ASSOCIATIVE NEURAL MODELS FOR BIOMIMETIC MULTI-MODAL LEARNING IN A MIRROR NEURON-BASED ROBOT

STEFAN WERMTER, CORNELIUS WEBER, MARK ELSHAW
Hybrid Intelligent Systems, School of Computing and Technology,
University of Sunderland,
Sunderland, SR6 0DD, UK
[Stefan.Wermter, Cornelius.Weber, Mark.Elshaw]@Sunderland.ac.uk
www.his.sunderland.ac.uk

By using neurocognitive evidence on mirror neuron system concepts the MirrorBot project has developed neural models for intelligent robot behaviour. These models employ diverse learning approaches such as reinforcement learning, self-organisation and associative learning to perform cognitive robotic operations such as language grounding in actions, object recognition, localisation and docking. In this paper we describe architectures based on an associative self-organising framework which were designed to combine multimodal inputs of language, vision and motor programs to produce complex robot behaviours.

1. Introduction

In this paper we will describe some research performed as part of the "biomimetic multimodal learning in a mirror neuron-based robot" (MirrorBot) project for neuroscience-based models for an intelligent robot. Theories and experiments in neuroscience have indicated that a neuroscience-oriented approach for multimodal processing is promising for new computational techniques to associate vision, language and motor control (Pulvermuller 1999, Rizzolatti and Arbib 1998, Wermter et al. 2001). In particular the neuroscience motivation for our models comes from the neurocognitive evidence on action verb processing in the brain and the mirror neuron system found in primates and humans.

Neurocognitive evidence on word processing shows that cortical assemblies have been identified in the cortex that activate in response to the performance of motor tasks at a semantic level (Pulvermuller 1999, Rizzolatti and Arbib 1998, Hauk and Pulvermüller 2004). The meaning of words is critical for determining the cortical populations or cell assemblies that form a functional unit that are activated to implement the cognitive task (Pulvermüller 1999, Wermter and Elshaw 2003). For instance, perception words are represented by perisylvian

assemblies and posterior cortex, and nouns related to animals activate the inferior temporal or occipital cortices (Pulvermüller 1999).

Neurocognitive evidence on action verb processing sees a division of representation in the brain based on whether the action is performed by a specific body part (Hauk et al. 2004, Pulvermüller 1999, Pulvermüller 2001). Various EEG experiments on processing of action verbs were carried out to test this hypothesis by looking at words related to the action performed by the leg, arm and face/head. The differences based on different body parts for action verbs processing can be seen from the average response times for lexical decisions. They are faster for face-associated words compared to arm-associated words, and the arm-associated words are faster than leg-associated words. Leg-words generated greater activation in the central brain region around the vertex, while face-words activated inferior-frontal areas, thereby suggesting that the relevant body part representations are differentially activated when words for different actions are being comprehended. This is schematically depicted in Figure 1, with the light circled cell assemblies shared by all body types and the darker assemblies being specific to the particular body type.

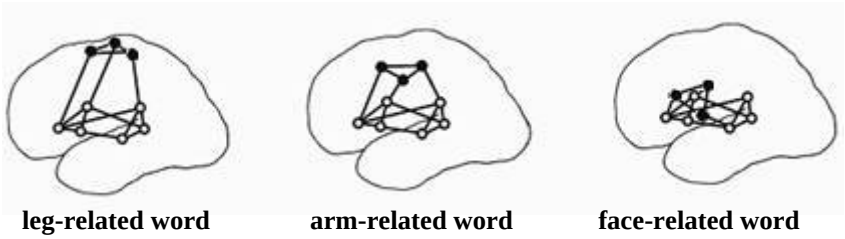


Figure 1. The cell assemblies found to be associated with the processing of action words based on body parts (Pulvermüller 2003).

Focusing on individual neurons, Rizzolatti and Arbib (1998) found two types of neurons located in the F5 motor cortex region of primates, the classical motor neurons which only respond to the performance of the action and the mirror neurons which respond not only when performing an action but also when seeing or hearing the action performed (Kohler et al. 2002, Rizzolatti and Arbib 1998).

This ability to understand actions allows the observer to learn through imitation, to predict the actions and to act accordingly (Gallese 1998). The

concepts from mirror neurons suggest that own actions, observed actions and language are very much interrelated since the same mirror neurons fire when any of these three modalities is an input.

In response to this neuroscience evidence we have developed and will describe various neural models for biomimetic multimodal learning in a mirror neuron-based robot. These models perform activities such as object localization, grounding of language in actions and robot docking. These models are based on diverse learning approaches that are proposed for distinct regions of the brain and we will describe our overall associator approach. Finally, we will consider a self-organising architecture to combine concepts from these neural models to produce a multimodal behaviour in a robot (Figure 2). This architecture takes as inputs language, vision and actions. This architecture is able to associate these so that it can produce or recognize the appropriate action. The architecture either takes a language instruction and produces the behaviour or receives the visual input and action at the particular time-step and produces the language representation (Wermter et al. 2004).

