Chapter 1
Hierarchies for embodied action perception

Dimitri Ognibene, Yan Wu, Kyuhwa Lee, and Yiannis Demiris

Abstract During social interactions, humans are capable of initiating and responding to rich and complex social actions despite having incomplete world knowledge as well as physical, perceptual and computational constraints. This capability relies on action perception mechanisms, which exploit regularities in observed goal-oriented behaviours to generate robust predictions, and reduce the workload of sensing systems. To achieve this essential capability, we argue that the following three factors are fundamental. Firstly, human knowledge is frequently hierarchically structured, both in the perceptual and execution domains. Secondly, human perception is an active process driven by current task requirements and context. This is particularly important when the perceptual input is complex (e.g. human motion) and the agent has to operate under embodiment constraints. Thirdly, learning is at the heart of action perception mechanisms, underlying the agent's ability to add new behaviours to its repertoire. Based on these factors, we review multiple instantiations of a hierarchically-organised biologically-inspired framework for embodied action perception, demonstrating its flexibility in addressing the rich computational contexts of action perception and learning in robotic platforms.

1.1 Introduction

When a boxer is facing an adversary, its action perception system operates under hard embodiment constraints. It should not only recognise the adversary's movements, but also to select appropriate response actions based on its prediction of the opponent's goals. To react in time, predicting only immediate movements is insufficient. The boxer needs to infer longer sequences of adversarial actions and the underlying intentions (e.g. moving the fight to

Department of Electrical and Electronic Engineering, Imperial College London, UK

the corner), and perform strategic movements to unveil the opponent's intentions while hiding its own. This example illustrates the human capabilities to actively perceive others' actions, to predict their intentions at different levels of abstraction and to learn from the observation of others' activities. Our research is interested in equipping robots with robust action perception capabilities to allow them to participate in rich social interactions.

This chapter reports on several experiments on robotic platforms investigating the essential factors to achieve robust action perception performance. We will argue that these factors include 1) the use of hierarchical knowledge representation and processing architectures; 2) the use of active perceptual systems, where sensors actively seek for the required data to process; 3) the prediction of the sensory consequences of the most probable actions; 4) the reuse of action execution knowledge for action perception.

The remaining of this section reports on the computational principles underlying these factors along with relevant neuroscience research that supports their role in the human action perception system.

Hierarchical action representations have long been adopted in AI and robotics [66] both at the planning and execution stages for coping with large search spaces and long term decisions that characterise real-world conditions. Different hierarchical frameworks for planning have been proposed such as options or angelic semantics [64]. Such frameworks share the presence of a relationship connecting each element in a higher or more abstract level to many elements of the lower levels. However, in each framework the semantics of the relationship can be different, for example the execution of one abstract element can represent a selective, parallel, sequential or order-independent execution of the connected lower level elements.

Hierarchical representations also present advantages for learning and adaptation [34]. They may allow for more efficient inference and learning with fewer samples by exploiting partial reuse [68]. At the sensory level, hierarchical processing is extremely helpful in integrating cues from several levels of abstraction while avoiding an expensive centralised computation (the "local administration advantage"[11]). Such systems have compact representations and exhibit good generalisation between objects with similar parts [19]. In biology, the hierarchical structure of the nervous system has been seen as a general principle that enables animals to behave efficiently in complex environments [63]. Evidence of a hierarchical nervous architecture is present in many vertebrates [36, 32] and invertebrates [47]. An extensive review on the evidence of a hierarchical organisation of action representation in the human brain, including goals (short-term) and intentions (long-term) is available in [28].

Active perception directs sensors and selects data to process using additional sources of information, such as task knowledge, and predictions based on previous inputs [1, 2, 3] . By selecting only the relevant input for the current task, active perception permits parsimonious use of the sensory and computational resources [42]. Active perception can also facilitate more efficient exploration of the environment [57] and enhance the system robustness to conditions in which relevant information is hidden when observed with passive sensors [51, 65]. Apart from reducing the computational costs, active perception can, in some cases, facilitate learning [52]. The active and top-down components of human perception have been confirmed by behavioural [53, 49, 67], imaging [5] and recording [8] evidence.

Prediction of the sensory consequences of actions enables a system to recognise actions in unseen contexts by utilising learned causal relationships between actions and their sensory consequences. Predictive approaches have been extensively studied in machine learning to exploit the capability of generative models to use both unlabelled and labelled data [7]. Generative models enable learning of hierarchical representations of the task structure using fewer samples since each layer of abstraction captures domain structure information that is exploited by the other levels [35]. Internal generation of expected results enhances perception with a more robust management of noise and missing observations (sensory substitution). By enabling incremental and anticipative recognition of actions, it extends the decision time available to produce effective behavioural responses. The existence of internal representations, prediction mechanisms and simulation machinery in the brain has been one of the most discussed subjects in the cognitive science literature. Recent studies [29, 33, 40, 56] collectively show that the presence of low level representations and simulation mechanisms strongly coupled with perception and motor control, is the foundation for building higher level processes and representations. This hypothesis is supported by modern imaging and recording evidences [4, 6, 58].

The reuse of action execution knowledge for action perception allows the observing agent to recognise others' actions based on the agent's own control experience of embodied task execution and vice versa. Evidence of shared mechanisms between execution and recognition has been demonstrated in recent neuroscience studies, such as the different activation of brain areas due to different level of motor proficiency in the observed action [9], and improved performance in recognition shown by subjects with higher motor proficiency [55]. The integration of shared mechanisms and internal generation of expected results leads to what is known as the "simulation theory of action perception" [23]. According to the simulation theory, the observer recognises an action by comparing it with its internal simulations. Simulations are generated from the perspective of the performer and produced through the motor systems of the observer. By using its own motor system, the observer can directly have access to a goal-based representations detached from raw observations. In neuroscience, the discovery of mirror-neurons - motor neurons active during both execution and perception - in primates [22] and humans [37, 20, 26, 43] provides support to the simulation theory of action understanding [10, 13, 61].

