results corresponding to the three models as described in "The Models", tracking our 'experimental trajectory,' along which we eventually converged to the architecture with SOM maps in visual subsystems and NG in the multimodal layer. We trained each model for 100 epochs and tested it with a novel set of inputs. For each run, the data set was randomly split to the training and testing subsets using the 70:30 ratio.

Model 1

In Model 1, the single SOM in the visual system is tested whether it can learn to represent all visual features simultaneously. We observe a high error in this system for the trajector features, because trajector positions overlap in the specific area. Errors for trajector color (37 %) and trajector shape (65 %) are rather high. Although the spatial location of the trajector is fuzzy, the error for this feature in the test set is the lowest (14 %). Low errors also result for base color (18 %) and base shape (28 %). We also test whether the level of fuzziness (shown in Fig. 2b) affects the error in the SOM map. All features except for the spatial location are not sensitive to the fuzziness level, as the errors vary

within a 3 % range. On the other hand, the error for spatial location correlates with the fuzziness starting from 3 % for fixed position of the trajector to 14 % for highly overlapping spatial locations.

The phonological RecSOM layer performs better compared to the visual layer because the phonetic features, being sequentially fed to the system, are not fuzzy. There are 0 % errors for base color, base shape, trajector color and spatial term. Error for the trajector shape is 1 %. On the other hand, there is a 22 % confusion error, which results in the confusion of the neuron response (see the illustrative Fig. 3 with only 6 × 6 neurons) and increases the error in the multimodal layer. There are only 4 winning neurons in case of confusion (the same neuron wins twice within one sentence). We observe that this problem can partially be eliminated using the decayed winner activation of winners (as described in "Phonological Subsystem"). It would be possible to solve this problem more reliably by excluding the winner from competition until the rest of the sentence (as, e.g., done in [16]).

The performance of the multimodal layer heavily depends on the effectiveness of unimodal layers. The errors for the representation of trajector color (8 %), base color (1 %) and base shape (2 %) are low. On the other hand, there are high errors for both the trajector shape (46 %) and spatial term (25 %). This is due to poor performance of the visual layer. The overall error of the system reaches 68 %.

Model 2

Model 2 processes 'what' and 'where' information using separate SOMs, and we identify a difference in accuracy between the two systems. The 'what' system outperforms the 'where' system, as documented by low errors for base color (1 %), base shape (8 %), trajector color (0 %) and trajector shape (5 %). We did not test the performance of the 'what' system for the spatial term simply because that information was not made available to this system. The errors for the 'where' and phonological systems are identical to Model 1, because these layers receive the same input as in Model 1. Notably, the additional 'what' layer changed the performance of the multimodal layer. Errors for base color (2 %) and base shape (4 %) in the multimodal layer remain the same as in Model 1, but lower errors are observed for trajector color (1 %) and trajector shape (5 %). On the other hand, the system exhibits a much higher error for the spatial term (71 %) compared to Model 1 (25 %). The multimodal SOM layer is probably not able to merge the information from three unimodal layers. The overall error is 75 %, caused by the problem with the

