Mapping language to vision in a real-world robotic scenario

DOI:
10.1109/TCDS.2018.2819359

Document Version
Accepted author manuscript

Citation for published version (APA):
Stepanova, K., Klein, F. B., Cangelosi, A., & Vavrecka, M. (2018). Mapping language to vision in a real-world robotic scenario. *IEEE Transactions on Cognitive and Developmental Systems*. https://doi.org/10.1109/TCDS.2018.2819359

Published in:
IEEE Transactions on Cognitive and Developmental Systems

Mapping the language to vision in a real-world robotic scenario

Karla Štěpánová[1], Frederico B. Klein[2], Angelo Cangelosi[2], and Michal Vavrečka[1]

[1] Czech Institute of Informatics, Robotics, and Cybernetics, Czech Technical University, Prague, Czech Republic
[2] School of Computing, Electronics and Mathematics,
Plymouth University, Plymouth, UK
{stepakar, vavrecka}@fel.cvut.cz
{frederico.klein, a.cangelosi}@plymouth.ac.uk
http://www.plymouth.ac.uk
http://www.fel.cvut.cz

Abstract Language has evolved over centuries and was gradually enriched and improved. The question, how people find assignment between meanings and referents, still remains open. There is plenty of computational models based on statistical co-occurrence of meaning-reference pairs. Unfortunately, this mapping strategy shows poor performance in the environment with higher number of objects or noise. Therefore we propose a more robust noise-resistant algorithm. We tested the performance of this novel algorithm both with simulated and physical iCub robot. We developed a testing scenario consisting of objects with varying visual properties presented to the robot accompanied by utterances describing the given object. Our results suggest that the proposed mapping procedure is robust, resistant against noise and shows better performance than one-step mapping for all levels of noise in the linguistic input as well as slower performance degradation with increasing noise. Furthermore, it increases the clustering accuracy of both modalities.

Keywords: cross-situational learning, symbol grounding, cognitive modeling, iCub robot, language acquisition

1 Introduction

The essential (and still not fully answered) question in language acquisition is how percepts are anchored in some arbitrary symbols. In other words, how words (symbols) get their meanings. This is a so called symbol grounding problem [17]. For many years, there has been a joint attempt of cognitive modeling, neuroscience, psychology and machine learning to understand how human solve this 'problem' [5]. The ability to learn language through perception and especially through visual grounding is not only important for understanding human cognition but is also applicable in many areas such as verbal control of interactive robots [24], automatic sports commentators [12], car navigation systems, for visually impaired, situated speech understanding in computer games [16], automated generation of weather forecasts [15], tutoring children foreign language [19], etc.

Despite the extensive research in the area of language acquisition, the question how the word-to-meaning mapping is learned remains open. Using the unsupervised approach, Li et al. [21] designed the DevLex model, consisting of two self-organising networks that are bidirectionally connected. Gliozzi et al. [14] proposed an alternative with a multimodal representation layer: their unsupervised feature-based model was used to account for early category formation in young infants. This approach postulates the unsupervised role of linguistic labels that can affect categorisation during the acquisition process, which has also been supported by Taniguchi et al. [42]. Vavrečka and Farkaš [46] recently introduced a multimodal architecture for grounding of spatial words using a biologically inspired approach (separate "what" and "where" visual subsystems) in which the visual scenes (two objects in 2D space in a spatial relation) are associated with their linguistic descriptions, hence leading to integration of modalities.

On the other hand, there is still not available fully unsupervised architecture, which would be able to deal with language grounding [41], particularly language grounding in a case where sentences have variable structure and when there is more than one object in a scene. Current state-of-the-art on variable length sentences is very restricted and deals only with the static scenes [25]. Most of the recent models based on deep networks are oriented towards the application in the image-to-text [18] or video-to-text [8] mapping and does not take into account the psychological aspects of language acquisition (e.g. mutual exclusivity). Moreover, these systems are trained in supervised manner without advantage of transfer learning.

The difficulty of the task was described in a well-known experiment done by Quine [30] who imagined the anthropologist meeting a native who pointed at the scene and said "*gavagai*". When the anthropologist is stimulated in a situation by seeing a rabbit, he will suppose that the word represents running rabbit in front of him, even though it could mean as well "*ground*", "*sun*", "*hello*", or whatever else. This problem is related to language relativity, as there are several objects and their features that are described by words [29]. A simplified version of this problem consists of a simple visual scene and separate words that are grounded based on statistical co-occurrence (cross-situational learning).

From a neuroscientific point of view, symbol grounding can be viewed as a process of finding mappings between primary unimodal visual and language brain areas. Where exactly the integration is performed is still the subject to research and existing literature provides only incomplete accounts of the cortical location of this convergence. For example the study of [3] provides evidence for the involvement of the left basal posterior temporal lobe (BA37) in the integration of language and visual information. Another studies (eg. Spitsyna et al. [37]) propose that access to verbal meaning depends on both anterior and posterior heteromodal cortical systems within the temporal lobe. The grounding of actions and motoric primitives is associated with the activity in dorsal stream and premotor cortex [6].

How the language could be developed in an unsupervised manner is also the important task in developmental robotics as the most of the language acquisition in human is fully unsupervised. One of the main long-term objectives of many teams worldwide is building the conversational robots, which will be able to participate in cooperative tasks mediated by a natural language. It has been shown how robots can learn new symbols using already grounded ones and their combination [4] and how to transfer knowledge between agents [47]. Cangelosi [4] has presented their research on language emergence and grounding in sensorimotor agents and robots. This model was further extended by Tikhanoff [44], who did iCub simulation experiments and focused on integration of speech and action. Grounding of higher order concepts in action was also explored by Stramandinoli at al. [39], who made use of recurrent neural networks. Sugita and Tani [40] in their paper describe the experiment dealing with semantic compositionality – the capability of a robot to use the compositional structure to generalize novel word combinations. The current state-of-the-art on grounding variable length sentences is very restricted and deals only with static scenes [32,25].

In this paper, we present our research in the area of language acquisition using a real-world robotic scenario. We implement an hierarchical cognitive architecture for language acquisition that includes both visual and language processing. Particularly, we chose to extend current models of cross-situational learning by allowing vision-to-language mapping in the case of non-equal number of classes and by taking into account situation-time dynamics.

This is accomplished by replacing one-shot mapping with sequential mapping and adding inhibitory mechanisms to the connections. The best mapped classes are gradually eliminated and the clusterization is adaptively changed. We see our work as an extension of the McMurray model [25] and we compare it with other single step mapping models. The mapping strategy presented in this article, was shown to be very robust as it can not only find the mapping under circumstances of very noisy real-world input, but also increase the clustering accuracy of both modalities. Recently, we have tested the proposed algorithm also on the task of clustering body parts from simultaneous tactile and linguistic input [38]. In that case, sequential mapping shown slower degradation with increasing noise level in linguistic input and outperformed one-step mapping for all data set sizes and all levels of noise.

The rest of the paper is structured as follows: In Section 2 we compare different mapping algorithms used in cross-situational learning. Particularly, in subsection 2.2 we provide a mathematical formulation of the newly proposed sequential mapping algorithm and in Section 2.3 we describe the whole cognitive architecture which incorporates unimodal processing of vision and language and finding their association through mapping algorithm. Performance of the proposed method on data from iCub humanoid robot and from iCub simulator is evaluated in Section 3. Finally, results are discussed in Section 4 with an outlook for a future work.

2 Materials and Methods

In this section, we will first present one-step and newly proposed sequential mapping algorithms (Sections 2.1 and 2.2). Afterwards, we described in a detail the whole cognitive architecture used to process data from individual modalities (vision and language). Finally, we describe iCub robotic platform and iCub simulator in Section 2.4 and provide a description of evaluation in Section 2.6.

