An Ontology-based approach for Robot and Ambient System collaboration

Emmanuel Dumont, Dan Istrate, Mohamed Chetouani

An Ontology-based approach for Robot and Ambient System collaboration

Emmanuel Dumont[1,2], Dan Istrate[1] and Mohamed Chetouani[2]

***Abstract*— In activity recognition, sensor based errors or uncertainties induce a wrong statement in the environment representation. To overcome these issues we introduce our ontology-based approach where a system, based on a mobile robot and an ambient system, is able to generate collaborative tasks when necessary to recognize activities. Using two ontologies to represent the environment, inconsistencies are detected and high level data are exchanged between both systems to correct and clarify ambiguous description of the environment. This process allows to generate adequate clarifying tasks by explicitly including knowledge and capabilities of each system. We expose our approach in two experiments where interaction between both systems are mandatory to respond a human "find an object" request in a first scenario and recognize multi user activities in a second scenario.**

I. INTRODUCTION

Robotic and ambient systems are more and more involved to improve wellness and health condition. Usability and acceptance of robotics approaches for Ambient Assisted Living are increasingly getting investigated [1]. In the field of Active Assisted Living (AAL), ubiquitous systems are employed and, sometimes, hardware limitations or uncertainties in data processing induce wrong statement and inaccurate decision (i.e. a blind spot on a mobile robot or a low confidence presence detection in a security system). To overcome those issues, several systems, such as robots and ambient systems in a cloud robotics networks, can be used in parallel to complete and reinforce each other environment representation [2], [3].

In activity recognition, robots or ambient systems are widely explored ([8]-[12]) combination of these systems, as proposed by Hu in [4], are getting investigated through smart environment and cloud robotics. In this paper, we introduce a ROS-based interactive platform including a mobile robot and an ambient system, integrating connected objects and sensors developed by several companies. Based on knowledge, perception and action capabilities, the platform generates adequate collaborative tasks using ontology based representation. From a user point of view, the interaction between the robot and the ambient system is transparent, (i.e. asking to the robot or the ambient system to find an object does not change the outcome of the scenario). In accordance with [5], we name "system of systems" this transparent global structure composed of the ambient system and the mobile robot.

[2]Laboratoire de BioMecanique et BioIngenierie UMR 7338, University of Technology of Compiegne, 60200 Compiegne, FRANCE

[1]Institut des Systemes Intelligents et de Robotiques ISIR UMR 7222, University of Pierre and Marie Curie, 75005 Paris, France emmanuel.dumont@isir.upmc.fr

In [6], a human and a robot interact together to find objects. The robot is able to handle ambiguities (i.e. more than one object fulfills the given description) using an ontology approach. Knowledge representation over an ontology grants a formal naming and description of the classes (Object, Human, Sensor, etc.), attributes (spatial, temporal and physical properties), and relationships between everything that physically or fundamentally exists in an environment allowing a formal context representation. An ontology grounds the sensed physical world with natural language, making the comparison of different types of sensors, and thus different types of information in a system, easier. Moreover communication between a system and a human is also facilitated as an information comes with its context [7]. Finally defining in an ontology offline classes, properties and relationships, and using this ontology as an online semantic, thus human readable, data collection allow to infer and reason on the stored knowledge, i.e. defining a property "*A recognized human is in the same room as the sensor used to recognized him*" will automatically infer, when a human is detected by a camera, "*This human is in the same room as the camera*". Thereby, from a basic online extracted information and the offline defined properties, an environment representation can be described.

II. RELATED WORK

Our work is based on the recent advancements in cloud robotics network and activity recognition. It relates on a collaboration between a robot and an ambient system, both able to detect and clarify inconsistencies in their environment representation. We tend to recognize activities with this system of systems and exploit a collaboration mechanism to perform tasks. Consequently, [8]-[10] introduce ambient systems composed of a wide set of sensors to recognize ADL. Tapia in [8] designed a system for complex activity recognition using *"tape on and forget"* state-changes sensors placed on doors, sinks, light switches, etc. This work demonstrates that pervasive simple sensors can be used to recognize activities. Storf is able in [9] to monitor and extract typical behavior of older person. The system detects deviation by employing a range of ambient and non intrusive sensors. Van Kasteren in [10] recorded a dataset in a house where 14 state-change sensors were installed on doors, cup-boards, refrigerator and a toilet flush sensor. Each activities were annotated by the subject. Probabilistic models are employed to determine features for each activity in order to detect them during future experimentation. Xia in [11] present a framework and algorithm to recognize ADL from the robot's

