Observational Learning Based on Models of Overlapping Pathways

Emmanouil Hourdakis and Panos Trahanias

Institute of Computer Science, Foundation for Research and Technology – Hellas (FORTH)
Science and Technology Park of Crete, GR70013, Crete, Greece
{ehourdak,trahania}@ics.forth.gr

Abstract. Brain imaging studies in macaque monkeys have recently shown that the observation and execution of specific types of grasp actions activate the same regions in the parietal, primary motor and somatosensory lobes. In the present paper we consider how learning via observation can be implemented in an artificial agent based on the above overlapping pathway of activations. We demonstrate that the circuitry developed for action execution can be activated during observation, if the agent is able to perform action association, i.e. relate its own actions with the ones of the demonstrator. In addition, by designing the model to activate the same neural codes during execution and observation, we show how the agent can accomplish observational learning of novel objects.

Keywords: Computational Brain Modeling, Observational Learning, Connectionist Model, Grasping Behaviors.

1 Introduction

Recent imaging experiments of the parieto-frontal cortex of Macaque monkeys performing grasping tasks have revealed the existence of an extended overlapping pathway of activations between action execution and action observation [1].These areas include the forelimb representations of MI and SI cortices (both activated at 50% during observation), ventral (F5) area of the premotor cortex (100% activation during observation), as well as areas of the inferior (IPL) and superior (SPL) parietal lobes (each with 50% activation during observation).

The functional roles of the above mentioned regions during grasping tasks and their activation during observation, have led neuroscientists to believe that primates are using the circuitry developed for action execution in order to simulate an observed behavior and its anticipated consequences [1]. In the current paper, we consider the activation results reported in [1] in order to develop a computational model that accomplishes observational learning of novel objects, i.e. a model that is able to associate known actions to novel objects only by observation. Similarly to their biological counterparts, this is accomplished by activating the same extended circuitry of regions used for action execution during action observation. To enable learning during observation, the model encodes an observed behavior using the neural codes stored for its execution.

T. Honkela et al. (Eds.): ICANN 2011, Part II, LNCS 6792, pp. 48–55, 2011.

The importance of the overlapping activations between action execution and action observation in primates is well established in the computational modeling community. Previous recording studies in the cerebral cortex of macaque monkeys have revealed the existence of a specific group of neurons that discharge when the primate performs or observes a goal directed behavior [2]. These overlapping activations have provided computational modelers an important foundation for implementing imitation mechanisms in artificial agents. In this context, Oztop and Kawato developed a biologically inspired model of action observation based on the process of inferring the mental states of conspecifics [3]. A more biologically faithful model has been developed by Fagg and Arbib that focused on the parietal-premotor associations during primate grasping [4], which was also extended by Arbib in the Mirror Neuron System [5] to include regions of the primary and supplementary motor cortices. Finally, a model exploiting the overlapping activations in an extended network of brain areas in order to implement the reaching component of grasping acts has been developed in our previous work using genetic algorithms to tune model parameters [6,7].

2 Computational Modeling

In the current section we provide the design and implementation details of a computational model that replicates the results described in [1] in order to accomplish observational learning of novel objects. To activate the same neural codes during execution and observation we need to track how input/output information propagates within the model. For this reason we identify certain pathways within the model, i.e. sets of regions that participate in a certain transformation of information from one type to another. In the current model we define three different pathways: *(i)* object recognition, *(ii)* proprioceptive association and *(iii)* behavior learning. The following section provides details on the implementation of the input, neuron model and network topologies, as well as the design of the specified three pathways.

2.1 Input Encoding

Our simulated agent receives three types of input: *(i)* information regarding the objects present in the scene, *(ii)* proprioceptive input that indicates the joint positions of its fingers and *(iii)* information portraying the demonstrator’s finger joint positions. All three aspects of input are encoded using population codes [8], i.e. each variable is encoded using a group of neurons, with each neuron being selective to a range of its values. For each object we encode a set of discriminative features, namely its XY axis ratio and number of corners. Behaviors (proprioception and demonstrator’s motion) are encoded based on the joint value for each finger at each time step.

