Toward a Context-Aware Human–Robot Interaction Framework Based on Cognitive Development

João Quintas, Gonçalo S. Martins, Luis Santos, Paulo Menezes, and Jorge Dias

***Abstract*—The purpose of this paper was to understand how an agent's performance is affected when interaction workflows are incorporated in its information model and decision-making process. Our expectation was that this incorporation could reduce errors and faults on agent's operation, improving its interaction performance. We based this expectation on the existing challenges in designing and implementing artificial social agents, where an approach based on predefined user scenarios and action scripts is insufficient to account for uncertainty in perception or unclear expectations from the user. Therefore, we developed a framework that captures the expected behavior of the agent into descriptive scenarios and then translated these into the agent's information model and used the resulting representation in probabilistic planning and decision making to control interaction. Our results indicated an improvement in terms of specificity while maintaining precision and recall, suggesting that the hypothesis being proposed in our approach is plausible. We believe the presented framework will contribute to the field of cognitive robotics, e.g., by improving the usability of artificial social companions, thus overcoming the limitations imposed by approaches that use predefined static models for an agent's behavior resulting in non-natural interaction.**

***Index Terms*—Active assisted living, adaptive systems, cloud robotics, context awareness, decision systems, human–machine systems, interaction design.**

Fig. 1. CaMeLi and GrowMeUp systems being tested by older adults.

I. Introduction

In our recent works [1]–[5], we designed and implemented two different approaches of artificial social companions (ASCs). Both approaches aimed to operate as assistive technology in real-world indoor environments. Their primary mission was to help older adults in managing activities of their daily life and staying connected with their social circle. In CaMeLi,[1] we designed and implemented a virtual partner capable of showing a variety of dialogues and a wide spectrum of animated facial expressions while providing a set of services to answer the user's needs/requests. Moreover, in GrowMeUp,[2] we designed and implemented a service robotic system able to learn an older person's needs and habits over time and enhance (grow up/scale up) its functionality to compensate for the degradation of their abilities and to support, encourage, and engage the older persons to stay active, independent and socially involved while carrying out their daily lives at home. In both systems, we implemented various degrees of human interaction and autonomy that aimed to perform cognitive-like functions and accomplish real-time goals in terms of interaction and self-sufficiency. In other words, the user and agent could interact through multiple modalities, which included speech commands, gestures, touch screen, or other modes. In both cases, we dedicated most of our efforts to developing perception capabilities, user interfaces, and the

Manuscript received November 12, 2017; revised February 14, 2018; accepted April 30, 2018. This work was supported by the GrowMeUp Project through the European Commission within the H2020-PHC-2014 under Grant 643647. This paper was recommended by Associate Editor A. Hussain. *(Corresponding author: João Quintas.)*

J. Quintas is with the Laboratory for Automation and Systems, Instituto Pedro Nunes, 3030-199 Coimbra, Portugal, and also with the Department of Electrical and Computer Engineering, Institute of Systems and Robotics, University of Coimbra, 3000-214 Coimbra, Portugal (e-mail: jquintas@ipn.pt).

G. S. Martins, L. Santos, and P. Menezes are with the Department of Electrical and Computer Engineering, Institute of Systems and Robotics, University of Coimbra, 3000-214 Coimbra, Portugal (e-mail: gmartins@isr.uc.pt; luis@isr.uc.pt; pm@deec.uc.pt).

J. Dias is with the Department of Electrical and Computer Engineering, Institute of Systems and Robotics, University of Coimbra, 3000-214 Coimbra, Portugal, and also with the Robotics Institute, Khalifa University, Abu Dhabi 127788, UAE (e-mail: jorge@deec.uc.pt).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2018.2833384

[1] http://www.cameli.eu/
[2] http://www.growmeup.eu/

integration of these core components that resulted in two fully functional systems (Fig. 1).

Although some challenges still remain in the design and implementation of interaction workflows, involving users in the design process allowed us to tailor the systems functionalities to their needs and requirements. However, implementing interactivity based on predefined user scenarios and action scripts is not sufficient to take into account the uncertainty associated with noisy inputs, variation in the conditions of the operating environment, or unclear expectations from the user. It is not realistic to expect that users always use the same interaction patterns and never commit a mistake or that environment conditions remain unchanged.

We believe that part of the solution to this problem is to incorporate redundancy and fall-back strategies in terms of interaction functionalities, resulting in the agent's self-adaptation to its context (e.g., user model and environment conditions).

Based on this background, this paper was performed aiming at understanding how an agent's performance is affected when interaction workflows are incorporated in its information model and decision-making process. Therefore, we developed a framework that captures the expected behavior of the agent into descriptive scenarios, translates these into the agent's information model and uses the resulting representation in probabilistic planning and decision making to control interaction.

Our expectation was that adopting this framework could reduce errors and faults in the agent's operation, resulting in improved performance while interacting with the user.

Other researchers dedicated their efforts to developing approaches that address the challenge of adapting an agent's interaction to the user and involving context; we summarize their efforts in the next section.

A. Related Work on Adaptive Artificial Social Companions

Regarding related work on adaptive ASCs, the approaches presented in [1] and [6]–[8] discuss the fundamentals of context-aware adaption for cyber-physical systems (including virtual and robotic agents). In these works, the focus was mainly on context features related to the conditions of the environment that characterize the situation, where the interaction between the agent and the user occurs. Alami *et al.* [9] discussed a decisional framework for human–robot interactive task achievement that aimed to allow the robot to produce behaviors that support its engagement vis-a-vis its human partner and to interpret human behaviors and intentions. On the other hand, state-of-the-art interaction models similar to that proposed by Sili *et al.* [10] typically refer to some degrees of adaptation for the customization of multimodal user interfaces, but explicit models must be provided to rule out the behavior of the system. Devin *et al.* [11] summarized the essential building blocks to design architecture for cognitive and interactive robots. The concepts presented may be generalized for human–machine systems overall.

Regarding user-adaptiveness, we have found, essentially, three types of user-adaptive ASC systems: systems that adapt without explicit knowledge about the user, systems that keep a static user model and systems that keep a dynamic user model. The works presented in [12] and [13] do not maintain an explicit model of the user. Instead, these systems achieve the user-adaptiveness as a collateral effect of their main goal. In fact, the system of [12] adapts to the user by monitoring accessible areas for vacuuming, and that of [13] adapts to the user by estimating their intention in the cooperative task of selecting ingredients for a recipe. Static user models, such as those in [14]–[16] can also be used for adaptation. These systems make use of immutable information on the user, such as their persona [14], [17], personality [15], and physical capabilities [16], to generate adapted behavior. The unchanging nature of the user models employed do not allow for systems to gain information on the user from direct interaction and hinder interaction to naturally fleeting characteristics of the user, such as their mood. Despite the lack of dynamism in the user model, these systems are very successful at adapting to these wider, unchanging traits of the user and achieve interesting results in their specific applications. Dynamic user models, such as those found in [18]–[20], can be used to adapt the system's behavior to the user's dynamic characteristics, thus achieving higher levels of adaptivity and potential interaction quality. The dynamic nature of the user model allows the system to learn from the user *in loco* while the interaction is taking place. Systems of this nature have been applied to strict human–robot interaction (HRI), such as in [19], or in roboticized versions of classical human–computer interaction problems, such as learning assistance for children [18] and in robotic recommender systems [21].