point of view. Two datasets (one using a humanoid robot and one with a non-humanoid robot) for egocentric robot activity recognition recorded were registered and analyzed.
Compared with probabilistic approaches, ontologies based modeling are adjustable and can be easily customized and scaled up during initialization and in use situation. Chen in [12] introduce an ontology-based approach; considering sensors, objects and activities; to model, represent and infer the current user's activity. Whereas Riboni in [13] propose to combine statistical inferencing and ontological reasoning to recognize ADL.
Finally, an interaction between a robot and an ambient system is introduced by Hu in [4]. These two systems perform a task of user identification: A pervasive camera indicates users' position to the robot able to navigate and identify them. Waibel in [14] introduces a platform where multiple robots are connected. Each robot collects data during the execution of a task and share them with others to improve the overall task performance using a simple learning algorithm. Manzi in [15] proposes a cloud robotic architecture where an assistive robot gets data from worn sensor to localize a human in an apartment. Flexibility in a smart environment facilitates the user acceptance. Amato in [16] designed a smart environment composed of a robot and an ambient system able to adapt itself to a new environmental and/or user condition.
The contributions of our paper are the implementation of a platform where a mobile robot and an ambient system collaborate, when mandatory, to achieve simple activity recognition tasks. Currently, we do not aim to recognize Activities of Daily Living (ADL). However as this platform will tend to recognize them, we emphasize some features of the system using a basic activity recognition scenario in 3. An ontology-based data collection approach is used to clarify ambiguities, by exploiting data exchanges between the two environment representations.

III. System Design and Knowledge Representation

The system of systems is composed of a robot and an ambient system. Both interact and collaborate with each other in order to fulfill a proactive or human-requested task. In this section, we describe the way the environment representation is stored and shared.

A. Architecture

In our work, the robot and the ambient system, have their own perception environment. The global architecture is detailed in the figure 1. Software wrappers have been developed under the ROS middleware for each family of sensors integrated. They process received data (such as images, low level numerical values, etc.) and extract similar features (i.e. face detection and recognition, location of a detected movement, etc.) regardless of the platform on which they are integrated. A data manager software is used to update the ontology and manage exchanged data (*Request*) with the other system using a simple TCP/IP communication over WiFi. The server/client implemented in the data manager

