Calculation of object position in various reference frames with a robotic simulator

Marcel Švec (svec.marcel@gmail.com) and Igor Farkaš (farkas@fmph.uniba.sk)
Department of Applied Informatics, Comenius University
84248 Mlynská dolina, Bratislava, Slovakia

Abstract

The brain encodes the space in various reference frames. The key role in spatial transformations is played by the posterior parietal cortex where neurons combine retinal location of visual stimulus with gaze direction to encode spatial information. This nonlinear dependence of neuronal responses, gain modulation, is considered a fundamental computational principle used in the brain. The important insight can be obtained through computational models, typically artificial neural networks. In this paper, we test the Zipser–Andersen model but use more realistic and variable stimuli, employing the simulated iCub robot. The multi-layer perceptron was able to successfully perform coordinate transformation from eye- to body-centered reference frame, using gaze information. Model achieves high accuracy of 2 to 4 degrees on testing data, depending on the dataset variability. We provide visualisation techniques for analysing the network, and the effects of gain modulation and shifting receptive fields. Our results confirm previous findings that hidden neurons use various intermediate codings that mediate transformations.

Keywords: reference frames; coordinate transformation; neural network; robotic simulator; gain field

Introduction

Determining the object position in space is always related to some point at known location. This relationship is captured by the concept of *reference frame* in which we can define a concrete coordinate system (Batista, 2002). Humans are able to use both egocentric and allocentric reference frames, which can be combined to support behavior according to the task (Burgess, 2006). In general, reference frame may be anchored to practically anything, to our head, hand, or any other object. Neuroscientists naturally adopted the concept of reference frames to better understand how the space is represented in the brain. Cells in the visual system respond to the stimuli located only in particular location called the *receptive field* in cases where the response is considered to be strictly sensory. The receptive field (also called response field) can also refer to the set of patterns that evoke neuron's activation.

Research on reference frames that has been progressing for the past 25 years suggests that reference frames do not always exist in an explicit form, but rather as some intermediate representations of space that are further processed for specific purposes, for instance to generate reaching commands. The process of converting sensory stimuli into the motor plans is referred to as *sensorimotor transformation*. These are often formalised in terms of spatial transformations from eye-centred (retinotopic) or head-centered coordinates into hand-centered coordinates.

Coordinate transformations are also a key component for learning the body schema (Hoffmann et al., 2011). In most cases, the authors use artificially generated inputs and outputs for training the model. In this paper, we take advantage of the simulated iCub robotic environment (Tikhanoff et al., 2008) that naturally provides embodied data of higher complexity for learning the task. The classical view, coming from geometry, applied commonly in robotics, assumes that coordinate transformations are computed explicitly and applied sequentially. On the contrary, in a novel view, being more consistent with neuroscientific data, coordinate transformations are computed implicitly and in parallel (Blohm & Crawford, 2009). Hence, we analyze the phenomenon of coordinate transformations in the context of the progressive cognitive robotics that offers a promising pathway to building autonomous systems. We are not aware of this type of work with iCub. For coordinate transformation, we use the original Zipser–Andersen model described below, and show that the implicit transformation can be learned equally well despite using more complex data. We also introduce visualization techniques that reveal model behavior.

Background

Andersen and Mountcastle (1983) discovered that neurons in area 7a of posterior parietal cortex (PPC) of monkeys combine retinal location of visual stimulus with gaze direction to encode spatial information. The role of PPC as a sensorimotor interface for visually guided eye and arm movements has been also supported by later findings (Buneo & Andersen, 2006; Khan, Pisella, & Blohm, 2012). Cells in PPC appear to nonlinearly combine information from different modalities, while their sensitivity is modulated by one modality (e.g. gaze direction) without changing their selectivity to the other modality (visual stimuli). This phenomenon was coined as *gain modulation* and the changes in neuron's sensitivity as *gain fields*. The subsequent studies of gain modulation have revealed that it is an extremely widespread mechanism that appears to be a fundamental computational principle behind coordinate transformations (Salinas & Thier, 2000; Salinas & Sejnowski, 2001; Blohm & Crawford, 2009).

The types of signals that could produce gain fields include gaze direction, head position, eye vergence, target distance, chromatic contrast or attention, all together leading to the suggestion that gain modulation is a general mechanism for *multimodal integrations* that underlie important cognitive functions like sensorimotor transformation, object recognition, motion processing or focusing attention (Salinas & Thier, 2000). The essential feature of gain fields is nonlinearity. However, the biophysical basis that allows neurons to combine information from two sources such that their output is close to the product of two functions is still unclear.

