Architecture and representation for handling dialogues in human-robot interactions

Eloy Retamino*†, Suraj Nair*‡, Aravindkumar Vijayalingam* and Alois Knoll*‡
*TUM-CREATE, Singapore.
†Institut de Robtica i Informtica Industrial, CSIC-UPC, Spain.
‡Robotics and Embedded System, Technische Universitat Munchen, Germany.

***Abstract*—Human-robot interaction is an important component for robots operating in human environments and verbal interaction is in many cases the most intuitive and effective solution for humans. Managing dialogues between physical agents interacting in a physical environment brings additional challenges to virtual dialogue systems (eg. Siri or Google Now). More channels of information are available, as gaze or hands movements, which can modify or support verbal information. Also exophoric references to different parts of the environment can occur along the conversation. In this article we focus on the problem of extending the representation of the dialogue context to a physical environment and using this representation for resolving exophoric references. We also describe an architecture for integrating an open source dialogue manager in a service robot. In this architecture, the aforementioned representation is jointly built by different modules and it's used by the dialogue manager to ground utterances happening in the conversation. Finally we describe several experiments performed for assessing the utility of this architecture with actual robots in physical scenarios.**

I. Introduction

There is a sustained effort on managing situated multi-modal interactions. We can summarize the additional challenges which arise in this sort of interactions as handling the extra channels of information available (eg. gaze, gestures, pointing) in comparison with purely verbal communication. That is, in the first place producing richer output using these extra channels, as for example synchronizing speech with eyes or arms movement or other actuators available. And in the second place understanding how information coming through these channels modifies verbal information. What is more, the environment itself plays a major role in situated interactions as there are constant references to objects contained in it. The meaning of spatial expressions (eg. at the end of, behind, near) is affected by the geometry of the environment and by the spatial configuration of the objects involved. All this non-verbal information plus the environment itself compound what it's known as context of situation. In opposition with the context of conversation, which refers to all the information explicitly communicated along the conversation. Robots interacting in physical environments should maintain these two contexts in order to correctly understand and ground human utterances.

In this work we present a representation of the context of situation which can be jointly built by different modules in the system. This representation, based on the concept of symbol anchoring [1], contains both geometrical and semantic information about the objects in the environment (including the human and the robot) which can be both used for enriching the output (eg. allow the robot to look at the human) and improving the interpretation of human utterances, in our case by resolving exophoric references over the representation. Furthermore, we present an architecture for building and maintaining this representation and how we have proceeded in order to integrate an open source dialogue manager – IrisTK [2] – in this architecture.

II. Related Work

As mentioned in the introduction, an increasing number of works are being focused on handling different aspects of situated multi-modal interactions. In a close direction to the one presented here, in [3] Iida et Al. analyzes the effect of gaze in the resolution of exophoric references. Also, in [4] Misu et Al. uses multi-modal inputs of speech, geo-location, gaze and dialog history to allow a dialogue system mounted on a vehicle to answer drivers' queries about their surroundings. The present work can complement these ones in which it proposes a representation which can be used to integrate information coming from different modalities and to better interpret human utterances.

III. Dialogue Manager

There are multiple dialogue managers available in the community (eg. TrindiKit [5], RavenClaw [6]). From among them we chose IrisTK [2]. There were several reasons for this decision: (1) it's actively maintained; (2) it's Java based and open source, which facilitated the integration in our architecture (based on the ROS[1] middleware in a Linux environment); (3) the representation used for modeling the state of the conversation is based on Harel statecharts [7], which we find very convenient as a middle term between pure FSM and Information State approaches, as TrindiKit.

IrisTK is comprised of a set of modules which communicate through an internal message passing system. We developed modules for speech recognition and speech synthesis which connect through ROS services with nodes already present in our architecture which provide these functionalities. The dialogue manager, which is another module in the IrisTK

[1] http://www.ros.org

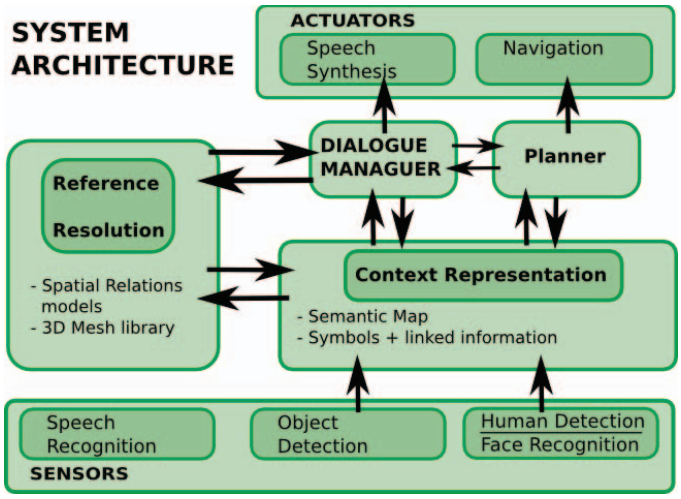

Fig. 1. System architecture. Arrows represent communication between modules. There should be an arrow from the Speech Recognition Module to the Dialogue Manager and a bidirectional arrow between the Planner and the Reference Resolution Module. All the communications are handled using ROS topics and services.

system, uses a variant of SCXML[2] language for modeling states and transitions (ie. the dialogue flow).