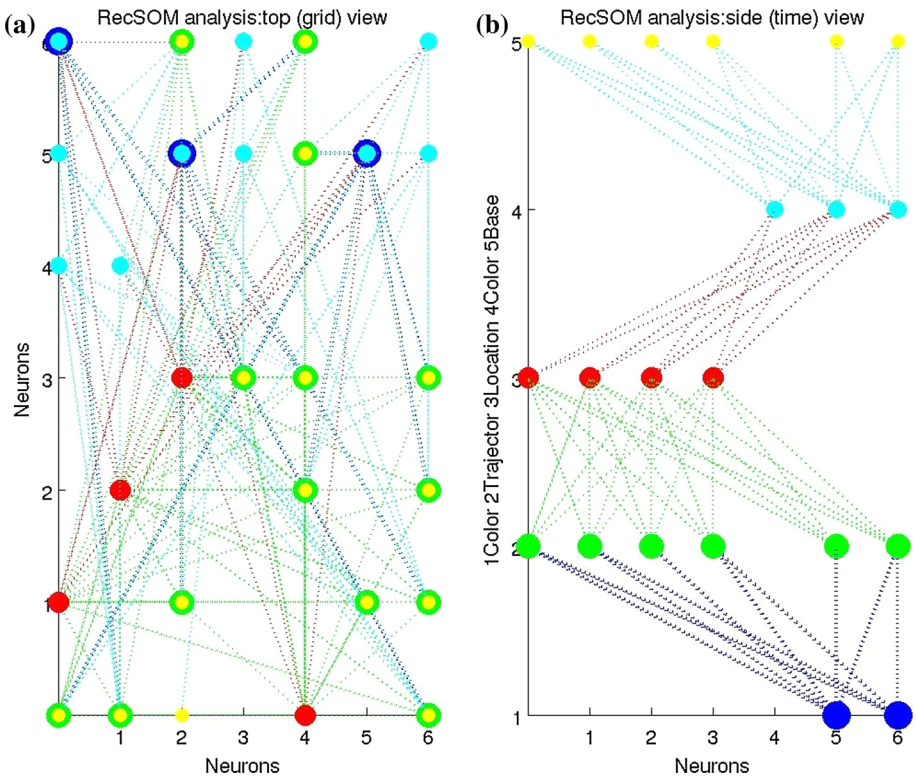

Fig. 3 Unit responses in the phonological layer of Model 1. If the same neuron responds to the same feature (e.g., *shape*) of the trajector and the base (shown by *overlapping dots*), it will increase the error for the whole scene/sentence as well. **a** Visualization of the RecSOM grid (time is represented *bottom-up* by the size of the *dot*; **b** The time course of sentence processing (*y*-axis) in the *bottom-up direction*

representation of the spatial term. A more detailed analysis is explained in the Discussion section.

Model 3

The simplification of inputs to the 'where' system is achieved by using monochromatic bounding boxes instead of object shapes and colors. This expectedly led to a lower error (8.3 % in the most complex and fuzzy scenario) compared to full retinal images (see Fig. 4). We do not compare the results for object features (shape and color), because there is no information about them provided to the 'where' system in Model 3. The analysis of the SOM structure revealed a better organization of specific clusters in favor of bounding box inputs for the spatial term representation. These results lead us to the conclusion that it is possible to simplify the information projected to the 'where' system to optimize the speed and effectiveness of the model. However, the simplification of the 'where' inputs does not affect the performance of the multimodal layer. There are similar results for the object features, spatial term (70 %) and also the overall error (74 %). Therefore, we tested the NG algorithm in the multimodal layer in further simulations trying to improve the accuracy.

Comparison of SOM and NG in Multimodal Layer

We compare the effectiveness of the SOM and NG algorithms in the multimodal layer for all three models. We observe a different type of clustering in the unimodal layers that are transferred to the multimodal layer, where the SOM is not able to adapt to the concatenated outputs from unimodal layers, apparently due to neighborhood constraints (Model 1SOM and 2SOM). The results of the NG algorithm (Model 1NG and 2NG) for the same input data confirm this hypothesis. The multimodal layer based on NG is able to correctly map all the object features except spatial term without any problem. There is a 0 % error for both simplified inputs (Model 3NG) and also for full retinal projections to the 'where' system (Model 2NG). The errors for the multimodal NG module and the single SOM in the visual layer (Model 1NG) are as follows: 1 % for base color, 2 % for base shape, 6 % for trajector color and 26 % for trajector shape. These results are significantly better than those for the multimodal SOM. Surprisingly, we observe the lowest error for the representation of the spatial term in the multimodal layer for NG algorithm and a single SOM visual layer (Model 1NG). There is a 12 % error compared to 24 % for Model 2NG (see Fig. 5) and 32 % for Model 3NG (see Table 2). The SOM algorithm leads to higher errors of the spatial term for both models, namely 25 % (Model 1SOM), 70 % (Model 2SOM) and 73 % (Model 3SOM). These results are contradictory, because Model 2SOM and 3SOM with separate 'what' and 'where' systems perform better for all features except the spatial term (see Discussion). Preliminary results of this comparison were also presented in Vavrečka, Farkaš and Lhotská [49].