In the first computational model, Zipser and Andersen

(1988) trained an artificial neural network to compute the head-centered position of target from eye-centered visual stimuli and gaze direction. Their network spontaneously developed visual receptive fields (RF), gain modulated by eye position similar to what had been observed in PPC in area 7a (Andersen & Mountcastle, 1983). Their artificial simuli used for training the network were simple but nicely illustrated the concept of gain fields as analysed in the hidden layer of the three-layer feedforward network.

Since then, computational models have been proposed to account for coordinate transformations. For instance, Xing and Andersen (2000) analyzed simple neural network models of the PPC and concluded that the gain field is an effective mechanism for performing coordinate transformations. Blohm and Crawford (2009) studied coordinate transformations in the context of visually guided reaching in 3D. Their four-layer neural network was successfully trained to perform visuomotor transformation from gaze-centered inputs to a shoulder-centered output used for reaching.

The model

The neural network performs transformation from the eye-centered (retinal) object position to the body-centered object position (attached to robot's waist),[1] modulated by the information about eye positions (which determine gaze direction).

Data generation

To generate the data in iCub simulator, we randomly moved iCub's eyes and randomly positioned an object in robot's visual space. Then we collected the data as shown in Figure 1.

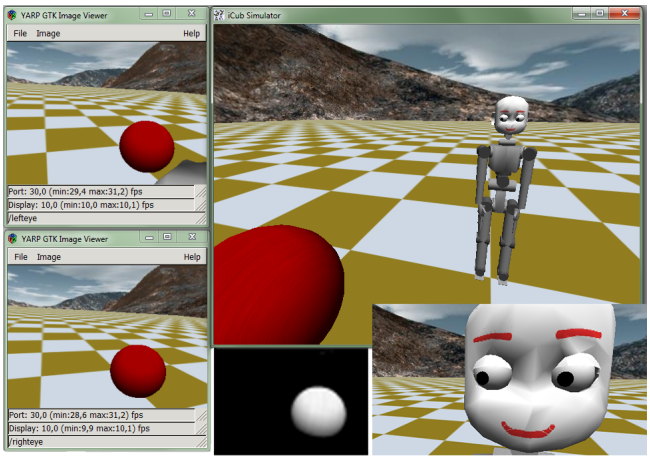

Figure 1: Generating the dataset in iCub simulator involves: setting the eye position, positioning the object in the scene, and collecting 3 pieces of data: retinal images (shown left), eye positions and target object position. We removed the background to get a B&W image.

[1]Our model differs in this detail from Zipser–Andersen model that uses head-centered coordinates. Howeveer, both models can be seen as equivalent, since the taget reference frames only differ in vertical coordinate.

During random eye positioning, we needed to determine iCub's field of view. Two cameras with a resolution 320×240 pixels use a simple pinhole projection with the focal lengths equivalent to 257 pixel units (in both directions), which yields a field of view of $\sim 64^\circ$. We kept both eyes parallel in the simulator (i.e. no convergence). In order to generate the data, we randomly placed an object in front of iCub and randomly moved its eyes by the same angle, regardless of the object position. We always checked that the object remained in the visual field and did not get under the ground. The simulator has three predefined object types: box, sphere and cylinder. We generated datasets with these 3 types and also datasets with only spheres. To make the data diverse enough, we generated objects with random sizes, within a reasonable scale range. We repeated the object setting procedure 1500 times to generate a sufficient number of training and testing patterns. These patterns covered the entire visual space quite densely, such that they could be assumed to be representative.

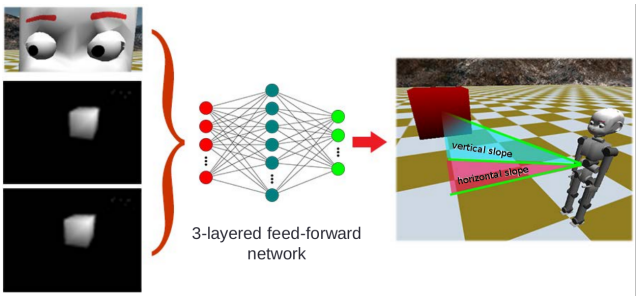

Figure 2: Learned transformation: from eye-centered object coordinates, and given eye-positions to object position in body-centered frame of reference. Robot's head is fixed.