In the next section, we introduce HAMMER (Hierarchical Attentive Multiple Models for Execution and Recognition [15, 41]) as a prototype of hi-

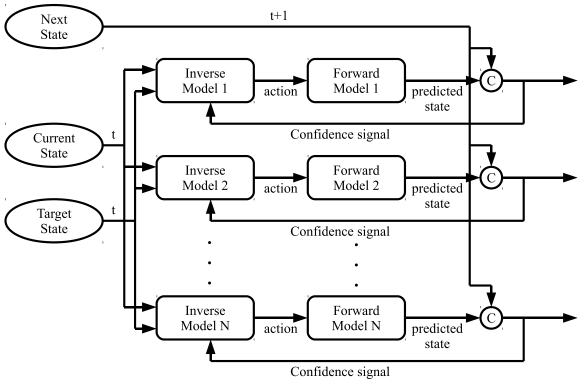

Fig. 1.1: The core of the HAMMER architecture consists of a distributed network of inverse and forward models that compete to predictively explain the ongoing demonstration

erarchical action recognition architectures which possess the aforementioned characteristics.

1.2 Hierarchical Attentive Multiple Models for Execution and Recognition (HAMMER)

The HAMMER architecture is a framework based on simulation theory, designed to empower robots with capabilities to understand and imitate human actions based on the four factors described in the previous section. This framework has been implemented in real-dynamics robot simulators [17, 14] and real robotic platforms [15, 41, 18]. Open source versions of the architecture have been freely released [60] with support for the NAO and iCub humanoids.

Fig. 1.1 shows the schematics of the HAMMER architecture. The basic building block of HAMMER consists of an inverse-forward model pair. The inverse model generates action commands from a set of input states aiming to advance the robotic agent towards a goal. This goal can be implicitly or explicitly specified in the model. The forward model provides an estimate of the upcoming states given the action commands and current state. Predictions of upcoming states in execution mode can be used to overcome delays, to handle input noise and as sensory substitution. When it is used to recognise actions, such predictions from each model are compared against the demonstrator's actual states.

For each inverse/forward model pair, the prediction of the demonstrator's next state is evaluated against the ground truth to provide an error signal. The error signal accumulated over time is used to compute the confidence value of the model pair, which is an indicator of how closely the demonstrated action matches the model. During execution, the confidence value is used to detect the actual context and/or hidden states. This enables the switching from one model to another according to the confidence indicator of the fittest model. The confidence signal can also be used as credit assignment for module training [30, 31]. The architecture recognises actions of others by comparing the observed movements with the different expected results produced by running its own motor models (”putting the observer in the shoes of the demonstrator”) in parallel while inhibiting the models from sending their generated commands to motor systems.

The HAMMER architecture incorporates a top-down allocation of sensory and computational resources. For action-execution, HAMMER can rely on the simple principle of “attention for action” and seek the information required by the current task. On the other hand, action-recognition poses new problems for the attention system since the observer does not know in advance what the observed task is. HAMMER maps simulations to attentional needs using the following principle: during the demonstration of actions, the information requested by the attention system of the observer are those needed to generate the internal simulated actions [16]. For example, the inverse model for executing an arm movement will request the state of the corresponding arm of the performer when it is used in perception mode. This principle is compatible with the pre-motor theory of attention in humans which states that the preparation or simulation of action enhances the perception of related stimuli [21].

Action imitation can be achieved by integrating recognition and execution in HAMMER. The system starts in recognition mode, and when a model with the highest confidence reaches a certain threshold, the command inhibition is deactivated to allow the model to reproduce the observed action if required. If no confidence value reaches the threshold within a time limit, a new motor model is learned to represent the postural and posture-object configurations of the observed action.

The HAMMER models can be connected in arbitrarily complex configurations. Their overt execution does not need to be mutually exclusive, i.e. models managing different joints[14] can be executed overtly in parallel. This arrangement has been extended to hierarchical structure as shown in Fig. 1.2 [15]: primitive models are combined to form higher, more complex sequences, with the eventual goal of achieving increasingly more abstract inverse models [41].

Using the underlying principles in HAMMER, Demiris [17] derived a set of testable predictions for the behaviour of biological systems. A key prediction states that mirror neurons in monkeys would not fire (or fire less) when the demonstrated movement was performed at speeds unattainable by the observer. Experiments subsequently reported in [24] showed that the amplitude of the motor evoked potentials (MEP) induced by transcranial magnetic stimulation (TMS) in humans observing a reaching-grasping action was modulated by the kinematics of the observed figure aperture.

Modelling human grasping action and its perception with HAMMER reproduced several interesting neuroscience observations: a) the computational grasping model reproduced some of the characteristics of human grasping including [39] an overshoot in the grip aperture at approximately 70% of the movement time[62]; b) the TMS-based results on the response of mirror neurons to different action timings and coordination properties reported in [25].

In the next section, we will present experiments demonstrating how the hierarchical generative approach to action perception may be used to cope with embodiment constraints in action perception.

1.3 Hierarchical action perception and abstraction

This section describes a HAMMER implementation [41] on a ActivMedia Peoplebot, and how its hierarchical representation is used to cope with the "correspondence problem" [50], the problem incurred by an imitator during imitation of actions produced by a performer with different embodiment.

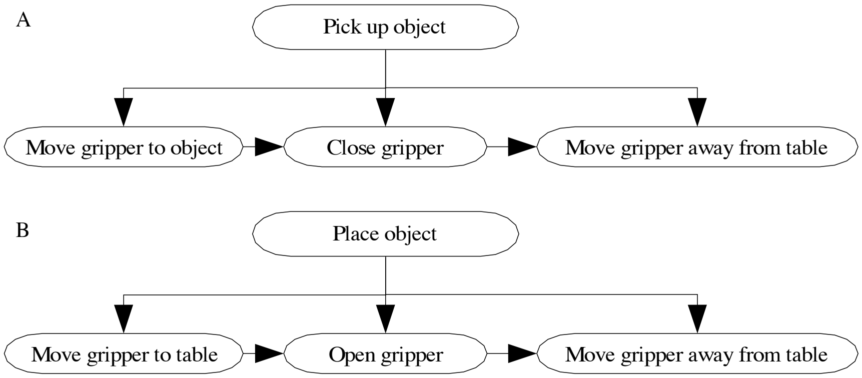

Fig. 1.2: Example arrangements of primitive inverse models into more complex inverse models: (A) Pick up object (B) Place object.

