Knowledge-based sensing/acting in mobile autonomous robots

Michele Ruta, Floriano Scioscia, Giuseppe Loseto, Eugenio Di Sciascio
Politecnico di Bari
via E. Orabona 4, Bari (I-70125), Italy
{michele.ruta,floriano.scioscia,giuseppe.loseto,eugenio.disciascio}@poliba.it

***Abstract*—The paper proposes a knowledge-based framework for mobile autonomous robots. It exploits data annotation for semantic-based context description. High-level event/situation detection and action decision are performed through a semantic matchmaking approach, supporting approximate matches and relevance-based ranking. The framework was fully implemented in a prototype built with off-the-shelf components, validated in a Search And Rescue (SAR) case study and evaluated in an early performance analysis, supporting the feasibility of the proposal. The work demonstrates novel analysis methods on data extracted by inexpensive sensors can yield useful results without requiring hefty computational resources.**

I. Introduction

Smart mobile robotic platforms are emerging through the integration of artificial intelligence technologies, low-power embedded processing platforms and a wide array of sensing micro- and nano-devices. The goal of fully autonomous wide-purpose robots, however, still remains elusive. Autonomy has been defined as "the extent to which a robot can sense its environment, plan based on that environment, and act upon that environment with the intent of reaching some task-specific goal (either given to or created by the robot) without external control." [1]. Transforming low-level raw environment data into high-level actionable knowledge is clearly one of the main issues. Captured data streams must be processed efficiently and effectively in order to endow retrieved information with a meaningful interpretation. Furthermore, flexible inference paradigms are needed in order to enable robots to take decisions even in the presence of incomplete, uncertain or contradictory information. Current knowledge-based approaches mostly require a full match between the robot's perceptions and situation/action templates or rules, in order to trigger a decision. This can have serious practical –or even ethical– implications when a robot is faced with a decision among partially satisfactory choices [2].

The combination of wireless networks of autonomous embedded devices with Semantic Web ideas and technologies is bringing about the so-called *Semantic Web of Things* (SWoT) [3]. This paradigm aims to enable novel classes of intelligent applications and services based on Knowledge Representation (KR). Semantically rich (compact) descriptions are associated to real-world objects and to data they retrieve. Hence, logic-based automatic inferences are exploited to infer implicit information starting from an explicit event and context detection. This can lead to increased autonomic capabilities of pervasive robotic devices as well as to self-coordination and self-orchestration of teams of robotic agents by means of high-level knowledge exchange.

In order to support the above vision, this paper introduces a knowledge-based framework for robot sensing/acting in pervasive environments. The proposed approach refers to a mobile autonomous robot gathering information through micro-devices attached on board or deployed in given environments and interconnected wirelessly. A semantic-based multisensor fusion is performed, by extracting relevant features from each data stream and annotating them as Description Logic (DL) conjunctive concept expressions, in compliance with an ontology modeling the reference knowledge domain. Web Ontology Language (OWL 2) [4] is adopted as reference language. The produced meaningful and structured annotations are exploited for context and event recognition as well as action determination, by means of a rigorous semantic matchmaking framework grounded on non-standard inference services [5]. The framework adopts the Open World Assumption (OWA): unspecified information is not considered as a constraint of negation, but simply as missing (*e.g.*, due to being unknown or irrelevant). Support for approximated matches with relevance score evaluation is a key feature of the proposed approach. It enables an autonomous robot to assess whether (and how much) an action is acceptable in a particular context, comparing and ranking different alternatives, when a full match with the current context description does not occur. Explanation capabilities provided by the underlying semantic matchmaking increase confidence in system behaviors.

The devised framework aims to show how novel analysis methods applied to data extracted by inexpensive off-the-shelf sensor devices can yield useful results without requiring hefty computational resources, unmanageable in mobile and pervasive contexts. The approach was validated in a Search And Rescue (SAR) case study. Performance was evaluated on a fully implemented prototype based on iRobot Create 2 programmable robotic platform and UDOO Quad single-board computer, in order to ensure the computational feasibility of the proposal.