2.1 One-step mapping in cross-situational learning

In most of the cross-situational learning models, word-to-referent mapping is found by directly using frequencies of referent and meaning co-occurrences, that is, the ones with the highest co-occurrence are mapped together [35,34,49]. These models suppose availability of the ideal associative learner who can keep a track and store all co-occurences in all trials, internally memorizing and representing the word–object co-occurrence matrix of input. This allows the learner to subsequently choose the most strongly associated referent [50,51]. These models do not see the mapping as dynamic competition but operate only with the static state. Even though some of them are using likelihoods of different words and referents to perform Bayesian inference [13,49], they do not take into account how the similarity of two word forms can affect learning although it has been shown that it affects learning in children [31]. Another shortcoming of these strategies is that they don't address how these similarities will affect learning in a dynamic competition.

The simplest one-step word-to-referent learning algorithm simply accumulates word-referent pairs co-occurences. This can be viewed as Hebbian learning: the connection between a word and an object is strengthened if the pair co-occurs in a trial. To extend this basic edea we can enable also forgetting by introducing a parameter η, which can capture the memory decay. This so called dumb associative-learning model (DAM) was implemented by Yu in [51]. Supposing that at each trial t we observe an object o_t^n and hear a corresponding word w_t^n (N_t possible associations), we can describe the update of the strength of the association between word model $L(i)$ and object model $K(j)$ as follows:

$$A(i,j) = \sum_{t=1}^{T} \eta(t) \sum_{n=1}^{N_t} \delta(w_t^n, i)\delta(o_t^n, j), \tag{1}$$

where T is the number of trials, δ is the Kronecker delta function (1 when both arguments are equal and 0 otherwise), w_t^n and o_t^n indicate the nth word–object association that the model attends to and attempts to learn in the trial t and $\eta(t)$ is the parameter controlling the gain of the strength of association. This parameter η can capture different cognitive mechanisms such as memory decay [51].

Now let's assume that the word $w(i)$ is modeled by the model L_i in the language domain and object (referent) $o(j)$ is modeled by the model $K_{m(i)}$ in the visual domain. Our goal is to find the corresponding model $K_{m(i)}$ from visual subdomain for each model L_i from language domain to assign them together. Indices $m(i)$ are found as follows:

$$\forall i : m(i) = \underset{i}{\operatorname{argmax}}\, A(i,j), \tag{2}$$

where A is the co-occurence matrix computed in the Eq. 1 (element $A(i,j)$ captures co-occurence between the word $w(i)$ and object $o(j)$).

In Fig. 1, one-step mapping is visualized as it was implemented in this paper.

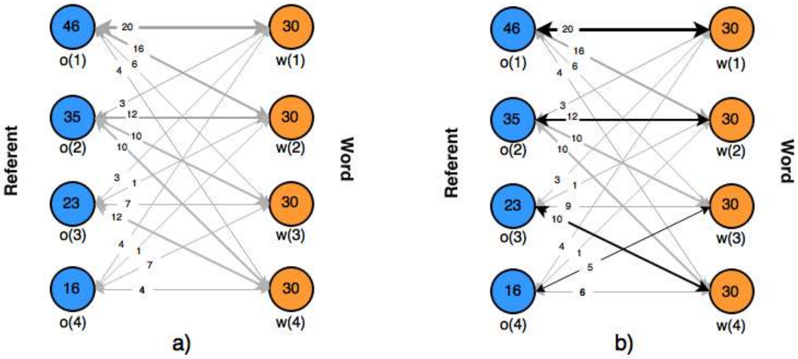


Figure 1: One-step mapping: (a) In the first stage, weights between objects (referents) and words are changed using the Hebbian learning: the connection between a word and an object is increased if the pair co-occurs in a trial [51]. (b) Afterwards, word-to-referent mapping is found in one-step - objects and words with the highest co-occurence are mapped together. The number on each connection between word $w(i)$ and object $o(j)$ refers to the number of co-occurences of the $o(j)$ and $w(i)$. In this example we suppose that there are 30 occurences of each word.

Modifications of the basic above-mentioned model includes e.g. work of Regier [31]. He proposes the model of mapping, which stems from competition models and incorporates two-way associations between words and referents. This enables model to capture both selective attention to individual words and referents as well as provide probability distribution over associated referents/words. Regier also shows in the article that for his model, learning of a novel word is most effective when memory interference is minimized [31].

2.2 Proposed sequential mapping

Since we know that learning is not static but it is rather a dynamic process, it seems reasonable to extend the basic idea of one-step cross-situational learning by incorporating dynamic competition mechanisms between words and referents to the model. This should be achieved by additional inhibitory connections. In this case, the process of finding word-referent associations resembles a Hebbian learning with inhibitory connections - once the word is associated to corresponding object (referent), links from this referent to other words are inhibited. This idea also corresponds to the fact that children prefer mapping where object has only one label to multiple labels - so called mutual exclusivity bias [22]. The inhibitory mechanisms and situation-time dynamics were already to some extend included into the model of cross-situational learning proposed by McMurray [25].

Even though our model shares some similarities with the model proposed by McMurray, it stems from different computational mechanisms. The proposed sequential mapping is able to capture both non-discrete assignment to individual clusters as well as dynamic competition mechanisms. The first mechanism is incorporated into the model by considering likelihoods that the observed data were generated by a given model instead of 1/0 assignment to models. In this way, similarities of individual meanings and referents as well as the likelihood of their recognition in each trial is taken into account. The second mechanism (dynamic competition) facilitates the sequential mapping as the best mapped classes are gradually justified with inhibitory connections to other classes (i.e. after a reliable assignment between a language and tactile model is found, inhibitory connections among this tactile model and all other language models are added). Thanks to this mechanism, mutual exclusivity principle (the fact that children prefer mapping where object has only one label to multiple labels [22]) is guaranteed.

The assignment between visual models K_j and language models L_i is found using a following iterative procedure:

1. Visual and language data are clustered separately and the corresponding posterior probabilities are found:

$$p(L_i|\boldsymbol{w}_t^n) = \frac{p(\boldsymbol{w}_t^n|L_i) * p(L_i)}{\sum_{i'} p(\boldsymbol{w}_t^n|L_{i'}) * p(L_{i'})}, \tag{3}$$
$$\forall i \in \{1, 2, ..., I\}, \forall t \in \{1, ..., T\}, \forall n \in \{1, 2, ..., N_t\}.$$
$$p(K_j|\boldsymbol{o}_t^n) = \frac{p(\boldsymbol{o}_t^n|K_j) * p(K_j) * \boldsymbol{k}(j)}{\sum_{j'} p(\boldsymbol{w}_t^n|K_{j'}) * p(K_{j'})}, \tag{4}$$
$$\forall j \in \{1, 2, ..., J\}, \forall t \in \{1, ..., T\}, \forall n \in \{1, 2, ..., N_t\},$$

   where I is the number of language models, J is the number of visual models, T is the number of trials and N_t is the number of possible object-word associations in the trial t.
2. For each datapoint the most probable visual and language clusters are selected and the datapoint is assigned to these clusters:

$$a_t^n = \underset{i}{\operatorname{argmax}}\, p(L_i|\boldsymbol{w}_t^n), \tag{5}$$
$$b_t^n = \underset{j}{\operatorname{argmax}}\, p(K_j|\boldsymbol{o}_t^n), \tag{6}$$
$$\forall t \in \{1, ..., T\}, \forall n \in \{1, 2, ..., N_t\}.$$

3. Co-occurence matrix $A(i,j)$ is computed:

$$A(i,j) = \zeta(i,j) * \sum_{t=1}^{K} \eta(t) \sum_{n=1}^{N_t} \delta(a_t^n, i)\delta(b_t^n, j), \tag{7}$$

where $\zeta(i,j)$ is the matrix storing the strength of the connections between visual model K_j and language model L_i, $\eta(t)$ is the parameter controling the gain of the strength of association.