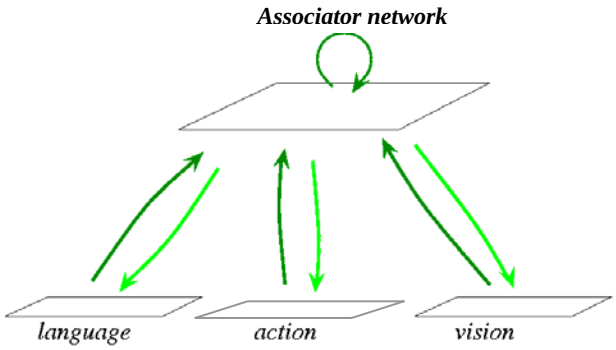


Figure 2. The feature extractor and associator architecture for multimodal integration.

2. Associator model for object localization

Our model for associative object localisation in Figure 3 consists of a "what" pathway on the left, and a "where" pathway on the right (Weber and Wermter 2003). The "what" pathway consists of an input area and a hidden area. The hidden area of the "what" pathway consists of two layers. The lower layer receives bottom-up connections W^{bu} from the input. These have been trained

according to a sparse coding Helmholtz machine and represent feature detectors similar to Gabor functions, but also with colour sensitive elements. The depicted top-down weights $\boldsymbol{W}^{td}$ were needed to train $\boldsymbol{W}^{bu}$, but are not used further on. The upper layer receives a one-to-one copy of the output of the lower layer cells (denoted by the 3 thin arrows in Figure 3). After it receives this initial input, it functions as an attractor network which solely updates its activations based on its previous activations. Each cell receives its input from all other neurons via recurrent weights $\boldsymbol{V}^{22}$. In addition, input arrives from the laterally connected area of the "where" pathway via weights $\boldsymbol{V}^{23}$.

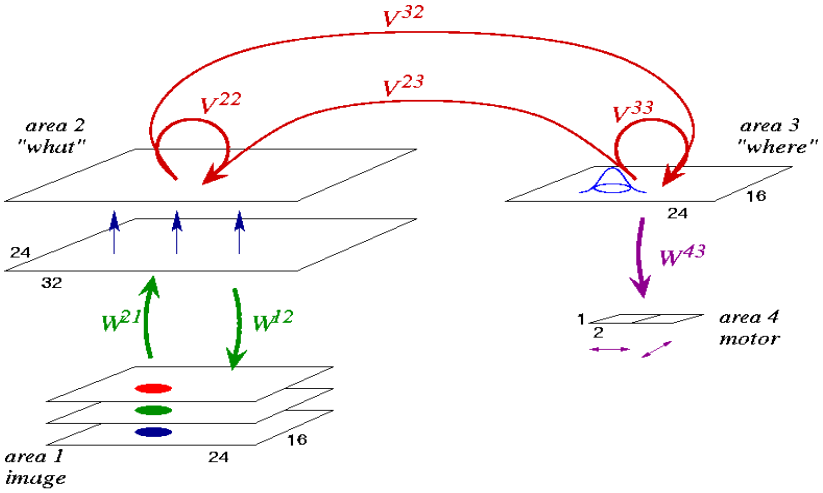


Figure 3. Associator model architecture. Left, the pathway of the lower visual system: below, the three red-, green- and blue-layers of the image, then the feature detecting cells and on top, the attractor cells of the “what” area. On the right, the “where” area displays the location of the object of interest. Small numbers denote simulated area sizes. Thick arrows denote trained weights. The two motor units in this simple setup control the pan- and tilt-position of the robot camera. Their weights $\boldsymbol{W}^{43}$ have been trained in an error driven fashion so that the camera focuses on the object.

The “where” pathway on the right of Figure 3 consists of just one area. The “where” neurons are fully connected via recurrent weights $\boldsymbol{V}^{33}$ and in addition receive input from the highest “what” layer via $\boldsymbol{V}^{32}$. In the following, we will refer to all connections $\boldsymbol{V}^{22}$, $\boldsymbol{V}^{33}$, $\boldsymbol{V}^{23}$ and $\boldsymbol{V}^{32}$ collectively as V^{lat}, because they are lateral weights and receive the same treatment, during training as well as

during activation update. The activation update of the "where" and highest level "what" neurons is governed by the following equation:

$$u_i(t+1) = \mathrm{f}\Big(\sum_l v_{il}^{lat} u_l(t)\Big) \quad , \quad \text{where } \mathrm{f}(x) = e^{\beta x} / (e^{\beta x} + n) \qquad (1)$$