In this implementation, two kinds of models were used:

- Primitive models constructed as a simple motor program (the inverse model) tightly coupled with a hand-coded forward model;

- Higher level models implemented as graphs: to create a hierarchy, graphs are handled recursively with a graph node being either a graph itself or a primitive (e.g. Fig. 1.2. Models connected serially are executed in serial manner, and those connected in parallel are executed in parallel.

A goal state is associated with each inverse model. During execution, a graph will execute each of its constituent nodes in turn until completion. At this point, the node will reset its confidence and the graph will continue execution of the subsequent node. For recognition, the sequential execution constraint is relaxed. Inverse models at all levels of the hierarchy are executed in parallel regardless of the recognition stage. At each step, every model signals its performance to all of his parent-models by propagating its confidence value. Each high-level model computes its confidence based on (and normalised against) the first model in the child model sequence whose confidence value has reached a certain threshold.

In these experiments, the Peoplebot had to learn to recognise and transport objects between two tables following a human demonstration of this task, with the two agents, human and Peoplebot having very different embodiments. State information was extracted using visual markers. The states consisted of the positions of the hand, tables and objects, the relative distances among them, their derivatives, and a boolean flag indicating “object in gripper”. The high-level abstract inverse model was constructed by learning primitive inverse models from human demonstration. 23 primitive inverse models were available to the architecture, while 20 repeated demonstrations performed at natural speeds and trajectories were conducted in the experiment [41]. Note that not all actions in Fig. 1.3 are absolutely necessary for the human demonstrator to achieve the final goal. Moreover for the robot, it is not easy to perceive some of them.

The confidence evolution plot of recognition with the abstraction mechanism (Figure 1.4.A), shows how the high-level inverse model recognises the other inverse models as being salient and incorporates them to achieve the highest confidence overall. Figure 1.4.B shows the high-level inverse model failing to achieve high confidence; without the abstraction mechanism, the motor pattern of the high-level inverse model is so fundamentally different to that performed by the demonstrator, that it fails to match.

The reported experiments (full details in [41]) show that an agent endowed with hierarchical action representations with abstraction capabilities can understand and imitate a composite action and its final goal even if it cannot directly execute it or does not know each single composing action. Moreover the actions which are recognised and can be executed may not contributed by themselves to the achievement of the goal, or may also miss some precondition to be executable. With a hierarchical representation the agent can use the recognised actions as clues for the rehearsal of a higher level action which will put the recognised lower level actions in the proper context for execution.

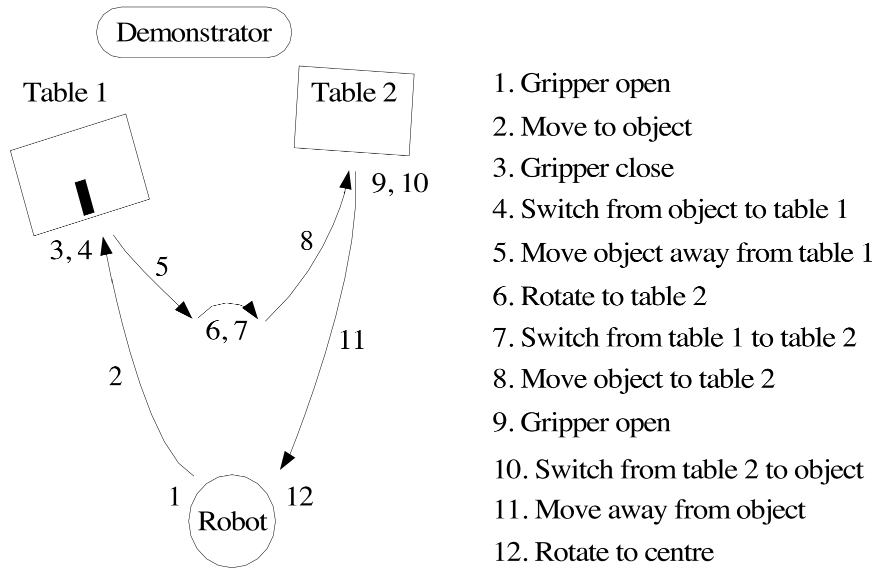
Demonstrator
Table 1
Table 2
3, 4
5
6, 7
8
9, 10
11
2
1
12
Robot
1. Gripper open
2. Move to object
3. Gripper close
4. Switch from object to table 1
5. Move object away from table 1
6. Rotate to table 2
7. Switch from table 1 to table 2
8. Move object to table 2
9. Gripper open
10. Switch from table 2 to object
11. Move away from object
12. Rotate to centre

Fig. 1.3: The sequence of primitive inverse models that constitute the abstract inverse model for moving an object from one table to another

1.4 Acquiring Hierarchical representations for integrated social and autonomous learning

The previous section demonstrated how hierarchically organised inverse and forward models were able to observe and imitate a sequence of actions. A fundamental question underlying this research is where do these models come from? In the following sections we will describe how these hierarchies can be learned, starting from learning primitive forward and inverse models, to learning action descriptions using stochastic context free grammars.