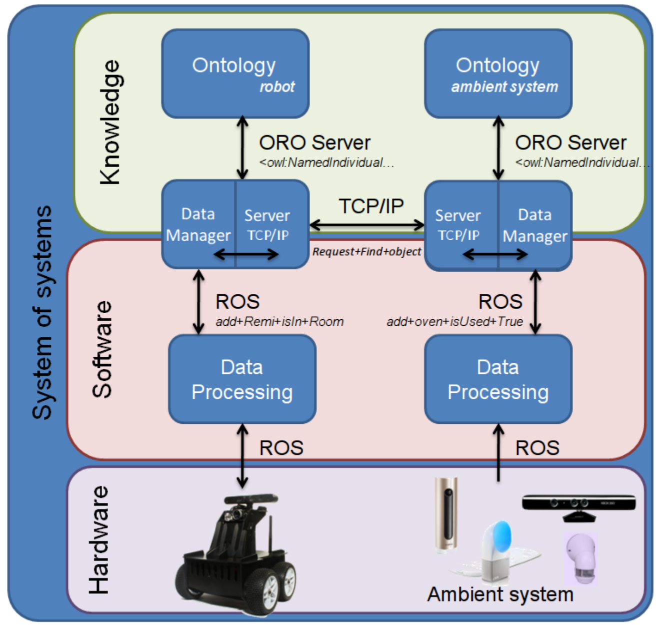

Fig. 1. Overview of the architecture

exchanges requests and data between the two systems. When a server receives a request, it transfers the received payload to the data manager by publishing in the adequate topic. Then the demand is processed and the answer (which contains information) is relayed to the client via the server.
Any extracted information is stored semantically in an ontology using the open source ORO platform [17]. "ORO relies on a dialect of RDF, OWL Description Logic, which is the decidable part of OWL". ORO functionalities include to insert (*Add*) or delete (*Remove*) facts (i.e. RDF[1] triples), to look up for concepts (*Find*) and check consistency: each time a data is added in an ontology, ORO checks if an inconsistency occurs. The communication protocol between ORO and the data manager uses another TCP/IP protocol, the ontologies are updated by executing standard SPARQL queries from ORO. The key benefits of this framework are 1) the simplification of expression statement in the ontology compared to the SPARQL queries. 2) Ontology reasoning with Pellet [18] on the high level data collection combined with the predefined properties, builds up the corresponding system environment representation in real time. So when a statement is added in the ontology using ORO Server, the ontology consistency is checked each time: a safe "add" option, provided by ORO, allows to safely add a statement in the ontology and check the consistency using Pellet. If the resulting ontology is inconsistent, ORO removes this statement and send back an error message. When a system adds an information in the ontology via ORO, this safe "add" command is employed. If an error is raised, a clarification process is initiated.
The mobile robot uses the ROS navigation stack for autonomous path planning and Simultaneous Location And

[1]Resource Description Framework, https://www.w3.org/RDF/

Mapping (SLAM). Before the scenario, a map of the apartment is created using the map service proposed by Xaxxon and implemented on ROS. Each room and point of interest is manually labeled on the map as a "waypoint" which corresponds to a Cartesian position on the map. Moving to a specific destination, such as a table, corresponds to navigate to the specified waypoint.

The ambient system is composed of industrial sensors: Infrared sensors, used as movement detection to indicate active human presence in a room (*unknown_human1 isIn Bedroom*) and power consumption sensors, to specify if an electric device (oven, microwave, TV) is turned on (*kitchen hasAnActivated oven*) both from Legrand, a Welcome camera, from Netatmo able to detect and recognize faces. Data are retrieved and updated in the ontology every minute (*human1 isIn livingRoom*), some Kinect devices from Microsoft used as RGB cameras for color detection and face recognition (*human1 isIn livingRoom, redObject isIn bedroom*), a sleeping connected object "Aura" from Withings indicates only if someone is sleeping every five minutes (*unknown_human1 isSleepingIn bedroom*). Each sensors were provided with an API. ROS wrappers were developped to be able to update and nourish the ontology.

The mobile robot is an Oculus Prime platform from Xaxxon. This robot is composed of a microphone, an RGB camera, a loudspeaker and an Orbbec Astra depth camera.

In both systems, a new sensor or connected object with a provided Application Programming Interface (API) can be easily included just by developing a ROS wrapper to ensure data compatibility in the ontology.

B. Knowledge Representation

Data representation can be done at two levels. Low level data (signals) reflect the sensed environment and can only be compared to an alike low level data (a sound can only be compared to another sound to recognize it). Robotic and ambient systems are limited when a low level data comparison is processed: stored data have to come from the same kind of sensors which implies that both systems share the same sensors architecture. High level data (here semantic data) represents a normalized information, i.e. presence of a recognized human, here the human recognition could have been done using a voice recognition, a face recognition or any human recognition method and thus, does not depend on the sensors, thereby the ambient system and the robot does not need to share the same kind of sensors. We chose to collect semantic high level data in an ontology to be able, using its already defined properties, to process a more complex information. A more complex information is an imbrication of several high level data i.e. the activity *cooking* can inferred from the information of the activation of the oven and the presence of someone in the kitchen.

In this subsection, a formalization of the environment representation is done for the robot and the ambient system using semantic, regardless from which sensor the information is extracted.

1) Classes, Properties and inference: The ontology used is slightly inspired from the *OpenRobots Common Sense Ontology* [17]. Classes represent the basic concepts of the apartment environment *Objects* (*Bed*, *Oven*, etc.), *Sensors* (*Camera*, *Microphone*, etc.) or *Systems* (*Robot* or *Ambient system*). Spatial properties are defined and spatial chained properties are set up (i.e. a chain of object properties between *isIn* and *isNextTo*: *human1 isNextTo Bed*, *Bed isIn Bedroom* implies *human1 isIn Bedroom*).

Activity recognition is a more complex high level of data. They are established as a triplet [Human hasActivityIn Location] in the ontology and need to be inferred from other collected data and a chain of object properties (i.e. the two statement "*kitchen hasActivated oven*" and "*human1 isIn kitchen*" infer the next triplet *human1 isCookingIn kitchen*; or if the human has not been recognized for the moment but sensors detects someone is sleeping in the bed: *unknown_human1 isSleepingIn bedroom*.