4. The best assignment is selected:

$$[im, jm] = \underset{i}{\operatorname{argmax}}\, \underset{j}{\operatorname{argmax}}\, A(i,j). \tag{8}$$

5. Inhibition connections are added between the assigned visual model K_{jm} and all language models L_i, where $i \neq im$ (mutual exclusivity):

$$\zeta(i, jm) = \zeta(i, jm) * (1 - z_1), \forall i \neq im, \tag{9}$$

where z_i is the parameter capturing the strength of the inhibition (in our experiment set to 1, which corresponds to total inhibition of the given connection).

6. Inhibition is added to the assigned visual model K_{jm} (a prior probability of the model is changed):

$$k(jm) = k(jm) * (1 - z_2), \tag{10}$$

where z_2 is the parameter capturing the inhibition of the assigned visual model (in our experiment is this parameter set to 1, which corresponds to total inhibition of the given model). It is possible to

7. The assigned points (datapoints which belong to both K_{jm} and L_{im}) are deleted from the dataset:

$$X = X \setminus \left\{ (\boldsymbol{o}_t^n, \boldsymbol{w}_t^n) \mid \underset{j}{\operatorname{argmax}}\, p(K_j|\boldsymbol{o}_t^n) == jm \wedge \underset{i}{\operatorname{argmax}}\, p(L_i|\boldsymbol{w}_j^n) == im \right\} \tag{11}$$

8. Repeat (1)-(7) until $X \in \emptyset$ (dataset is empty) or $\|\boldsymbol{k}\| > 0$ (some of the visual models are not totally inhibited)

The proposed algorithm where words are assigned to corresponding referents in a sequential manner is visualized in the Fig. 2.

In the ideal case, the unambiguous mapping between the two clusterizations will be found. In the real case (where clusterizations in visual and language layer are not optimal), none or more than one model from visual layer will be assigned to one cluster L_i in language layer or vice versa.

2.3 Specific architecture

Our multimodal hierarchical architecture consists of multimodal and unimodal part. The unimodal part has two layers performing separate processing of localist inputs - visual objects and auditory word-forms. Both unimodal layers are subsequently mapped one to each other in the upper multimodal layer (see Fig. 3).

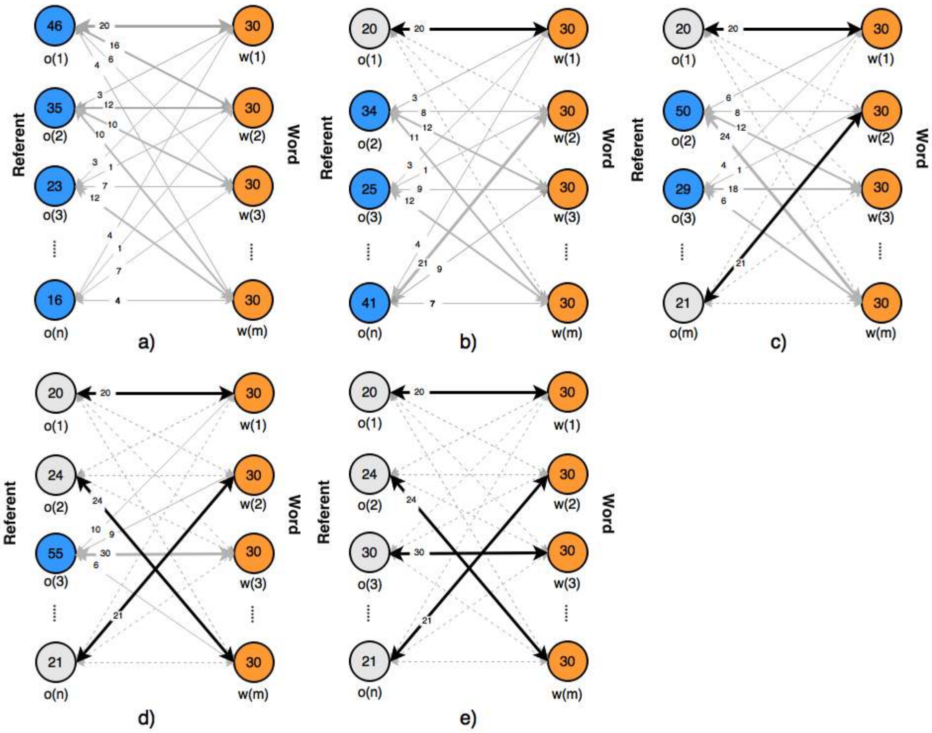


Figure 2: Sequential mapping: The toy example of sequential mapping is shown to clarify the mechanism of finding object-word assignment. In this example we suppose that there are 30 occurences of each word. Dotted line marks the inhibitory connection between the object $o(j)$ and word $w(i)$, black line corresponds to the already found mapping. The number on each connection between word $w(j)$ and object $o(i)$ refers to the number of co-occurences of the $o(j)$ and $w(i)$. Objects $o(j)$ and words $w(i)$ are assigned to corresponding models based on the given clustering mechanism.

Visual layer Each datapoint (object $\boldsymbol{o_t^n}$) can be considered as a triplet of continuous-valued vectors for each visual feature: $\boldsymbol{o_t^n} = (\boldsymbol{x}_{t,n}^{size}, \boldsymbol{x}_{t,n}^{colour}, \boldsymbol{x}_{t,n}^{shape})$. This enables us to write the visual dataset as: $X^{vis} = [X^{size} X^{colour} X^{shape}]$ and process data for each visual feature separately. For processing visual data was used Gaussian mixture model, which is a convex mixture of d-dimensional Gaussian densities $l(\boldsymbol{x}^k|\boldsymbol{\theta}_j^k)$, where $k \in \{\text{size, colour, shape}\}$. In this case, each visual model K_j^k is described by a set of parameters $\boldsymbol{\theta}_j^k$. The posterior probabilites $f(\boldsymbol{\theta}_j^k|\boldsymbol{x}^k)$ are computed as following:

$$f(\boldsymbol{\theta}_j^k|\boldsymbol{x}^k) = \sum_{j=1}^{J_k} r_j^k l(\boldsymbol{x}^k|\boldsymbol{\theta}_j^k), \tag{12}$$

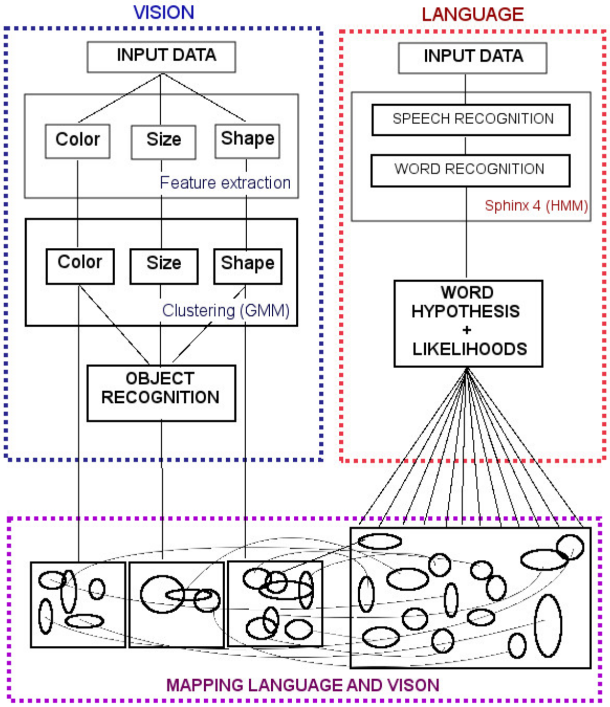


Figure 3: Proposed multimodal architecture

$$l(\boldsymbol{x}^k|\boldsymbol{\theta}_j^k) = \frac{1}{\sqrt{(2\pi)^d}\sqrt{|\boldsymbol{S}_j^k|}} \exp[-\frac{1}{2}(\boldsymbol{x}^k - \boldsymbol{m}_j^k)^T(\boldsymbol{S}_j^k)^{-1}(\boldsymbol{x}^k - \boldsymbol{m}_j^k)], \tag{13}$$

where $k \in \{\text{size, color, shape}\}$, $\boldsymbol{x}^k$ is a set of d-dimensional continuous-valued data vectors, r_j^k are the mixture weights, J_k is the number of visual models for each visual feature k, parameters $\boldsymbol{\theta}_j^k$ are cluster centers $\boldsymbol{m}_j^k$ and covariance matrices $\boldsymbol{S}_j^k$.