The remainder of the paper is as follows. Section II discusses relevant related work. The proposed framework is explained in Section III, while Section IV describes the implemented robot prototype. Section V reports an illustrative example from the SAR case study. Section VI describes the

978-1-5090-6724-4/17 $31.00 © 2017 IEEE
DOI 10.1109/IRC.2017.83

experiments, and conclusion closes the paper.

II. Related work

Semantic data mining is a relevant issue for robotic applications. The survey [6] reviews works on ontology-based rule mining, classification and clustering. Ontologies are useful to bridge the semantic gap between raw data and applications, as well as to provide data mining algorithms with prior knowledge to guide the mining process or reduce the search space. They have been successfully used in all steps of a typical data mining workflow. In ontology-based information extraction (OBIE) [7] they usually represent also the output of data mining, like in this paper. Providing mining results with rich, formal and machine-understandable meaning [8] makes them useful for further inference processes.

Robot data sources are a combination of on-board sensors and micro-devices interconnected through short-range wireless links. In the latter case, reference technologies include sensor-enabled RFID (Radio Frequency IDentification) tags [9] and Semantic Sensor Networks (SSNs) [10]. Semantic Web research addressed the issue of describing sensor and data features through ontologies. *SSN-XG* [11] was adopted as reference in this work, being among the most rigorous, relevant and widely accepted proposals.

Event and activity recognition is a crucial aspect of autonomous robot platforms. The majority of approaches exploits classical Machine Learning techniques, but the lack of machine-understandable characterization of outputs is a prominent limit for a possible exploitation in fully autonomic robotic scenarios and applications. Semantic-based proposals also exist, not only for supporting subsequent inferences on action decisions [12], but also, *e.g.*, for gleaning environment mapping clues [13]. The DL-based framework for recognizing and describing events in [14] includes three abstraction layers: sensors, contexts and situations. Modeling, however, is mainly oriented to "on/off" sensors and sequences of events. Dealing with quantitative data was not properly addressed as done in this work. Furthermore, the matching algorithm is based on Subsumption, hence only full matches are recognized. The same limitation affects ontology-based approaches in [15] and [16] for office and home activity detection. The paper aims to show how detection approaches can exploit imprecise data more effectively by accepting a properly-determined approximated match whenever a full one is not available [17].

Robot task selection is another key application area for semantic-based techniques. Hierarchical task/subtask composition has been widely adopted to implement planning via inference services and logic programs. In [18], robot components, capabilities, and actions are modeled in a DL Knowledge Base (KB). A requested action is matched to a robot by decomposing it into a tree of sub-actions and matching each elementary action with the robot's capabilities; similarly, in order to possess a capability a robot must provide all required components and sub-capabilities it depends on. This recursive approach allows enumerating missing components and capabilities in case of failure, thus providing useful explanation. Nevertheless, the Subsumption-based matching can yield only "yes/no" answers, limiting the robot usefulness. Supporting approximate matches would provide "good enough" solutions in many situations, like in the example reported in [18] where a robot, though capable of setting the table with regular cutlery, will reply it cannot set the table with silverware if it is not explicitly modeled as a subclass of cutlery in the KB. A similar limitation affects the approach in [19], which uses service composition to orchestrate a robot with other ambient intelligence devices in its surroundings: the process hopelessly fails when a full match is not achieved. Actually, non-standard inference services supporting approximate matches would generate useful plans –even though imperfect– in many practical service composition scenarios [20], [21]. The proposed approach supports approximate matches while retaining logic explanation capabilities, so combining the best of both worlds.