Model architecture

In order to use the dataset as an input for the neural network, we converted each pattern (pixelwise) into the set of real numbers in the interval $[0,1]$. Camera images from the left and right eyes were flipped in both directions and downscaled to 64×48 pixels. For better performance, we also removed the background.[2] The processed image is illustrated in Figure 1 (white ball on black background). The image input was hence represented by $2\times64\times48$ = 6144 neurons. Eye positions and object position were represented by biologically plausible population coding[3] that lends itself to robustness and good generalization (Averbeck, Latham, & Pouget, 2006). Eye positions are represented by eye tilt and eye version. Eye tilt is encoded by 11 neurons with preferred directions uniformly distributed over the interval $[-35^\circ, 15^\circ]$ and eye version by

[2]This diminishes our motivation to use realistic data, but what is still preserved is the varying object size and the shading. We assumed that the image segmentation component performed figure–ground separation.

[3]In population coding, a value x is represented as a vector of activations y_i of neurons with equidistantly shifted centres x_i of their Gaussian RFs with the same width σ: $y_i(x) = \exp(-(x-x_i)^2/2\sigma^2)$.

21 neurons distributed over the interval $[-50^\circ, 50^\circ]$. The object position (network output) is represented by two slopes (shown in Figure 2), horizontal (19 neurons) and vertical (19 neurons), making it a 2.5D model that calculates the direction to the object from iCub's chest, rather than the distance. The preferred directions of output neurons are for both coordinates uniformly distributed over the interval $[-90^\circ, 90^\circ]$.

We used a multi-layer perceptron with full connectivity between layers. The input layer consisted of $6144 + 11 + 21 = 6176$ neurons, the hidden layer had 64 neurons (result of experimentation in the range 50–300; performance with smaller values was limited) and the output layer contained 38 neurons. It turned out to be useful to also optimize the slope k of the neuron's activation function $f(net) = 1/(1+\exp(-k.net))$. For the hidden layer we used $k_H = 0.05$ and for the output layer $k_O = 0.1$, found experimentally.

Input balancing

Since there were more input units encoding the retinal image than those encoding the eyes position (6144 versus 32), we decided to proportionally modify the weights of input–hidden connections in order to guarantee good functioning of the model (in the original Zipser–Andersen model this was not an issue). Here we solved this problem by using the appropriate scaling coefficient. This can be expressed as

$$net = c_{\text{ret}} \sum_{i=1}^{N_{\text{ret}}} w_i r_i + c_{\text{pos}} \sum_{j=1}^{N_{\text{pos}}} w_j e_j \qquad (1)$$

where *net* is the input to a hidden neuron, c_{ret} and c_{pos} are the coefficients used for balancing retinal inputs r_i and eye-position inputs e_j, N_{ret} and N_{pos} are the numbers of units encoding the corresponding modality. We calculated the two coefficients to correspond to the desired ratio R:E, where R is the desired size of retinal inputs contribution and E is the desired contribution of eye-position inputs. The equations for calculating both coefficients are

$$c_{\text{ret}} = \frac{R(N_{\text{ret}} + N_{\text{pos}})}{N_{\text{ret}}(R+E)}, \quad c_{\text{pos}} = \frac{E(N_{\text{ret}} + N_{\text{pos}})}{N_{\text{pos}}(R+E)}. \qquad (2)$$

We chose R:E = 2:1, but there was no significant difference in network performance for slightly different ratios. Even the original network (i.e. without balancing) was able to successfully perform the transformations, but by setting this ratio we achieved faster training, better accuracy and weight profiles that were nicer for visualisation purposes.

Results

We tested several versions of backpropagation (BP) algorithm (using the FANN simulator; (Nissen, 2005)) and stopped the training when the mean-squared-error decreased below the value 5×10^{-4}, found experimentally. In the first trials, we achieved the best training performance with RPROP algorithm (Riedmiller & Braun, 1993) which performed approximately 8 times faster than standard BP and about 10 times faster than QuickProp algorithm (Fahlman, 1988). Adding the momentum to standard BP dramatically increased the speed of training. Inspired by Qian (1999), we used values of the learning rate ($\alpha = 1.5$) and the momentum ($\mu = 0.9$), with which the online BP outperformed RPROP. The additional disadvantage of RPROP and QuickProp algorithms is that they often generated large weights and were thus not suitable for visualisation purposes. This is an interesting point, because it indicates that the right choice of the training algorithm may be important for the purposes of studying the internal structures of the network.