2) Spatial Knowledge: When two humans exchange information about the localization of an object, relative position are employed "on the living room table", "close to the computer", etc. When two systems exchange information about the localization of an object, absolute and relative position in a space can be employed when the same perspective is shared. However ambiguities arise when a system cannot take the perspective from another point of view, i.e. an overhead camera has not the same point of view as an embedded camera on a robot. Here, some specific objects, which are not supposed to move daily (bed, tables, sofa, etc.), are set up in every ontology during initialization. When exchanging spatial knowledge, a global perspective point of view is adopted and relative properties are used:

[Object] isIn [Room]. An object or a human is detected in a specific room. This object property is functional "for each individual x, there can be at most one distinct individual y such that x is connected by *isIn* to y"[2]. In other words, a first statement "*human1 isIn aRoom*" and the statement "*human1 isIn anotherRoom*" will lead to an inconsistency of the ontology so that, *human1* can only be in on room at a time.

[Object] isOn [Object]. An object, or a human, is on another object when a sensor detects the first one where a stationary known object should be (i.e. a bed, a sofa, etc.).

[Object] isNextTo [Object]. An object, or a human, is next to another human or object when the applied metrics between them is small.

3) Ontology Fusion: During the achievement of a task by a system, inconsistencies or uncertainties may occur during the feature extraction. In this paper, to clarify a situation means to detect that an inconsistency or an uncertainty occurred in a system. The system in question requests to the other to complete or correct the raised problem by sharing knowledge about a set of data. Here, this interaction, which consists of an exchange of data between the two systems, can be done if each system's environment representation

[2]See www.w3.org/TR/owl2-syntax/#Functional_Object_Properties

is semantically stored in its ontology and if necessary sets of data are shared during a clarification process. In term of software implementation, when a system request to the other an information, the request is immediately treated: the information is sought in the system ontology and if found, sent back to the requester. If not found, the ambient system will answer that "*nothing is found*" whereas the robot will seek for the missing information. An information always depends on a location, i.e. "Is there someone in the bed ?". For each location there is a corresponding waypoint on the robot's map so that the robot can navigate to this point autonomously. For every system, when a new request is received during the process of the previous one, it is put in a queue (FIFO), and will be put out when the previous request will be completely treated.

IV. Use Case Scenarios

In the previous section, we proposed an architecture allowing two systems to represent their environment. This paper focuses on a smart environment working with a mobile robot to achieve tasks they cannot do alone. This section analyzes two main situations where interactions between the systems are mandatory: a system seeks for a missing information in its ontology ; a system adds an information in its ontology which leads to an inconsistency. We present two scenarios selected to address these issues and report the resulting interactions. Figures 2 and 3 describe the interactions between the ambient system and the robot that occur when one of the system cannot find any required information to fulfill a task.

A. Scenario 1: Cooperation between the ambient system and the robot to find an object

In this scenario the cooperation between the ambient system and the robot is tested in a task where a human requests and asks to the robot to bring him/her to a specific object. Here, the robot and the ambient systems does not have any sensor alike (color detection is only integrated in the ambient system), moreover, the robot initially has no information on the object and no way to get some by itself. So, both systems must collaborate, by exchanging information, to find the object and reach the goal. Those interaction are totally transparent for the user which does not know if the robot is able or not to achieve alone the requested task.

The first step, as described in the figure 2, consists to launch the system and set up the goal *"Find and reach the red object"* which is, here a red-colored object. This object is on a table in sight range for the ambient system's RGB camera. In the second step, after initialization, the ambient system, using its RGB camera, extracts the relative location of the asked red object :(*Object isIn bedroom* and *Object isOn table*) then pushes corresponding data into its ontology. The robot, pulls everything about the red object from its ontology. As the robot has no cameras and cannot find it using its other sensors, nothing is found (cameras used for color detection purpose are only integrated in the ambient system; by default the robot has its own depth cameras only used, here, for a navigation purpose). As no information of the object was found in the robot's ontology, it requests information about the object to the ambient system. From the ambient system point of view, as the robot is not seen by the camera, giving him the object's absolute coordinates would mean nothing for it as the perspective of the two systems are different and thus, the necessary Cartesian system is not common for the two systems. However, the room and position of some objects like the table have been set up initially in each system. So, stored data (*Object isIn bedroom* and *Object isOn table*) in the ambient system's ontology can be exchanged and directly stored in the robot ontology as such.