III. Semantic-enhanced robotic framework

The proposed framework enables robot sensing and acting in complex environments by exploiting Knowledge Representation and Reasoning (KRR). The process starts with raw data extracted from the environment and leads to the decision of the most appropriate action, in a continuous evaluation and feedback process. KRR allows dealing with several issues, including: multi-sensor fusion; production of high-level semantic-based context annotations out of low-level numerical data streams; recognition of conditions/events of interest; matching the most appropriate action for the current situation in a dynamic way. The approach relies on a Knowledge Base in a restriction of the Web Ontology Language (OWL 2) [4] corresponding to $\mathcal{ALN}$ (Attributive Language with unqualified Number restrictions), a medium-expressiveness language of the Description Logics family. $\mathcal{ALN}$ was selected to support the efficient execution of non-standard inference services used subsequently for semantic matchmaking, as explained below. Basically, a KB is composed of: (i) a TBox $\mathcal{T}$ (*Terminological Box* a.k.a. *ontology*), modeling conceptual knowledge for the reference domain through *concepts* (a.k.a. *classes*) representing sets of objects, *properties* (a.k.a. *roles*) representing relationships between pairs of concepts and *axioms* used to limit the way the meaning of concepts and properties can be interpreted; (ii) an ABox (*Assertion Box*), which uses the TBox elements to describe *individuals* (a.k.a. *instances*) pertaining to a specific application or problem. In the proposed framework, the KB represents a logical core endowment of a robot, through which it can interpret data, phenomena and events, as well as take decisions dynamically. This enables the robot to act effectively and to increase progressively its own knowledge from past and present observations.

The overall sensing/actuation framework is depicted in Figure 1. The main steps are described as follows:

1. Data gathering. The robot acquires raw numerical data from its environment through sensors, either on-board ones or via wireless connections.

2. Data mining. It starts with a preprocessing step, splitting

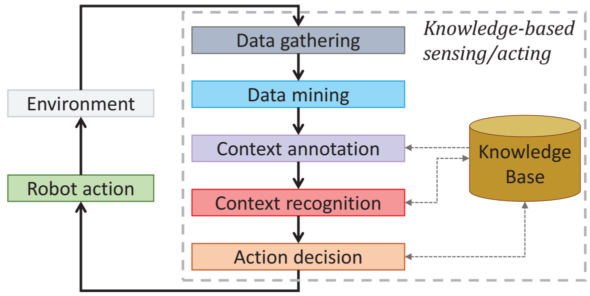
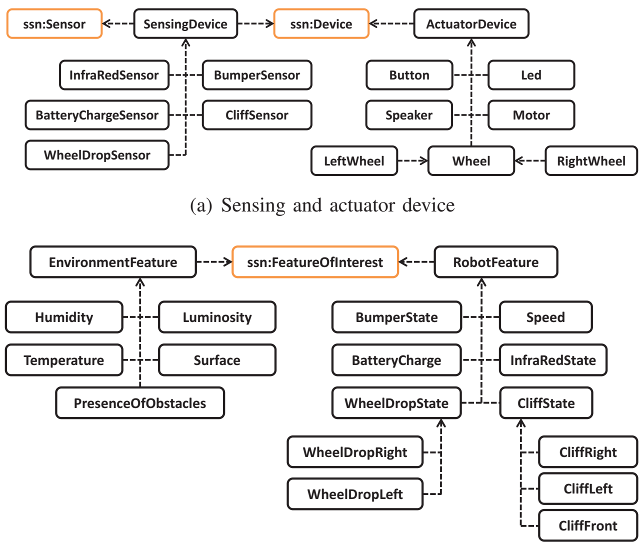


Fig. 1. Sketch of the proposed knowledge-based robot framework



(a) Sensing and actuator device

(b) Feature of Interest

Fig. 2. Robot ontology excerpt (all depicted relationships are `rdfs:SubClassOf`)

data streams in observation windows. For each window, relevant features are computed, according to the sensor type. Statistical features are always produced, including static ones on a single window (average, standard deviation, kurtosis, skewness, etc.) and dynamic ones w.r.t. previous windows (moving average, derivative, etc.). The data mining framework

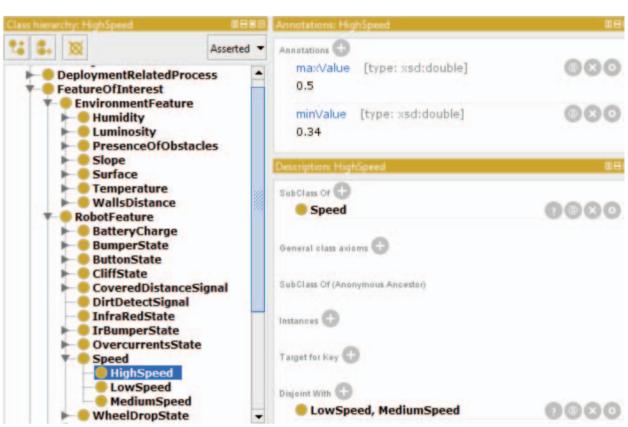


Fig. 3. Ontology-based data annotation

is open to computing further features for specific data sources, based on domain knowledge and depending on the needs of the robotic platform application.