1.4.1 Learning primitive models through motor babbling

First we are faced with the problem of how to learn the models at the lowest part of a hierarchy, i.e. primitive inverse and forward models. We have developed a system that learns primitive forward and inverse models through motor babbling [12], a learning method that associates randomly executed motor commands and their effects on environment [27]. The system learns a

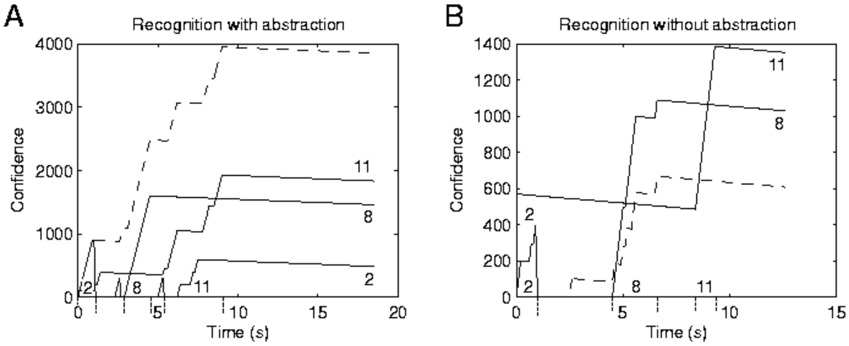

Fig. 1.4: Graphs of confidence over time for four inverse models in recognition mode. The dashed series is the high-level inverse model for moving an object between two tables. The other three inverse models are numbered as in Figure 1.3. Graph (A) shows the confidences of these inverse models recognised using the abstraction mechanism. The high-level inverse model, represented by series 1, incorporates the other inverse models as salient features of the demonstration, achieves the highest overall confidence and thus is successfully recognised. Graph (B) shows the confidences of the inverse models when recognition is performed without the abstraction mechanism. The high-level inverse model fails to incorporate the other inverse models, achieves a low overall confidence, and thus is not recognised.

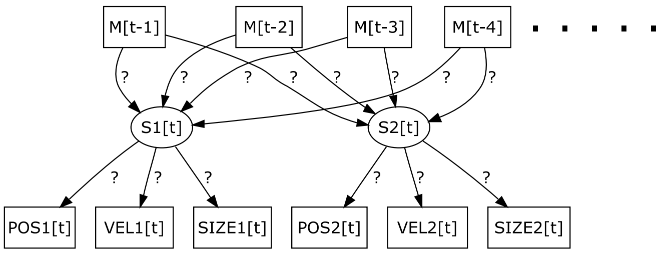

Fig. 1.5: The Bayesian network for the ActivMedia Peoplebot's gripper forward model. The robot has to learn the mappings (indicated by question marks) between sequences of motor commands (top row) and resulting states of the gripper (perceived through a camera, bottom row), despite the inherent sensory delays of real-robotic systems.

forward model implemented with a Bayesian network [54] as shown in Fig. 1.5 without prior knowledge of its motor system or the external environment. The forward model represents a probability distribution of the states of the robot and other objects in the environment after *d* time-steps from the previous motor commands. The network structure is learned online by performing a search through the set of possible structures (with different delays in motor commands and different observation nodes for each possible object) and choosing the one which maximises the log-likelihood of the observed experiment results. Expectation maximisation (EM) is used with the inference stage performed with the junction-tree algorithm [54]. Moving objects in the scene are automatically detected and tracked by clustering the low-level image features of the visual input [48]. An inverse model is derived by the forward model by exploiting the Bayesian representation (see Fig. 1.6).

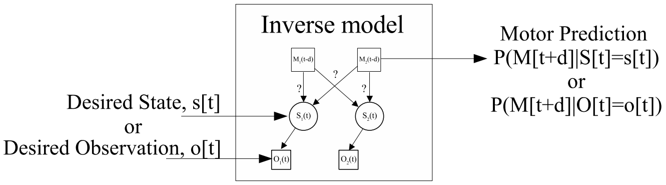

Fig. 1.6: Using the learnt Bayesian network as an inverse model. Evidence is supplied to the observations or the state, and the task is to infer the probability distribution of motor commands.

The learned HAMMER inverse-forward model enables the robot to imitate simple human hand movements by replacing the robot's observations of its own movements by those of a human demonstrator shown in Fig. 1.7.

1.4.2 Learning action sequences by demonstration

Having learned the primitive inverse and forward models, imitation can be used to learn sequences of these models in order to complete more complex tasks. An early study [15] used two ActiveMedia Peoplebots facing each other. One robot executed a sequence of actions while the other observed and learned this sequence of actions. The observer initially is equipped with basic action primitives to control its gripper (open, close, rise and lower) but does not possess any high level action model. A typical experiment consisted of the robot demonstrator executing a random sequence of basic action primitives, with the imitator robot observing, storing the observed sequence of inverse models in working memory and subsequently replicating. These early experiments demonstrated how sequence learning can occur, but attempted

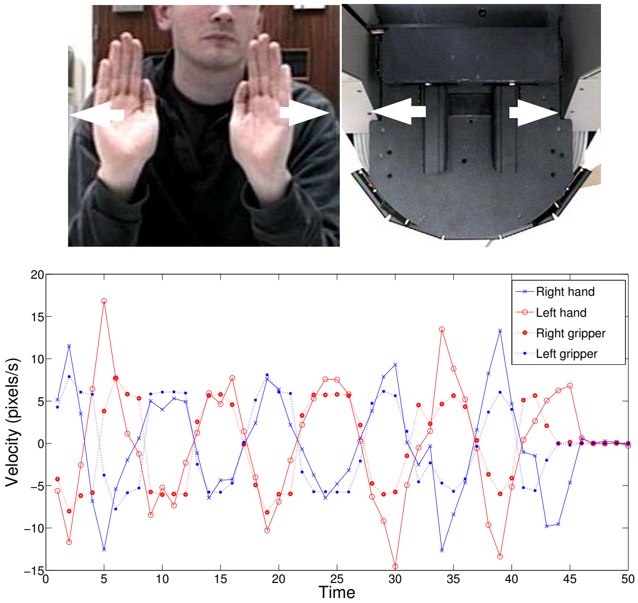

Fig. 1.7: Imitation using a single inverse model. The top images are corresponding frames from the demonstration (left) and imitating (right) sequences. The graphs show the trajectory of the demonstrating hands, and the corresponding imitating trajectory of the grippers.

no further processing in the models in working memory other than simply storage. For the purposes of this chapter, an interesting aspect is how low level sequences can be generalised to new situations and how we can infer action hierarchies from these observations, in order to utilise their benefits as advocated in this chapter. In order to do so, we first turn our attention to how observations can lead to generalizable primitives; the benefit of the next algorithm (OSILA) is that it can generalise from a single demonstration, but it lacks certain types of expressiveness (for example, it cannot readily learn to represent recursion). The final section will describe an algorithm that enables the learning of more expressive abstract representations involving probabilistic grammars.