In a similar manner, other authors developed approaches dedicated to generating task-oriented interactions of service robots or attempted to modeling the duration of the user interest during interaction with an artificial agent. Kim and Yoon [22] defended that "to obtain appropriate human aid for conducting tasks, a robot should be capable of generating meaningful questions regarding the task procedures in real time and applying the results to modify its task plans or behaviors." They concluded that few studies addressed the integration of robot task management and HRI in high-level task planning. For that purpose, they proposed a script-based scheme for task planning and HRI that supported the planning and is generated by it. Zhang *et al.* [23] proposed a hidden semi-Markov model to track the change of users' interests. They were motivated by the observation that "users' preferences often change over time" but "most existing approaches that address time information remain primitive," thereby justifying their use of a probabilistic approach. Another application example, by Cheng *et al.* [24], proposed a semantic Web-based context ontological reasoning service for multimedia conferencing process management that automatically selected the appropriate means of notifications based on the conference time and the participant contact details. This last example demonstrates the relevance of research on context-based interaction approaches for improving automatic intelligent systems.

B. Summary

Based on the information from our survey and our previous work [6]–[8], [25], we found that implementing realistic interaction workflows on interactive agents (e.g., virtual assistants and social robots) is extremely challenging because of the limitations imposed by the selected system architecture and the availability of perception features that operate correctly in multiple environments (i.e., contexts). On the other hand, the dominant approach used to capture the user's perspective regarding the expected behavior of the agent is based on describing different situations in the form of static user scenarios and is unable to describe all possible variability in operation conditions. In the following sections, we present, in Section II, the conceptual approach for the context-aware HRI (CAHRI) framework. Details on implementation aspects are presented in Section III. We present and discuss our experimental results in Section IV. Section V concludes this paper summarizing our major findings and identifying relevant topics for future work.

II. Context-Aware Human–Robot Interaction Framework

Our understanding of ASCs is tightly coupled with their capacity for self-development over time. We consider that an ASC should be capable of including new information into its knowledge base for later use when interacting with new users or operating in new contexts (i.e., "learning new things" during its life cycle).

To achieve self-development over time, we require a strategy that can translate the user's descriptions into knowledge representation and apply this knowledge during the agent's operation in such a manner that ensures an expected functionality in the face of context changes.

Thus, we examined current cognitive development theories for inspiration on how to design our framework to allow for the artificial agent to develop similarly to the human mind or at least take into consideration some basic factors (e.g., scalability of the knowledge representation, representation of the user model, and representation of context model).

A. Cognitive Development Theories

Troadec and Martinot summarized the last two decades of cognitive development theories in [26]. The overall conclusion is that the study of the mind suffered a shift from the classical conception as rational, abstract, universal, central, nonbiological, a-historic, emotionless, asocial to a new conception of the mind as positioned, framed by real time, guided by daily routines, and culture-dependent. In summary, cognition is now thought to be context-dependent and strictly related to biologic principles. Further, in their book, we found three main models for context-dependent cognitive development: 1) *the developmental niche* from Charles Super and Sara Harkness; 2) the *ecocultural theory* from John Berry; and 3) the *ecological model* from Urie Bronfenbrenner. The first two models are more focused on systemic approaches to the influence of cultural and societal aspects on the mental development of the individual. Hence, context is defined in

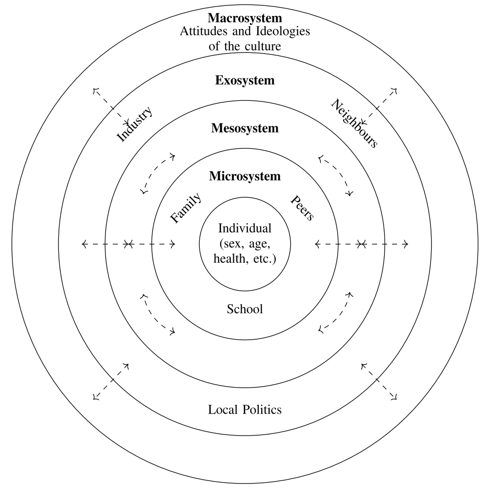


Fig. 2. Ecological model of development.

terms of cultural variables. The third model involves the entire ecological system in which development occurs, including the biological and genetic aspects of the person (Fig. 2). Other theoretical approaches and authors, such as Hoc [27], have proposed models of human cognitive activity especially in the case of human–machine cooperation. Moreover, as extended by Pacaux-Lemoine and Itoh [28], they have also distinguished between the interactions of one agent with another agent and the interaction with the context (more related to the task they have to perform to control a situation). Such models are also useful in distinguishing different levels of activity and different levels of information, with the goal of achieving a reliable and robust interaction workflow. From this concise overview, we may assume the following.

1) An ASC must also be capable of context-dependent development (i.e., its perception and knowledge representation must take into account how to represent a context model).
2) The ecological model better fits the design and implementation of ASC because it allows us to conceptualize the individual (represented by our user model) and its relationship with different contexts (represented by our knowledge model).

B. Proposed Architecture

Inspired by the previous model, we propose a CAHRI framework consisting of three major blocks (Fig. 3): 1) decision process; 2) knowledge model (i.e., upper ontology and scenario ontology); and 3) user model.

1) Decision Process: In command-driven approaches, we explicitly describe the protocols for interaction, which impose a limitation on the agents' interactivity (i.e., it will only execute predefined rigidÎ interaction patterns). CAHRI aims to overcome such limitation by using a knowledge representation that allows us to represent known interaction plans in the form of asserted graphs, which can be completed as the agent infers new relationships in data (e.g., using a reasoner). This information will be later used in the decision process, which, when formulated as a probabilistic graphical model, simplifies

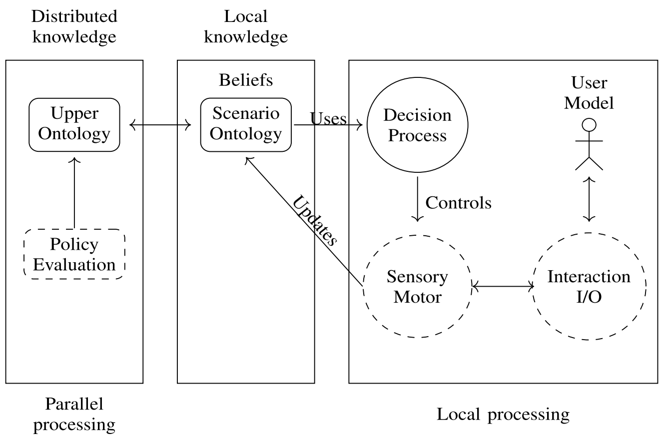


Fig. 3. Overall system architecture.