3. Context annotation. For each data source, computed features are mapped to an OWL conjunct by means of an ontology $\mathcal{T}$ modeling the domain conceptualization along properly defined patterns. In particular, as shown in Figure 2, SSN-XG [11] was extended to model more detailed annotations of robot features and context entities. Basically, each parameter is represented as a classes/subclasses taxonomy featuring all its significant configurations in the domain of interest. Figure 3 depicts an example OWL ontology excerpt, modeling the *Speed* parameter: its value can be annotated with one of the three subclasses *HighSpeed*, *LowSpeed* and *MediumSpeed*, for which numerical data ranges (in m/s) are defined within the ontology through OWL annotation properties (as shown in the upper right area of the picture). The robot processing unit finally builds a complete annotation of the current context, by joining the above logical concepts in a conjunctive expression. In this way, context annotation is independent from the particular domain. Once defined, the robot ontology can be used as an upper ontology and exploited in different scenarios to annotate specific objects or events. If needed, derived domain ontologies can be also defined to enrich the description of a reference environment, including novel concepts created or imported from other existing vocabularies.

4. Context recognition. The obtained semantic annotation connotes the observed environment in a detailed, structured and meaningful way, based on input sensor data. Represented information, however, is still of rather low level. The robot is now able to exploit this *context profile* in order to recognize particularly relevant events, conditions and aspects currently in the environment, and to represent them by means of more abstract high-level concepts in the ontology. In the proposed knowledge-based approach, the recognition problem is solved by means of *semantic matchmaking*, which basically ranks a set of resources (in this case, *context templates*, stored as individuals in the ABox of the KB) according to relevance w.r.t. a request (the context profile). Standard reasoning services for matchmaking include *Subsumption* and *Satisfiability*. Unfortunately, they can only detect *full* and *disjoint* (a.k.a. *partial*) *matches*, respectively. This is inadequate for advanced autonomic scenarios, because full matches seldom occur and disjointness is frequent when dealing with complex descriptions. The framework exploits *Concept Abduction* and *Concept Contraction* non-standard inference services [5] to produce a fine-grained resource ranking with a logic-based explanation of outcomes. Given a request R disjoint with an available resource S, Contraction detects which part G (for *Give up*) of R is conflicting with S. If one retracts G from R, a contracted version K (for *Keep*) of the original request is obtained, such that it is compatible with S. If R and S are compatible but S does not match R fully, Abduction identifies what additional features H (for *Hypothesis*) should be assumed in S in order to reach a full match. A *penalty function* associates a semantic distance metric to both Contraction and Abduction.

This induces the final ranking: matchmaking will return the context description with lowest distance w.r.t. the annotation generated from raw data. The framework includes the *Mini-ME* mobile reasoner, [5] providing efficient implementations of the above inferences for low-energy computing architectures on moderately expressive DLs: this tool is fit for battery-powered mobile robot platforms.

5. Action decision. This step is carried out in a conceptually similar way w.r.t. the previous one. Semantic matchmaking is computed between the recognized high-level context description and a set of *action templates*, stored as individuals in the KB. The matchmaking outcome determines the action to perform as the semantically closest one w.r.t. the current context. The obtained distance score represents the robot's overall confidence in its decision. Explanation features of the semantic matchmaking approach (outlining the contents of G and H) provide justification to the robot's choices, so increasing human trust in the system.

IV. Prototype implementation

A prototypical testbed, shown in Figure 4, was implemented following the proposed semantic-based robotic framework. It includes three basic components:

1. *Robot Platform*: a robot endowed with motors, sensors and actuators was implemented starting from the *iRobot Create 2 Programmable Robot*[1] base stack.