1.4.3 Learning generalisable action templates from single observation

Humans can learn new tasks from a single demonstration; an one-shot imitation learning algorithm (OSILA) was proposed in [69] to tackle the problem of learning (through a single observation) action primitives that can be adapted to new contexts. It stores observed actions as human-readable movement templates and re-adapts them according to the constraints of the new contexts. OSILA (figure 1.9) conjectures a spatial relationship between the template and the applied environment. Inference is used to locate relevant invariant landmarks or invariant control points (ICPs) in both contexts. Subsequently, it uses a minimum distortion function based on Thin Plate Splines (TPS) warping to define a mapping between the two spaces. This allows the definition of a set of candidate waypoints in the applied space extracted from the observed action. The adaptation mechanism is based on the use of a warping energy measure that reduces the deformation of the performed action with different environmental structures.

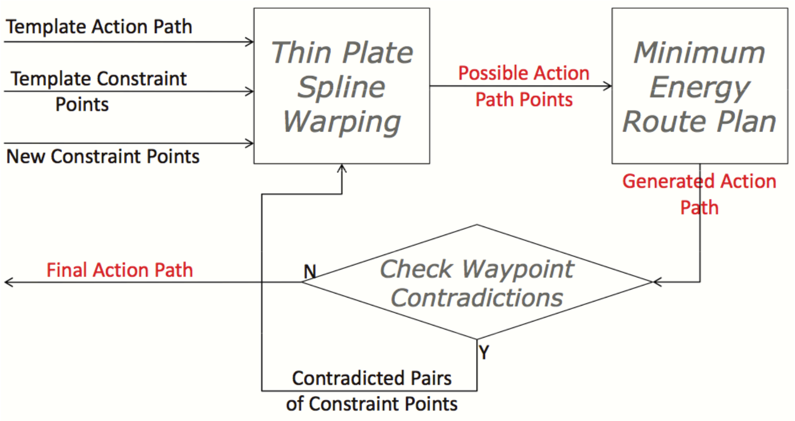
Template Action Path
Template Constraint Points
New Constraint Points
Thin Plate Spline Warping
Possible Action Path Points
Minimum Energy Route Plan
Generated Action Path
Final Action Path
N
Check Waypoint Contradictions
Y
Contradicted Pairs of Constraint Points

Fig. 1.8: Adapting a OSILA-learned template of an inverse model to a new context.

An inverse model in OSILA reproduces the action using the visual state information of the new context and adapting the previously-observed action template, setting a threshold for tolerable warping energy when matching available hypotheses (templates). This way the algorithm has a principled way for selecting when to learn new primitives or use combinations of the already learnt primitives. Experiments conducted in [69] show that the trajectories stably generated by OSILA resemble the paths produced by humans under similar circumstances. Experimental scenarios using the icub humanoid robot included the game of tic-tac-toe (figure **??**), where the robot learned (through one-shot demonstration) different templates of movements (for example making an O in one position of the board, that it could subsequently generalise to other positions of the board).

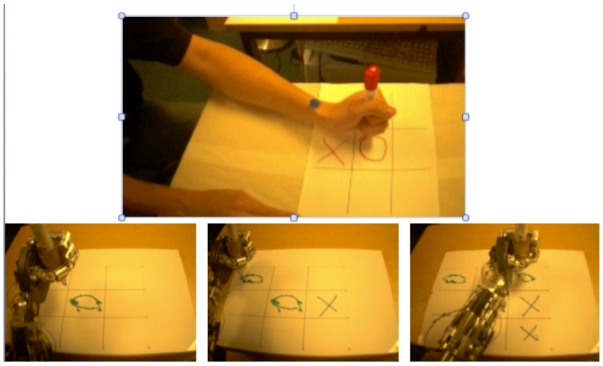

Fig. 1.9: Using OSILA to adapt learned templates to new board positions using the icub humanoid in a tic-tac-toe game.

1.4.4 Learning action hierarchies using probabilistic grammars

The previously described learning mechanisms does not explicitly tackle the problem of learning a hierarchical structure which is particularly important both for generalisation and for keeping resource requirements bounded. They also do not explicitly focus on the advantages of hierarchical representations to boost noise robustness when perceiving complex and long action sequences.

We have studied these issues [46] using a generative approach to learn by observation task representations in the form of Stochastic Context Free Grammars (SCFG). SCFG-based representations allow to express complex hierarchies of actions in a compact and efficient manner. During recognition, they take into account the uncertainties of actions common in real-world settings in a probabilistic manner which makes this framework highly scalable. SCFGs are also capable of recognising arbitrary lengths of action sequences composed of finite set of action symbols using recursive expressions. SCFGs essentially extend the Context-Free Grammars (CFG) framework [59] by associating a probability to each production rule, which enables all parse trees to be assigned with probability values based on the production rules used. In [46], the terminal symbols of the grammar are generated by primitive action detectors,while non-terminal symbols can be thought of as sequences of primitive actions. In an example scenario, “take out all the objects in a bag and give them to a human”, the robot must repeatedly perform high-level actions such as “take out objects” and “give them to a human”, which are composed of lower-level actions such as “locate”,“approach”,“grasp”,“move” and “release”. Researchers have argued that understanding everyday human behaviour requires this representational power [59, 38] and the recognition using a direct pattern-matching approach against all possible behaviours is not a computationally efficient approach.As described below, the HAMMER-like SCFG restrains the set of candidate behaviours through the use of higher level grammatical production rules and predicts future observations using the available information online.