The comparison of the overall accuracy (overall error) is shown in Fig. 6 and Table 2. The best results are obtained for 'what' and 'where' subsystems and the NG algorithm in the multimodal layer (Model 2NG). There is a 25 % error compared to 70 % overall error for the multimodal SOM in the most complex scenario. Hence, the better, albeit not perfect, results are achieved with NG by sacrificing the topographic organization of responses in the multimodal layer.

The last analysis is dedicated to the comparison of SOM (Model 3SOM) and NG (Model 3NG) algorithms in the

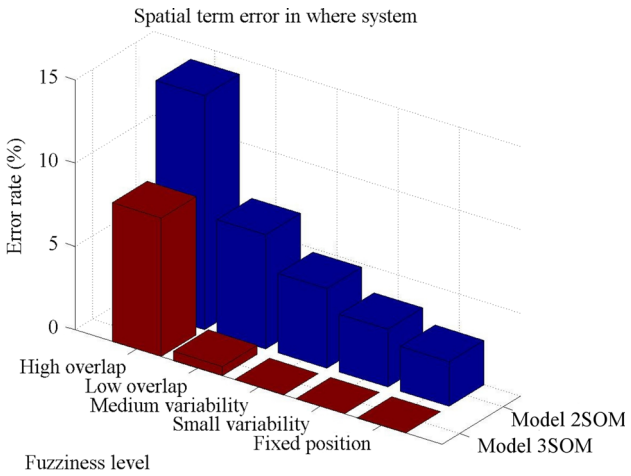

Fig. 4 Visualization of spatial term errors in the 'where' layer for full retinal inputs (*blue*) and for bounding box inputs (*red*) as a function of the fuzziness level of trajector spatial location

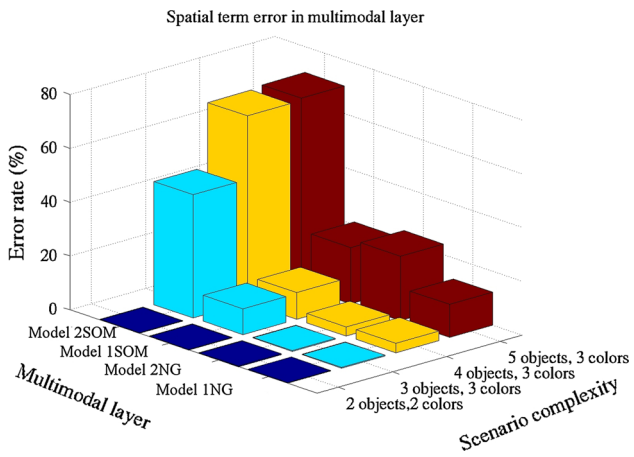

Fig. 5 Comparison of the errors in the multimodal layer for the representation of the spatial term. Model 1NG (NG in the multimodal layer and a single SOM in the visual system) performs best

Table 2 Summary of error rates for specific layers and models

Subsystem	Model	TrajCol	TrajShape	SpatTerm	BaseCol	BaseShape	Overall
Where	1,2SOM; 1,2NG	39.3	68.2	14.2	19.2	30.5	91.7
	3SOM; 3NG	-	-	8.3	-	-	-
What	2,3SOM; 2,3NG	0.4	5.3	-	0.9	0.8	-
Phono	1,2,3SOM; 1,2,3NG	0.0	1.2	0.2	0.0	0.0	12.3
Multimodal	1SOM	8.3	46.0	24.6	0.5	2.0	68.3
	2SOM	0.9	5.4	70.3	1.9	3.8	74.7
	3SOM	0.9	4.1	72.7	1.2	1.6	75.3
	1NG	5.6	26.4	12.3	0.6	1.7	41.5
	2NG	0.0	0.3	24.0	0.0	0.0	24.3
	3NG	0.0	0.0	31.5	0.0	0.0	31.5