2.2 Neurons and Connections

The layout of the model is shown in Fig 1. Each area is replicated using a distinct neural network, comprising of interconnected neurons that are modelled based on the Leaky Integrate and Fire model suggested by [9].

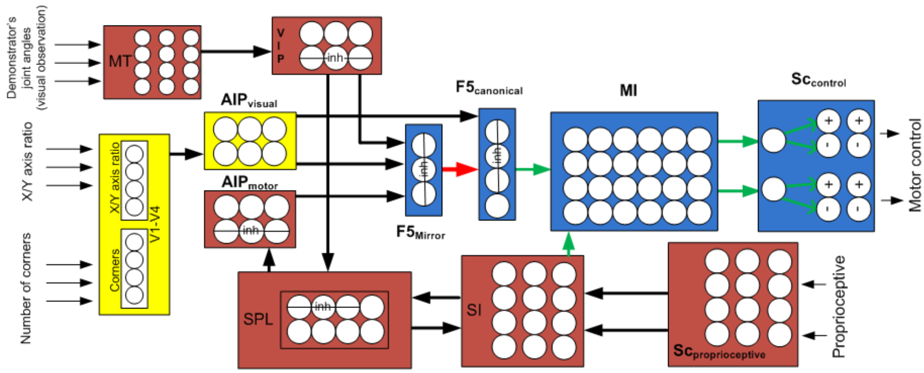

Fig. 1. Layout of the proposed model. The three pathways are marked with different colors; object recognition: yellow; proprioceptive association: red; behavior learning: blue. Different types of synapses are also marked with different colors: STDP: black, reinforcement: red, GA: green. The lines crossing the neurons in some networks (e.g. SPL) indicate the existence of lateral inhibitory connections in the respective networks

The inter and intra connectivity of each network is modelled using Spike Timing Dependent Synaptic Plasticity (STDP, [10]). Since STDP is an associative learning rule, the role of each network is to bind the information from its input to a common neural code. To form the associations between two neural networks, we use excitatory STDP synapses sparsely created from the neurons of one network towards the neurons of another. In addition, STDP is used to promote the competition between the neurons of the same networks using lateral-inhibitory connections. This is implemented as inhibitory STDP connections densely formed among the neurons of the same network (networks that employ lateral inhibitory connections are marked with a line crossing their neurons in Fig. 1, e.g. SPL network). The lateral-inhibitory connections ensure that the dominant firing neurons of a network will suppress the stimulation of less active cells.

New behaviors are taught using a derivation of the BCM learning rule for the spiking neuron model [11]. This is implemented in the connections between the $F5_{canonical}$-$F5_{mirror}$ neurons, and is described later in section 2.4.

2.3 Model Pathways

Following the design principle of pathways mentioned in the introduction, this section provides details about the implementation of the *(i)* object recognition, *(ii)* proprioceptive association and *(iii)* behavior learning pathways.

Object Recognition Pathway. The first entry point of information in the current model is through regions $V1V4_{corners}$ and $V1V4_{XYaxisRatio}$. Those two networks are responsible for encoding the properties of the demonstrated object into population code. The output from those two networks is associated in region AIP_{visual} which during training forms neuronal clusters in response to its inputs. This is accomplished by connecting neurons that are close together with excitatory links, and neurons that are distant from each other with inhibitory synapses. The $V1V4_{corners}$ and $V1V4_{XYaxisRatio}$

networks are densely connected to the AIP_{visual} region, so that when an object is viewed by the agent more than one cluster of neurons is activated. These compete during training (through their inhibitory connections), and the dominant cluster suppresses the activation of others. To ensure that diverse objects are clustered in different topological regions in the network space of AIP_{visual}, the weights of all synapses of a neuron in AIP_{visual} from the $V1V4_{corners}$ and $V1V4_{XYAxisRatio}$ networks are normalized in the [0..1] range. As a result, when a certain neuron in the input strengthens its connections with a specific cluster, it also suppresses the strength of the connections between that cluster and the remaining neurons in the input.

Proprioceptive Association Pathway. This pathway includes regions $Sc_{proprioceptive}$, SI, SPL, $AIP_{motor,}$ VIP and MT, and is assigned two tasks:*(i)* to form the neural codes that represent the motion of the fingers of our cognitive agent and *(ii)* to build a correspondence between the agent's and the demonstrator's actions (through the VIP-SPL circuit).