Table 1: Testing errors as a function of dataset variability.

Data set	Error
Boxes, spheres and cylinders at various sizes	$4^\circ \pm 3.5^\circ$
Spheres of various sizes	$3^\circ \pm 3^\circ$
Spheres of fixed size	$2^\circ \pm 2^\circ$

The training dataset contained 1000 patterns, the testing data set 500 patterns which varied according to the variability. We summarize the testing errors in Table 1. The best results were for these model parameters: the population curves with $\sigma_T = 5$ for eye-tilt neurons, $\sigma_V = 7$ for eye-version neurons, and $\sigma_O = 10$ for output neurons coding the object position. We did not find any significant correlation between the error size for a particular pattern and the position of eyes or the object. The distribution of testing errors was positively skewed (i.e. towards smaller errors). These results reveal that even in this more realistic scenario with the iCub robot the modelled transfomation is accurate and generalizes well. The model performance on the more complex datasets is comparable to that of monkeys trained on saccades (Robinson, Noto, & Bevans, 2003).

In the following we analyze the hidden layer of the trained network, focusing on three aspects: receptive fields, gain modulation and the reference frames. We illustrate the model properties using one hidden neuron (unit 4) and show that all hidden units learn various intermediate reference frames.

Receptive fields

After the network learned to accurately perform the transformation, we examined the hidden layer for the effect of gain modulation and shifting RFs. For this purpose, we first visualised the RFs of hidden units by plotting their incoming weights. We found a wide variety of RFs but these could be roughly divided into three groups as shown in Figure 3.

In group A, we can distinguish continuous area with positive weights contrasting with an area of smaller or negative weights (e.g. neuron 4). Group B has RFs divided into two parts, usually with stronger weights on the sides. In group C, we were not able to find any continuous area and the RF was hard to interpret without further investigation. Quantitatively, in our network with 64 hidden units we found 41 units of type A, 15 of type B and 8 of type C. These numbers are specific for the given network and would be slightly different if we

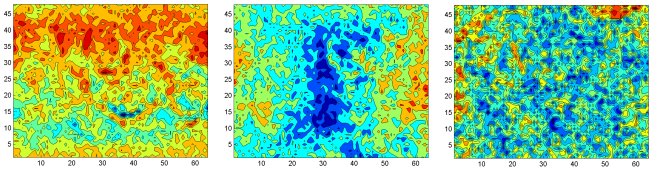

Figure 3: Examples of receptive fields: continuous (left), divided into two parts (middle), unspecified (right).

repeated the training process. Based on the analysis of well-trained models, we may conclude that the majority of units have developed continuous RFs for particular area(s) in the visual space.

Gain modulation

Gain modulation is revealed when a modulatory input changes the response amplitude of a neuron to the other input, without modifying its selectivity (Salinas & Sejnowski, 2001). To examine this effect, we recorded the responses of hidden units to the fixed visual stimuli and different eye positions, which were changed in a systematic way with a 10° step in vertical direction (tilt) and 20° step in horizontal direction (version). Hence, we considered 25 different eye positions arranged in a 5×5 grid.

We repeated this process with 9 different retinal images that depicted the object at particular locations, arranged analogically in a 3×3 grid, from top left to bottom right region of the image. Thus, all together there were 9×25 different configurations of visual stimuli and eye positions that lead to neuron's response profile (i.e. a vector of 225 responses). We investigated the response profiles of the hidden units, and illustrate here the model behaviour using one example: response profile of neuron 4, shown in Figure 4.

Panels A–I indicate the object locations relative to iCub gazing straight ahead (A means the object was up left; I means bottom right). Empty magenta circles show how the neuron would respond only to the visual stimuli without the influence of eye position. In every panel, filled blue circles represent unit responses to the visual stimuli modulated by corresponding eye position. Top left circle denotes response when gazing top left, bottom right circle corresponds to gazing bottom right. The plus sign means that the effect of modulation is excitatory. The effect of gain modulation is evident in all panels. For instance, in panel D, blue bottom circles illustrate that the unit is active even though its response to purely visual input is weak. Gain modulation has the same direction as the RF (see also Figure 6 and the associated text), meaning that the RF is sensitive to the object at the bottom and the effect of gain modulation is highest when gazing down.