The data just harvested by the ambient system are transferred to the robot which, in turn, updates its ontology. At this point, the two ontologies contain the same data concerning the object. The robot is now able to locate the object and navigates as close as possible to its position on the table, so as close as possible to the table. Finally, when the table is reached, the robot interacts with the human by indicating *"The object is here"* with its loudspeakers.

B. Second scenario: Activity recognition

In this scenario the ambient system and the robot try to recognize humans activities in an apartment. The experiment takes place in a domestic environment where three people have an activity: the first one (H1) is sleeping in the bedroom, the second one (H2) is in the sofa watching TV, the third one (H3) is in the kitchen cooking. Following ethical rules, cameras are not installed in rooms where privacy should be preserved. Thus, cameras and microphone, including those which equip the robot, are banned in the bedroom, toilets and the bathroom.

For each human, three data are permanently assessed by the system of system {*Identity; Location; Activity*}. As previously mentioned, the objective for the two systems is to collaborate to recognize activities and to have in the ontology: *H1 isSleepingIn bedroom*, *H2 isWatchingTVIn livingRoom*, *H3 isCookingIn kitchen*. However, this goal cannot be reached as the sleeping human cannot be identified due to the ethical rule.

Moreover, at the beginning of the experiment in the ambient system, H2 and H3 are voluntarily mixed up in order to observe an ambiguous situation where one person is detected in two different rooms.

The ambient system is composed of two RGB cameras (one set up in the living room, one set up in the kitchen), a power consumption sensor and the Aura sensor in the bed. The robot is now equipped with an RGB camera for face detection and recognition and a depth camera for its navigation system.

First, as detailed in the initialization part in the figure 3, the ambient systems detects: a) An unidentified human is sleeping in the bedroom (*unknown_human1 isSleepingIn bed*), b) an identified human is cooking in the kitchen using the oven (*kitchen hasActivated oven*, *human1 isIn kitchen*), c) an identified human is watching tv in the living room (*livingRoom containsActivated tv*, *human1 isIn livingRoom*).