Parameters in the neuronal transfer function f are the slope $\beta = 2$ and the sparseness factor n = 8 which leads to little average neuronal activity, if there is no input. The lateral weights are trained from the bottom-up input. Their purpose is to memorise the incoming activities $\boldsymbol{u_i(t=0)}$ as activation patterns which they maintain. Learning maximises the log-likelihood to generate the incoming data distribution by the internal activations $\boldsymbol{u_i(t)}$ if Eq. (1) is applied repeatedly:

$$\Delta v_{il}^{lat} \approx \sum_t (u_i(t=0) - u_i(t))\, u_i(t-1). \qquad (2)$$

The contribution to learning can be seen in the difference term where $\boldsymbol{u_i(t=0)}$ is the data while $\boldsymbol{u_i(t)}$ is produced through recurrent application of Eq. (1) which corresponds to a continuous-valued attractor basin of the attractor network.

First, the weight matrices $\boldsymbol{W^{td}}$ and $\boldsymbol{W^{bu}}$ were trained on small patches randomly cut out from natural images. Lateral weights V^{lat} were then trained with $\boldsymbol{W^{td}}$ and $\boldsymbol{W^{bu}}$ fixed. For this, within each data point (an image patch), an artificially generated orange fruit was placed at a randomly chosen position. The "where" area received a Gaussian hill of activity on the location which corresponds to the one in the input where the orange is presented.
The representation of the image with an orange obtained through $\boldsymbol{W^{bu}}$ on the lower layer was copied to the upper layer cells. This together with the Gaussian hill on the "where" area was used as target training vector. Relaxations were done for four time steps, which had been discovered in empirical tests.

Figures 4 and 5 show the relaxation of the network activities after initialisation with sample stimuli. In all cases, the "where" area neuron's activations were initialised to zero at time $t = 0$. The relaxation procedure therefore completes a pattern which spans both, the "what" and the "where" area, but which is incomplete at time $t = 0$, as can be seen in Figure 4.

All weights in the model have been trained on the basis of real images and are therefore irregular. Localisation quality may vary at slightly different object locations within the image. The third frame in Figure 5, for example, leads to a blurred "where" representation. If information from the second frame would be

taken into account, this may be cleaned up. However, for simplicity and consistency with the training procedure, the algorithm processes only one frame at a time.

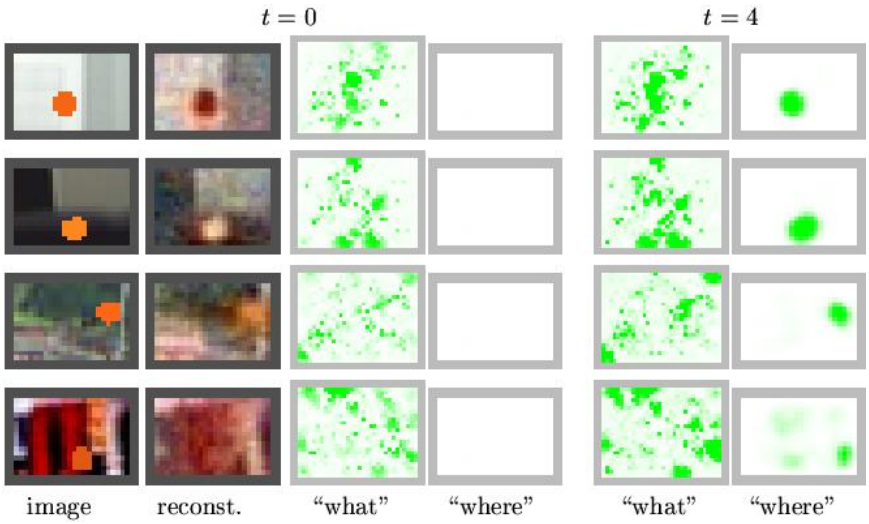


Figure 4. Each row shows the network response to a color image which contains an artificially generated orange fruit. From left to right: the image, the reconstruction of the image via $\boldsymbol{W}^{12}$ (cf. Figure 3), the representation on the "what" area (used for the image reconstruction), the initial zero activities on the "where" area at time $t = 0$. Then the activations on the "what" and "where" areas at $t = 4$ time steps which is the relaxation time used for training. The estimated position of the orange on the "where" area is correct in the upper three rows.