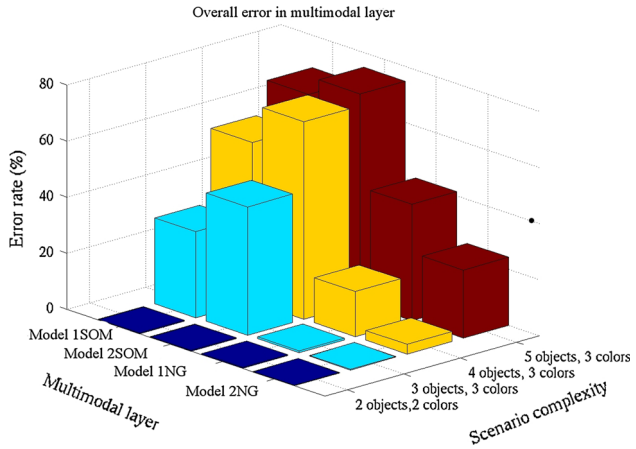

Fig. 6 Errors in the multimodal layer for whole scene (overall) representation. Model 2NG based on 'what' and 'where' visual system and NG in multimodal layer performs best

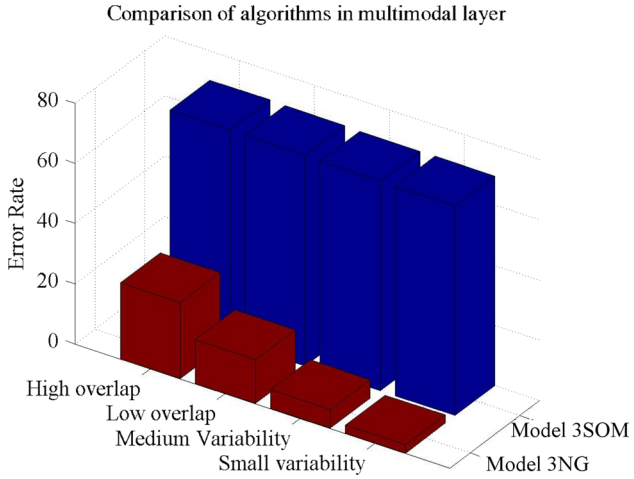

Fig. 7 Errors in the multimodal layer for SOM (Model 3SOM) and NG (Model 3NG) algorithms as a function of the fuzziness level of the trajectors' spatial location (see Fig. 2b)

multimodal layer that have to process different levels of spatial fuzziness. Fig. 7 reveals a lower error for NG at all levels of fuzziness and the high errors for SOM regardless of the fuzziness level (70 %). Hence, the multimodal SOM is unable to unambiguously represent neither fuzzy nor distinct inputs.

Discussion

We analyze the presented models in the context of theoretic assumptions, especially the perceptual theory of cognition and conceptual approaches to knowledge representation. We also discuss various aspects of our model, its relation to other models and the features of visual feature-binding and temporal synchrony [6].

Architecture

We should also compare our architecture with the system for the representation of spatial relations developed by Regier [34]. The main difference lies in the unsupervised manner of our architecture compared to the supervised approach adopted by Regier. His system is composed of specific modules for the calculation of angle between trajector and base, an object's intersection and dynamic properties in motion inputs. It resembles the designer's approach described in Ziemke [53] as there is modular architecture engineered for the specific task. Our system is more generic and biologically inspired as the subsystems copy the information processing in human brain (unsupervised learning, 'what' and 'where' pathways, multimodal integration). The advantage of Regier's system is the ability to represent dynamic spatial relations (around, through, etc.). On the other hand, our unsupervised architecture based on RecSOM [51] in a visual subsystem and the growing-when-required networks [24] in the phonological and multimodal layer was able to process visual sequences (around, through, outside, over and under) and it reached 88 % accuracy [48].