Mixture of Gaussians is trained by the EM algorithm [7]. An output of this layer for each data point $\boldsymbol{x}_i^k$ is the vector $\boldsymbol{y}_i^k$ of J_k output parameters describing the data point (the likelihood that the data point belongs to each individual cluster in a mixture). This corresponds to the fuzzy memberships (distributed representation).

For simpler evaluation we made use of localist representation (Winner-takes-all), where only the cluster with the highest cluster membership probability is considered for further processing (see Eq. (5)-(6)):

$$M(K_j^k|O) = \begin{cases} 1 & \text{if } j = \text{argmax}_{j'}\, f(K_{j'}^k|O) \\ 0 & \text{if } j \neq \text{argmax}_{j'}\, f(K_{j'}^k|O) \end{cases} \tag{14}$$

where $k \in \{$ size, color, shape$\}$, $j \in \{1, 2, ..., J_k\}$.

Language layer The linguistic input are spoken sentences describing the image in the format: <size> <color> <shape> (eg. "Small red triangle"). Afterwards, individual word-forms are extracted from the audio input and compared to pre-learned language models - the log-scale scores $p(\boldsymbol{w}_t^n|L_i)$ of the audio matching the model is computed. Based on these data, posterior probability can be computed:

$$p(L_i|\boldsymbol{w}_t^n) = \frac{p(\boldsymbol{w}_t^n|L_i) * p(L_i)}{\sum_{i'} p(\boldsymbol{w}_t^n|L_{i'}) * p(L_{i'})}, \quad (15)$$
$$\forall i \in \{1, 2, ..., I\}, \forall t \in \{1, ..., T\}, \forall n \in \{1, 2, ..., N_t\},$$

where I is the number of language models, T is the number of trials (sentences) and N_t is the number of word-forms in the trial (sentence) t.

An output of this layer for each data point $\boldsymbol{w}_t^n$ is the vector $\boldsymbol{y}_i$ of I output parameters describing the data point (the likelihood that the data point belongs to each individual language model). This corresponds to the fuzzy memberships (distributed representation). Linguistic and visual inputs are processed simultaneously.

Mapping - Model 1 and Model 2 After both visual and language data are clustered, the mapping between the two layers must be found. For each cluster L_i in the language layer a corresponding cluster K_j^k in visual layer (for each feature $k \in \{\text{size, colour, shape}\}$) is found. The mapping is found as following: for each j and k we find cluster $L_{kmax_{jk}}$ from language layer which will be assigned to the cluster K_j^k from the visual layer. In the paper, we compare two different models how to find indices $kmax_{jk}$. We compared one-step mapping (see Section 2.1) and newly proposed sequential mapping (see Section 2.2).

The exact algorithm used to find mapping between visual and language models in a sequential manner is described in the detail in the Algorithm 1. Indices $m(i)$ are found sequentially. In each step, the best mapped data are excluded and the rest of data is reclustered using Gaussian mixture models. Afterwards, one-step mapping is performed (see Alg. 1). An extension of the algorithm for a variable length sentence is described in the Appendix.

2.4 iCub robotic platform and iCub simulator

For the experiment we used a simulated [43] and a physical [26] iCub robot. The iCub (Fig. 1 (c)) is an open-source humanoid robot with the size of a three and a half year-old child, fully articulated hands as well as a head-and-eye system which makes him ideal for cognitive experiments. The iCub simulator has been designed to reproduce, as accurately as possible, the physics and the dynamics of the robot and its environment [43]. The simulator and the actual robot have the same interface supporting YARP [27] which is a robot platform for interprocess communication and control of the physical and simulated robot in a real-time.

Algorithm 1 Sequential mapping: fixed grammar

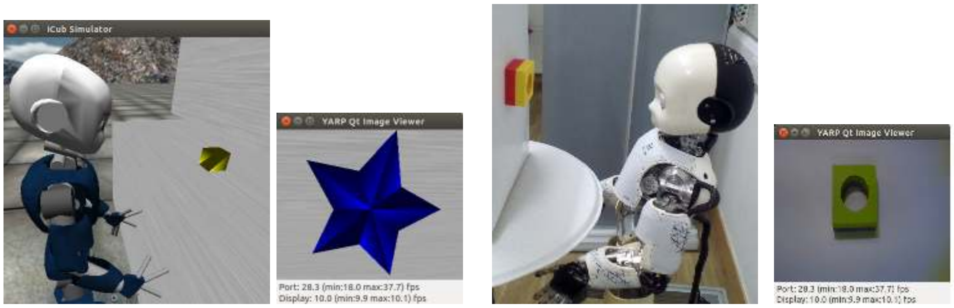
```
Inputs:
    language clusters $L_i$ $(i \in 1:I)$, visual clusters $K_j^k \sim N(\boldsymbol{m}_j^k, \boldsymbol{S}_j^k)$,
    $j \in 1 : J_k$, input data $\boldsymbol{x}^k$ for each feature $k \in \{$size, colour, shape$\}$,
    number of clusters $J^k$ for each feature $k$
Output:
    mapping between all visual classes $K_j^k$ and language classes $L_i$

for $k \in \{$size, colour, shape$\}$ do
    $NCl \leftarrow J^k$
    while $NCl > 0$ and $\boldsymbol{x}^k$ is not empty do
        assign each data point from $\boldsymbol{x}^k$ to visual and language cluster (Winner-takes all, see Eq. (14))
        for $j = 1 : NCl$ do
            for $i = 1 : I$ do
                $A_{ij} \leftarrow$ how many times was class $i$ actually classified as $j$
            end for
        end for
        $[im, jm] \leftarrow \text{argmax}_i \, \text{argmax}_j \, A_{ij}$
        $\boldsymbol{x}_{del}^k \leftarrow$ data points assigned to both $K_{jm}^k$ and $L_{im}$
        $\boldsymbol{\Theta}_{new_{NCl}}^k \leftarrow N(\boldsymbol{x}_{del}^k)$ learn Gaussian on the to be deleted data
        $\boldsymbol{X}^k \leftarrow \boldsymbol{X}^k \setminus \boldsymbol{X}_{del}^k$ delete all data points assigned to both $K_{jm}^k$ and $L_{im}$
        $NCl \leftarrow NCl - 1$
        relearn $K^k \sim N(\boldsymbol{m}^k, \boldsymbol{S}^k)$ on new data $\boldsymbol{x}^k$ with $NCl$ number of clusters
    end while
end for
cluster visual data using new $\boldsymbol{\Theta}_{new}^k$ parameters (cluster centres $\boldsymbol{m}^k$ and covariance matrices $\boldsymbol{S}^k$ and perform One-step mapping (Model 1)
```

2.5 Input data description and preprocessing

The input to our model consisted of visual and language data. The visual scene was composed of an object in a center of the scene with a variable position. Visual features (size, shape and color) of an object also varied. We developed two separate datasets for training and testing purpose. Real-world dataset has visual sensory data acquired from the cameras of the physical iCub robot who observed simple objects placed on the white board in front of his eyes (see Fig. 4, (c) and (d)) (204 instances, 3 sizes, 5 colors and 7 shapes). Simulated dataset is made in iCub simulator (see Fig. 4, (a) and (b)) as a Blender generated virtual objects (432 instances, 3 sizes, 6 colors and 6 shapes).

The spoken language input were sentences pronounced by a non-native English speaker describing the image in the format: <size> <color> <shape> (eg. "Small red triangle") and were processed simultaneously with the visual input.