Using SCFGs to recognise an action involves selecting a parse tree (that is a hierarchical structure using the connections represented in the grammar) that best explains the observation, i.e. the parsed action sequence with the highest probability. In [45], it was applied in a real-world scenario where an icub humanoid robot uses task-independent action templates in the form of SCFGs to recognise human behaviours .

The algorithm proposed in [46] exploits the confidence values computed by the primitive action detectors during both learning and recognition to deal with ambiguities inherent in perception. During parsing, the algorithm computes probability distributions of the possible parse trees based on (noisy) symbols observed so far, and updates the distribution after each new input. The probability distribution over the parse tree permits to derive the expectation of the future inputs in a compact way.

During learning, the algorithm starts with a naive grammar containing all input sequences. It then builds grammar hypotheses using “Substitute” and “Merge” operators to find the grammar with the minimum description length [44] that maximises the posterior probability. The algorithm actively searches for frequently occurring sub-sequences of actions to infer the hierarchical structure which allows more compact and generalised representations while offering robustness to observations containing errors. Thus, erroneous sequences are assigned lower probability values than frequently occurring sequences. Furthermore, the confidence values of the primitive action detectors are considered to emphasise symbols with less ambiguity. The Substitute operator replaces a partial sequence of symbols in the right-hand side of the rule and groups them into a new symbol, thus building a structural hierarchy. The Merge operator is applied on two symbols in such a way that both symbols are replaced with a single symbol. This process turns the current representation into a more generalised, compact representation. Both operators are applied until the grammar with the minimum description length is found.

The algorithm was tested on artificial data sets and on real videos of humans solving the tower of Hanoi game (figure 1.10. In the artificial data set, the tested data was generated by a grammar model ($a^n cb^n$) with various levels of noise by substituting and inserting terminal symbols in the input strings with a random symbol. Each symbol was also assigned with varying confidence values. As compared to other state-of-the-art algorithms, the algorithm proved to be able to produce grammars that were more compact and more robust to noise and execution errors. In the real-world experiment, videos of humans solving the Towers of Hanoi puzzle were used as input (figure 1.10). The primitive action detectors were designed using HMMs that can recognise different actions of moving a disk between two poles. In this experiment, the algorithm showed to be able to acquire good representation of the task despite the fact that the observations contained several errors.

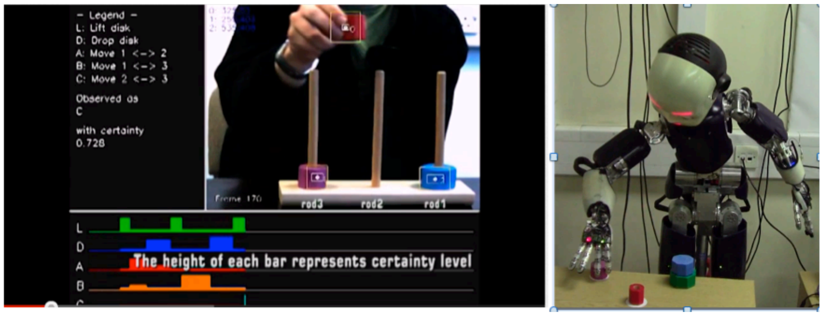

Fig. 1.10: Observing and learning new hierarchical behaviours in the Towers of Hanoi game, using stochastic context free grammars.

These results show an interesting aspect of hierarchical action representations that is advantageous for learning by observation: learned hierarchical structures can effectively deal with observation ambiguities. Moreover, the experiments demonstrate that the chosen hierarchical approach is able to learn a generalised task representation that is able to recognise unforeseen, more complex actions with the same task type, e.g. playing the Towers of Hanoi puzzle with a larger number of disks than those demonstrated.

1.5 Conclusions

In this chapter, we argued for the computational benefits of hierarchies for embodied action perception. We presented the HAMMER architecture that we use as a framework to empower our robots with capabilities to understand, learn from and imitate human action.

The experiments reported in this chapter described the essential roles played by hierarchical representations implemented in/with HAMMER in action perception and social learning. We argued that:

- hierarchical representations allow internal representations of a task to match external demonstrations of the task when performed by others, even when the embodiment characteristics of the demonstrator and the imitator are different. In general, hierarchical representations allow the demonstrated and predicted information to be compared at the appropriate of level of abstraction, providing flexibility and robustness to execution variability.

- hierarchical representations can be learned through a combination of low-level inverse/forward model learning using techniques such as motor babbling and low-level imitation, while more abstract hierarchical representations can be constructed using more grammatical tools such as stochastic context free grammars.

While the reported experiments demonstrated several successful applications of HAMMER in human robot interactions, natural social interactions pose far more complex challenges. Action perception and imitation capabilities are crucial for enabling robots to behave in unstructured social environment and for natural interactions in dynamic contexts. In these contexts, the information is far richer and more complex; and the set of behaviours that the robot will need to recognise and discriminate will be far more wide-ranging. The role played by hierarchical representations will grow in importance as our robots increasingly tackle more challenging social interaction scenarios.

