A Framework Inspired by Cognitive Memory to Learn Planning Domains From Demonstrations

Luca Buoncompagni and Fulvio Mastrogiovanni

All the authors are affiliated with the Department of Informatics, Bioengineering, Robotics and Systems Engineering, University of Genoa, Via Opera Pia 13, 16145, Genoa, Italy.
Corresponding authors' e-mails: luca.buoncompagni@edu.unige.it.

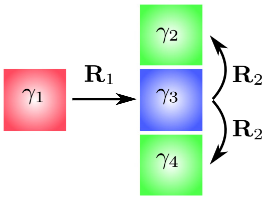
Abstract We introduce a framework for acquiring structured knowledge from human-lead demonstrations and generate task planning domains for robots. It is based on a novel algorithm that builds symbolic models of environmental states as structured memory items, which are stored and retrieved after reasoning processes. The paper addresses the formalisation of memory items and its management over time through cognitive-like functions, i.e., encoding, storing, retrieving, consolidating and forgetting. Based on the two simple scenarios, we present preliminary results and we discuss the benefits and limitations of our approach.

Keywords: Knowledge Acquisition, Structured Concepts Learning, Human-Robot Collaboration, Description Logic.

1 Introduction

Robots should be able to *bootstrap* knowledge by observing humans [12], which might communicate verbally with supervising purposes. A robot needs to focus on the *important* changes of the environment in order to build an agnostic structure, which will be used for reproducing the observed task in other contexts. State of the art approaches use imitation learning for mapping observations and interactions into motion primitives at the trajectory and symbolic levels [2]. Recurrent Neural Networks [10], Reinforcement Learning [13] and Deep Learning [15] have been used to actuate a robot based on demonstrations, and the latter work relies on cognitive aspects to acquire knowledge, i.e., with *attention* factors. However, it is challenging to address the issue of storing tasks structures that are communicable to users, which might improve them through dialogues for instance. Also, learning black-box like structures strongly limits the integration of state of the art symbolic task planners and imitation learning techniques.

We present an approach to structure models of the observed environment into a memory, which can be used with reasoning and planning purposes. We used Description Logic (DL) [1] to manage a general-purpose memory, which contains *items* and have *functions* to store and retrieve observations deduced through interaction. This paper introduces a formal framework to investigate methods to acquire communicable knowledge into the robot's memory for supporting its



Facts		*Beliefs*	*Category*
$\gamma_1\colon\Lambda_1$,	$\gamma_1,\gamma_3\colon\mathbf{R}_1$,	$\epsilon_t,\gamma_3\colon\mathbf{R}_1^{\Lambda_1}$,	$\varPhi_t \sqsubseteq \bar{\varPhi}$
$\gamma_2\colon\Lambda_2$,	$\gamma_3,\gamma_2\colon\mathbf{R}_2$,	$\epsilon_t,\gamma_2\colon\mathbf{R}_2^{\Lambda_3}$,	$\sqcap \geqslant 1\ \mathbf{R}_1^{\Lambda_1}.\Lambda_3$
$\gamma_3\colon\Lambda_3$,	$\gamma_3,\gamma_4\colon\mathbf{R}_2$	$\epsilon_t,\gamma_4\colon\mathbf{R}_2^{\Lambda_3}$	$\sqcap \geqslant 2\ \mathbf{R}_2^{\Lambda_3}.\Lambda_2$
$\gamma_4\colon\Lambda_2$,			

Figure 1. Input facts based on an interface specifying Λ_1≡`RedBox`, Λ_2≡`GreenBox`, Λ_3≡`BlueBox`, and $\mathbf{R}_1$≡`alignedWith`, $\mathbf{R}_2$≡`connectTo`. From facts, SIT derives the beliefs of the scene ϵ_t, which are used to build the $\varPhi_t$ category.

actions. In particular, we consider a scenario where knowledge is memorised online and stored into a structured tasks representation domain.

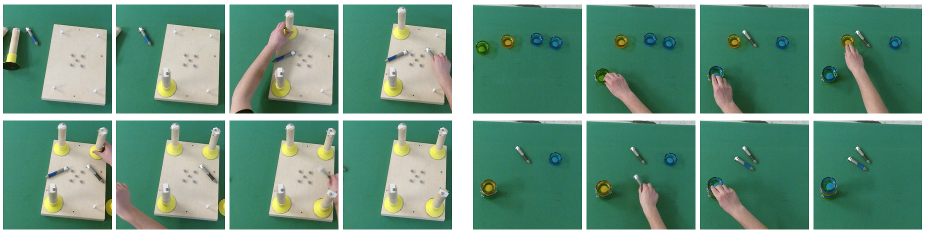
2 Memory Items

We developed the Scene Identification and Tagging (SIT) algorithm [7], which generate a symbolic representation of a *scene* acquired while observing a human-lead demonstration. The algorithm models scene *categories* from *beliefs* about the environment, which are computed with perception modules that provide *facts*. Facts and beliefs describe the environment only for the current instant of time, while categories are stored in – and retrieved from – the memory. SIT uses scene beliefs for (*i*) creating a new category from observation, and (*ii*) classifying the current scenes with respect to categories previously learned, if any.