In our model, the representations take advantage of the two or three unimodal layers of units. The phonological layer represents unique labels (linguistic terms), whereas the visual 'where' subsystem represents fuzzy information about the spatial locations of objects in the external world and the 'what' subsystem captures shapes and colors of objects in a fixed foveal position. The multimodal level integrates the outputs of these unimodal layers. The grounded meaning is simultaneously represented by all layers (phonological, visual and multimodal), making this approach resemble the theory of Peirce [30] who defined basic components of a sign—representamen and interpretant. Our model represents the sign hierarchically guaranteeing better processing and storing of representations, because the sign (the multimodal level) is modifiable from both modalities (the sequential 'representamen' via the phonological level and the parallel 'interpretant' via the visual level). This feature makes the units in the higher layer bimodal (i.e., they can be stimulated by any of the primary layers) and their activation can be forwarded for further processing. Bimodal (and multimodal) neurons are known to be ubiquitous in the association areas of the brain [39]. The multimodal layer is formed by exploiting the concept of self-organized conjunctive representations that have been hypothesized to exist in the brain with the purpose of binding the features such as various perceptual properties of objects [26]. We adhere to the view that conjunctive neurons, as an alternative to mechanisms of temporal synchrony, are the plausible connectionist approach for addressing the binding problem [29]. Here, we extend the concept of binding by linking the subsymbolic and symbolic information. Hence, each output unit learns to represent a unique combination of perceptual and symbolic information.

Visual Feature-Binding

Our Models 2 and 3 propose the unsupervised solution to the visual feature-binding, based on the integration of the 'what' and 'where' pathways. With respect to the visual feature-binding [6], the model is based on convergent hierarchical coding, also called combination coding [35]. The neurons react only to combinations of features, that is, to an object of a particular shape and color at a particular retinal position (localist representation). Hierarchical processing implies that increasingly complex features are represented by higher levels in the hierarchy. Complex objects and situations are constructed by combining simpler elements. On the other hand, the convergent hierarchical coding requires as many binding units as there are distinguishable objects. It should result in a combinatorial explosion for large-scale simulations. Our model is able to represent 840 combinations, but it can also suffer from the combinatorial explosion because we represent pairs of objects instead of separate entities in the primary visual layers. In case of 10 objects, 5 colors in 4 spatial locations, we would need to represent 2450 object pairs in a primary 'what' system, instead of 50 separate objects. It is also possible to add a separate layer for the color processing, in which case there will only be 10 objects presented in the 'what' system. Alternatively, we could represent the features in the activity of a population of neurons distributed within and across levels of the cortical hierarchy as the distributed representation [8], although some authors have raised the question whether the combinatorial explosion is really a problem [10]. It is estimated that the number of objects, scenarios, colors and other features in the brain is approximately 10 million items. It is obviously beyond the limits of recent cognitive systems, but it is below the number of neurons in the mammalian visual cortex, so the combination coding could be a sufficient method. It could also be possible to adopt Neural Modeling Fields [31], the unsupervised learning method based on Gaussian mixture models that arguably does not suffer from combinatorial complexity. The application of this theory to the area of symbol grounding resulted in 95 % accuracy of the system that learned the repertoire of 112 actions [5].

Temporal Synchrony

Our model is able to map the words in the sentence with the fixed grammar to the objects in the environment without any prior knowledge (temporal synchrony). Previous models of symbol grounding [2–5] deal with the lexical level, but our model goes beyond words because it can represent sentences in RecSOM. The ability of temporal synchrony can be considered as an extension of the symbol grounding. Cangelosi et al. [2] recommend to ground-specific words at the first stage (sensorimotor toil) and then compositionally chain them at the grounded language level (symbolic theft). There are separate objects presented to their system within a training phase, grounding basic object features. Our approach can be considered an alternative to this theory. We also ground words in the first stage, but unlike the mentioned approach, we present sentences as linguistic inputs to be bound with proper features from the visual subsystem (shape, color and location). Compared to the classic sensorimotor toil experiments based on the grounding of two features, our system is able to ground 5 features simultaneously, which speeds up the process of symbol grounding (faster acquisition of the grounded lexicon). Tikhanoff [43] proposed an architecture (and implemented it in iCub robot) that was able to understand basic sentences, but it was based on supervised learning. Our model is a proof of concept that unsupervised architectures can also find proper mapping between multiple visual and lexical features. We are able to

build representations solely from sensory inputs, arguing that the co-occurrence of inputs from the environment is a sufficient source of information to create an intrinsic representational system.