The process that allows the formation of the *proprioceptive codes* of the pathway involves the $Sc_{proprioceptive}$, SI, SPL, AIP_{motor} and VIP regions. $Sc_{proprioceptive}$ contains three sub-populations, each assigned to one of the index, middle and thumb fingers and projects to the SI network which also contains neuron sub-populations assigned to specific fingers. In turn each sub-population in the SI network projects to a different cluster of neurons in SPL. Finally, SPL projects to the AIP_{motor} network, which through its connections with $F5_{mirror}$ provides information on the proprioceptive codes of the agent when generating a behavior. The neural codes formed in regions SI, SPL and AIP_{motor}become progressively sharper (i.e. less distributed and concentrating more on the peaks of their tuning curves) as they are projected from the one region to the other.

In addition to the formation of the proprioceptive codes, the proprioceptive association pathway is also responsible for the *action association* function. This is accomplished using the SI-SPL-VIP circuitry. SPL, apart from SI, also accepts connections from VIP, i.e. the region encoding a distributed representation of the demonstrator's active fingers. These synapses (VIP-SPL) undergo a competition process which aims at associating the neural representations of the agent's fingers with the neural representation formed for the demonstrator's fingers. This is accomplished through the MT network which projects directly to VIP, with excitatory STDP synapses, resulting in VIP encoding a distributed representation of the demonstrator's motion. This representation is associated with the actions of the agent, using the synapses between VIP and SPL based on a competition mechanism, that normalizes the weights of all synapses (in respect to the sum of their presynaptic weights) leading to the same neuron in SPL from VIP in the [0..1] range. Because of the above process, after learning a behavior, the agent is able to identify it during observation using the same neural codes used for its execution.

Behavior Learning Pathway. The behavior learning pathway exploits information from the previous two pathways in order to observe and execute a behavior using the same networks. The entry point for the behavior learning pathway is in the $F5_{mirror}$ network which accepts connections from: (a) AIP_{motor} that provides information about the motor behavior that is being executed by the agent,(b) VIP network encoding the current demonstrated behavior, and (c) AIP_{visual} which provides information about the viewed object. To make mirror neurons respond only to transitive actions, i.e. only when an object is present in the scene, we normalize the input current sent to the $F5_{mirror}$neurons from the AIP_{motor}, AIP_{visual} and VIP regions, to appropriate ranges, and thus force the neurons in the $F5_{mirror}$ network to become active when the AIP_{visual} (object present) and at least one from the AIP_{motor} (executing) or VIP (observing) networks is active. More details on how the $F5_{mirror}$ neurons are programmed, as well as a discussion on related theories regarding the formation of mirror neurons are given in [12].

2.4 Motor Control of the Simulated Agent

The $Sc_{control}$ neural network, which controls the fingers of our simulated agent, accepts neuron signals from the MI neural network which encodes in a distributed manner the input signals received from $F5_{canonical}$. The $F5_{canonical}$ network contains three neurons, each corresponding to a specific finger in the simulated body of the agent. The $F5_{canonical}$-MI-$Sc_{control}$-SI-$Sc_{proprioceptive}$ circuitry (motor control circuitry) is evolved using genetic algorithms so that when a neuron in the $F5_{canonical}$ network is active the finger assigned to that neuron will move. The complete layout of the motor control circuitry is described more thoroughly in [12].

New behaviours are taught through the same circuit for execution and observation, using the connections between the $F5_{mirror}$-$F5_{canonical}$ neural networks. The synapses between those networks are adjusted in a series of observation/execution cycles (execution phase), using reinforcement learning. To calculate the reward signal, we assign a binary variable the value of 0 or 1 depending on whether the demonstrator's and observer's fingers are moving. The reward signal is simply the subtraction of the two variables, rescaled to the -0.5..0.5 range. The synapses of the $F5_{mirror}$-$F5_{canonical}$ neurons are updated using a derivation of the BCM learning rule for spiking neurons [11].