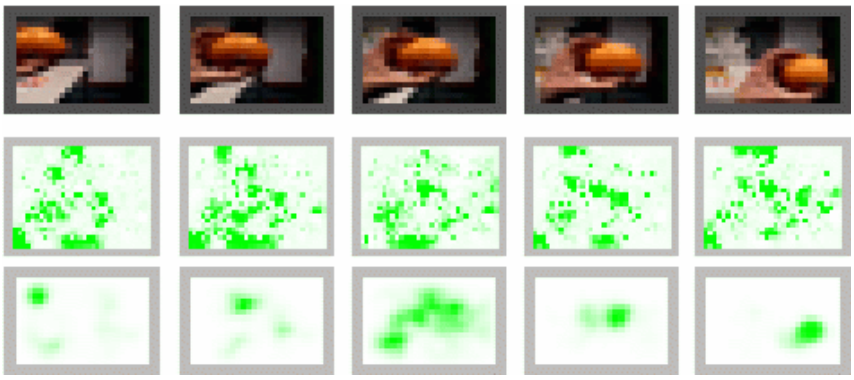


Figure 5. Localisation for real images taken from the robot camera. In each column, the top row shows an image, the middle row shows the response on the "what" area and lower row the response of the "where" area at iteration time $t = 4$ to the presentation of the image in the upper row.

3. Associator architecture for robot docking

After the successful completion of the localization associator network it was expanded as a learning approach to perform the docking behaviour (Weber et al. 2004b). This extended model uses neural vision and reinforcement learning as a solution for robotic docking, which moves the PeopleBot robot toward a table so that it can grasp an object. As the robot has a short non-extendable gripper and wide “shoulders” it must approach the table at a perpendicular angle so that the gripper can reach over it.

Our robot selects the action which leads to the largest expected reward. This makes the action selection network the core part. The model includes four motor neurons that are “on” depending on if the robot is to move forward, backward, left or right, respectively. The peripheral vision module is trained before the action selection network so that it can supply the necessary visually obtained perception as input.

Overall, we have three training phases: stages one and two perform the training for the object localization outlined above, stage three trains the weights W^c and W^m from the conceptual space to the critic of the motor outputs, respectively (Figure 6). During training, the motor units are guided by the firing rate of one “value function” unit which assigns a fitness value to any state. Together, these four neurons are trained by reinforcement learning, in which a scalar reinforcement signal is given only at the end of each training action sequence. The value of the signal is positive, if the robot docks at the object in parallel to the table, or negative, if the robot's shoulders bump into the table at an angle or if the object is lost out of sight.

The input to the action selection network is the robot's visual perceptual state, defined by its relative position to the target, an orange fruit at the border of a table as shown in Figure 7. The localization model described in the previous section is used to provide the location of the object for docking. Additional input to the action selection network is the robot rotation angle φ, supplied by the robot's internal odometry.

Each trial thus constitutes a robot's experience about either reaching the goal or loosing the target or hitting the table. It consists of a sequence of actions

and several learning steps. The more trials have been done, the better the robot performs.

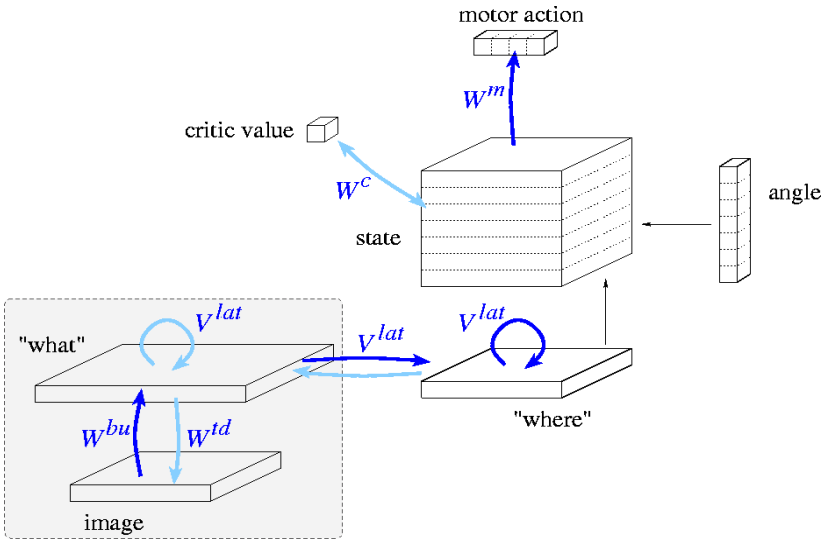
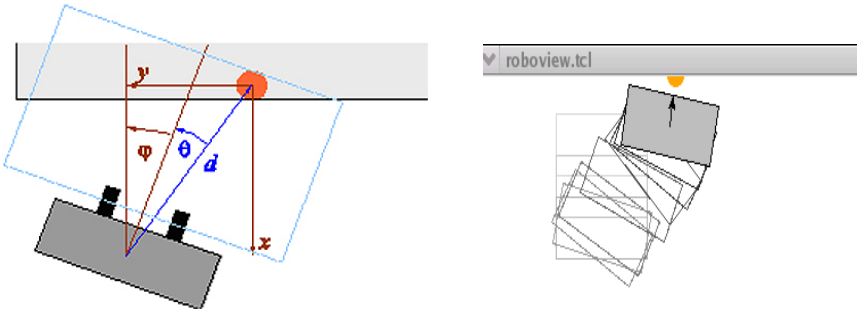