Performance

The analysis of the model behavior revealed that the trajector shape and the spatial term representations are the most difficult subtasks for visual unimodal systems. The difficulty is caused by the variability and fuzziness of these inputs. The correct representation of the trajector shape requires a separate unimodal 'what' system. The errors for (both SOM and NG) Model 1, 2 and 3 confirm the necessity of the 'what' system in the complex environment because we observe a 60 % increase of errors in the model without a separate 'what' system. On the other hand, the error for the spatial term in Model 2 and 3 reflects some problems with an increasing number of inputs from different subsystems to the multimodal layer, because there is a lower error for Model 1 compared to Model 2 and 3 (both SOM and NG). The problem could reside in the number of dimensions. The multimodal module receives a 1300-dimensional input in Model 1 and a 1925-dimensional input in Model 2 and 3. The increase of dimensionality together with a localist unimodal output function may decrease the effectiveness for the spatial term representation, although other features are represented better in a high-dimensional space. This contradiction has to be investigated in greater detail.

The results for specific algorithms in the multimodal layer confirm our hypothesis that the SOM algorithm, based on the fixed neighborhood function, is not able to adapt to the joint distribution of the outputs from unimodal layers. The SOM-based models aim at the topology-preserving property for the input data, but they are weak with regard to properly representing clusters with different non-uniform data distributions [18]. On the other hand, the NG algorithm is not subject to topographic constraints and, thus, leads to better clusters. Our results are also in line with Pezzulo and Calvi [32], who conclude that perceptual symbols may not be topographically organized, although some parts of the perceptual and motor areas show topographic hierarchical organization. Grounding models based on topographically organized connectionist networks (e.g., [17]) to simulate the perceptual symbol system also exist, but our results do not confirm this assumption for more complex inputs.

The mapping in our models is actually a clustering process that makes the system also vulnerable to errors in the input space. Successful clustering presumes that at least one modality provides distinct activation vectors for different classes to drive the clustering process (i.e., the classes are well separable in the corresponding input subspace). On the other hand, the occurrence of both phonological and visual fuzzy inputs is rare in the real world, so our system could be considered a step toward solving the symbol grounding problem (at least at this small scale).

Conclusion

We have created an unsupervised connectionist system that is able to extract constant attributes and regularities from the environment and link them with abstract symbols. The meaning is non-arbitrarily represented at the conceptual level that guarantees the correspondence of the internal representational system with the external environment. We can also conclude that it is advantageous to follow the biologically inspired hypothesis about the processing of visual information in separate subsystems. The question for future research is to find a proper way of output coding from the unimodal layers to increase system accuracy and to scale up the model. The main advantage of our model is the hierarchical representation of the sign components.

Acknowledgments This work has been supported by the research program MSM 6840770012 of the CTU in Prague, SAIA scholarship and GAČR Grant P407/11/P696 (M.V.) and by VEGA Grant 1/0439/11 (I.F.).