Speech recognition CMU Sphinx (an open-source flexible Markov model-based speech recognizer system) was used for speech recognition [20]. Sphinx


iCub Simulator
YARP Qt Image Viewer
Port: 28.3 (min:18.0 max:37.7) fps
Display: 10.0 (min:9.9 max:10.1) fps
YARP Qt Image Viewer
Port: 28.3 (min:18.0 max:37.7) fps
Display: 10.0 (min:9.9 max:10.1) fps

(a) iCub simulator (b) Blender object (c) physical iCub (d) Real object

Figure 4: Experiment design and corresponding input data

itself offers large vocabulary, but we created our own task-specific smaller vocabulary using online IMtool that produces a dictionary based on a CMU dictionary and matches its language model.

There is a probabilistic output from the CMU Sphinx. The 10 best hypothesis for a matching model with corresponding scores were saved for each utterance (those are log-scale scores of the audio matching the model). Because the scores for hypothesis of each word in the sentence were needed for further evaluation, the words were pronounced with the large pauses and the end of the sentence was marked by the word "STOP". An output of the language layer is a I-dimensional continuous valued vector, where I is a number of language clusters (corresponding to the number of possible utterances). This vector contains 10 non-zero values and the rest is zero.

Image processing The image inputs are processed using standard MATLAB functions. First, the image is morphologically opened with a disk-shaped structuring element (*imopen*) to remove the noisy background of an image, then all grayish pixels are removed and the image is converted from the true color RGB to the grayscale intensity image by eliminating the hue and saturation information while retaining the luminance (*rgb2gray*). Finaly, the intensity image is converted to a binary image using the threshold computed by Otsu's method (*threshold*). Example of the preprocessed image is shown in the Fig. 5.

Afterwards, the properties of image regions are measured using the function *regionprops*. Individual visual features (shape, color, size) are subsequently processed separately. Following features have been used: Color (3 features: Average RGB of the selected region), Size (6 features: Parimeter of an object, distance from the centroid to the left corner of the bounding box, width and length of the bounding box), Shape (13 features: Area, centroid, major axis length, eccentricity, orientation, convexArea, FilledArea, EulerNumber, EquivDiameter, Solidity, Extent, Perimeter). To obtain shape features we automatically cropped and resized the image to equalize the size of objects.

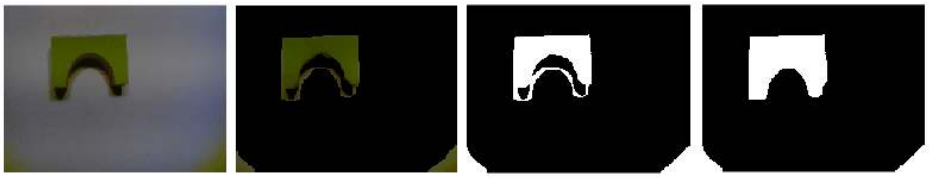


Figure 5: Image processing - original image, removal of the background, converting to BW image and filling the holes

In spite of the fact that our visual model is mainly mathematical and implemented in a very 'machine vision' sort of way, the bases of its processing follow biological correlates of mammal vision. More specifically, from the neuroanatomical point of view, this corresponds to the processing of the visual input in the separate higher visual centra in the brain, specifically to the independent processing of the information about position and indentification of an object in the ventral (”what”) and dorsal (”where”) neural pathways respectively [28]. Individual object properties are identified in the separate visual centra of the occipital lobe.

2.6 Evaluation

In order to evaluate performance of clusterisation of visual data achieved by unsupervised GMM, we compare our results to supervised version of the GMM algorithm. Furthermore, we provide comparison with SOM algorithm and GWR algorithms as state-of-the-art alternatives to our approach.

In a case of supervised GMM algorithm, GWR and SOM, data were divided to training and validation dataset in the ratio 70:30 For unsupervised GMM, HMM and k-means algorithms, we computed the accuracy in a different manner. After performing the clustering of the data, each cluster is assigned to the class that appears most frequently in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned data points (compared to manual true labels) and dividing this by the total number of data points.

3 Experimental results

The first part of the results is dedicated to the performance of our model in the real-world scenario. The robot interacts with a human in noisy condition that distorts speech input.

3.1 Vision

At the first stage we evaluated the Vision subpart of our model. There were several algorithm compared, namely the GMM algorithm, supervised GMM algorithm, *k*-means, SOM and GWR (growing when required neural gas) algorithm [1]) [23]. Both SOM and GWR had 100 nodes. The results for real-world dataset and simulated dataset with Blender objects can be seen in the Table 1. Even though SOM and GWR are considered to be unsupervised algorithms, we adopted technique for labeling inputs, so they should be compared with supervised algorithms. There is also overestimated number of clusters (number of nodes corresponds to the number of clusters). It indicates that these algorithms are partly overfitting the data so we divided the set to testing and validation data.

	Real-data			Blender		
Accuracy [%]	Size	Colour	Shape	Size	Colour	Shape
GMM sup.	83.3 ± 0.0	99.0 ± 0.0	81.4 ± 0.0	98.6 ± 0.0	97.9 ± 0.0	93.1 ± 0.0
GMM unsup.	76.2 ± 6.8	76.1 ± 9.1	56.1 ± 6.2	74.2 ± 10.1	60.9 ± 9.0	64.3 ± 7.2
K-means	67.8 ± 6.2	81.2 ± 1.1	53.1 ± 4.2	66.3 ± 0.2	77.1 ± 10.7	72.8 ± 6.9
SOM	69.6 ± 5.6	78.9 ± 6.8	54.2 ± 4.1	66.1 ± 4.2	81.7 ± 7.6	59.3 ± 6.2
GWR	89.9 ± 2.1	99.5 ± 0.4	76.6 ± 1.4	88.9 ± 0.7	98.1 ± 0.9	94.2 ± 0.6

Table 1: Comparison of clusterization and classification accuracy of visual data. The mean and standard deviation from 100 repetitions is visualised.

3.2 Mapping

The performance of one-step mapping (vision and language are mapped in one step based on the frequency of co-occurrence) and sequential mapping (see Alg. 1) is shown in Table 2. We calculated accuracy both for real-world data from physical iCub and for Blender objects placed in the iCub simulator. Language accuracy for Blender dataset is much higher compared to the real-world data as tutor was speaking directly to the microphone.

	Real-data			Blender		
Accuracy [%]	Size	Colour	Shape	Size	Colour	Shape
Vision	76.2 ± 6.8	76.1 ± 9.1	56.1 ± 6.2	74.2 ± 10.1	60.9 ± 9.0	64.3 ± 7.6
Language	70.6 ± 0.0	82.4 ± 0.0	77.5 ± 0.0	98.1 ± 0.0	96.5 ± 0.0	98.1 ± 0.0
One-step mapping	54.1 ± 4.1	58.2 ± 10.3	52.2 ± 4.9	67.3 ± 8.2	56.2 ± 6.1	61.9 ± 3.2
Sequential mapping	74.2 ± 15.1	87.1 ± 10.2	72.9 ± 5.1	96.1 ± 31.2	95.2 ± 1.2	92.1 ± 0.9

Table 2: Comparison of One-step mapping and Sequential mapping for data from iCub simulator (Blender) and physical iCub (real-data). The mean and standard deviation from 100 repetitions is visualised.

Tolerance of the sequential mapping to noise in the language data is visualized in the Fig. 6 for visual data from iCub simulator in combination with language data processed by Sphinx 4. The noise to the language data is added subsequently and evenly to all classes (given proportion of language inputs was randomly changed to the random word). The noise was added artificially, but can be interpreted either as a noise in the data or mistakes in labeling perceived objects. We grouped them together into a misclassification variable. The visual data are let intact so the only cause of the observed variations in the accuracy is initialization. As can be seen, the accuracy of sequential mapping remains very stable even though the accuracy of language decreases and outperforms both language and vision for almost all values of the misclassification.