IrisTK is by default thought for being used in situated interactions. It comes with modules for performing human detection and face recognition, and it's possible to store a context of situation. This context will be conformed by an identifier for each human being detected, which allows for example to remember the information already told to that person. The pose of the human can be store as well. This is sufficient in simple scenarios in which, though the interaction is situated, there are no exophoric references. Also in more complex architectures in which several modules must jointly maintain the context (eg. object detection, robot localization, human detection, face recognition) keeping the representation within the dialogue manager is not an optimal approach.

IV. Context of Situation

The approach we had followed for representing the context of situation is based on the concept of symbol anchoring [1]. Symbols represent objects in the environment (including the robot itself and detected humans). Each sensor module in the system (eg. object detection, human detection, face recognition, robot localization) is in charge of asserting symbols in the representation and maintaining the "anchor" with the corresponding perceptual information. The way of maintaining the anchor differs from module to module. For example the face recognition module is responsible for checking the identity of detected humans and inferring if a detected human is already present in the representation or should be asserted. In the case of objects, the object detection module currently uses a very simple logic, considering that all the objects are static. Thus if the same object is detected in another location it will be considered a different one and asserted to the representation.

[2]http://www.w3.org/TR/scxml/

One advantage of keeping the representation agnostic to the characteristics of any concrete module is that new modules can be easily plugged in or their logic for maintaining the asserted symbols modified without affecting the rest of the architecture.

Information relevant to the interaction can be linked to any of the symbols in the representation by any module (including the dialogue manager). This information can be semantic or geometric. For example if the object detection module detects a mug, it would associate the label "mug" and the object pose to its symbol. If during the interaction with a human the dialogue manager learns that that concrete mug belongs to that human, that information can be linked as well to the symbol. It's easy to notice that before the dialogue manager can assert that a concrete mug belongs to the person the robot is interacting with, the exophoric reference to the mug in the conversation must be resolved first, ie. it's needed to find out which symbol in the representation corresponds to that mug. In the next section we describe how this process is accomplished.

The other element in representation of the context of situation (aside from object symbols) is a semantic map of the environment. This map consists of an occupancy grid with labeled regions and a topological map indicating connections between these regions [8]. The map is used for navigation and for reducing the context when resolving references. That is, if some region is mentioned in the reference, just objects detected inside this region are considered for the resolution (eg. "the mug in the kitchen"). Otherwise just the objects in the region where the interaction is happening are considered.

Reference Resolution

In the architecture there is a dedicated module for resolving references, the *Reference Resolution Module (RRM)*. Its core functionality is to assess spatial relations between objects in the context representation. We will define a spatial relation as a tuple $< reg, rel, obj1, obj2 >$. Where *rel* is one of the supported spatial prepositions (ie. near, far, left, right, behind, in front of, above, below, on) and *obj1*, *obj2* are symbols in the representation. *reg* is the label of a semantic region in the map. The assessment of spatial relations requires that every symbol has linked a pose and a class label. Each class label has associated a 3D mesh which, jointly with the object poses, is used in geometrical computations for the assessment. Each spatial relation has an associated computational model which given two 3D meshes and a semantic region returns a number between 0 and 1 proportional to the applicability of the relation for the two objects. A description of the used computational models can be found in [9]. The variables relevant for the evaluation of spatial relations in the models are the objects dimensions, their relative pose and the dimensions of the semantic region in which the objects are placed.

The RRM functionality for assessing spatial relations was exploited for two different purposes in the performed experiments. In the first case, given a spatial preposition, the class labels of two objects and a semantic region (eg. "kitchen, on, table, mug"), it's required to infer the identity of these two objects. That is, their symbols in the representation. In this

Fig. 2. a) Robotino playing a game with a human. b) Turtlebot searching for a box. c) Visualization of the detected objects in a moment of the search and the region the robot was exploring (red cells).

case the module will assess the spatial preposition for every pair of objects in the representation of the right classes and return those which obtained the highest applicability.