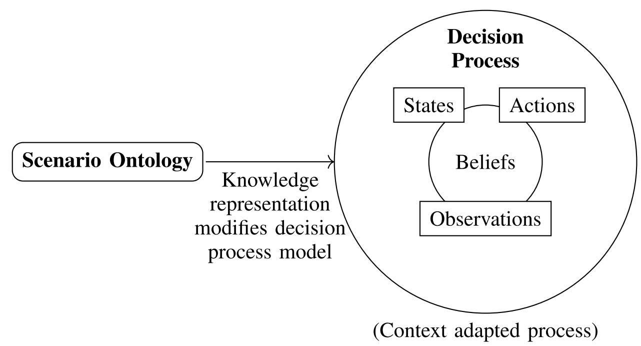


Fig. 4. Context representation in the scenario ontology modifies the components of the decision process at each moment in time.

the integration between knowledge representation and execution. Moreover, the decision process can add complementary information about the interaction protocols by determining the likelihood of certain interaction workflows [i.e., policies in a partially observable Markov decision processes (POMDPs)] to occur.

As described in [7], the mathematical formalism of POMDP is well-suited to our problem because we require an approach that takes into consideration aspects regarding limitations in *a priori* planning (i.e., we cannot plan every possible course of actions *a priori*) and the limited capability of measuring the state of the world (i.e., limited perception capability). These two aspects introduce uncertainty into the decision process; such uncertainty is not fully considered by other approaches commonly used in decision making (e.g., decision trees, influence diagrams, multicriteria decision making, or Markov chains). On the one hand, we assume interaction workflows follow a Markovian process [i.e., an interaction workflow depends solely on the preceding state of the system (i.e., context)]. On the other hand, our problem addresses decision making (i.e., choose the right actions); thus, it addresses planning and control, not exclusively addressing perception or actuation.

Therefore, in our approach, we define a POMPD model for each context and represent the resulting policies (i.e., interaction workflows) in the scenario ontology, as depicted in Fig. 4. During execution, an ASC will adapt its decision process to different contexts by querying its knowledge model for the most suitable scenario ontology.

In this paper, we summarize the components involved in a POMDP; however, we do not present the details of the associated mathematical formalism that can be found, for example, in [8]. It suffices to note that a POMDP model is defined by the tuple $\{\mathcal{S}, \mathcal{A}, \mathcal{O}, \Omega, T, R\}$, in which each variable specifies the state of the world, the set of actions, the finite set of observations, the observation function that expresses the relationship between the state and the observations, the transitions function that expresses the likelihood of transitioning from state s with action a to new state s', and the reward function, respectively.

The goal of the POMDP solver is to find a value function (VF) $V(b)$ that represents the optimal policies over the belief distribution b, where b is defined with parameters $p_1, p_2, \ldots, p_N$, the beliefs of corresponding state, where N is the number of states. Moreover, $V(b)$ is defined as

$$V(b) = \sum_{i=1}^{N} v_i p_i \tag{1}$$

where $v_1, v_2, \ldots, v_N$ are the coefficients of a linear function. For a finite horizon T, (1) is a piecewise linear and convex VF $V_T(b)$ and can be represented by the maximum of a finite set of linear functions

$$V_T(b) = \max_k \sum_{i=1}^{N} v_i^k p_i \tag{2}$$

where $v_1^k, v_2^k, \ldots, v_N^k$ denote the parameters of the kth linear function.

2) Knowledge Model: Our approach adopts an ontological representation for capturing and storing knowledge regarding concepts and their relationships. This type of representation allows us to capture the types of knowledge required to fully represent the cognitive model, including concepts related to the person, environment, physical interaction, social interaction, and machine/robot interaction and algorithms.

The knowledge model (Fig. 5) captures the relevant information involved in the HRI process. We define the upper ontology for this framework based on four main entities: 1) machine; 2) human; 3) interaction; and 4) context. From these entities, we can define other entities as associated subclasses and establish relationships between entities that encode the semantics of their associations. A more detailed representation is explained in our previous work [8], of which we provide an updated iteration resulting from the current experimental application.[3]

Moreover, this set of concepts extends the core ontology for robotics and automation [29] and can be proposed to be included in the standard ontology for autonomous robots under development by the autonomous robotics working group.[4] In this paper, the focus is on the concepts related to the context-based human–machine interaction.

The entities defined and their relationships allow for the representation of the components of the system involved in the interaction process at each time.

[3]This model is available to be incorporated or extended by other representations at http://www.contextawarerobotics.org/cahri/kr/im-cahri.owl.

[4]https://standards.ieee.org/develop/wg/Autonomous_Robotics.html

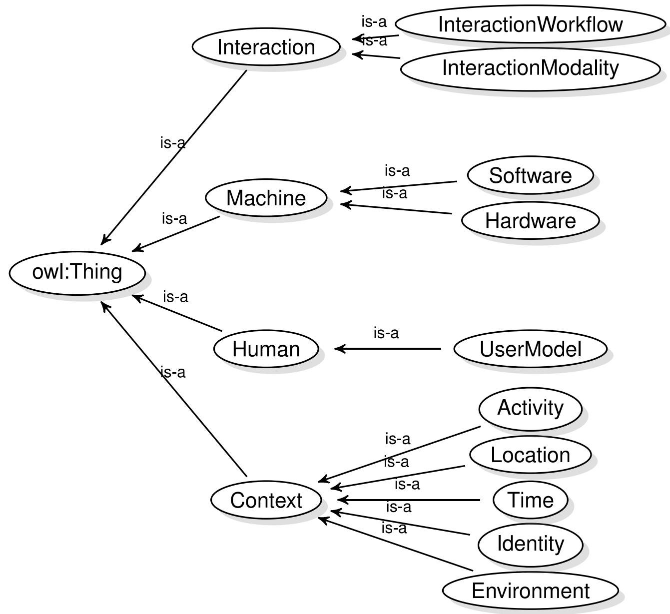


Fig. 5. Knowledge model used in the context-based HRI framework.

In addition to defining classes' taxonomy, we define object properties (OPs) and data properties (DPs) that will establish the relationships between individuals of each class. Our current model includes the following OPs.