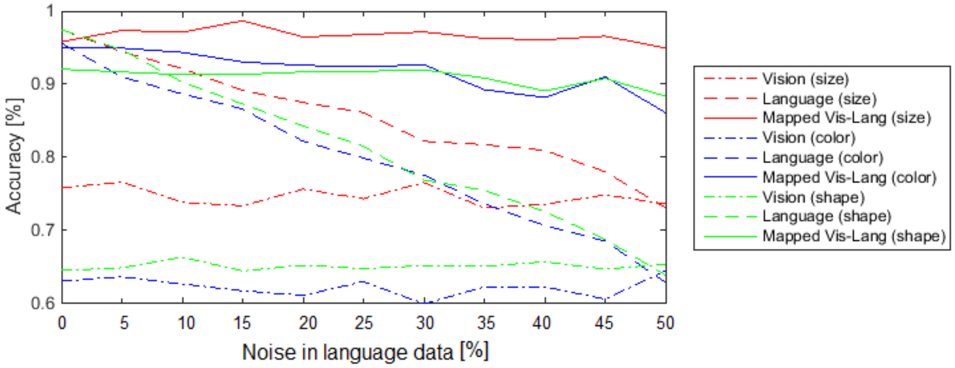


Figure 6: Dependence of mapping accuracy on the misclassification in the language data for fixed length sentence (mean values over 50 repetitions are visualized). Different colours correspond to different visual features (red – size, blue – colour, green – shape). Visual data are generated in Blender and acquired through iCub simulator, language data are processed using Sphinx 4.

4 Discussion and Conclusion

Current models of vision-to-language mapping often make use of cross-situational learning while relying directly on statistical co-occurrence of meaning-referent pairs - the ones with the highest co-occurrence are mapped together (e.g. [35]). This approach show poor performance in cases of higher number of objects or noise. Therefore we extended this basic model and introduced a new more robust and noise-resistant mapping procedure. Our approach incorporates situation-time dynamics, mutual exclusivity and is able to deal with non-equal number of classes in individual subdomains.

Mathematical formulation of the newly proposed mapping is provided (see Section 2.2) and results on both simulated and real-world data from iCub robot are compared to one-step mapping (see Table 2). It was shown, that the method

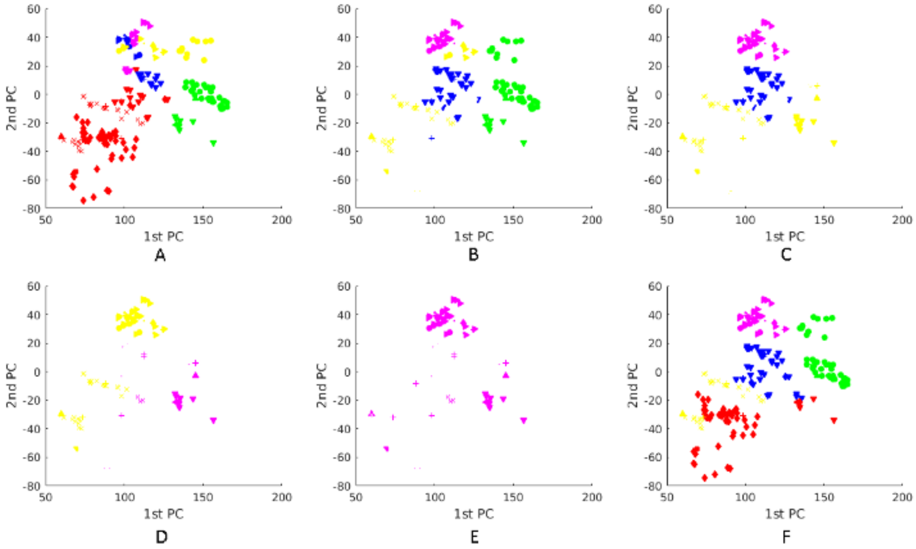


Figure 7: Sequential mapping procedure on real-world visual data from physical iCub robot. Data are plotted in the space of first two principal components. Individual colours distinguish separate clusters found at each iteration. Subfigures (A)-(E) show sequential elimination of data points. Initial clusterisation of visual data and clusterisation after sequential mapping can be seen in subfigure (A) and (F) respectively. For visualisation purposes, only 100 datapoints were plotted.

is able to find mapping between language and vision, it improves the accuracy of both individual subdomains and shows very good resistance to noise or misclassification in language (see Fig. 6). How to map in an unsupervised manner several clusterings (e.g. for vision, action, language) is not only important question in cognitive modeling, but also in general machine learning, where data acquired from different sensors or in different situations can be independently clustered and mapped one to each other. More detailed discussion of the results follows.

The trivial one step mapping can be imagined as a basic Hebbian learning. Our extension, can be liken to the Hebbian learning with inhibitory connections. In recent years, few approaches to find an alternative way to the basic approach appeared [52,25]. Our model partially stems from McMurray approach, who showed that associative learning can be sufficient for language acquisition and that the main components of this type of learning are online competition of models and pruning incorrect associations which makes it possible to gradually improve associations between models.

The mutual exclusivity principle is guaranteed in our method thanks to the inhibitory connections which are gradually created among models. Once the mapping between referent and meaning is found, the connection from a given meaning to other referents is inhibited. Dynamic competition is addressed in the following way: when any association of meaning and referent is found, other models compete again for resources. Firstly, well mapped data are deleted, and afterwards, the resting data are reclustered. Furthermore the likelihoods can associate each data point to many separate models instead of binary membership. The similarity of two word forms can affect learning similar to children development [31]. The algorithm also enables mapping together data in a case where we have uneven number of clusters in both subdomains.

On the other hand, the principle of mutual exclusivity is not suitable for further stages of language acquisition, namely learning of polysemic words. A polyseme is a word or a phrase with different but related senses (e.g. wood as a piece of tree or the area with trees). The homonyms are subset of the polysemes, but the difference between homonyms and polysemes is subtle and fuzzy. Homonyms represent a group of words sharing similar spelling (homographs) and the same sound (homophones) but have different and unrelated meanings, e.g. homograph bank standing for embankment or place where money is kept. The learning of polysemic words violates the principle of mutual exclusivity as the dynamic competition does not allow to map two words to the same visual model. We are aware of this problem and we would like to extend it in the future iteration of the model. Similar to humans that has to learn polysemic words as an exceptions we will incorporate this principle to the next version of our architecture.

In the next part we analyze the ability of our architecture to deal with ambiguous inputs. The mapping will find a reliable labeling for the visual input data (more generally for data from any other modality) with a possibility to incorporate fuzziness of this mapping. For some concepts finding an unambiguous mapping is very easy, for others it is much more difficult or impossible (such as abstract words , e.g. the love has no dominating colour, but sky is usually blue). Since the mapping is established only among the clusters where it makes sense, dealing with a lot of redundant information is avoided. Similar idea is used in classification algorithms which use sparse matrices (e.g. [9,10]). We also analyzed strong and weak points of the algorithms adopted in our architecture. First, the ability of different algorithms to classify unimodal visual data was compared. As expected, for data which are well separated and mainly spherically distributed (this is generally a case for simulated and artificially generated data), k-means algorithm outperformed GMM algorithm. On the other hand for non-spherical real data performed generally better GMM algorithm (see Table 1 for comparison of performance on simulated data placed in an iCub simulator and data from real iCub robot cameras). Since the unsupervised algorithms are highly dependent on the initialization, it can be seen, that the standard deviation of data is quite high even though 20 repetitions were averaged. We should conclude that the performance of the algorithms in our tasks reflects both their fundamental advances and limitations. We are still missing the algorithm that is able to cope with highly variable datasets in term of their statistical properties.

Afterwards, we focused on the mapping between vision and language and compared two different approaches: one-step mapping to sequential mapping which in a stepwise manner finds the best mapped clusters while constantly relearning clusterization of visual data. As can be seen on the results in Table 2, the novel sequential mapping led to an improvement of effectiveness compared to the method which maps vision to language in one step. This can be seen in more accurate mapping which leads to better estimation of the clustered data labels and consequently to the lower classification error for all of the evaluated datasets and features.