References

1. Dorffner G, Hentze M., Thurner G. A connectionist model of categorization and grounded word learning. In: Koster C, Wijnen F, editors. Proceedings of the groningen assembly on language acquisition (GALA'95), 1996.
2. Cangelosi A, Greco A, Harnad S. From robotic toil to symbolic theft: grounding transfer from entry-level to higher-level categories. Conn Sci. 2000;12(2):143–62.
3. Cangelosi A, Parisi D. The processing of verbs and nouns in neural networks: insights from synthetic brain imaging. Brain Lang. 2004;89(2):401–08.
4. Cangelosi A, Riga T. An embodied model for sensorimotor grounding and grounding transfer: experiments with epigenetic robots. Cogn Sci. 2006;30(4):673–89.
5. Cangelosi A, Tikhanoff V, Fontanari JF, Hourdakis E. Integrating language and cognition: a cognitive robotics approach. IEEE Comput Intell Mag. 2007;2(3):65–70.
6. Feldman J. The neural binding problem(s). Cogn Neurodyn. 2012; doi:10.1007/s11571-012-9219-8.
7. Fontanari JF, Tikhanoff V, Cangelosi A, Ilin R, Perlovsky LI. Cross-situational learning of object-word mapping using neural modeling fields. Neural Netw. 2009;22:579–85.
8. Goldstein EB. Wahrnehmungspsychologie. Heidelberg: Spektrum Akademischer Verlag, 2002.
9. Gliozzi V, Mayor J, Hu J-F, Plunkett K. Labels as features (not names) for infant categorization: a neurocomputational approach. Cogn Sci. 2009;33(4):709–38.
10. Ghose GM, Maunsell J. Specialized representations in visual cortex: a role for binding? Neuron 1999;24:79–85.
11. Greco A., Caneva C. Compositional symbol grounding for motor patterns. Front Neurorobot, 2010;4(111), doi:10.3389/fnbot.2010.00111.