Figure 6 Associator network for action selection. Thick arrows denote trained weights W of which only the dark ones are used during performance, while those depicted bright are involved in training. The "what"-"where" network is the same as in Figure 3, but the object location is used here together with the robot angle as input to an action-critic network that learns the strategic docking manoeuvre by adapting $\boldsymbol{W^m}$ and $\boldsymbol{W^c}$. The shaded part of the network uses the same learning principle as the associator architecture for multimodal integration, described in the next chapter.

The state description (Figure 6) consists of a Gaussian covering several state space units simultaneously. A critic weight w_j^c is thus updated close to weight $w_{j'}^c$ if j and j' are neighbours in the state space; analogously motor weights w_{ij}^m. This topological relation exploits the fact that similar states imply similar optimal actions and speeds up learning. A successful example of simulated docking performance is depicted in Figure 7b which displays a successful movement.

The experiments described in this section of the paper have shown the suitability of associator models for learning robot docking behaviour. Such models offer great flexibility compared to approaches for robot behaviour that

rely on preprogramming. They also indicate that such models could be linked by an associative architecture to achieve multimodal input fusion to achieve complex behaviours based on a goal given as a language instruction.



(a) (b)

Figure 7 a). The figure depicts the table and the target on it. The robot is shown with short black grippers and its field of vision is outlined by the dotted line. Real world coordinates (x, y, φ) specify the position and rotation angle of the robot and the perceived position of the target within the robot's visual field is then defined by angle θ and distance d. 7 b). The simulated trained robot during the docking to the orange object on the edge of a table.

4. Associator architecture for multimodal integration

An overall self-organising associator architecture for multimodal integration of vision, action and language was designed that contains many of the features of the models described above (Elshaw et al. 2004, Weber et al. 2004b). Furthermore, the approach takes inspiration from the mirror neuron system by a student robot recognising and producing the behaviours of a teacher. In our approach a student robot learns from a teacher robot how to perform three separate behaviours 'pick', 'lift' and 'go' based on multimodal inputs. These behaviours were selected as they are typical ones that a robot has to perform and are complex in that the same visual input would require different actions based on the language instruction.

First, a robot simulator was produced with a teacher robot performing 'go', 'pick' and 'lift' actions continuously in an environment (Figure 8). The student robot observed the teacher robot performing the behaviours and was trained by

receiving multimodal inputs. These multimodal inputs were the higher-level visual inputs of the x and y coordinates and the rotation angle φ of the teacher robot, the motor directives of the robot and the language instruction.

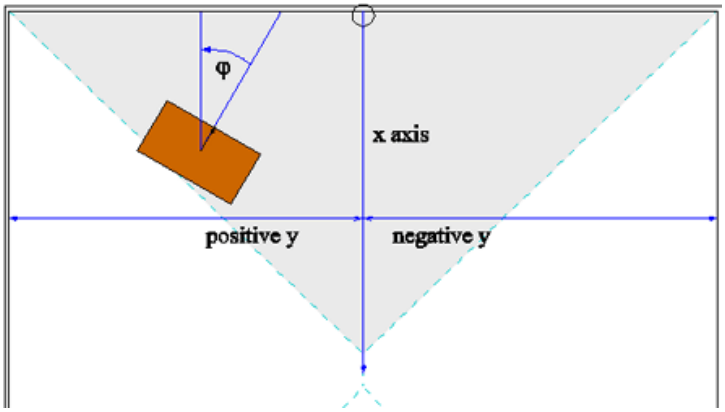


Figure 8. The simulated environment containing the robot. The x, y and φ coordinates of the robot are taken with respect to the nearest wall. Dashed lines indicate the borders of areas "belonging" to a wall. The area belonging to the top wall is depicted in light grey.

The simulated teacher robot performs the three behaviours in reoccurring loops and in the following order: the behaviour represented by the word 'go' involves moving around the environment until it reaches a wall and then turns away from the wall at a set angle. It switches to the docking behaviour outlined above represented by the word 'pick' if it comes close to the target at the top of the arena. The final behaviour follows the behaviour represented by the word 'lift' involving moving backward to leave the table and then turning around to face toward the middle of the arena. When receiving the multimodal inputs, the student robot was required to learn these behaviours so that it could recognise them in the future or perform them based on a language instruction.

The imitation model (Figure 9) used an associator network based on the Helmholtz machine approach (Hinton et al. 1995). The Helmholtz machine generates representations of data using unsupervised learning. Bottom-up weights W^{bu} generate a hidden representation $\vec{r}$ of some input data $\vec{z}$. Conversely, top-down weights W^{bu} $\tilde{z}$ reconstruct an approximation of the data $\vec{z}$ from the hidden representation.