The accuracy of multimodal mapping outperforms either vision, language or both of them. This is an important finding, since the sequential mapping doesn't improve only accuracy of visual clustering, but can as well fix mistakes in the language recognition, which provides the labels. Furthermore this result suggests that we are able not only to find mapping between more clusterings, but we can also improve clusterisation accuracy by combining individual classifiers. This is not easily seen on the presented dataset as there is high recognition accuracy of Sphinx software (especially in the case of sentences recorded for the simulated dataset). Therefore we also tested whether the noise in the language data affects the correct mapping between vision and language. The result can be seen in the Fig.6. Even though the noise is added to the language data, the ability to find the mapping remains nearly intact. The mapping accuracy decreases only very slightly and remains around 90% since the accuracy of language recognition drops from original approx. 95 % to approx. 70% (depends on the specific visual feature).

We also analyzed how complexity of environment affects accuracy of our architecture. We suppose that the performance of the one-step mapping will decrease with an increasing complexity of the task (more clusters, higher overlap and higher dimensionality) as it is more difficult to find reliable clustering of the data in. This hypothesis is supported by our preliminary results on clustering body parts from simultaneous tactile and linguistic input [38] and by the results presented in this paper in the Table 2. The quality of one-step mapping correlates with quality of visual data clustering. For Blender data, the worst performance was achieved for mapping words to visual feature Shape (52 ± 5 %) and for physical iCub for feature Colour (62 ± 3 %). The feature Shape has the highest number of clusters (10) and the feature Colour has the second highest number of clusters (9) and the highest overlap of the clusters for the physical iCub. Real-world tasks are much more complex and we can expect tens of different object shapes. In that case, the performance of clustering is crucial and one-step mapping wouldn't be able to provide a reliable mapping. It can be seen from our results, that the proposed mapping which enables gradual re-estimation of models parameters and works in a dynamical fashion, achieves much higher accuracy even for the cases where the one-step mapping fails. We suppose that the mapping accuracy of the proposed method decreases slower with the decreasing accuracy of clustering of individual modalities. Unfortunately this factor was not studied in our restricted scenario but our preliminary results on mapping tactile and linguistic input [38] support this hypothesis. We plan to investigate this phenomenon more deeply in future research.

The language dataset differs considerably from the natural language. On the other hand the dataset reflects some characteristics from the findings of Werker et al. [48] as infant-directed words are usually kept short with large pauses between words. Moreover Brent and Siskind [2] showed that frequency of exposure to a word in isolation predicts better whether that word will be learned than the total frequency of exposure to that word. Also Snow in her paper [36] found out that mothers' speech to 2-years-olds is much simpler and less redundant than their speech to 10-years-old. Which indicates that young children have available a sample of speech which is simpler, more redundant, and less confusing than normal adult speech.

The proposed algorithm was tested on language to vision mapping and also on language to tactile mapping [38], and it can be easily extended to language to any other modalities mapping. The mapping between multiple modalities and words was already researched in [42]. Fazly [11] proposed a probabilistic model of cross-situational learning where he considered sentences containing both objects and their motion. Monaghan [29] studied differences between cross-situational learning of nouns and verbs on human participants as an extension of work of Tomasello[45,33]. He consider learning of verbs same difficult as learning of as nouns when presented in syntactic context. He noticed that nouns are learnt quicker but both verbs and nouns can be acquired simultaneously.

It is worth notion that the main goal of our study is to analyze mapping between modalities. Hence the processing of the individual modalities does not stem from the state of the art algorithms. We keep them deliberatively simple for better understanding of cross situational learning . We suppose that the sequence mapping should be applied to any outputs from auditory and visual subsystems.

Acknowledgment

This work was in part supported by European EU FP7 research project TRADR (Project Nr. 609763); CAPES Foundation, Ministry of Education of Brazil (scholarship BEX 1084/13-5); CNPq Brazil (scholarship 232590/2014-1); and UK EPSRC project BABEL (EP/J004561/1 and EP/J00457X/1). ADD CAK

References

1. Belmonte Klein, F.: GWR and GNG Classifier - File Exchange - MATLAB Central. http://uk.mathworks.com/matlabcentral/fileexchange/57798-gwr-and-gng-classifier (2016)
2. Brent, M.R., Siskind, J.M.: The role of exposure to isolated words in early vocabulary development. Cognition 81(2), B33–B44 (2001)
3. Büchel, C., Price, C., Friston, K.: A multimodal language region in the ventral visual pathway. Nature 394(6690), 274–277 (1998)

4. Cangelosi, A., Greco, A., Harnad, S.: From robotic toil to symbolic theft: grounding transfer from entry-level to higher-level categories. Connection Science 12.2, 143–162 (2000)
5. Coradeschi, S., Loutfi, A., Wrede, B.: A short review of symbol grounding in robotic and intelligent systems. KI - Künstliche Intelligenz 27(2), 129–136 (2013), http://dx.doi.org/10.1007/s13218-013-0247-2
6. Culham, J.C., Valyear, K.F.: Human parietal cortex in action. Current opinion in neurobiology 16(2), 205–212 (2006)
7. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. Journal of the royal statistical society. Series B (methodological) pp. 1–38 (1977)
8. Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2625–2634 (2015)
9. Elhamifar, E., Vidal, R.: Robust classification using structured sparse representation. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. pp. 1873–1879. IEEE (2011)
10. Elhamifar, E., Vidal, R.: Sparse subspace clustering: Algorithm, theory, and applications. IEEE transactions on pattern analysis and machine intelligence 35(11), 2765–2781 (2013)
11. Fazly, A., Alishahi, A., Stevenson, S.: A probabilistic computational model of cross-situational word learning. Cognitive Science 34(6), 1017–1063 (2010)
12. Fleischman, M., Roy, D.: Grounded language modeling for automatic speech recognition of sports video. In: ACL (2008)
13. Frank, M.C., Goodman, N.D., Tenenbaum, J.B.: Using speakers' referential intentions to model early cross-situational word learning. Psychological Science 20(5), 578–585 (2009)
14. Gliozzi, V., Mayor, J., Hu, J.F., Plunkett, K.: Labels as features (not names) for infant categorization: A neurocomputational approach. Cognitive Science 33(4), 709–738 (2009)
15. Goldberg, E., Driedger, N., Kittredge, R.: Using natural-language processing to produce weather forecasts. IEEE Expert 9(2), 45–53 (1994)
16. Gorniak, P., Roy, D.: Speaking with your sidekick: Understanding situated speech in computer role playing games. In: AIIDE. pp. 57–62 (2005)
17. Harnad, S.: The symbol grounding problem. Physica D: Nonlinear Phenomena 42(1-3), 335–346 (1990)
18. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3128–3137 (2015)
19. Kennedy, J., Baxter, P., Belpaeme, T.: Comparing robot embodiments in a guided discovery learning interaction with children. International Journal of Social Robotics 7(2), 293–308 (2015)
20. Lamere, P., Kwok, P., Gouvea, E., Raj, B., Singh, R., Walker, W., Warmuth, M., Wolf, P.: The cmu sphinx-4 speech recognition system. In: IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2003), Hong Kong. vol. 1, pp. 2–5. Citeseer (2003)
21. Li, P., Farkas, I., MacWhinney, B.: Early lexical development in a self-organizing neural network. Neural networks 17(8), 1345–1362 (2004)
22. Markman, E.M.: Constraints children place on word meanings. Cognitive Science 14(1), 57–77 (1990)