12. Grossberg S. Competitive learning: from interactive activation to adaptive resonance. Cogn Sci 1987;11(1):23–63.
13. Hammer B, Micheli A, Sperduti A, Strickert M. Recursive self-organizing network models. Neural Netw. 2004;17(8–9): 1061–85.
14. Hinton G, Osindero S, Teh Y. A fast learning algorithm for deep belief nets. Neural Comput. 2006;18:1527–54.
15. Jacobs RA, Jordan MI, Barto AG. Task decomposition through competition in a modular connectionist architecture: the what and vision tasks. Cogn Sci. 1991;15(2):219–50.
16. James DJ, Miikkulainen R. SardNet: A self-organizing feature map for sequences. Adv Neural Inf Process Syst 1995;7:577–84.
17. Joyce D, Richards L, Cangelosi A, Coventry KR (2003) On the foundations of perceptual symbol systems: specifying embodied representations via connectionism. In: Detje F, Drner D, Schaub H, editors. The logic of cognitive systems. Proceedings of the fifth international conference on cognitive modeling, Universitaetsverlag Bamberg, pp. 147–52.
18. Kim B, Sang-Woo B, Minho L. Growing fuzzy topology adaptive resonance theory models with a pushpull learning algorithm. Neurocomputing. 2011;74(4):646–55.
19. Kohonen T. Self-Organizing Maps, 3rd edn. Berlin: Springer; 2001.
20. Li P, Farkaš I, MacWhinney B. Early lexical development in a self-organizing neural network. Neural Netw. 2004;17(8–9): 1345–62.
21. Li P, MacWhinney B. PatPho: a phonological pattern generator for neural networks. Behav Res Methods Instrum Comput. 2002;34:408–15.
22. Malach R, Levy I, Hasson U. The topography of high-order human object areas. Trends Cogn Sci. 2002;6(4):176–84.
23. Marocco D, Cangelosi A, Fischer K, Belpaeme T. Grounding action words in the sensorimotor interaction with the world: experiments with a simulated iCub humanoid robot. Front Neurorobot. 2010;4(7), doi:10.3389/fnbot.2010.00007.
24. Marsland S, Shapiro J, Nehmzow U. A self-organising network that grows when required. Neural Netw. 2002;15(8–9):1041–58.
25. Martinetz T, Berkovich S, Schulten K. "Neural-gas" network for vector quantization and its application to time-series prediction. IEEE Trans Neural Netw. 1993;4(4):558–69.
26. Mel B, Fiser J. Minimizing binding errors using learned conjunctive features. Neural Comput. 2000;12:247–78.
27. Miikkulainen R. Dyslexic and category-specific aphasic impairments in a self-organizing feature map model of the lexicon. Brain Lang. 1997;59:334–66.
28. Newell A, Simon HA. Human problem solving. Englewood Cliffs: Prentice-Hall; 1972.
29. O'Reilly RC, Busby RS, Soto R. Three forms of binding and their neural substrates: alternatives to temporal synchrony. In: Cleeremans A, editor. The unity of consciousness: binding, integration, and dissociation. Oxford: Oxford University Press, 2003; 168–92.
30. Peirce, C.S. Collected papers of Charles Sanders Peirce. In Hartshorne C, editor. Harvard University Press, 1931.
31. Perlovsky LI. Neural networks and intellect: using model-based concepts. Oxford University Press, New York, 2001.
32. Pezzulo G, Calvi G. Computational explorations of perceptual symbol systems theory. New Ideas Psychol. 2011;29:275–297.
33. Pylyshyn Z. Computation and cognition: towards a foundation for cognitive science. Cambridge: MIT Press; 1984.
34. Regier T. The human semantic potential: spatial language and constrained connectionism. Cambridge: MIT Press; 1996.
35. Riesenhuber M, Poggio T. Neural mechanisms of object recognition. Curr Opin Neurobiol. 2002;12:162–168.
36. Roy D. Grounding words in perception and action: computational insights. Trends Cogn Sci. 2005;9:389–396.
37. Roy D, Pentland A. Learning words from sights and sounds: a computational model. Cogn Sci 2002; 26:113–146.
38. Steels L, Kaplan F. Situated grounded word semantics. In: Proceedings of the 16th international joint conference on artificial intelligence, vol 2. 1999; p. 862–67.
39. Stein B, Meredith M. Merging of the senses. Cambridge: MIT Press; 1993.
40. Sugita Y, Tani J. Learning semantic combinatoriality from the interaction between linguistic and behavioral processes. Adapt Behav 2005; 13(1):33–52.
41. Taddeo M, Floridi L. The symbol grounding problem: a critical review of fifteen years of research. J Exp Theor Artif Intell. 2005;17(4):419–45.
42. Tikhanoff V, Cangelosi A, Fitzpatrick P, Metta G, Natale L, Nori F. An open-source simulator for cognitive robotics research: the prototype of the iCub humanoid robot simulator. In: Performance metrics for intelligent systems (PerMIS) workshop, 2008; p. 57–61.
43. Tikhanoff, V. Development of cognitive capabilities in humanoid robots. PhD thesis. School of Computing, Communications & Electronics, University of Plymouth, UK, 2009.
44. Tiňo P, Farkaš I, van Mourik J. Dynamics and topographic organization in recursive self-organizing map. Neural Comput. 2006;18:2529–67.
45. Ungerleider LG, Mishkin M. Two cortical visual systems. In: Ingle DJ et al. editors. Analysis of visual behavior. MIT Press, Cambridge; 1982.
46. Vavrečka M. Symbol grounding in context of zero semantic commitment (in Czech). In: Kelemen J, Kvasnička V, editors. Kognice a umělý život VII. (1st ed.) Opava : Slezská univerzita 2006; 365–377.
47. Vavrečka, M. Grounding of spatial terms (in Czech). In: J. Kelemen J, Kvasnička V, editors. Kognice a umelý život VII, Opava: Slezsk univerzita, 2007; p. 365–77.
48. Vavrečka M. Application of cognitive semantics in the model of the spatial terms representation (in Czech). PhD thesis, Masaryk University in Brno, Czech Republic (2008).
49. Vavrečka M, Farkaš I, Lhotská L. Bio-inspired model of spatial cognition. In Lecture notes in computer science 7062 LNCS (Part 1), 2011;443–450.
50. Vesanto J, Himberg J, Alhoniemi E, Parhankangas J. Self-Organizing Map in Matlab: the SOM Toolbox. In: Proceedings of the Matlab DSP conference, 2000; p. 35–40.
51. Voegtlin T. Recursive self-organizing maps. Neural Netw 2002; 15(8–9):979–91.
52. Vogt P, Divina F. Social symbol grounding and language evolution. Interact Stud. 2007;8:31–52.
53. Ziemke T. Rethinking grounding. In: Riegler A, Peschl M, von Stein A, editors. Understanding representation in the cognitive sciences. New York: Plenum Press; 1999. p. 177–90.