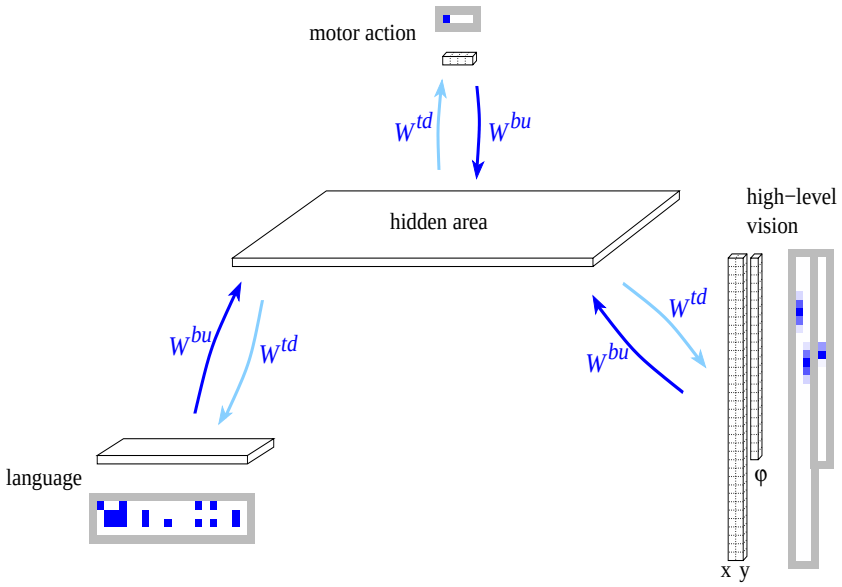


Figure 9. The associator model for robot imitation learning. A stimulus example is shown near each of the three input areas.

Both sets of weights are trained by the unsupervised wake-sleep algorithm which uses the local delta rule. In the wake phase, a full data point $\vec{z}$ is presented which consists of the full motor, language and higher-level visual components. The linear hidden representation $\vec{r} = W^{bu}\vec{z}$ is obtained from which a competitive version $\vec{r}^{\,c}$ is obtained by taking the winning unit of $\vec{r}$ (given by the strongest active unit) and assigning activation values under a Gaussian envelope to the units around the winner. Thus, $\vec{r}^{\,c}$ is effectively a smoothed localist code. The reconstruction of the data is obtained by $\tilde{\vec{z}} = w^{td}\vec{r}^{\,c}$ and the top-down weights from units j to units i are modified according to

$$\Delta w_{ij}^{td} = \eta r_j^c (z_i - \tilde{z}_i) \qquad (3)$$

with an empirically determined learning rate $\eta = 0.001$. The learning rate was increased 5-fold whenever the active motor unit of the teacher changed.

In the sleep phase, a random hidden code $\vec{r}^{\,s}$ is produced by assigning activation values under a Gaussian envelope centred on a random position on

the hidden layer. Its linear input representation $\vec{r}^{\,s} = W^{td}\vec{r}^{\,s}$ is obtained, and then the reconstructed hidden representation $\tilde{r}^{\,s} = W^{bu}\vec{z}^{\,s}$ is obtained from which a competitive version $\tilde{r}^{\,c}$ is obtained by assigning activation values under a Gaussian envelope centred around the winner. All weights W^{td} and W^{bu} were rectified to be non-negative at every learning step and the bottom-up weights W^{bu} of each hidden unit were normalised to unit length.

The hidden layer of the associator network in Figure 9 that acted as the student robot's cortex had 16 by 48 units. In the wake phases of training it received multimodal inputs $\vec{z}$ based on observing the actions of the teacher robot performing the three behaviours.

These inputs included first the higher-level vision which represents the x and y coordinates and rotation angle φ of the teacher robot. The x, y and φ coordinates in the environment were represented by two arrays of 36 units and one array of 24 units, respectively. For a close distance of the robot to the nearest wall, the x position was a Gaussian of activation centred near the first unit while for a robot position near the middle of the arena the Gaussian was centred near the last unit of the first column of 36 units. The next column of 36 units represented the y coordinates so that a Gaussian centred near the middle unit represented the robot to be in the centre of the environment along the y axis. Rotation angles φ from -180^{o} to 180^{o} were represented along 24 units with the Gaussian centred on the centre unit if $\varphi = 0^{o}$.

The bottom-up weights from units i to units j are modified according to

$$\Delta w_{ij}^{bu} = \in (w_{ji}^{bu} - z_i^s)\,\tilde{r}_j^c \qquad (4)$$

with an empirically determined learning rate $\varepsilon = 0.01$.