23. Marsland, S., Shapiro, J., Nehmzow, U.: A self-organising network that grows when required. Neural Networks 15(8), 1041–1058 (2002)
24. Mavridis, N.: A review of verbal and non-verbal human-robot interactive communication. Elsevier Journal of Robotics and Autonomous Systems 63, 22–35 (January 2015)
25. McMurray, B., Horst, J.S., Samuelson, L.K.: Word learning emerges from the interaction of online referent selection and slow associative learning. Psychological review 119(4), 831 (2012)
26. Metta, G., et al.: The icub humanoid robot: an open platform for research in embodied cognition. In: Proceedings of 8th workshop on performance metrics for intelligent systems, ACM. pp. 50–56 (2008)
27. Metta, G., Fitzpatrick, P., Natale, L.: Yarp: yet another robot platform. International Journal on Advanced Robotics Systems 3(1), 43–38 (2006)
28. Mishkin, M., Ungerleider, L.G., Macko, K.A.: Object vision and spatial vision: two cortical pathways. Trends in neurosciences 6, 414–417 (1983)
29. Monaghan, P., Mattock, K., Davies, R.A.I., Smith, A.C.: Gavagai is as gavagai does: Learning nouns and verbs from cross-situational statistics. Cognitive Science 39(5), 1099–1112 (2015), http://dx.doi.org/10.1111/cogs.12186
30. Quine, W.V.: On the reasons for indeterminacy of translation. The Journal of Philosophy 67(6), 178–183 (1970)
31. Regier, T.: The emergence of words: Attentional learning in form and meaning. Cognitive science 29(6), 819–865 (2005)
32. Roy, D.K.: Learning visually grounded words and syntax for a scene description task. Computer Speech and Language 16.3, 353–385 (2002)
33. Schwartz, R.G., Terrell, B.Y.: The role of input frequency in lexical acquisition. Journal of child language 10(01), 57–64 (1983)
34. Siskind, J.M.: A computational study of cross-situational techniques for learning word-to-meaning mappings. Cognition 61(1), 39–91 (1996)
35. Smith, K., Smith, A.D., Blythe, R.A., Vogt, P.: Cross-situational learning: a mathematical approach. Lecture Notes in Computer Science 4211, 31–44 (2006)
36. Snow, C.E.: Mothers' speech to children learning language. Child development pp. 549–565 (1972)
37. Spitsyna, G., Warren, J.E., Scott, S.K., Turkheimer, F.E., Wise, R.J.: Converging language streams in the human temporal lobe. The Journal of Neuroscience 26(28), 7328–7336 (2006)
38. Stepanova, K., Hoffmann, M., Straka, Z., Cangelosi, A., Vavrecka, M.: Where is my forearm? clustering body parts from simultaneous tactile and linguistic input. In: KUZ XVII: Cognition and artificial life (accepted) (2017)
39. Stramandinoli, F., Marocco, D., Cangelosi, A.: The grounding of higher order concepts in action and language: a cognitive robotics model. Neural Networks 32, 165–173 (2012)
40. Sugita, Y., Tani, J.: Learning semantic combinatoriality from the interaction between linguistic and behavioral processes. Adaptive Behavior 13(1), 33–52 (2005)
41. Taddeo, M., Floridi, L.: Solving the symbol grounding problem: a critical review of fifteen years of research. Journal of Experimental and Theoretical Artificial Intelligence 17.4, 419–445 (2005)
42. Taniguchi, A., Taniguchi, T., Cangelosi, A.: Multiple categorization by icub: Learning relationships between multiple modalities and words. In: IROS2016 Workshop on Machine Learning Methods for High-Level Cognitive Capabilities in Robotics (2016)

43. 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: Proceedings of 8th workshop on performance metrics for intelligent systems, ACM. pp. 57–61 (2008)
44. Tikhanoff, V., Cangelosi, A., Metta, G.: Integration of speech and action in humanoid robots: icub simulation experiments. IEEE Transactions on Autonomous Mental Development 3.1, 17–29 (2011)
45. Tomasello, M., Akhtar, N.: Two-year-olds use pragmatic cues to differentiate reference to objects and actions. Cognitive Development 10(2), 201–224 (1995)
46. Vavrečka, M., Farkaš, I.: A multimodal connectionist architecture for unsupervised grounding of spatial language. Cognitive Computation 6.1, 101–112 (2014)
47. Vogt, P.: Language evolution and robotics: Issues on symbol grounding. Artificial cognition systems 176 (2006)
48. Werker, J.F., McLeod, P.J.: Infant preference for both male and female infant-directed talk: a developmental study of attentional and affective responsiveness. Canadian Journal of Psychology/Revue canadienne de psychologie 43(2), 230 (1989)
49. Xu, F., Tenenbaum, J.B.: Word learning as bayesian inference. Psychological review 114(2), 245 (2007)
50. Yu, C., Smith, L.B.: Rapid word learning under uncertainty via cross-situational statistics. Psychological Science 18(5), 414–420 (2007)
51. Yu, C., Smith, L.B.: Modeling cross-situational word–referent learning: Prior questions. Psychological review 119(1), 21 (2012)
52. Yurovsky, D., Yu, C., Smith, L.B.: Competitive processes in cross-situational word learning. Cognitive Science 37(5), 891–921 (2013)

A Appendix

Here we describe an extension of the proposed algorithm presented in the Section 2.2 for a case where we have sentence with variable structure. This mean, that we cannot directly associate words from sentence to individual visual features and therefore associations to all visual features must be taken into account. The whole algorithm is described in Alg. 2 in a form of pseudocode.

The Algorithm 2 can be further extended when we incorporate language model and corresponding probabilities of sequence of individual visual features.

Algorithm 2 Sequential mapping – variable sentence

```
Inputs:
    language clusters $L_i$ $(i \in 1 : I)$, visual clusters $K_j^k \sim N(\boldsymbol{m^k}, \boldsymbol{S^k})$, $j \in 1 : J^k$ for each feature $k$, visual input data $\boldsymbol{x_t} = \{\boldsymbol{x}_t^1, \ldots, \boldsymbol{x}_t^K\}$ and corresponding language data $W_t = \{w_t^1, \ldots, w_t^{N_t}\}$ for each trial $t$, number of clusters $J^k$ for each feature $k$
Output:
    mapping between all visual classes $K_j^k$ and language classes $L_i$

while $\sum J^k > 0$ and $\boldsymbol{x}$ is not empty do
    $l_t^n \leftarrow$ assign each word $w_t^n$ from each sentence $t$ to a language cluster (Winner-takes all, see Eq. (14), $l_t^n = \mathrm{argmax}_i(P(w_t^n|L_i))$
    for $k \in$ {size, colour, orientation, texture, shape} do
        $v_t^k \leftarrow$ assign each datapoint $x_t^k$ to a visual cluster (Winner-takes all, see Eq. (14), $v_t^k = \mathrm{argmax}_j(P(x_t^k|K_j^k))$
        for $j = 1 : J^k$ do
            for $i = 1 : I$ do
                $T_{ij}^k \leftarrow$ how many times did visual class $i$ coocurred with language class $j$ $(T_{ij}^k = \sum_{t;v_t^k==j} \sum_n \delta(l_t^n, i))$, where $\delta$ is Kroenecker delta
            end for
        end for
    end for
    $[km, im, jm] \leftarrow \mathrm{argmax}_k \, \mathrm{argmax}_i \, \mathrm{argmax}_j \, T_{i,j}^k$ (the visual cluster $K_{jm}^{km}$ is mapped to the language cluster $L_{im}$)
    $\boldsymbol{x_{del}^{km}} \leftarrow$ data points assigned to both $K_{jm}^{km}$ and $L_{im}$
    $\Theta_{new,J^k}^k \leftarrow N(\boldsymbol{x_{del}^k})$ learn Gaussian on the to be deleted data
    $\boldsymbol{X}^{km} \leftarrow \boldsymbol{X}^{km} \setminus \boldsymbol{X}_{del}^{km}$ delete all data points assigned to both $K_{jm}^{km}$ and $L_{im}$
    $J^{km} \leftarrow J^{km} - 1$
    relearn $K_j^{km} \sim N(\boldsymbol{m^k}, \boldsymbol{S^k})$ on new data $\boldsymbol{x^{km}}$ with $J^{km}$ number of clusters
end while
cluster visual data using new $\Theta_{new}^k$ parameters (cluster centres $\boldsymbol{m^k}$ and covariance matrices $\boldsymbol{S^k}$) and perform One-step mapping (Model 1)
```