1) (OP) hasActivityMission.
2) (OP) hasActuator (Domain: robot/Range: actuator).
3) (OP) hasContext (transitive).
4) (OP) hasEnvironmentCondition.
5) (OP) hasIdentity.
6) (OP) hasInteraction.
7) (OP) hasInteractionWorkflow.
8) (OP) hasRequirement (transitive).
9) (OP) hasSensor (Domain: robot/Range: sensor).
10) (OP) isActivityMissionOf (inverse of hasActivityMission).
11) (OP) isEnvironmentConditionOf (inverse of hasEnvironmentCondition).
12) (OP) isInteractionWorkflowOf (inverse of hasInteractionWorkflow).
13) (DP) policyGraph (string).

3) User Model: We can say that the decision process and the knowledge model are fundamental parts of any automatic system. Nevertheless, if the purpose of the system is to interact with humans, it becomes clear that models about the users are relevant to understand actions, intentions and more globally the context. To this end, any user model can be used as long as it provides the necessary inputs for the decision process. Hereafter, we describe an example for the user model adopted in our framework, which we refer to as the Bayesian user model. The main goal of the model is to infer a vector of the user's characteristics, $\mathbf{C} \in \mathbb{N}^n$, taking as input a vector of evidence $\mathbf{E} \in \mathbb{N}^m$. The main output of the system is the distribution

$$P(\mathbf{C}|\mathbf{E}, I) \propto P(\mathbf{C})P(\mathbf{E}, I|\mathbf{C}) \tag{3}$$

which encodes the user I's characteristics revealed by the evidence. By performing maximum *a posteriori* estimation [30] over this distribution, we can obtain the characteristics that each user is the most likely to exhibit.

The instantiation process is split into modules, with each module inferring one of the characteristics of the $\mathbf{C}$ vector. For each model, a Bayesian model is used to infer the characteristic. Once instantiated, the inferred characteristics can be used by the underlying system for interaction.

The model learns by fusing tuples of the form

$$T_i = (L_i, \mathbf{E}, h_i) \tag{4}$$

where $L \in \mathbb{N}$ is the label obtained for characteristic C_i, via maximum *a posteriori* estimation, and h_i is the entropy of the distribution $P(\mathbf{C}_i)) \approx P(\mathbf{C_i}|\mathbf{E}, I)$, to a global likelihood that is propagated across the system.

This fusion process yields a new likelihood obtained via

$$\begin{aligned} &P(\mathbf{E} = \mathbf{e}, I = u|\mathbf{C} = L_i)_{k+1} \\ &\quad = \frac{1}{\mu}\left(P(\mathbf{E} = \mathbf{e}, I = u|\mathbf{C} = L_i)_k + \frac{1}{h}\sigma_c\right) \end{aligned} \tag{5}$$

where e is the combination of evidence that generated the classification, c is the label obtained, μ is a normalization factor, k denotes the previous and updated likelihoods, u is the identity of the current user, and σ_c is an impulse function that is equal to 1 on c.

This fusion mechanism is able to accommodate both hard and soft labels. If hard evidence is received by the system, then L_i is set to the corresponding value and $P(\mathbf{C})$ becomes nonzero only for the corresponding value, thus conveying the certainty of hard evidence.

III. Framework Implementation

We adopted a technical implementation process based on the *behavior driven development* methodology, wherein we achieve the following.

1) Describe the agent's behavior by creating user stories that explain different scenarios of operation.
2) Create an upper ontology (classes, properties) that captures the information that is common across the domains of all scenarios associated with the feature (im-cahri-top, corresponds to the knowledge model presented in Section II).
3) Create a lower ontology (instances) that represents the scenarios as graphs.
4) Use the lower ontology to model the plan for interaction workflows in the decision process algorithm to adapt the actions in run time according the plan for a given situation (scenarios can interchange; thus, the context of operation changes).

A. Describing the Agent's Behaviors: Gherkin Scenarios and Creating Scenario Specific Ontologies

We describe the agent's desired behavior by creating user stories that explain different operation scenarios for some relevant application scenario. In our case, we refer to the conclusions from the SocialRobot and GrowMeUp projects in [31], which identified natural robot–user communication (e.g., via facial recognition, voice commands, and audio/video conferencing) as *user needs*. The interaction design of both systems was framed in well-identified requirements for the robotic platform that was developed. Of particular interest, in terms of HRI, we highlight the following.

1) The robot should be able to guide or follow someone in the environment.
2) The robot should be able to track a person (Haar-like features).
3) The robot should be able to perceive person poses.
4) The robot should be able to identify a person (via face recognition or reading some ID tag).

These requirements were later extended into use case scenarios in [32]. Specifically, we consider "Scenario 4: Face Recognition, Navigation, and Tracking—Elderly Care Centre Use" that sets the extended background for the two features (in Fig. 6) using the Gherkin scenario pattern as described in [33].

Persona: George is an 81-year old man having some light memory problems and some difficulties in balancing by walking; he is used to staying alone at home. After a fall, during the night, George decided that it was better for him to stay in an elderly care center because the only person who could take care of him was his daughter, who lives far away in another city, and he is not a very communicative person, making him reluctant to ask for support from his neighbors.

User Scenario: In the elderly house one morning George decided to walk to the small, sunny and warmer living room instead of going to the big and colder one at the main entrance. SocialRobot identified him sitting there alone and asked him if he would like to tell his friend Kostas to join him. George responded that he would like to have his friend Kostas around. SocialRobot went around the elderly center and found his friend Kostas, a 78-years old man who has similar disabilities and behaviors as George. Both became friends in the elderly care center. SocialRobot asked Kostas if he wanted to join George in the small living room because he was sitting there alone. After Kostas answered yes and SocialRobot accompanied him in the small sunny living room. George and Kostas were happy to be together chatting and enjoying the sun. SocialRobot recorded that they both like this room, and next time, it will inform them again if it finds one of them sitting there alone.

```
Feature: Person Detection and Face Recognition

  In order to identify the different people around
  the elderly center
  As a SocialRobot
  I need to perform face recognition while moving
  around

  Background:
    Given SocialRobot moves around the elderly
    center
    And the light conditions will be different in
    distinct divisions of the building

  Scenario: Person detection and face recognition
    in dimmed light
    Given the robot is moving around detecting
    people using the haar like features algorithm
    And the robot is crossing a division with
    ambient light below 200-350 luxes
    When the robot selects hog like features
    algorithm to detect people in its way
    And the robot selects haar like features
    algorithm to detect faces
    And the robot selects eigenfaces algorithm
    to identify the person
    Then, the robot should identify the person

--------------------------------------------------

Feature: User-Adaptive Guided Visit

  In order to guide a user on a visit to a new
  location
  As a SocialRobot
  I need to move between pre-defined points, while
  recognizing user characteristics and adapting to
  them

  Background:
    Given SocialRobot is standing on a fixed point
    And is running a person detection algorithm
    And starts an interaction when a new person
    is detected in its vicinity

  Scenario: Adaptation to proxemics
    Given the user feels the robot is invading
    their personal space
    When the robot has determined that the user
    is uncomfortable with its position
    Then, the robot should update its model of
    the user
    And adapt its positioning accordingly

  Scenario: Adaptation to hearing impairment
    Given the user feels that they cannot hear
    the robot correctly
    When the robot has determined that the user
    is not hearing it correctly
    Then, the robot should update its model of
    the user
    And adapt its speaking volume accordingly
```

Fig. 6. Elderly care center user scenario—Gherkin scenario example.