Since SIT is based on symbols in an ontology, it can be defined with a general-purpose *input interface*, based on (*i*) a set of DL concepts $\bar{\Lambda} \sqsubseteq \{\Lambda_1 \dots \Lambda_n\}$ describing entities in the environment (e.g., `RedBox`), and (*ii*) a set of DL role $\bar{\mathbf{R}} \sqsubseteq \{\mathbf{R}_1 \dots \mathbf{R}_m\}$ representing relationship among entities of type $\bar{\Lambda}$. Thus, the input facts are role assertions at a specific time instant, i.e., a role $\gamma_1,\gamma_3\colon\mathbf{R}_1$, which relates the DL instances γ_1 and γ_3, that are classified in $\bar{\Lambda}$ (e.g., $\gamma_1\colon\Lambda_1$). Figure 1 shows a simple 2D example and possible input facts required by SIT. In this case, a fact is γ_1,γ_3: `alignedWith`, where γ_1: `RedBox` and γ_3: `BlueBox`.

For each fact, SIT computes a belief that contributes to the description of the scene ϵ_t. Beliefs are computed through *reification*, which defines a DL role $\mathbf{R}_i^{\Lambda_j}$ with a symbol deduced from the concatenation of the symbols defining $\mathbf{R}_i$ and Λ_j, e.g., $\mathbf{R}_1^{\Lambda_3} \equiv$ `alignedWithBlueBox`. With beliefs about ϵ_t, SIT can create a new DL concept $\varPhi_t$ that represents a scenes category in the ontology, which is defined with conjunctions of cardinality *restrictions*, as shown in the last column of Figure 1. In the example of Figure 1, the model of the environment at time t is expressed as a scene category $\varPhi_t$ where: "at least 1 `BlueBox` is `alignedWith` a `RedBox`, and at least 2 `GreenBox` are `connectedTo` a `BlueBox`". Remarkably, each $\varPhi_t$ is defined with respect to the universal scene $\bar{\varPhi}$, which contains all the possible scenes that can be represented with an input interface $\langle\bar{\Lambda}, \bar{\mathbf{R}}\rangle$.

SIT checks the consistency among categories restrictions through DL reasoning, which generates a graph, i.e., the robot's *memory*. In the memory, each



(a) A demonstration of an assembly task. (b) A demonstration of objects stacking.

Figure 2. Salient scenes acquired during two demonstrations sequentially performed and arranged over time.

node is an item describing a scene category, while each edge identifies a logic implication among them (Figure 3). Such a graph does not only represents relations among sub-scenes (i.e., $\Phi_i \sqsubseteq \Phi_j$), but it can also be used to classify a scene ϵ_t with respect to previously generated categories, i.e., $\epsilon_t\!:\!\Phi_{t-i}$.

For instance, if at t_2 a new block is introduced in the scene of Figure 1 (acquired at t_1), SIT will perform one-shot learning to generate a new category Φ_2. Then SIT would classify the new scene ϵ_2 in the categories Φ_1 and Φ_2, which are related to time t_1 and t_2 respectively. This occurs because ϵ_2 has beliefs that also respect the restrictions learned from ϵ_1 and stored in Φ_1. In other words, SIT infers that the second category implies the first, i.e., $\Phi_2 \sqsubseteq \Phi_1$, that is related to an edge in the memory graph (e.g., Figure 3).

Moreover, SIT provides a normalised *similarity* value to describe the classification of ϵ_t in more categories. This value is low when few beliefs of ϵ_t satisfy the restrictions of a category Φ_{t-i}, and high otherwise. Based on such a value, the SIT *output interface* is a sub-graph of the memory containing each node Φ_j that (*i*) has all the restrictions satisfies by the beliefs of the current scene ϵ_t, and (*ii*) does not have too many unspecified restrictions for the other beliefs of ϵ_t (e.g., the once introduced by the new block when ϵ_2:Φ_1 is evaluated).

3 Memory Capabilities

Since we want to use SIT when demonstrations hold for a reasonably long interval of time, and the robot perceives inputs fact about the scenes observed with a suitable frequency, we define a *consolidation* score for each node in the memory graph Φ_t, and five functions inspired by cognitive models. Remarkably, since SIT performs one-shot learning, it might occur that the knowledge in memory *overfits* a particular demonstration. In our framework, the consolidating and forgetting capabilities are used to avoid this issue by implementing an attentive behaviour that identifies the important items to maintain in memory, e.g., the once that do not involve the pens in Figure 2.