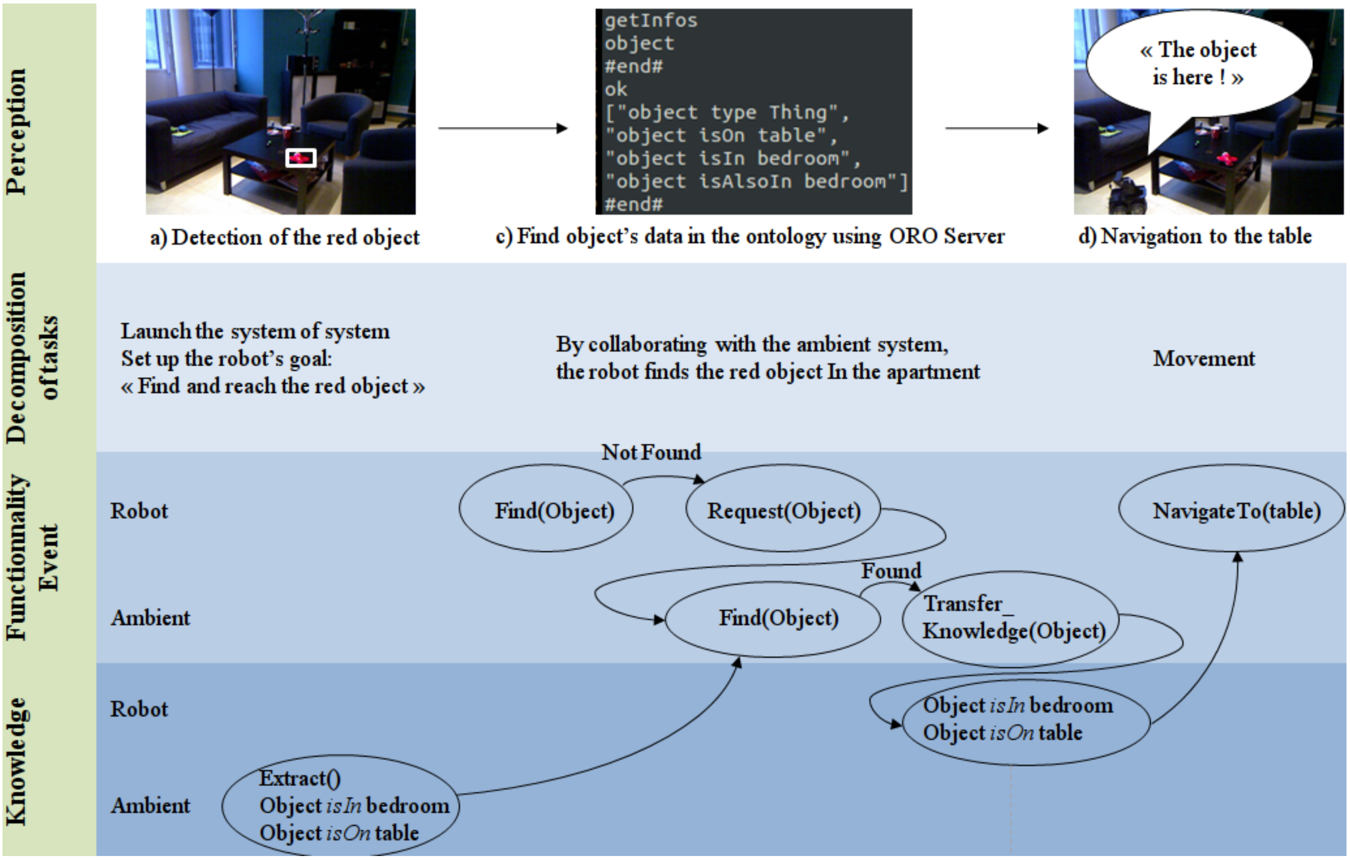

Fig. 2. Global overview of the system of systems knowledge transfer: Scenario where the robot work in with the ambient system to help a human to find a red-colored object in the apartment

Voluntarily, the two humans in the living room and in the kitchen are the same. So, when the system adds the last statement (*human1 isIn livingRoom*) in its ontology, as someone cannot be in two different places at the same moment (see the description of *isIn* in section III-B.2 for more information), an alarm is triggered due to the inconsistency and do not add this last statement in the ontology.

Finally the ambient system requests the robot to check every data [*?who isIn livingRoom*] and [*?who isIn kitchen*]. In other words, the ambient system requests the robot to share every information concerning who is in the kitchen and in the living room.

As the robot's ontology is initially empty, it navigates from its position to the living room and the kitchen (in order of request) to check the human identity. Then, using its loudspeaker, it asks to the person to look at him to be able to see the human's face and be able to recognize him. When the recognition process ends, the robot stores every extracted data in its ontology before sending them back to the ambient system. The same process is repeated again for the next request from the ambient system.

As the robot is in an "answering to request" mode, if an error occurs during this scenario and the robot's ontology is also inconsistent (e.g. one more time, if the same human is recognized in two different room), the robot will not request to the ambient to share data with him and will not send the wrong statement but a *not found* answer instead without updating its ontology.

After the two identifications have been done, the ambient systems checks every human triplet. As the sleeping human is not identified it asks to the robot to get this last information. The robot is banned from the bedroom, thus it sends back to the ambient system an *not allowed room* answer to let the ambient system knows about the rule concerning the bedroom. The ambient system will not request again an identification for the bedroom and thus, avoiding any future request for information about the human which is sleeping.

V. Conclusions and Perspectives

In this paper, we proposed an approach where a mobile robot and an ambient system platform collaborate to prevent wrong inferences from errors or inconsistencies from extracted data. The system of systems, designed to recognize activities of daily living, can use any kind of sensor to infer an activity as long as a software wrapper is implemented to ensure compatibility with the system. Features are extracted as high level data sent to an ontology for classification and reasoning. Activities are described in the ontology as new sets of object properties before activating the system.

While our system aims to recognize and analyze activities, only the approach used for clarification is described here. We scaled the validating tests corresponding to our needs (see

Initialization	
Ambient System	Add information: [*unknown_Human0 isSleepingIn bed*] [*kitchen hasAnActivated oven, human1 isIn kitchen*]: The Information [*human1 isCookingIn kitchen*] is inferred. [*livingRoom containsActivated tv, human1 isIn livingRoom*] leads to an inconsistency:
Clarification	
Ambient system	**request**(*[?who isIn livingRoom]*)
Robot	Receives the requests, **navigate**(*livingRooom*), when location is reached: **speech**(« *Can you look at me please ?* ») when the face is recognized add information: [*human1 isIn livingRoom*], then **sendback**[*human1 isIn livingRoom*]
Ambient System	Received data from the robot: [*human1 isIn livingRoom*] Everything corresponding to [*?who isIn livingRoom*] is removed in the ontology and the returned data is added: **add**([*human1 isIn livingRoom*]) **request**(*[?who isIn kitchen]*)
Robot	Receives the requests, **navigate**(*kitchen*), when location is reached: **speech**(« *Can you look at me please ?* ») when the face is recognized add information: [*human1 isIn kitchen*], then **sendback**[*human1 isIn kitchen*]
Ambient System	Received data from the robot: [*human1 isIn kitchen*] Everythingcorresponding to [*?who isIn kitchen*] is removed in the ontology and the returned data is added: **add**([*human1 isIn livingRoom*])