In the Gherkin scenario of Fig. 6, we write each sentence as similarly as possible to an ontology triple format (i.e., subject-predicate-object). The domain-specific ontology related to the previous example is illustrated in Fig. 7. This ontology represents the corresponding instances in the scenario and their relationships by means of a graph.

The resulting assertions are represented using OWL; see the listing in Fig. 8 for a snippet of the representation for the asserted axioms referring to individual *context1*.

B. Using Knowledge in Decision Process

The last step on our framework consists of using the lower ontology to model the plan for interaction workflows.

To this end, we train the decision process model (i.e., POMDP) for the specific scenario. Assuming that our model converges for an infinite horizon, it is possible to define a *policy graph* that can be used latter at run time (i.e., planning phase). This result will be stored as the value for the instances of class *InteractionWorkflow* in Fig. 5 that correspond to *interactionPolicy1* in the example illustrated above in Fig. 7.

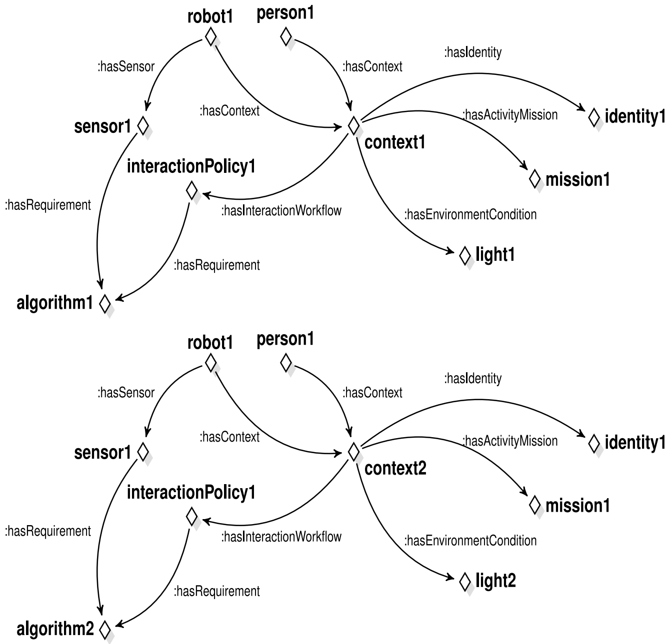


Fig. 7. Gherkin scenario (feature1-scenario 1) represented in terms of an ontology, instantiating classes from the upper ontology defined in Fig. 5.

```
<owl:NamedIndividual
rdf:about="http://www.contextawarerobotics.org/
cahri/kr/feature1-scenario1.owl#context1">
     <rdf:type rdf:resource=
     "http://www.contextawarerobotics.org/
     cahri/kr/im-cahri.owl#Context"
     />

     <im-cahri:hasActivityMission rdf:resource=
     "http://www.contextawarerobotics.org/
     cahri/kr/feature1-scenario1.owl#mission1"
     />

     <im-cahri:hasEnvironmentCondition
     rdf:resource=
     "http://www.contextawarerobotics.org/
     cahri/kr/
     feature1-scenario1.owl#lightCondition1"
     />

     <im-cahri:hasIdentity rdf:resource=
     "http://www.contextawarerobotics.org/
     cahri/kr/feature1-scenario1.owl#identity1"
     />

     <im-cahri:hasInteractionWorkflow
     rdf:resource=
     "http://www.contextawarerobotics.org/
     cahri/kr/
     feature1-scenario1.owl#interactionPolicy1"
     />
 </owl:NamedIndividual>
```

Fig. 8. Snippet of OWL representation for the Gherkin scenario illustrated in Figs. 6 and 7.

The great advantage of using an ontology representation is that after asserting a set of axioms, we can use a *Reasoner* to infer new knowledge from the relationships between instances (i.e., this process is also known as classifying the ontology).

```python

import rdflib
import rdfextras

def getRobotContext(graph):

    qres = graph.query("""
    PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
    PREFIX owl: <http://www.w3.org/2002/07/owl#>
    PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
    PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
    PREFIX im-cahri: <http://www.contextawarerobotics.org/cahri/kr/im-cahri.owl#>
    PREFIX : <http://www.contextawarerobotics.org/cahri/kr/feature1-scenario1.owl#>

    SELECT DISTINCT ?robot ?context
    WHERE {
    ?robot rdf:type im-cahri:robot .
    ?robot im-cahri:hasContext ?context .
    }""")


    for row in qres:
        print("%s hasContext %s" % row)

    return
```

Fig. 9. SPARQL query to obtain all contexts for all robots defined in the knowledge representation using rdflib for a Python implementation.

In other words, we only need to define explicitly that the *interactionPolicy1* requires *algorithm1* as the Reasoner would infer that *interactionPolicy1* also requires *sensor1*, given that *hasRequirement* is a transitive OP. Another advantage is the scalability and flexibility of merging different ontologies into a "unique" knowledge base. The result of inferring new knowledge by the Reasoner can be made permanent by adding the inferred axioms to initially asserted ones. This result is particularly useful for applications where a Reasoner is not available or for improved searching because inference can become impractical for large ontologies.

In runtime, we may use any programming library that can manipulate RDF/RDFS/OWL (e.g., rdflib, Sesame for Python, or Java implementations, respectively) to query our knowledge representation (also known as a triple store), using SPAQRL language, to obtain useful information from our asserted axioms and conduct the interaction workflow (i.e., following the policy graph stored previously). An example of these types of queries is illustrated in Fig. 9.

IV. Experimental Design, Results, and Discussion

The experimental validation of this paper replicated the conditions of the scenarios described in Fig. 6. The goal of the experiment was to answer our research question, studying the effects of integrating a decision process that selects interaction

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE I
RELEVANT CHARACTERISTICS OF ENVIRONMENT CONDITIONS (CONTEXT OF OPERATION). *DEPENDING ON SENSOR RANGE

Environment	Description	luminance range (lux)
Room	natural day light	[50, 400]
Room	artificial light	[10, 250]
Room	normal light	[10, 50]
Room	dimmed light	[1, 10]
Room	total darkness	0
Outdoor	day light	> 1000
Other	direct-contact with lamp	up to 3000*

workflows to automatically adapt to different environment conditions (i.e., context) aiming to: 1) improve the usability of an interactive agent and 2) make the human–machine interaction component of the system more robust (i.e., fewer failures).