For the language region representations of phonemes were presented. This approach used a phoneme representation consisting of 20 phonetic features, which produced a different binary activation pattern in the language input region

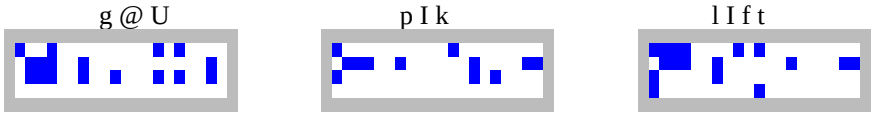


Figure 10. The phonemes and the corresponding 4 x 20-dimensional vectors representing ‘go’, ‘pick’ and ‘lift’.

for each phoneme. A region of 4 rows by 20 columns was used to represent the words with the first row representing the first phoneme and the second row the second phoneme etc., so that the order in which they appear in the word is maintained (Figure 10).

As final part of the multimodal inputs the teacher robot motor directives were presented on the 4 motor units (forward, backward, turn right and turn left) one for each of the possible actions with only one active at a time. The activation values in all three input areas were between 0 and 1.

During training the student robot received all the inputs, however when testing, either the language area for recognition or the motor inputs for production were omitted. Recognition was verified by comparing the units which are activated on the language area via W^{td} (Figure 9) with the activation pattern belonging to the verbal description of the corresponding behaviour. For action production the robot continuously received its own current x, y and φ coordinates and the language instruction of the behaviour to be performed. Without motor input it had to produce the appropriate motor activations via W^{td} which it had learnt from observing the teacher to produce the required behaviour.

The associator model based robot recognised the ‘go’-, ‘pick’- and the ‘lift’ behaviour, while the teacher robot was looping between the three behaviours as done during training. For testing the recognition of a behaviour, the teacher robot was placed at random start positions and performed a corresponding action. Furthermore, the trained student robot could successfully recreate a behaviour based on a language input as can be seen from Figure 11.

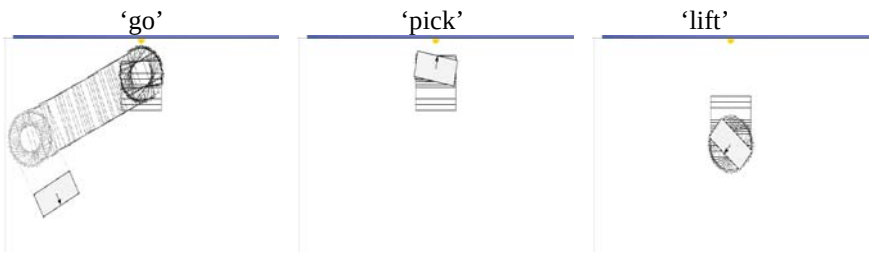


Figure 11 The simulated trained student robot performance when positioned at the same point in the environment but instructed with different language input.

The robot's ability to both recognise an observed behaviour and perform the behaviour that it has learnt by imitating a teacher shows the model was able to recreate some concepts of the mirror neuron system. The student robot produces similar regional unit activation patterns when observing the behaviour and performing it, as seen in Figure 12. In doing so it combined features of the individual models outlined in this paper to achieve the grounding of language and visual information in action in a multimodal approach.

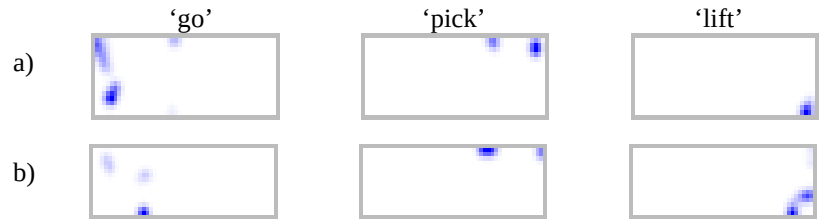


Figure 12. Hidden layer activations for the associator network summed up during short phases while the student robot a) correctly recognises the behaviours and b) performs them based on a language instruction.

6. Discussion

First, learning was done in simulation, then the trained models were transferred to the robot, where the performance of the docking has also been demonstrated. The lower-level vision network ($\boldsymbol{W}^{m}$ and $\boldsymbol{W}^{c}$ in Figure 6) was trained with "real" natural images, i.e. small patches randomly cut out from a set of colour images depicting natural scenes. We have used simulation primarily to ease the generation of data material for training but then transferred the simulation to a real robot environment. The "what"-"where" association network ($\boldsymbol{W}^{lat}$ in Figure 6) was trained supervised and therefore requires controlled conditions. We have used a simple orange coloured disc, pasted into a natural image to denote the target object during learning. During performance, the network localises real orange fruit sufficiently robustly. The actor-critic network ($\boldsymbol{W}^{m}$ and $\boldsymbol{W}^{c}$ in Figure 6) was trained with a simulator. Due to the robustness of the docking (if the robot should turn left, then it should do so in a large area), the learnt weights can also control the real robot. Actor-critic learning advances relatively quickly, and reasonable performance is achieved after a few successful trials of the simulated agent. Therefore, learning using real hardware is also realistic. Finally, the performance of the associator architecture for multimodal integration (all weights in Figure 9) increases slowly compared to the actor-critic network.