Fig. 3. Data based transcription of the interaction between the human, the robot and the ambient system, inconsistencies clarification

the two scenarios proposed in IV). However no evaluation like activity recognition analysis is done yet. Future works involve to use this platform to analyze ADL and IADL to assess the elderly's dependency. Several improvement are being investigated: Fuzzy ontologies approach to include more information about the data itself i.e. confidence coefficient, degree of truth, etc. Human Robot Interaction is also a possibility to clarify a situation. Being able to phrase a question at the end of the process when the global inference has a low confidence index.

ACKNOWLEDGMENTS

This work was funded within the framework of EBIOMED Chair part of IUIS (Institut Universitaire d'Ingénierie en Santé).

REFERENCES

[1] A. QUEIRS, et al, Usability, accessibility and ambient-assisted living: a systematic literature review. Universal Access in the Information Society, 2015, vol. 14, no 1, p. 57-66.

[2] JJ. KUFFNER, et al, Cloud-enabled robots, In: IEEE-RAS international conference on humanoid robotics, 2010, TN

[3] K. KAMEI, Nishio, et al, Cloud networked robotics, IEEE Network, 2012. 26(3):2834.

[4] N. Hu, Multi-user identification and efficient user approaching by fusing robot and ambient sensors, in Robotics and Automation (ICRA), 2014 IEEE International Conference on (pp. 5299-5306). IEEE.

[5] M. JAMSHIDI, (ed.). Systems of systems engineering: principles and applications, CRC press, 2008.

[6] R. ROS, Which one? grounding the referent based on efficient human-robot interaction, In : RO-MAN, 2010 IEEE. IEEE, 2010. p. 570-575.

[7] R. STEVENS, Why use an ontology?, 2013 Ontogenesis.

[8] E. Tapia, S. Munguia, et al. Activity recognition in the home using simple and ubiquitous sensors, In International Conference on Pervasive Computing, pp. 158-175. Springer Berlin Heidelberg, 2004.

[9] H. STORF, T. Kleinberger, et al. An event-driven approach to activity recognition in ambient assisted living, In European Conference on Ambient Intelligence, pp. 123-132. Springer Berlin Heidelberg, 2009.

[10] T. Van Kasteren, N. Athanasios, et al. Accurate activity recognition in a home setting, In Proceedings of the 10th international conference on Ubiquitous computing, pp. 1-9. ACM, 2008.

[11] L. Xia, I. Gori, et al. Robot-centric activity recognition from first-person rgb-d videos, In Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on. IEEE, 2015. p. 357-364.

[12] L. Chen, J. Hoey, C. D. Nugent, et al. Sensor-based activity recognition, IEEE Transactions on Systems, Man, and Cybernetics, 2012 Part C (Applications and Reviews), 42(6), 790-808.

[13] D. Riboni and C. Bettini, COSAR: hybrid reasoning for context-aware activity recognition, Personal and Ubiquitous Computing 15.3 (2011): 271-289.

[14] M. WAIBEL, et al. Roboearth, IEEE Robotics & Automation Magazine, 2011, vol. 18, no 2, p. 69-82.

[15] A. MANZI, et al. Design of a cloud robotic system to support senior citizens: The KuBo experience, Autonomous Robots, 2017, vol. 41, no 3, p. 699-709.

[16] G. AMATO, et al. Robotic ubiquitous cognitive ecology for smart homes, Journal of Intelligent & Robotic Systems, 2015, vol. 80, p. 57.

[17] S. Lemaignan, ORO, a knowledge management platform for cognitive architectures in robotics, in Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on. IEEE, 2010. p. 3548-3553.

[18] E. SIRIN, Pellet: A practical owl-dl reasoner, in Web Semantics: science, services and agents on the World Wide Web, 2007, vol. 5, no 2, p. 51-53.