A. Experimental Design

We structured the experiment according a multivariate analysis of variance, a repeated measures design in which the samples are analyzed by all different approaches being studied, comparing the difference between means. In this experiment, we were focused on the analysis of the effects on specificity for each approach used. We considered as our primary variables the light intensity and number of persons in the scene. Considering that this experiment would lead to a very large number of images to analyze (i.e., a large population), we decided to use random samples of data to perform our trial. The sample size was calculated using the "Test 1 Mean: 1-Sample, 1-Sided" [34] method, which is useful for tests concerning whether a given result is equal to a reference value. In our case, we want to measure the result for specificity of the different algorithms; hence, based on previous work [8], we used the values for the null hypothesis mean (μ_0) equal to 0.5, the true mean (μ) equal to $\mu_0 \pm 0.05$, the error standard deviation equal to 0.2, the power equal to 0.85 and the Type I error rate α equal to 0.1. The resulting minimum sample size n was equal to 73 images.

B. Experimental Setting

In our experimental setting we prepared the environment in such terms that it could replicate typical living room conditions. The data collection procedure included acquisition of visual data and light conditions (i.e., rgb camera plus light sensor). Consequently, the resulting dataset considers the typical changes of the environment of operation as they are observed in relevant application environments (i.e., not in the controlled environment of a laboratory). More specifically, we performed an initial characterization of light conditions considering different variations of light intensity and illumination source, as summarized in Table I. Finally, we conducted our data collection in a room environment with normal and dimmed light conditions with luminance between 0 and 20 lux.

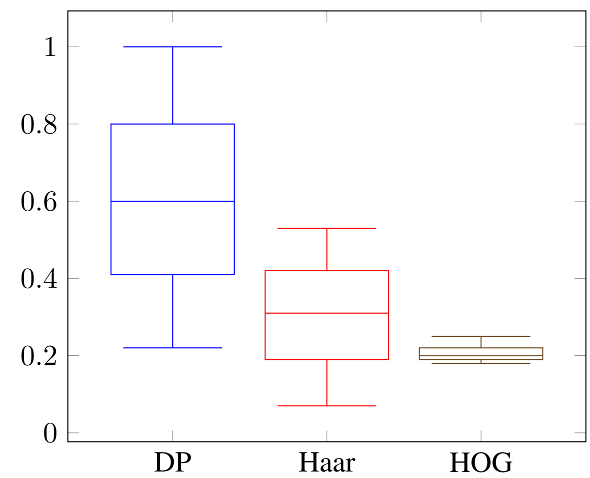


Fig. 10. Box plot for specificity.

C. Experimental Implementation

First, we used Protege[5] for designing and working with ontologies. The next step was to study the behavior of the algorithms using our previous approach described in [8], where we used the INRIA dataset[6] for people detection and measured precision, recall, f-measurement, and computational time for the Haar-like features and histograms of oriented gradients (HOG) algorithms. From this paper, our problem was defined as POMDP and solved for an infinite horizon that converged for a tolerable range of marginal improvement for the resulting policy graph. To achieve this solution, we used Anthony Cassandra's POMDP solve.[7] The resulting policy graph was incorporated into the specific ontology for our particular scenarios as the literals of the DP *policyGraph* in InteractionWorkflow class (instanciated in interactionPolicy1). Following this initial setup, we collected a dataset of aggregated visual and light information (i.e., video with 78 frames plus time-stamped light data in an additional file). We selected two algorithms that are commonly used for person detection—Haar-like features and HOG. These two approaches implemented the same functionality, but their performance differs depending of illumination conditions.

D. Results and Discussion

We analyzed the data that corresponded to three runs for each video. In the first run, we used the selected action for decision process; in the second run, we used only the Haar detection algorithm; and in the third run, we used only HOG detection.

The results for the statistical analysis of the detections outcomes were compiled into Table II and Fig. 10.

In Fig. 11, we present some examples of frames acquired and a visualization of the recorded hits, misses, and errors for each run.

The experimental setup described before allowed us to obtain results that confirm the second objective. From the

[5]https://protege.stanford.edu
[6]http://pascal.inrialpes.fr/data/human/
[7]http://www.pomdp.org/code/index.html

Fig. 11. Examples of frames from the scenario dataset. From top-left to bottom-right: we see the first two frames and the last, where we have false positives for Haar and HOG algorithms, but the action from the decision process is check-light, which contributes with less errors; the next three frames represent examples of correct detections by the Haar or HOG algorithms and the action selected by the decision process corresponding to most probable algorithm to work under the detected light conditions; at the bottom we summarize the overall results for true positives/negatives in light green, false positives in light red and false negatives in yellow. The first row represents the periods where light conditions were considered normal or dark. The following rows group the results for the decision process, Haar and HOG algorithms.

TABLE II
STATISTICAL MEASURES OF THE OVERALL PERFORMANCE FOR DECISION PROCESS COMPARED TO SINGLE SELECTION OF BODY DETECTION ALGORITHMS

Metric	DP	Haar	Hog
Recall	0,19	0,15	0,48
Specificity	0,60	0,31	0,20
Precision	0,44	0,18	0,49
Negative predictive value	0,31	0,28	0,19
Fall-out	0,40	0,69	0,80
False negative rate	0,81	0,85	0,52
False discovery rate	0,56	0,82	0,51
Accuracy	0,34	0,23	0,37
F1 score	0,27	0,16	0,48

obtained results, two main advantages can be observed from the statistical measures: first, the specificity value for DP is on average 2.5 times the specificity for the Haar and HOG algorithms when used in single operation (i.e., getting less errors resulted in a higher value for the true negative rate); second, precision for the DP is 11% less than that of the HOG algorithm, which showed the best overall performance. Attending to these results, we confirmed the second statement in our hypothesis.

Nevertheless, the main limitation observed from the statistical measures is that our approach resulted in lower recall. This limitation may be due to limited variations in the environment conditions, which may have not covered in sufficient detail the behavior of the overall system (i.e., our test focused mainly the operation in a room with normal and dimmed light conditions). Analyzing Fig. 11, we observe that for constant "dark-light" conditions, these observations resulted in the decision process constantly selecting action "check-light" (i.e., using the policy graph from interactionPolicy1). This action corresponds to a sensing action instead of trying to perform detection. Comparing the outcome of this action with the two other options, we observe that performing Haar detection would result in an equivalent recall rate but with much less precision and specificity. Alternatively, using HOG detection would result in higher recall but at the same time lose specificity. Overall, assuming the issues involved in interactive features and based on the lessons learned from previous works [2], preventing erroneous detections is as relevant as the hit rate. By proving our second objective, we can claim the implicit demonstration of the first part of our hypothesis. Given that the usability of an agent is intrinsically related with not only performing the correct action but also not performing the wrong one, it becomes trivial that our approach can achieve this first objective. Nevertheless, we will plan for future work gathering more information to better corroborate this claim.