2. *Control Unit*: a *UDOO Quad*[2] single-board computer was exploited to manage both basic robot functionalities and semantic-based tasks. It is equipped with: (i) *UDOObuntu 2.0 Minimal Edition*[3] operating system, a customized version of Ubuntu OS 14.04 LTS for embedded boards; (ii) *ROS* (Robot Operating System) *Indigo* [22] framework; (iii) Java 8 SE Runtime Environment (32-bit ARM build 1.8.0_91-b14). The board is connected to the robot through a *serial-to-USB* cable and is powered by a portable 15000 mAh power bank to avoid absorbing energy from the Create 2 internal battery and increase the overall robot working time between charges.

3. *Sensor devices*: in addition to the sensors embedded into the robot (bumpers, wheel drop, slope, proximity, speed, odometer, battery level) further micro-devices can be connected to the UDOO board to gather data from the environment or exchange information with other robots.

In particular, according to the architecture in Figure 5, a Java-based management software was developed to run on the robot control unit. It consists of the following packages:

- `it.poliba.sisinflab.ros`: implements all ROS-based functionalities (*e.g.*, movement and robot management). *rosjava*[4] libraries were used to communicate with ROS, *irobotcreate2ros*[5] library was exploited for robot I/O and the *rosjava_actionlib*[6] library was used to move the robot toward a target location;

- `it.poliba.sisinflab.sensors`: includes classes to gather data from sensors connected to the board or dipped in a given environment;

- `it.poliba.sisinflab.datamining`: implements the algorithm described in Section III in order to support semantic-based annotation and interpretation of raw data;

- `it.poliba.sisinflab.owl`: provides basic functionalities to load the reference KB, to manage all generated OWL annotations and to invoke the Mini-ME reasoner [5] implementing non-standard inference services.

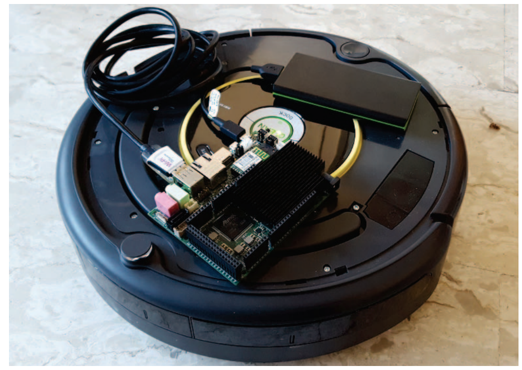


Fig. 4. Prototypical testbed

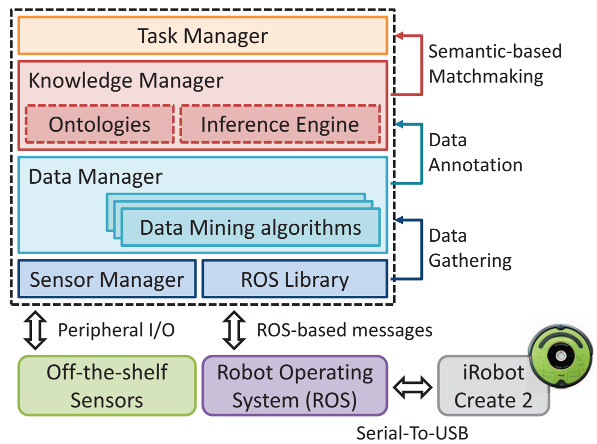


Fig. 5. Reference architecture of a robot control unit

V. Illustrative example

A *Search And Rescue* (SAR) case study is described to clarify practical applications of the proposed approach. SAR is a well-known robotic challenge aiming to develop and exploit autonomous rescue robots able to help human rescuers in case of natural or man-made disasters (*e.g.*, earthquakes, wildfires, explosions). Several SAR tasks can be performed by robots:

[1]http://www.irobot.com/About-iRobot/STEM/Create-2.aspx

[2]http://www.udoo.org/udoo-dual-and-quad –equipped with quad-core ARM Cortex A9 at 1 GHz clock frequency, ARM Cortex M3 coprocessor, 1 GB DDR3 RAM.

[3]http://www.udoo.org/udoobuntu-2-minimal-edition

[4]http://github.com/rosjava/rosjava_core

[5]http://github.com/CentroEPiaggio/irobotcreate2ros

[6]http://github.com/ernestmc/rosjava_actionlib

rubble penetration and removal, reconnaissance and mapping, object identification and victim extraction.