References

1. John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. Active vision. *International Journal of Computer Vision*, 1(4):333–356, january 1988.
2. R. Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):966–1005, 1988.
3. D.H. Ballard. Animate vision. *AI*, 48:57–86, 1991.
4. M. Bar and I. Biederman. Localizing the cortical region mediating visual awareness of object identity. *Proc Natl Acad Sci U S A*, 96(4):1790–1793, Feb 1999.
5. M. Bar, K. S. Kassam, A. S. Ghuman, J. Boshyan, A. M. Schmid, A. M. Schmidt, A. M. Dale, M. S. Hmlinen, K. Marinkovic, D. L. Schacter, B. R. Rosen, and E. Halgren. Top-down facilitation of visual recognition. *Proc Natl Acad Sci U S A*, 103(2):449–454, Jan 2006.
6. Moshe Bar. The proactive brain: using analogies and associations to generate predictions. *Trends Cogn Sci*, 11(7):280–289, Jul 2007.
7. Christopher M. Bishop and Julia Lasserre. Generative or discriminative? getting the best of both worlds. *BAYESIAN STATISTICS 8, pp. 324.*, 8:3–24, 2007.
8. Timothy J Buschman and Earl K Miller. Top-down versus bottom-up control of attention in the prefrontal and posterior parietal cortices. *Science*, 315(5820):1860–1862, Mar 2007.
9. B. Calvo-Merino, D.E. Glaser, J. Grzes, R.E. Passingham, and P. Haggard. Action observation and acquired motor skills: An fmri study with expert dancers. *Cerebral Cortex*, 15(8):1243–1249, August 2005.
10. Raymond H Cuijpers, Hein T van Schie, Mathieu Koppen, Wolfram Erlhagen, and Harold Bekkering. Goals and means in action observation: a computational approach. *Neural Netw*, 19(3):311–322, Apr 2006.
11. Richard Dawkins. *Growing points in ethology*, chapter I. Hierarchical organisation: A candidate principle for ethology. Growing points in ethology., pages 7–54. Oxford, 1976.

12. Anthony M. Dearden and Yiannis Demiris. Learning forward models for robots. In *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30-August 5, 2005*, pages 1440–1445, 2005.
13. Y. Demiris. Prediction of intent in robotics and multi-agent systems. *Cognitive Processing*, 8(3):151–158, 2007.
14. Y. Demiris and G. M. Hayes. Imitation as a dual-route process featuring predictive and learning components: a biologically-plausible computational model. In *Imitation in Animals and Artifacts*. MIT Press, 2002.
15. Y. Demiris and M. Johnson. Distributed, predictive perception of actions: a biologically inspired robotics architecture for imitation and learning. *Connection Science*, 15(4):231–243, 2003.
16. Y. Demiris and B. Khadhouri. Hierarchical attentive multiple models for execution and recognition of actions. *Robotics and Autonomous Systems*, 54(5):361–369, 2006.
17. Yiannis Demiris. *Movement Imitation Mechanisms in Robots and Humans*. PhD thesis, Department of Artificial Intelligence, University of Edinburgh, May 1999.
18. Yiannis Demiris and Bassam Khadhouri. Content-based control of goal-directed attention during human action perception. *Interaction Studies*, 9(2):353–376, 2008.
19. B. Epshtein and S. Ullman. Semantic hierarchies for recognizing objects and parts. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.
20. L. Fadiga, L. Fogassi, G. Pavesi, and G. Rizzolatti. Motor facilitation during action observation: a magnetic stimulation study. *Journal of neurophysiology*, 73(6):2608–2611, 1995.
21. S. Fagioli, B. Hommel, and R.I. Schubotz. Intentional control of attention: Action planning primes action-related stimulus dimensions. *Psychological research*, 71(1):22–29, 2007.
22. V. Gallese, L. Fadiga, L. Fogassi, and G. Rizzolatti. Action recognition in the premotor cortex. *Brain*, 119(2):593, 1996.
23. Vittorio Gallese and Alvin Goldman. Mirror neurons and the simulation theory of mind-reading. *Trends in Cognitive Sciences*, 2(12):493 – 501, 1998.
24. M. Gangitano, F.M. Mottaghy, and A. Pascual-Leone. Phase-specific modulation of cortical motor output during movement observation. *Neuroreport*, 12(7):1489, 2001.
25. M. Gangitano, F.M. Mottaghy, and A. Pascual-Leone. Modulation of premotor mirror neuron activity during observation of unpredictable grasping movements. *European Journal of Neuroscience*, 20(8):2193–2202, 2004.
26. Valeria Gazzola and Christian Keysers. The observation and execution of actions share motor and somatosensory voxels in all tested subjects: Single-subject analyses of unsmoothed fmri data. *Cerebral Cortex*, 19(6):1239–1255, 2009.
27. A. Gopnik and A.N. Meltzoff. *Words, Thoughts, and Theories.* MIT Press, 55 Hayward Street, Cambridge, MA 02142., 1997.
28. S.T. Grafton et al. Evidence for a distributed hierarchy of action representation in the brain. *Human movement science*, 26(4):590–616, 2007.
29. Rick Grush. The emulation theory of representation: motor control, imagery, and perception. *Behav Brain Sci*, 27(3):377–96; discussion 396–442, Jun 2004.
30. M. Haruno, D.M. Wolpert, and M. Kawato. Mosaic model for sensorimotor learning and control. *Neural Computation*, 13(10):2201–2220, 2001.
31. M. Haruno, D.M. Wolpert, and M. Kawato. Hierarchical mosaic for movement generation. 1250:575–590, 2003.
32. W. R Hess. *The functional organization of the diencephalon.* Grune & Stratton, 1957.
33. G. Hesslow. Conscious thought as simulation of behaviour and perception. *Trends in cognitive sciences*, 6(6):242–247, 2002.
34. G.E. Hinton. Learning to represent visual input. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 365(1537):177, 2010.