The second case is slightly different. Given a semantic region *reg*, an object *obj1* located inside of *reg* and a spatial preposition *rel*, the module is asked where an object *obj2* of class label *class* could placed in *reg* if the spatial relation$< reg, rel, obj1, obj2 >$ stands. In this case *obj1* is fixed and existence of obj2 is hypothesized. That is, the question posed to the RRM is 'if an object *obj2* of class label *class* was in region *reg* and the relation $< reg, rel, obj1, obj2 >$ was true, where *obj2* would be most probably located in *reg*?'. In this case the module assesses *rel* for an object of class label *class* being in every position of a 2D grid covering the region *reg*. It returns a 2D grid containing the result of the assessment for each position. This way of accessing the RRM is used by the planner in one of the experiments for guiding the search of an object based on semantic information.

V. Experiments

In this section are described the experiments performed for testing the present representation and architecture. Though a larger amount of data will be required in order to extract well supported conclusions from these experiments, we believe it is worth to include in this article a description of them and the partial results obtained in order to show the utility of the architecture and how it can fulfill its function of handling interactions in situated environment and resolving exophoric references. To support this conclusion we provide links to videos showing runs of the experiments on actual robots[3].

Two sets of experiments were performed. The first of them was focused on assessing the dialogue manager performance and the ability of the robot for engaging people into conversation. The other one was focused on testing the performance of the Reference Resolution Module and its utility for resolving exophoric references.

The Game Master Robot

This experiment was run using a Festo Robotino 3 platform in the installations of TUM-CREATE in Singapore. Pocket Sphinx[4] was used for performing ASR and CereVoice[5] for TTS. In order to make the interactions more natural, a microphone mounted on the robot platform was used for capturing voice. On top of the platform we mounted a tablet showing the virtual face showed in Fig 2a. It was possible to control the eye direction and the lips were synced with the speech.

The robot was continually navigating along the corridors of the office looking for humans to interact with. All the participants in the experiments were workers of the company uninstructed and even unaware that an experiment was being run. In the experiment, when a human is detected, the *face recognition module* is activated for deciding if the human is already stored in the representation or if a new symbol must be added. The *human detection module* links a pose to the symbol representing the human and continuously updates it. If the human is not known, the dialogue manager is called with the goal of asking his name. Once the human tells his name, it's linked to the symbol and is used in future interactions to personalize the conversation and letting the human know that the robot remembers him. If the human is known, the robot tries to engage him to play a simple game that involves the human guessing a number between 1 and 10. Every time the human tells a number the robot gives him feedback by telling him if it was lower or higher than the correct number. After the interaction, the number of times the human needed to guess the number is linked to his symbol. Also if during the interaction the human indicated that he was bored or didn't want to play, this information is also linked to the symbol. The next time the human is found this information is used to recall how many guesses he needed last time or for not interacting with him at all. An example illustration is shown in Fig 2a.

The human and robot poses are used to make the robot eyes look at the human. This action enforces the impression that the robot is aware of the human and that it's attending him. This in turn, along with the appealing interface, seems to have an effect in catching the human attention and therefore improving the quality of the interaction. However more experimental data would be needed in order to assess this effect.

The main source of misunderstanding during the interaction

[3]Game master robot: *https://youtu.be/6Au5u5K4Pt4*. Look for an object: *https://youtu.be/ovtoHcvGsjk*

[4]http://www.speech.cs.cmu.edu/pocketsphinx/

[5]https://www.cereproc.com

was the ASR module. As the microphone was installed on the robot platform, Pocket Sphinx performance was limited. With a high frequency when the human told one number, Pocket Sphinx recognized a different one. This made the robot feedback inconsistence and made people interacting with him to want to finish the game before guessing the number.

The Look for an Object Scenario

This experiment was run using a turtlebot in the Institute of Industrial Robotics (CSIC-UPC) in Barcelona, Spain. In this case human utterances were communicated to the robot using a keyboard. The Stanford CoreNLP Toolkit[6] was used to extract spatial relations from the input sentences. Object detection was performed using Artags and the Alvar detector[7].

In the experiments, the robot goal was to find an object of a certain class. The object could be anywhere inside the environment which was limited to a single room. Before starting the search the robot asked for information which could help it to find the object. This information consisted of a description of the object location formed by a series of spatial relations with other objects in the environment. The participants of the experiment were told about this fact and also about which spatial relations they could use. The description was used by the robot for performing indirect search. That is, whenever an object was detected, it was checked if an object of its class was related to the goal object in the description. If that was the case the reference resolution module was asked for the region in which the searched object could be. The returned grid was binarized using a threshold (usually of 0.7). The points over the threshold were explored. When there was no regions to explore, the robot performed a random exploration of the environment. This process continued until the object was found or the whole environment was explored.