While it requires this useful control algorithm to be already present, it can perform learning whenever the robot is active.

7. Conclusion

By combining various neuroscience inspired models for a biomimetic learning in a mirror neuron-based robot we have produced robot behaviours based on visual object localization, language grounding and docking action. The associator architecture was able to combine multimodal inputs of vision, language and motor in order to recognise and produce three behaviours. In doing so we have been able to reproduce an important property of the mirror neuron system and action verb processing. In conclusion we believe mirror-neuron based learning holds a lot of potential for learning robots in the future.

Acknowledgement

This work is part of the MirrorBot project supported by the EU in the FET-IST programme under grant IST-2001-35282.

References

Elshaw M., Weber C., Zochios A., Wermter S. (2004) An Associator Network Approach to Robot Learning by Imitation through Vision, Motor Control and Language. *Proceedings of the International Joint Conference on Neural Networks*, Budapest, Hungary, pp. 591-596.

Elshaw M., Wermter S., Watt P. Self-organisation of Language Instruction for Robot Action. *Proceedings of the International Joint Conference on Neural Networks*. Oregon, USA, pp. 22-27, July 2003.

Gallese, V. and Goldman, A. (1998) Mirror neurons and the simulation theory of mind-reading. *Trends in Cognitive Science*, 2(12), 493-501.

Foster, D., Morris, R. and Dayan, P. (2000) A model of hippocampally dependent navigation, using the temporal difference learning rule. *Hippocampus*, 10, 1-16.

Hauk, O., Johnsrude, I. and Pulvermüller, F. (2004) Somatotopic representation of action of action words in human motor and premotor cortex. *Neuron*, 41, 301-307.

Hauk O. and Pulvermüller F., Neurophysiological Distinction of Action Words in the Fronto-Central Cortex (2004) *Human Brain Mapping*, 21, 191-201.

Hinton, G. E., Dayan, P., Frey, B. J. and Neal, R. (1995) The wake-sleep algorithm for unsupervised neural networks. *Science*, 268, 1158-1161.

Keysers, C., Kohler, E., Umiltà, M.A. Fogassi, L., Nanetti, L. and Gallese, V. (2003) Audio-visual mirror neurones and action recognition. *Exp. Brain Res.*, 153, 628-636.

Kohler, E., Keysers, C., Umilta, M., Fogassi, L., Gallese, V. and Rizzolatti, G. (2002) Hearing sounds, understanding actions: Action representation in mirror neurons. *Science*, 297, 846-848.

Pulvermüller, F. (1999) Words in the brain's language. *Cognitive Neuroscience*, 22(2), 253-336.

Pulvermüller, F. (2003) *The neuroscience of language: On brain circuits of words and serial order*, Cambridge, UK: Cambridge University Press.

Pulvermüller, F. (2001) Brain reflections of words and their meaning. *Trends in Cognitive Science*, 5(12), 517-524.

Rizzolatti, G. and Arbib, M. 1998, Language within our grasp. *Trends in Neuroscience*, 21(5), 188-194.

Weber C., Wermter S. (2003) Object Localisation using laterally connected "What" and "Where" associator networks. *Proceedings of the International Conference on Artificial Neural Networks*, Istanbul, Turkey, pp. 813-820.

Weber C., Elshaw M., Zochios A., Wermter S. (2004a) A Multimodal Hierarchical Approach to Robot Learning by Imitation. *Fourth International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems*, Genoa, Italy, pp. 131-134.

Weber C., Wermter S., Zochios A. (2004b) Robot Docking with Neural Vision and Reinforcement. *Knowledge Based Systems*, 12(2-4), 165-72.

Wermter S., Austin J., Willshaw D., Elshaw M. (2001) Towards novel neuroscience-inspired computing. In Wermter S., Austin J. and Willshaw D. (Eds) Emergent Neural Computational Architectures based on Neuroscience. Springer, Heidelberg, Germany. pp. 1-19.

Wermter S., Elshaw M. (2003) Learning robot actions based on self-organising Language Memory. *Neural Networks*, 16(5-6), 691-699.

Wermter S., Weber C., Elshaw M., Panchev C., Erwin H., Pulvermüller F., (2004) Towards Multimodal Neural Robot Learning. *Robotics and Autonomous Systems Journal*, 47(2-3), 171-175.