In the proposed approach semantic-based enhancements are used *to suggest, according to the detected conditions in the disaster area (*e.g., *surface status and slope, presence of obstacles, temperature), which victims or objects should be extracted, minimizing risks and dangerous tasks*. Let us consider a simple example where the rescue robot penetrates into small to medium buildings collapsed after a natural disaster in a urban area. *In the first building, a theatre, there are three different rescue targets: a piano, an ancient vase and an unconscious person.* Each target is characterized by a semantic-based annotation defined with respect to the reference ontology reported in Section III.

Target annotations, reported in Figure 6, consist of a set of features describing both object characteristics (*e.g.*, size, weight) and safety conditions needed for a correct rescue, expressed as properties *detected* by a robot. Just as an example, considering a theatre furniture, a *piano* is defined as a resistant and bulky object so during the rescue it is suggested to avoid steep slopes and to use robots with a full battery charge, due to the possible long time and energy needed to extract it. Moreover, it is a wooden item, so high temperature or humidity can damage the object. It is important to notice that all high-level classes (*e.g.*, `LowSlope`, `FullCharge`, `Smooth` surface) are defined in the KB starting from low-level concepts (subclasses of `RobotFeature`) mapping the raw data gathered from robot sensors.

```
Piano ≡ ResistantItem and Bulky and Wooden
Bulky ≡ (hasSize only HighSize) and (hasWeight
only Heavy) and (detectsSlope only LowSlope) and
(detectsBatteryCharge only FullCharge)
Wooden ≡ (detectsTemperature only (not
HighTemperature)) and (detectsHumidity only (not
HighHumidity))
AncientVase ≡ FragileItem and Lightweight and
HeatSensitive
FragileItem ≡ (detectsSurface only SmoothSurface)
and (detectsWallDistance only Faraway) and
(detectsObstacle only (not Obstacle))
Lightweight ≡ (hasWeight only Light)
HeatSensitive ≡ (detectsTemperature only
LowTemperature)
UnconsciousPerson ≡ Human and HeatSensitive and
WaterSensitive and LifeThreatening
WaterSensitive ≡ (detectsHumidity only LowHumidity)
LifeThreatening ≡ (detectsSurface only Smooth) and
(detectsSpeed only HighSpeed)
DisasterArea ≡ (detectsSlope only LowSlope)
and (detectsWallDistance only MediumDistance)
and (detectsObstacle only FewObstacles) and
(detectsTemperature only MediumTemperature)
and (detectsHumidity only HighHumidity) and
(detectsSpeed only HighSpeed) and (detectsSurface
only RoughSurface) and (detectsBatteryCharge only
HalfCharge)
```

Fig. 6. Annotations of the reference scenarios

While the robot moves in the theatre, it collects data in the field progressively creating an annotation referred to a given ontology. When one or more objects are detected, the disaster area annotation (reported in Figure 6) is processed by the Mini-ME reasoner [5] and compared with stored objects descriptions. The goal is to reach an identification and proceed to the rescue and extraction from the area. Exploiting the *Concept Abduction* and *Concept Contraction* inference services [5], the reasoner ranks the target items according to observed conditions obtaining the following scores: piano 0.355; ancient vase 0.485; unconscious person 0.245. In this case, the robot selects the fainted person as reference target for rescue, exhibiting the lowest semantic distance, *i.e.*, the highest similarity between safety requirements and current conditions in the area. Note that the disaster area is affected by high humidity, a condition contrasting with rescue description for both piano and humans. Approaches based on rules or standard inferences would discard both tasks, selecting one with less common features w.r.t. current situation. After the first extraction, rescuers clear the rubble, remove bulky obstacles and reduce the temperature of the environment. The robot detects these new conditions and builds a second annotation, also considering its battery charge is now reduced to 50%. According to the new context, it selects the ancient vase as reference target.