Fig 2b corresponds to a case in which the robot was looking for a box. The description in this case was: "the box is near a table. the box is close to a bin. the box is close to a second bin". In Fig 2c is shown a visualization of the region the robot was exploring after detecting a table and two bins.

VI. CONCLUSIONS AND FUTURE WORK

Human-Robot interaction is a crucial issue in order to seamless integrate service robots in human environments. This interaction can be performed using different interfaces as text or graphical UIs. Though this interfaces may suffice, and in some cases optimize the interaction, for giving a robot simple commands or request encyclopedic information, complete and engaging interactions would require the ability of holding verbal conversations. In the present article we have described our effort for dealing with one of the problems specific to dialogues situated in physical environment, the representation of the context of situation and exophoric reference resolution upon of it. The representation combines semantic and geometrical information necessary for resolving the references. The modular structure of the proposed architecture allows different modules to assert and maintain specific kind of symbols into the representation. As semantic information can be linked to the symbols, the dialogue manager can easily store relevant information learned through conversations with humans and retrieve it in future interactions.

One limitation of the dialogue management in our current architecture is that just purely exophoric references had been considered. For handling actual conversations the inference mechanisms exposed along the article should be used along with algorithms for resolving anaphoric expressions (eg. resolving pronouns). Other way for improving the architecture would be using ontologies for storing the context representation. Ontologies, aside from providing a more structured frame for storing symbols and relations between them, allows to perform inferences and detect inconsistencies in the stored knowledge. Other issue which has been left aside in the article is the use of clarification dialogues for resolving ambiguities in the references. That is, after evaluating possible grounding for a reference, more than one object can have a high value of applicability. In these cases we just take the object with the highest one. The correct way to proceed would be to call the dialogue manager for asking the human relevant questions which allow to disambiguate the reference.

[6]http://nlp.stanford.edu/software/corenlp.shtml

[7]http://virtual.vtt.fi/virtual/proj2/multimedia/index.html

Acknowledgments

This work was partially funded by the Singapore National Research Foundation under its Campus for Research Excellence And Technological Enterprise (CREATE) programme and by the Spanish Ministry of Economy and Competitiveness under project TaskCoop DPI2010-17112.

REFERENCES

[1] S. Coradeschi and A. Saffiotti, "An introduction to the anchoring problem.," *Robotics and Autonomous Systems*, vol. 43, no. 2-3, pp. 85–96, 2003.

[2] G. Skantze and S. A. Moubayed, "Iristk: a statechart-based toolkit for multi-party face-to-face interaction.," in *ICMI* (L.-P. Morency, D. Bohus, H. K. Aghajan, J. Cassell, A. Nijholt, and J. Epps, eds.), pp. 69–76, ACM, 2012.

[3] R. Iida, M. Yasuhara, and T. Tokunaga, "Multi-modal reference resolution in situated dialogue by integrating linguistic and extra-linguistic clues," in *Proceedings of 5th International Joint Conference on Natural Language Processing*, 2011.

[4] T. Misu, A. Raux, I. Lane, J. Devassy, and R. Gupta, "Situated multimodal dialog system in vehicles," in *Proceedings of the 6th Workshop on Eye Gaze in Intelligent Human Machine Interaction: Gaze in Multimodal Interaction*, GazeIn '13, (New York, NY, USA), pp. 25–28, ACM, 2013.

[5] S. Larsson and D. Traum, "Information state and dialogue management in the trindi dialogue move engine toolkit," *Natural Language Engineering*, vol. 6, pp. 323–340, 2000.

[6] D. Bohus and A. I. Rudnicky, "Ravenclaw: dialog management using hierarchical task decomposition and an expectation agenda.," in *INTERSPEECH*, ISCA, 2003.

[7] D. Harel, "Statecharts: A visual formalism for complex systems," *Sci. Comput. Program.*, vol. 8, pp. 231–274, June 1987.

[8] H. Zender and G.-J. Kruijff, "Multi-layered conceptual spatial mapping for autonomous mobile robots," in *Papers from the AAAI Spring Symposium on Control Mechanisms for Spatial Knowledge Processing in Cognitive / Intelligent Systems* (H. Schultheis, T. Barkowsky, B. Kuipers, and B. Hommel, eds.), no. SS-07-01 in Papers from the AAAI Spring Symposium, (Menlo Park, CA, USA), pp. 62–66, AAAI, AAAI Press, 3 2007.

[9] E. R. Carrión and A. Sanfeliu, "Human-robot collaborative scene mapping from relational descriptions," in *ROBOT (1)*, pp. 331–346, 2013.