More in detail, (*i*) the *encoding* functionally generates input fact based on a contextualisation of sensory data, e.g., to extract spatial relations based on

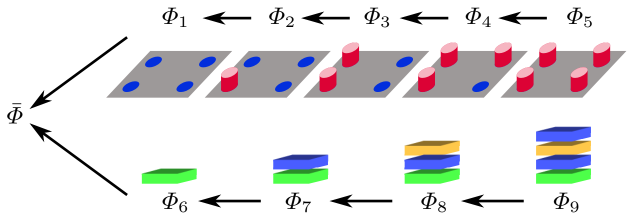


Figure 3. The memory after having observed the two demonstrations in Figure 2, and the representation of the scene categories Φ_t sored in the graph at the end of the experiment.

the centre of mass and shape of objects. During (*ii*) the *storing* function, SIT attempts to classify ϵ_t and, if it successes, each Φ_{t-i} node in the output graph will increase their consolidating score. Otherwise, a new category Φ_t will be derived from beliefs and added to the memory. (*iii*) The *retrieving* function uses DL queries to classify categories when a scene is requested through beliefs. Similarly to storing, also retrieving affects the consolidating scores. (*iv*) The *consolidate* function traverses the memory and normalises the scores of each node based on its neighbours and time trace decay theory [11]. Whereas (*v*) the *forgetting* function removes the nodes with a low score and restructures the edges of the graph consistently.

4 Preliminary Results

The consolidation score does not only rank categories for retrieving purposes, but it also allows to implement a forgetting function for removing categories that are not relevant to the demonstrated task. We preliminary tested our system with the hypothesis that often observed scenes are more important (and might never be forgotten), than sporadic configurations of facts (which can be neglected). We tested this approach in two scenarios involving different types of demonstrations. One consists of assembling the four legs of a table (Figure 2a), while the second in stacking four objects on top of each other (Figure 2b). For both scenarios we considered the $\bar{\mathbf{R}} \sqsubseteq \{\texttt{connectedTo}\}$ role assertion, which is estimated from objects' centre of mass. Whereas $\bar{\Lambda} \sqsubseteq \{\texttt{Support}, \texttt{Leg}, \texttt{Pen}\}$ was considered in the first scenario, while $\bar{\Lambda} \sqsubseteq \{\texttt{Box}, \texttt{Pen}\}$ in the second. In each scenario, we consider an object not related to the task (i.e., `Pen`), which is used to increase the scenes variability with configurations not strictly related to the demonstrated task.

Figure 3 shows the memory graph after the observation of the demonstrations partially shown in Figure 2. We notice that all the categories restricting some pens in the scene have been forgotten since they were not persistent during the overall demonstrations. In our scenario, SIT generates a memory that supports planning techniques because it is possible to find the differences among a ij-th category pair, and perform the actions required to change the classification of ϵ_t from Φ_i to Φ_j (e.g., with simulations [5]). Without using a consolidating and forgetting approach, we obtained a memory graph including all the demonstrated scenes, and many nodes were not directly related to the task. This would not only strongly limit the performances of SIT over time, but it also requires to deploy sophisticated reasoning and planning techniques since the graph becomes more complex. Instead, with the forgetting policy configured for our tests, SIT represents the tasks in a manner that is effective for planning purposes. However, to generalise this result for many different scenarios is still an open issue.

We implemented the SIT algorithm in a ROS architecture based on the ARMOR service [3], and we investigate a scenario where a human could refine the robot knowledge through dialogues during the demonstration. In [8], we addressed this complex human-robot interaction with a relatively simple system since we exploited the transparent representation that SIT generates. More generally, we obtained such representation because we based SIT into a symbolic representation that is familiar to users.

Nonetheless, using a symbolic formalism also allows us to design SIT with a general-purpose input interface, which supports multimodality and that can be used to generate graphs that contextualise facts differently, e.g., for implementing semantic and episodic memory types [14]. On the other hand, our symbolic input interface also leads to the main drawback of our framework since it does not allow to use sensory data directly, and it requires a prior symbolic set of environmental features $\langle \bar{\Lambda}, \bar{\mathbf{R}} \rangle$ to be accurately perceived over time, e.g., using [4] and [6]. Nonetheless, our framework gives a formal platform to investigate the generation of planning domain through demonstrations also under uncertainties since the approach presented in this paper is compliant with the SIT extension based on fuzzy logic [9].