35. GEOFFREY E. HINTON and ZOUBIN GHAHRAMANI. Generative models for discovering sparse distributed representations. *Phil. Trans. R. Soc. Lond. B*, 352:1177–1190, 1997.
36. C.F. Honeycutt and T.R. Nichols. The decerebrate cat generates the essential features of the force constraint strategy. *Journal of neurophysiology*, 103(6):3266, 2010.
37. Marco Iacoboni, Roger P. Woods, Marcel Brass, Harold Bekkering, John C. Mazziotta, and Giacomo Rizzolatti. Cortical mechanisms of human imitation. *Science*, 286(5449):2526–2528, 1999.
38. Y.A. Ivanov and A.F. Bobick. Recognition of visual activities and interactions by stochastic parsing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):852–872, 2000.
39. M. Jeannerod. Intersegmental coordination during reaching at natural visual objects. volume 9, pages 153–168. Lawrence Erlbaum Associates, Inc. Hillsdale, NJ, 1981.
40. M. Jeannerod. The representing brain: Neural correlates of motor intention and imagery. *Behavioral and Brain sciences*, 17(02):187–202, 1994.
41. M. Johnson and Y. Demiris. Abstraction in recognition to solve the correspondence problem for robot imitation. In *Proceedings of TAROS*, pages 63–70. Citeseer, 2004.
42. T. Kato and D. Floreano. An evolutionary active-vision system. In *Proceedings of the 2001 Congress on Evolutionary Computation.*, volume 1, pages 107–114vol.1, 27-30 May 2001.
43. Christian Keysers and Valeria Gazzola. Social neuroscience: Mirror neurons recorded in humans. *Current biology*, 20:353–354, 2010.
44. P. Langley and S. Stromsten. Learning context-free grammars with a simplicity bias. In *Machine Learning: ECML 2000: 11th European Conference on Machine Learning, Barcelona, Catalonia, Spain, May/June 2000. Proceedings*, pages 321–338. Springer, 2000.
45. K. Lee and Y. Demiris. Towards incremental learning of task-dependent action sequences using probabilistic parsing. In *IEEE First Joint International Conference on Development and Learning and on Epigenetic Robotics (ICDL-EPIROB 2011)*, Frankfurt am Main,, August 2011.
46. K. Lee, T. K. Kim, and Y. Demiris. Learning reusable task representations using hierarchical activity grammars with uncertainties. In *IEEE International Conference on Robotics and Automation (IEEE ICRA 2012)*, St. Paul, Minnesota, USA, May 2012.
47. Eckehard Liske. The hierarchical organiztion of mantid behaviours. In F. R. Prete, , Harrington Wells, Patrick H. Wells, and Lawrence E. Hurd, editors, *The praying mantids*. Johns Hopkins University Press, 1999.
48. Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of Imaging Understanding Workshop*, pages 121–130, 1981.
49. George L. Malcolm and John M. Henderson. Combining top-down processes to guide eye movements during real-world scene search. *Journal of Vision*, 10:1–11, 2010.
50. C.L. Nehaniv and K. Dautenhahn. *The Correspondence Problems*, chapter 2, pages 41–61. The MIT Press, 2002.
51. D. Ognibene, N. Catenacci Volpi, and G. Pezzulo. Learning to grasp information with your own hands. In *Proceedings of 12th Conference Towards Autonomous Robotics Systems (TAROS 2011)*, 2011.
52. Dimitri Ognibene, G. Pezzulo, and G. Baldassarre. How can bottom-up information shape learning of top-down attention control skills? In *Proceedings of 9th International Conference on Development and Learning*, 2010.
53. J. K. O'Regan and A. No. A sensorimotor account of vision and visual consciousness. *Behav Brain Sci*, 24(5):939–73; discussion 973–1031, Oct 2001.
54. J. Pearl. *Causality: models, reasoning and inference*. Cambridge Univ Press, 2000.

55. G. Pezzulo, L. Barca, A.L. Bocconi, and A.M. Borghi. When affordances climb into your mind: Advantages of motor simulation in a memory task performed by novice and expert rock climbers. *Brain and Cognition*, 73(1):68–73, 2010.
56. Giovanni Pezzulo, Lawrence W Barsalou, Angelo Cangelosi, Martin H Fischer, Michael Spivey, and Ken McRae. The mechanics of embodiment: A dialogue on embodiment and computational modeling. *Frontiers in Psychology*, 2(00005), 2011.
57. Rajesh P.N. Rao and DanaH Ballard. An active vision architecture based on iconic representations. *Artificial intelligence*, 78(1-2):461–505, October 1995.
58. Leila Reddy and Nancy Kanwisher. Coding of visual objects in the ventral stream. *Current Opinion in Neurobiology*, 16(4):408 – 414, 2006. ¡ce:title¿Sensory systems¡/ce:title¿.
59. MS Ryoo and JK Aggarwal. Recognition of composite human activities through context-free grammar based representation. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1709–1718. IEEE, 2006.
60. Miguel Sarabia, Raquel Ros, and Yiannis Demiris. Towards an open-source social middleware for humanoid robots. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robotics*, 2011.
61. K. Shanton and A. Goldman. Simulation theory. *Wiley Interdisciplinary Reviews: Cognitive Science*, 1(4):527–538, 2010.
62. Gavin Simmons and Yiannis Demiris. Object grasping using the minimum variance model. *Biol Cybern*, 94(5):393–407, May 2006.
63. Herbert A. Simon. The architecture of complexity. *Proceedings of the American Philosophical Society*, 106(6):467–482, Dec 1962.
64. Richard S. Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 211:112–181, 1999.
65. Mototaka Suzuki and Dario Floreano. Evolutionary active vision toward three dimensional landmark-navigation. In *From Animals to Animats 9*, 2006.
66. A. Tate. Generating project networks. In *, Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-77),*, page 888893, Cambridge, MA, USA,, 1977. , Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-77), pp. 888893, Cambridge, MA, USA, Morgan Kaufmann.
67. Benjamin W. Tatler, Mary M. Hayhoe, Michael F. Land, and Dana Ballard. Eye guidance in natural vision: Reinterpreting salience. *Journal of Vision*, 11(5):1–23, 2011.
68. G. Theocharous, K. Murphy, and L.P. Kaelbling. Representing hierarchical pomdps as dbns for multi-scale robot localization. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 1, pages 1045–1051. IEEE, 2004.
69. Yan Wu and Yiannis Demiris. Towards one shot learning by imitation for humanoid robots. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 2889–2894. IEEE, 2010.