We can arrive at consistent conclusion when looking at hidden–output connections and output neurons (see Figure 5). Consider an output neuron whose activity indicates that the object is located close to the ground. This neuron is fed by the population of hidden neurons, each of which can be thought of as indicating a specific position of the object. The output neuron thus collects these indications and responds. Let us look at the hidden neuron 4 whose RF suggests that it responds to objects located on the ground. Let us consider various combinations of the visual stimulus and eye positions. The object cannot be on the ground when iCub looks up and sees an object. When iCub gazes straight ahead, the object is on the ground only when its projection falls on the top part of the retina. When iCub gazes down, the object is always on the ground. We would expect that the output neuron indicating this position will have strong connections from hidden neuron 4. This is actually what one can see in Figure 5 when looking at the weights from hidden neuron 4 to the output units. Looking down corresponds to $-35°$ (of the range $[-90°, 90°]$), therefore the strongest connections are in the left part close to the centre (components 6–9).

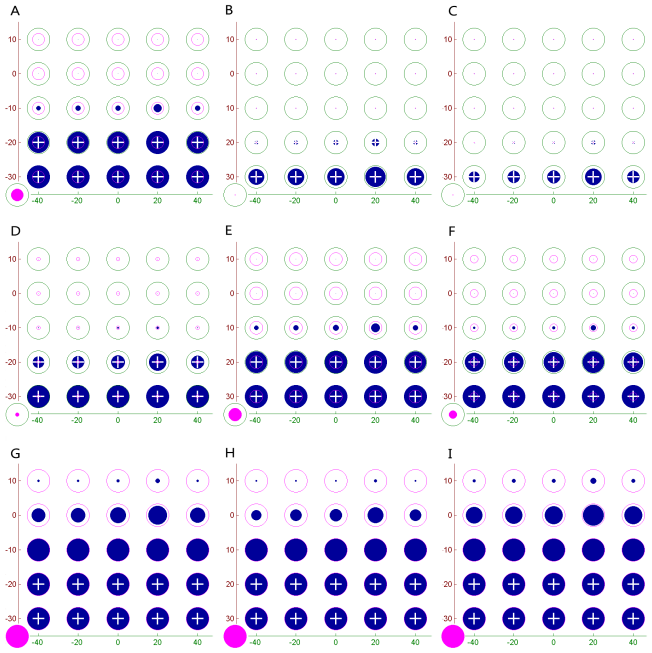

Figure 4: Gain modulation of hidden neuron's response. For explanation see the text.

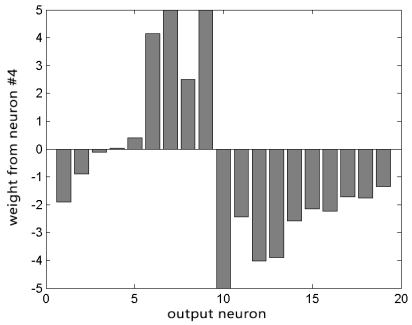

Figure 5: Weights of connections between neuron 4 and output neurons encoding vertical object position.

RF–GF differences Following the procedure described in Xing and Andersen (2000), we also analyzed the hidden neurons in terms of direction differences between the RF and the gain field (GF). Unit's RF is defined as the input area that evokes more than 50% of unit's maximal response. We examined the relationship between unit's gain and the RF by comparing their directions. The GF direction points to the best-tuned unit relative to the central eye position, and RF direction is calculated as the center of mass of the unit's response across the input map. The angle between these two directions, i.e. RF–GF, serves for testing whether GF and RF are aligned in the same or opposite way. We show RF–GF direction differences for all hidden units in Figure 6. We can see that the majority of hidden units has this difference close to zero, which implies that RF and GF are alignedThis also replicates results of Model 1–4 in Xing and Andersen (2000).

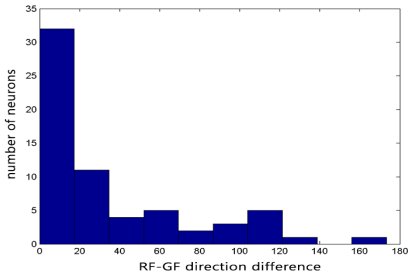

Figure 6: Histogram of RF–GF direction differences.

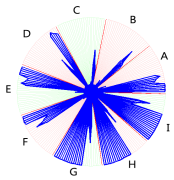

Figure 7: Star plot visualisation of the response profile (neuron 4). Each neuron response is plotted on one axis.