5 Conclusions

We presented a framework to acquire knowledge through interaction and produce a transparent robot memory that can represent planning domains. The memory allows encoding, storing, retrieving, consolidating and forgetting models of environmental states based on reasoning and contextualisation. With two proof of concept scenarios, we discussed a flexible framework for further investigating memory capabilities. Also, we introduced some open issues and limitations.

References

1. Baader, F., Calvanese, D., McGuinness, D., Patel-Schneider, P., Nardi, D., et al.: The description logic handbook: Theory, implementation and applications. Cambridge university press (2003)
2. Billard, A., Calinon, S., Dillmann, R., Schaal, S.: Robot Programming by Demonstration. In: Siciliano, B., Khatib, O. (eds.) Springer Handbook of Robotics, pp. 1371–1394. Springer Berlin Heidelberg, Berlin, Heidelberg (2008)
3. Buoncompagni, L., Capitanelli, A., Mastrogiovanni, F.: A ros multi-ontology references service: Owl reasoners and application prototyping issues. In: Proceedings of the 5th Italian Workshop on Artificial Intelligence and Robotics A workshop of the

XVII International Conference of the Italian Association for Artificial Intelligence (AI*IA 2018). vol. 2352, pp. 36–41. CEUR-WS, Trento, Italy (nov 2018)
4. Buoncompagni, L., Carfì, A., Mastrogiovanni, F.: A software architecture for multimodal semantic perception fusion. In: Proceedings of the 5th Italian Workshop on Artificial Intelligence and Robotics A workshop of the XVII International Conference of the Italian Association for Artificial Intelligence (AI*IA 2018). vol. 2352, pp. 18–23. CEUR-WS, Trento, Italy (nov 2018)
5. Buoncompagni, L., Ghosh, S., Moura, M., Mastrogiovanni, F.: A Scalable Architecture to Design Multi-modal Interactions for Qualitative Robot Navigation. In: Ghidini, C., Magnini, B., Passerini, A., Traverso, P. (eds.) AI*IA 2018 – Advances in Artificial Intelligence. pp. 96–109. Lecture Notes in Computer Science, Springer International Publishing, Trento, Italy (nov 2018)
6. Buoncompagni, L., Mastrogiovanni, F.: A software architecture for object perception and semantic representation. In: Proceedings of the 2nd Italian Workshop on Artificial Intelligence and Robotics A workshop of the XIV International Conference of the Italian Association for Artificial Intelligence (AI*IA 2015). vol. 1544, pp. 116–124. CEUR-WS, Ferrara, Italy (sep 2015)
7. Buoncompagni, L., Mastrogiovanni, F.: Teaching a robot how to spatially arrange objects: Representation and recognition issues. In: 2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN). pp. 1–8. New Delhi, India (Oct 2019)
8. Buoncompagni, L., Mastrogiovanni, F.: Dialogue-based supervision and explanation of robot spatial beliefs: a software architecture perspective. In: 2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN). pp. 977–984. IEEE, Nanjing, China (Aug 218)
9. Buoncompagni, L., Mastrogiovanni, F., Saffiotti, A.: Scene learning, recognition and similarity detection in a fuzzy ontology via human examples. In: Proceedings of the 4th Italian Workshop on Artificial Intelligence and Robotics A workshop of the XVI International Conference of the Italian Association for Artificial Intelligence (AI*IA 2017). vol. 2054, pp. 10–15. CEUR-WS, Bari, Italy (nov 2017)
10. Duan, Y., Andrychowicz, M., Stadie, B., Ho, J., Schneider, J., Sutskever, I., Abbeel, P., Zaremba, W.: One-shot Imitation Learning. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. pp. 1087–1098. NIPS'17, Curran Associates Inc., New York, USA (dec 2017)
11. Jonides, J., Lewis, R.L., Nee, D.E., Lustig, C.A., Berman, M.G., Moore, K.S.: The Mind and Brain of Short-Term Memory. Annual review of psychology 59, 193–224 (sep 2008)
12. Kunze, L., Burbridge, C., Hawes, N.: Bootstrapping Probabilistic Models of Qualitative Spatial Relations for Active Visual Object Search. In: 2014 AAAI Spring Symposium Series (Mar 2014)
13. Stulp, F., Theodorou, E.A., Schaal, S.: Reinforcement Learning With Sequences of Motion Primitives for Robust Manipulation. IEEE Transactions on Robotics 28(6), 1360–1370 (Dec 2012)
14. Vernon, D.: Artificial Cognitive Systems: A Primer. MIT Press (Oct 2014)
15. Zhang, T., McCarthy, Z., Jow, O., Lee, D., Chen, X., Goldberg, K., Abbeel, P.: Deep Imitation Learning for Complex Manipulation Tasks from Virtual Reality Teleoperation. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). pp. 1–8. Brisbane, Australia (May 2018)