3 Results

The experimental setup consists of two simulated robots. The first is assigned the role of the demonstrator and the second the role of the observer. The experiments consist of two phases, execution and observational learning. During the execution phase the demonstrator exhibits two behaviors to the observer, each associated with a different object. During the observational learning phase a novel object is shown to the observer and the demonstrator exhibits one of the behaviors taught during the execution phase. The goal for the observer is to identify the perceived behavior and associate it with the novel object on the scene.

Behavior Learning. After training, given a known object the agent is able to select and execute the correct behavior without any assistance from the demonstrator (i.e. the MT and VIP networks are inactive). During the execution phase the model activated regions SPL, IPL, MI and SI at a lower rate compared to the activation level during execution (Fig 2).

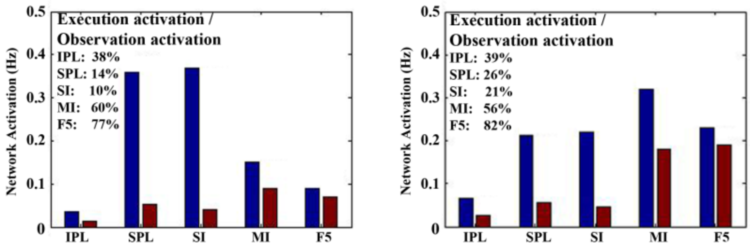

Fig. 2. The activations of the IPL, SPL, SI, MI and F5 networks, during the execution(blue bars) and sole observation cycle (red bars). The left activation plot shows the network activations during the first behavior (close middle and thumb), while the right plot shows the network activations during the second behavior (close index).

Computationally we can attribute the network activations during sole observation to the active visual input of the model, originating from regions $V1V4_{corners}$, $V1V4_{XYaxisRatio}$ and MT. During observation, the agent is still shown the object and demonstrated with the associated behavior and consequently connections between the SPL-SI, SI-MI and $SPL\text{-}AIP_{motor}$ networks activate the latter networks in each pair.

More importantly, a comparison of the active neurons during observation and execution indicates that the above mentioned networks activate the same neurons during the two phases (Fig 3).

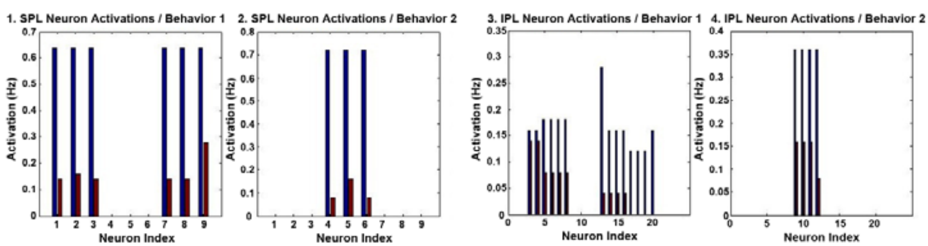

Fig. 3. Neuron activations for the execution (blue bars) and sole observation(red bars) phases for the SPL network during the first (plot 1, close middle and thumb) and second (plot 2, close index) behavior and IPL network during the first (plot 3, close middle and thumb) and second (plot3, close index) behavior

Observational Learning. During this phase the goal is to assess the ability of the agent to associate a novel object with the one of the behaviors taught during the execution phase and subsequently be able to execute this behavior whenever this object is presented.

Fig 4 illustrates the behavior executed by the agent when viewing a novel object before (left) and after (right) the observational learning phase of training.

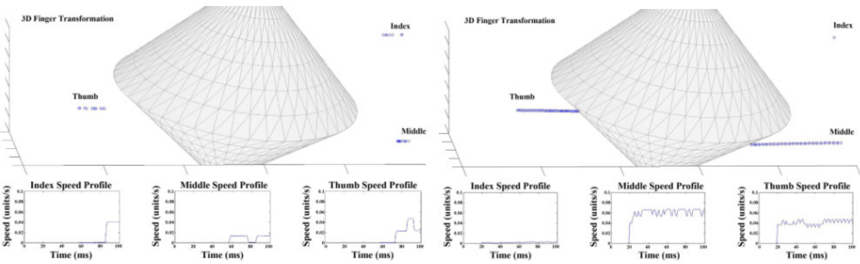

Fig. 4. Trajectories and speed profiles of the three fingers of the agent in response to a novel object before (left) and after (right) the observational learning phase. Both plots show the world coordinates of the three finger tips, along with a wireframe, transparent version of the object. (above), and the velocity profiles for the index, middle and thumb fingers during the 100ms cycle (below).