Variability of response profiles We observed that the response profiles of many hidden units are more complex than the one shown in Figure 4. To explore if there are any characteristic profiles or clusters of profiles, we used a visualisation method based on star plots. For illustration, visualisation of the response profile of neuron 4 is shown in Figure 7. We then trained a one-dimensional self-organizing map (SOM) with 64 units to topographically organize response profiles (Kohonen, 1982). Figure 8, nicely reveal gradual changes in these profiles both in terms of (direction/position) selectivity and in terms of amplitude (line length, gain modulated). This suggests that there is a *continuum* of responses profiles (rather than a discrete set) that emerged in the hidden layer as a result of learned transformation. We can also say that the response profiles appear to be specialized in similar manner as the RFs due to the effect of gain modulation.

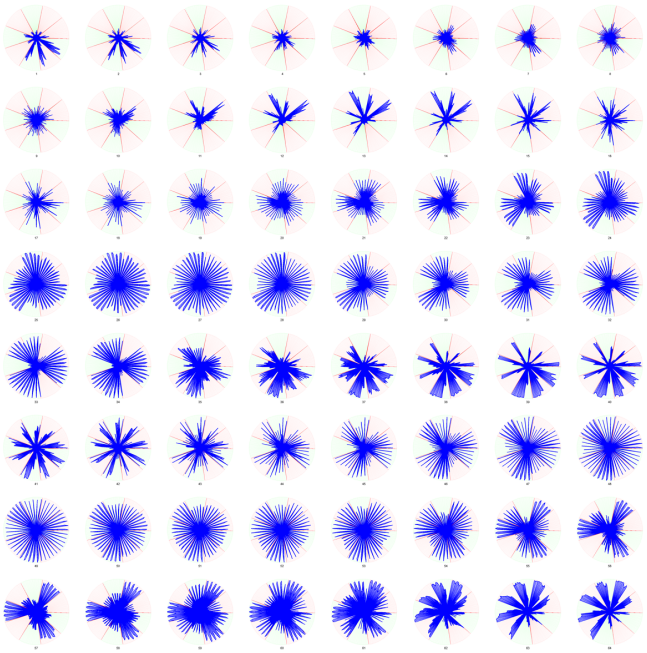

Figure 8: Response profiles, topographically sorted by the SOM. Every circle represents the response profile of one hidden unit.

Reference frames

Here we examined how the centre of mass of the RF shifts for different gaze directions. To determine the centre of RF, we swapped the organizion of units' response profiles to get a grid of 25×9 responses. In order to determine the reference frames used by the population of hidden neurons, we computed the absolute shifts of RFs and their standard deviations for all units and put them into histograms in Figure 9. Absolute shifts close to zero are interpreted as encoding in eye-centered reference frame. Absolute shifts close to one indicate body-centered reference frame. Since we observe none of these situations, we conclude that the hidden layer encodes the object position in intermediate coordinates between eye- and body-centered reference frames.

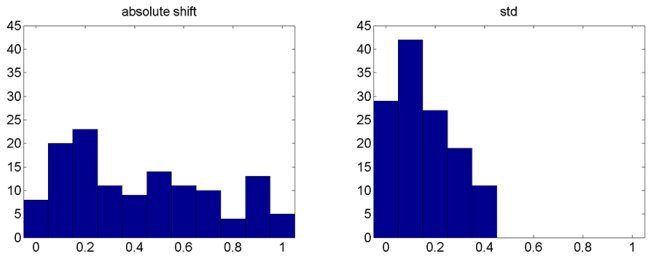

Figure 9: Histograms of absolute RF shifts (left) and their standard deviations (right). The vertical axis denotes the number of units with the given value of the shift. Both horizontal and vertical shifts are superimposed, so the sum of bars in each histogram is $2 \times 64 = 128$.

Conclusion

We replicated experiments with Zipser–Andersen model, using more complex (more realistic) data generated with the iCub simulator. For learning the model, we achieved the best results with the standard version of BP with the momentum term. The network was able to successfully perform the transformation task with testing accuracy within 2° in case of a more homogeneous set of objects, and with accuracy of 4° in case of more variable object sets. This implies that coordinate transformations could be successfully realized using the data from iCub simulator. We examined the hidden layer by means of visualisation techniques that revealed the nonlinear effect of gain modulation and shifting receptive fields. The results of the reference frame analysis indicate that the hidden neurons encode object position in the intermediate reference frames between eye- and body-centered coordinates. It is interesting that these reference frames are actually an emergent process that results from error minimization within a supervised learning task. It is possible that similar emergent processes could take place in the brain, possibly implemented by mechanisms other than BP that is considered biologically implausible. However, alternatives exist that avoid explicit error propagation between layers (O'Reilly, 1996) and share some features also with unsupervised Hebbian learning.