This experiment was designed with a clear intention of proofing the concept that incorporating redundancy and fall-back strategies in interaction functionalities should result in the agent's self-adaptation to its context. Therefore, these results are compared mainly in terms of the specificity of the CAHRI framework in relation to the previous results of mainstream research projects in this field—CaMeLi and GrowMeUp. In these particular examples, the feature for person detection was implemented using only the Haar-like features algorithm. Hence, using CAHRI can improve their specificity for this feature in near 2.5 times. Because we did not focus on implementing new classification methods that could be compared to other mainstream approaches (e.g., classifiers for people detection), a thorough comparison between the performances of different classification approaches was not covered in this paper. Nevertheless, we foresee that existing systems and mainstream research results may improve using the proposed framework. For example, in a related work in progress experiment, we are using YOLO [35] for practical assisted living applications in a home environment. In this setting, we are observing YOLO has high recall for person recognition. However, regarding object recognition it falls lower than required for practical application. We believe this situation could be improved if each neural network is previously trained to perform in a specific context and then we use our framework to select the best neural network for the context of operation.

V. Conclusion

The purpose of this paper was to understand how an agent's performance is affected when interaction workflows are incorporated in its information model and decision-making process. To achieve this objective, we must overcome current limitations of information sharing in decision processes and find computationally effective methods to build complex decision processes involved in the interaction process. We hypothesized that part of our solution could incorporate redundancy and fall-back strategies in terms of interaction functionalities that could result in the agent's self-adaptation to its context (e.g., user model and environment conditions). This incorporation would also result in fewer errors during operation. In our experimental validation, we considered how to improve the person detection feature in an environment with changing lighting conditions (i.e., environment context changes). This feature is particularly relevant because it is considered by end-users as a core functionality and was previously implemented in the two ASC systems (i.e., CaMeLi virtual assistant and GrowMeUp social robot), as demonstrated in our previous work. The results confirmed that our approach can indeed improve the agent's performance, maintaining precision while improving specificity. However, we must recognize that we still face some challenges in designing and implementing interaction workflows. Involving the users during the design process is relevant to the identification of needs and capturing requirements, but implementing interaction workflows based on predefined user scenarios and static action scripts is not sufficient to take into account uncertainty associated with noisy inputs, variation in the conditions of the operating environment, or unclear expectations from the user. Hence, this framework represents a contribution to the field of cognitive robotics by improving the usability of ASCs. Our future work will continue to develop this framework and will focus on usability validation and implementation of a distributed processing approach for planning algorithms.

References

[1] J. Quintas *et al.*, "Context-based understanding of interaction intentions," in *Proc. 21st IEEE Int. Symp. Robot Human Interact. Commun. (IEEE RO-MAN)*, Paris, France, Sep. 2012, pp. 515–520.

[2] C. Tsiourti *et al.*, "A virtual assistive companion for older adults: Design implications for a real-world application," in *Proc. SAI Intell. Syst. Conf.*, Sep. 2016, pp. 556–566.

[3] D. Portugal *et al.*, "SocialRobot: An interactive mobile robot for elderly home care," in *Proc. IEEE/SICE Int. Symp. Syst. Integr. (SII)*, Nagoya, Japan, Dec. 2015, pp. 811–816.

[4] G. S. Martins, L. Santos, and J. Dias, "Towards a context-aware adaptable services framework with applications in elderly care," in *Proc. Int. Conf. Soc. Robot. Workshop Improving Qual. Life Elderly Robot. Assistive Technol.*, Oct. 2015, pp. 1–6.

[5] G. S. Martins, L. Santos, and J. Dias, "The GrowMeUp project and the applicability of action recognition techniques," in *Proc. 3rd Workshop Recognit. Action Scene Understand. (REACTS2015)*, Sep. 2015, pp. 1–14.

[6] J. Quintas, P. Menezes, and J. Dias, "Auto-adaptive interactive systems for active and assisted living applications," in *Proc. IFIP Adv. Inf. Commun. Technol. Technol. Innov. Cyber Phys. Syst. (DoCEIS)*, Costa da Caparica, Portugal, Apr. 2016, pp. 161–168.

[7] J. Quintas, P. Menezes, and J. Dias, "Context-based decision system for human–machine interaction applications," in *Proc. IEEE Int. Conf. Syst. Man Cybern. (SMC)*, Budapest, Hungary, Oct. 2016, pp. 3906–3911.

[8] J. Quintas, P. Menezes, and J. Dias, "Information model and architecture specification for context awareness interaction decision support in cyber-physical human–machine systems," *IEEE Trans. Human–Mach. Syst.*, vol. 47, no. 3, pp. 323–331, Jun. 2017.

[9] R. Alami, A. Clodic, R. Chatila, and S. Lemaignan, "Reasoning about humans and its use in a cognitive control architecture for a collaborative robot," in *Proc. Cogn. Archit. Human Robot Interact. Workshop (HRI)*, Mar. 2014, pp. 1–4.

[10] M. Sili, M. Gira, M. Müllner-Rieder, and C. Mayer, *Interaction Model-Based User Interfaces: Two Approaches, One Goal—Comparison of Two User Interface Generation Approaches Applying Interaction Models*. Cham, Switzerland: Springer, Dec. 2015, pp. 185–197.

[11] S. Devin, G. Milliez, M. Fiore, A. Clodic, and R. Alami, "Some essential skills and their combination in an architecture for a cognitive and interactive robot," in *Proc. 2nd Workshop Cogn. Archit. Soc. Human Robot Interact.*, Mar. 2016, pp. 1–5.

[12] H. G. Kim, J. Y. Yang, and D. S. Kwon, "Experience based domestic environment and user adaptive cleaning algorithm of a robot cleaner," in *Proc. 11th Int. Conf. Ubiquitous Robots Ambient Intell. (URAI)*, Kuala Lumpur, Malaysia, Nov. 2015, pp. 176–178.

[13] C.-M. Huang and B. Mutlu, "Anticipatory robot control for efficient human–robot collaboration," in *Proc. Human Robot Interact.*, Christchurch, New Zealand, 2016, pp. 83–90.

[14] I. Duque, K. Dautenhahn, K. L. Koay, L. Willcock, and B. Christianson, "A different approach of using personas in human–robot interaction: Integrating personas as computational models to modify robot companions' behaviour," in *Proc. 22nd IEEE Int. Symp. Robot Human Interact. Commun.*, Gyeongju, South Korea, Aug. 2013, pp. 424–429.