VI. Performance Evaluation

Performance evaluation of the proposed approach has been carried out exploiting the prototypical testbed described in Section IV for the case study reported above. Experiments were conducted using the robot in the same disaster area (simulated in a laboratory room), containing 8 different targets to identify, and varying the conditions randomly for 10 different scenarios. Each test was split in the following tasks to identify and evaluate specific features featuring processing performance:

1) *Load KB*: loading and initializing the reference KB when the robot starts;
2) *Data gathering*: reading raw data coming from all connected sensors and storing them in a memory buffer;
3) *Data annotation*: building a semantic annotation w.r.t. the reference KB connoting the observed environment according to sensor data;
4) *Matchmaking*: performing non-standard inference services to identify most suitable rescue target according to the conditions of the detected area.

Each test was repeated three times and average values were taken. Results for processing times are reported in Figure 7. Ontology initialization is the slowest; however, this is not a problem, since loading occurs once per robot up-time session. On the contrary, data management and matchmaking tasks were very fast thanks to the optimized data structures and inference algorithms. Memory usage values, obtained with Oracle *JConsole* monitoring tool included in Java 8 JDK, are shown in Figure 8. Results refer to the first test scenario where the robot performs a rescue task for about 20 minutes. Robot management software requires very low memory on average:

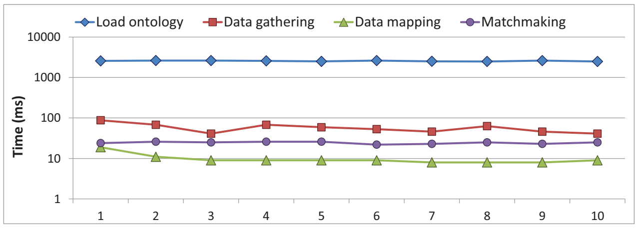


Fig. 7. Processing time

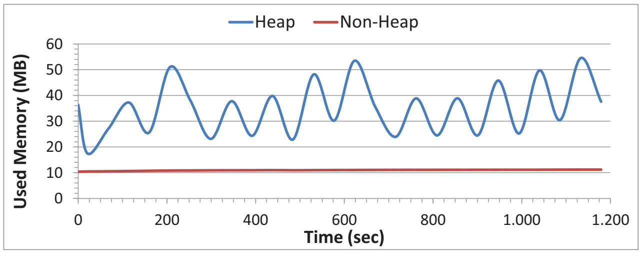


Fig. 8. Memory Usage

only 34.88 MB (Heap) and 10.97 MB (Non-Heap). Memory peak values were always under 55 MB, *i.e.*, reasonable values for embedded systems. Falling edges in the chart correspond to frequent and aggressive calls to the JVM garbage collector due to stricter memory constraints on embedded devices imposing to have as much free memory as possible at any time.

VII. Conclusion

This paper proposed a novel knowledge-based framework for autonomous pervasive robots. The approach includes sensor data mining, context recognition and action determination. It exploits a reference domain ontology to annotate data, while context and task templates are modeled as KB individuals. The framework was implemented in a fully functional prototype based on iRobot Create 2 robotic platform and UDOO Quad single-board computer. Early performance analysis results showed the feasibility of the proposal.

Future work includes: (i) improving the current prototype by increasing robot sensing and acting capabilities through the integration of further devices; (ii) creating a different prototype based on an off-the-shelf drone platform; (iii) studying knowledge exchange and coordination among teams of heterogeneous robots. Finally, the proposed approach will be tested in larger case studies and results compared with existing state-of-the-art approaches.

Acknowledgment

The authors acknowledge partial support of Apulia region cluster project PERSON.

References

[1] J. Beer, A. D. Fisk, and W. A. Rogers, "Toward a framework for levels of robot autonomy in human-robot interaction," *Journal of Human-Robot Interaction*, vol. 3, no. 2, pp. 74–99, 2014.

[2] B. Deng, "The robot's dilemma," *Nature*, vol. 523, no. 7558, pp. 24–26, 2015.

[3] M. Ruta, F. Scioscia, and E. Di Sciascio, "Enabling the Semantic Web of Things: framework and architecture," in *2012 IEEE Sixth International Conference on Semantic Computing*. IEEE, 2012, pp. 345–347.

[4] W3C OWL Working Group. OWL 2 Web Ontology Language Document Overview (Second Edition). W3C Recommendation 11 December 2012. [Online]. Available: http://www.w3.org/TR/owl2-overview/

[5] F. Scioscia, M. Ruta, G. Loseto, F. Gramegna, S. Ieva, A. Pinto, and E. Di Sciascio, "A mobile matchmaker for the Ubiquitous Semantic Web," *International Journal on Semantic Web and Information Systems*, vol. 10, no. 4, pp. 77–100, dec 2014.