The brain must be able to integrate different sources of information which can significantly differ in terms of the number of afferent pathways, to avoid dominance of one modality to the expense of the other. Some theories and computational models can be found in Makin, Fellows, and Sabes (2013) and references therein. Given the higher dimensionality of input data we optimized the integration of different modalities in a more straightforward, albeit hardwired manner.

Acknowledgments

This work was supported by the project 1/0898/14 of the Slovak Grant Agency for Science (VEGA). M. Švec was a master student at the Department of Applied Informatics. We thank three anonymous reviewers for their comments.

References

Andersen, R., & Mountcastle, V. (1983). The influence of the angle of gaze upon the excitability of the light-sensitive neurons of the posterior parietal cortex. *Journal of Neuroscience*, *3*(3), 532–548.

Averbeck, B., Latham, P., & Pouget, A. (2006). Neural correlations, population coding and computation. *Nature Reviews Neuroscience*, *7*, 358-366.

Batista, A. (2002). Inner space: Reference frames. *Current Biology*, *12*(11), 380–383.

Blohm, G., & Crawford, J. (2009). Fields of gain in the brain. *Neuron*, *64*(5), 598–600.

Buneo, C., & Andersen, R. (2006). The posterior parietal cortex: sensorimotor interface for the planning and online control of visually guided movements. *Neuropsychologia*, *44*(13), 2594–2606.

Burgess, N. (2006). Spatial memory: how egocentric and allocentric combine. *Trends in Cognitive Sciences*, *10*(12), 1–7.

Fahlman, S. (1988). *An empirical study of learning speed in backpropagation networks* (Tech. Rep. No. CMU-CS-88-162). Pittsburgh, PA: Carnegie Mellon University.

Hoffmann, M., Marques, H., Arieta, A., Sumioka, H., Lungarella, M., & Pfeifer, R. (2011). Body schema in robotics: a review. *IEEE Transactions on Autonomous Mental Development*, *2*(4), 304-324.

Khan, A., Pisella, L., & Blohm, G. (2012). Causal evidence for posterior parietal cortex involvement in visual-to-motor transformations of reach targets. *Cortex*, *49*, 2439–2448.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, *43*, 59-69.

Makin, J., Fellows, M., & Sabes, P. (2013). Learning multisensory integration and coordinate transformation via density estimation. *PLOS: Comput. Biology*, *9*(4).

Nissen, S. (2005). *Fast artificial neural network library.* Available from `http://leenissen.dk/fann/wp/`

O'Reilly, R. (1996). Biologically plausible error-driven learning using local activation differences: The generalized recirculation algorithm. *Neural Computation*, *8*(5), 895–938.

Qian, N. (1999). On the momentum term in gradient descent learning algorithms. *Neural Networks*, *12*(1), 145-151.

Riedmiller, M., & Braun, H. (1993). A direct adaptive method for faster backpropagation learning: The Rprop algorithm. In *IEEE international conference on neural networks* (pp. 586–591). IEEE Press.

Robinson, F., Noto, C., & Bevans, S. (2003). Effect of visual error size on saccade adaptation in monkey. *Journal of Neurophysiology*, *90*(1), 1235-1244.

Salinas, E., & Sejnowski, T. (2001). Gain modulation in the central nervous system: Where behavior, neurophysiology, and computation meet. *Neuroscientist*, *7*, 430–440.

Salinas, E., & Thier, P. (2000). Gain modulation: A major computational principle of the central nervous system. *Neuron*, *27*, 15–21.

Tikhanoff, V., Fitzpatrick, P., Nori, F., Natale, L., Metta, G., & Cangelosi, A. (2008). The iCub humanoid robot simulator. *Advanced Robotics*, *1*(1), 22-26.

Xing, J., & Andersen, R. (2000). Models of the posterior parietal cortex which perform multimodal integration and represent space in several coordinate frames. *Journal of Cognitive Neuroscience*, *12*(4), 601–614.

Zipser, D., & Andersen, R. (1988). A backpropagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature*, *331*, 679–684.