Finally, we note that after the observational learning phase, the agent is still able to execute the two behaviours learned during the execution phase. This is due to the fact that in the AIP_{visual} network, different objects activate different clusters of neurons. Thus when the novel object is presented during the observational learning stage, it activates a different cluster in the AIP_{visual} network. Consequently, learning of the new behaviour during the observational learning stage will employ a different set of synapses between the AIP_{visual} and $F5_{canonical}$ networks and will not interfere with the synapses used in previous behaviours.

4 Discussion

Recent neuroscientific experiments investigated the activation of regions in the brain cortex of primates when observing or executing a grasp behavior [1]. Results from these studies indicate that the same pathway of regions was activated during observation and execution of a behavior. This has led researchers to believe that during observation the primate is internally simulating the observed act, using its own cognitive circuits to comprehend it. The current paper, based on the neuroscientific results of [1], suggests how this extended overlapping pathway can be reproduced in a computational model of action observation/execution, and used for implementing learning during sole observation. Results from the model evaluation indicate that by shunting the motor output of an observing agent, there are several regions in the computational model that are activated at a lower firing rate compared to their activation during execution. Computationally, and maybe biologically, the lower activations that those networks exhibit are attributed to the fact that during sole observation the proprioceptive input of the agent is not available. As a result, during observation the number of active afferent projections towards each network is smaller, and therefore its activation is lower. The activations during observation are accomplished using an action correspondence circuitry, within the proprioceptive association pathway, that allows the agent to learn to associate its own joint angles with the joint angles of the demonstrator during the execution phase. The fact that this circuitry activates the same neural codes during observation and execution facilitates learning during observation.

The current work constitutes an initial attempt towards the derivation of a detailed computational model of observational learning using the overlapping pathway of activations between execution and observation. We are currently exploring how the aforementioned activations in the somatosensory cortex and parietal lobe can be used in order to extend the observational learning capabilities of an agent to learning novel actions only by observation.

References

1. Savaki, H.E., Raos, V., Evangeliou, M.N.: Observation of action: grasping with the mind's hand. Neuroimage 23(1), 193–201 (2004)
2. Rizzolatti, G., Craighero, L.: The mirror neuron system. An. Rev. of Neuroscience 27 (2004)
3. Oztop, E., Wolpert, D.M., Kawato, M.: Mental state inference using visual control parameters. Cognitive Brain Research 22, 129–151 (2005)
4. Fagg, A.H., Arbib, M.A.: Modeling parietal-premotor interactions in primate control of grasping. Neural Networks 11, 1277–1303 (1998)
5. Oztop, E., Arbib, M.A.: Schema design and implementation of the grasp-related mirror neuron system. Biological Cybernetics 87, 116–140 (2002)
6. Hourdakis, E., Maniadakis, M., Trahanias, P.: A biologically inspired approach for the control of the hand. Congress on Evolutionary Computation (2007)
7. Hourdakis, E., Trahanias, P.: A framework for automating the construction of computational models. Congress on Evolutionary Computation (2009)
8. Rieke, F., Warland, D., de Ruyter van Steveninck, R., Bialek, W.: Spikes: Exploring the Neural Code. The MIT Press, Cambridge (1999)
9. Stein, R.B.: Some models of neuronal variability. Biophysical Journal 7, 37–68 (1967)
10. Song, S., Miller, K.D., Abbott, L.F.: Competitive Hebbian learning through spike-timing-dependent synaptic plasticity. Nature Neuroscience 3, 919–926 (2000)
11. Baras, D., Meir, R.: Reinforcement learning, spike-time-dependent plasticity and the BCM rule. Neural Computation 19, 2245–2279 (2007)
12. Hourdakis, E., Savaki, H., Trahanias, P.: Computational modeling of cortical pathways involved in action execution and action observation. Journal of Neurocomputing 74, 1135–1155 (2011)