[6] D. Dou, H. Wang, and H. Liu, "Semantic data mining: A survey of ontology-based approaches," in *Semantic Computing (ICSC), 2015 IEEE International Conference on*. IEEE, 2015, pp. 244–251.

[7] D. C. Wimalasuriya and D. Dou, "Ontology-based information extraction: An introduction and a survey of current approaches," *Journal of Information Science*, 2010.

[8] A. Pinto, F. Scioscia, G. Loseto, M. Ruta, E. Bove, and E. Di Sciascio, "A semantic-based approach for machine learning data analysis," in *Semantic Computing (ICSC), 2015 IEEE International Conference on*. IEEE, 2015, pp. 324–327.

[9] X. Jia, Q. Feng, T. Fan, and Q. Lei, "RFID technology and its applications in Internet of Things (IoT)," in *Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on*. IEEE, 2012, pp. 1282–1285.

[10] L. M. Ni, Y. Zhu, J. Ma, Q. Luo, Y. Liu, S. C. Cheung, Q. Yang, M. Li, and M. Wu, "Semantic Sensor Net: An Extensible Framework," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 4, no. 3/4, pp. 157–167, Apr. 2009.

[11] M. Compton, P. Barnaghi, L. Bermudez *et al.*, "The SSN Ontology of the W3C Semantic Sensor Network Incubator Group," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 17, 2012.

[12] K. Ramirez-Amaro, M. Beetz, and G. Cheng, "Understanding the intention of human activities through semantic perception: observation, understanding and execution on a humanoid robot," *Advanced Robotics*, vol. 29, no. 5, pp. 345–362, 2015.

[13] W. Sheng, J. Du, Q. Cheng, G. Li, C. Zhu, M. Liu, and G. Xu, "Robot semantic mapping through human activity recognition: A wearable sensing and computing approach," *Robotics and Autonomous Systems*, vol. 68, pp. 47–58, 2015.

[14] A. Scalmato, A. Sgorbissa, and R. Zaccaria, "Describing and recognizing patterns of events in smart environments with Description Logic," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1882–1897, 2013.

[15] T. A. Nguyen, A. Raspitzu, and M. Aiello, "Ontology-based office activity recognition with applications for energy savings," *Journal of Ambient Intelligence and Humanized Computing*, vol. 5, no. 5, pp. 667–681, 2014.

[16] L. Chen, C. D. Nugent, and H. Wang, "A knowledge-driven approach to activity recognition in smart homes," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 24, no. 6, pp. 961–974, 2012.

[17] M. Ruta, F. Scioscia, M. Di Summa, S. Ieva, E. Di Sciascio, and M. Sacco, "Semantic matchmaking for Kinect-based posture and gesture recognition," *International Journal of Semantic Computing*, vol. 8, no. 04, pp. 491–514, 2014.

[18] L. Kunze, T. Roehm, and M. Beetz, "Towards semantic robot description languages," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 5589–5595.

[19] K. Qian, X. Ma, X. Dai, and F. Fang, "Knowledge-enabled decision making for robotic system utilizing ambient service components," *Journal of Ambient Intelligence and Smart Environments*, vol. 6, no. 1, pp. 5–19, 2014.

[20] G. Loseto, F. Scioscia, M. Ruta, and E. Di Sciascio, "Semantic-based Smart Homes: a Multi-Agent Approach," in *13th Workshop on Objects and Agents (WOA 2012)*, ser. CEUR Workshop Proceedings, G. V. Flavio De Paoli, Ed., vol. 892, sep 2012, pp. 49–55.

[21] M. Ruta, G. Zacheo, L. A. Grieco, T. Di Noia, G. Boggia, E. Tinelli, P. Camarda, and E. Di Sciascio, "Semantic-based resource discovery, composition and substitution in ieee 802.11 mobile ad hoc networks," *Wireless Networks*, vol. 16, no. 5, pp. 1223–1251, 2010.

[22] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.