[15] A. Aly and A. Tapus, "A model for synthesizing a combined verbal and nonverbal behavior based on personality traits in human robot interaction," in *Proc. ACM/IEEE Int. Conf. Human Robot Interact.*, Tokyo, Japan, Mar. 2013, pp. 325–332.

[16] S. D. Klee *et al.*, "Personalized assistance for dressing users," in *Proc. 7th Int. Conf. Soc. Robot. (ICSR)*, Paris, France, Oct. 2015, pp. 359–369.

[17] A. Madureira *et al.*, "Using personas for supporting user modeling on scheduling systems," in *Proc. IEEE 14th Int. Conf. Hybrid Intell. Syst. (HIS)*, Dec. 2014, pp. 279–284.

[18] G. Gordon and C. Breazeal, "Bayesian active learning-based robot tutor for children's word-reading skills," in *Proc. 29th AAAI Conf. Artif.*, Austin, TX, USA, Jan. 2015, pp. 1343–1349.

[19] A. B. Karami, K. Sehaba, and B. Encelle, "Adaptive artificial companions learning from users feedback," *Adapt. Behav.*, vol. 24, no. 2, pp. 69–86, 2016.

[20] G. H. Lim, S. W. Hong, I. Lee, I. H. Suh, and M. Beetz, "Robot recommender system using affection-based episode ontology for personalization," in *Proc. 22nd IEEE Int. Symp. Robot Human Interact. Commun.*, Yeongju, South Korea, Aug. 2013, pp. 155–160.

[21] J. A. Konstan and J. Riedl, "Recommender systems: From algorithms to user experience," *User Model. User Adapt. Interact.*, vol. 22, nos. 1–2, pp. 101–123, 2012.

[22] Y. Kim and W. C. Yoon, "Generating task-oriented interactions of service robots," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 8, pp. 981–994, Aug. 2014.

[23] H. Zhang, W. Ni, X. Li, and Y. Yang, "Modeling the heterogeneous duration of user interest in time-dependent recommendation: A hidden semi-Markov approach," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 2, pp. 177–194, Feb. 2018.

[24] B. Cheng, S. Zhong, and J. Chen, "Context ontology-based reasoning service for multimedia conferencing process intelligence," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 12, pp. 3219–3232, Dec. 2017.

[25] J. Quintas, P. Menezes, and J. Dias, "Cloud robotics: Towards context aware robotic networks," in *Proc. 16th IASTED Int. Conf. Robot. (Robo)*, Nov. 2011, pp. 420–427.

[26] B. Troadec and C. Martinot, *Le Développement Cognitif: Théories Actuelles De La Pensée En Contextes*. Paris, France: Belin, 2003.

[27] J. Hoc, "Towards a cognitive approach to human machine cooperation in dynamic situations," *Int. J. Human Comput. Stud.*, vol. 54, no. 4, pp. 509–540, 2001.

[28] M.-P. Pacaux-Lemoine and M. Itoh, "Towards vertical and horizontal extension of shared control concept," in *Proc. IEEE Int. Conf. Syst. Man Cybern.*, Kowloon, China, Oct. 2015, pp. 3086–3091.

[29] E. Prestes *et al.*, "Towards a core ontology for robotics and automation," *Robot. Auton. Syst.*, vol. 61, no. 11, pp. 1193–1204, 2013.

[30] J. F. Ferreira and J. Dias, *Probabilistic Approaches for Robotic Perception*. Cham, Switzerland: Springer Int., 2014.

[31] E. Christodoulou *et al.*, "SocialRobot D1.1 specification of user needs and requirements," SocialRobot, Lisbon, Portugal, Rep. D1.1, 2012.

[32] E. Christodoulou *et al.*, "SocialRobot D1.2 use case scenarios development and ethical privacy considerations," SocialRobot, Lisbon, Portugal, Rep. D1.1, 2012.

[33] S. Rose, M. Wynne, and A. Hellesoy, *The Cucumber for Java Book: Behaviour-Driven Development for Testers and Developers* (Pragmatic Programmers). Dallas, TX, USA: Pragmatic Bookshelf, 2015. [Online]. Available: https://books.google.pt/books?id=zQ2voQEACAAJ

[34] B. Rosner, *Fundamentals of Biostatistics*. Boston, MA, USA: Cengage Learning, 2010.

[35] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, pp. 1–10, Jun. 2015.

João Quintas received the M.Sc. degree in electrical and computer engineering from the University of Coimbra, Coimbra, Portugal, in 2009, where he is currently pursuing the Ph.D. degree in electrical and computer engineering (specialization in automation and robotics).

He is a Project Manager and a Researcher with Instituto Pedro Nunes, Coimbra. Since 2008, he has been acting as a Technical Manager, a Project Manager, and a Researcher in European and national projects. His current research interests include robotics, ambient intelligence, human–machine interaction, and active assisted living applications. He is a member of the Autonomous Robotics Ontology Working Group developing IEEE 1872.2 Standard and the IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems developing IEEE 7000 related standards.

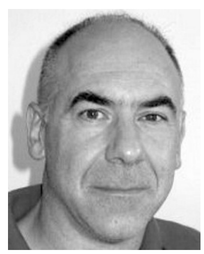
Gonçalo S. Martins received the M.Sc. degree in electrical and computer engineering from the University of Coimbra (UC), Coimbra, Portugal, in 2015, where he is currently pursuing the Ph.D. degree in electrical and computer engineering (specialization in automation and robotics).

He is currently a Researcher with the Institute of Systems and Robotics, UC. His current research interests include user-adaptive systems, machine learning, and social robotics.

Luis Santos received the Ph.D. degree in electrical and computer engineering (specialization in automation and robotics) from the University of Coimbra (UC), Coimbra, Portugal, in 2014.

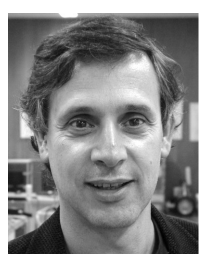
He is currently an Invited Research Assistant with UC. He has been serving as a Project and Technical Manager for the H2020 GrowMeUp Project (GA 643647) since 2015. His current research interests include social robotics, human motion understanding, data analytics, and explainable artificial intelligence.

Dr. Santos was a recipient of the Marie Curie Fellowship in 2014.



Paulo Menezes received the Ph.D. degree in electrical and computer engineering (specialty in informatics) from the University of Coimbra, Coimbra, Portugal.

He is an Assistant Professor with the University of Coimbra, where he is a Senior Researcher with the Institute of Systems and Robotics. His current research interests include computer vision, HMI/HRI, immersive and interactive environments, and human behavior and emotional analysis. He has been involved in several national and European research and development projects in the above areas.



Jorge Dias received the Habilitation and the Ph.D. degrees in electrical engineering (specialization in control and instrumentation) from the University of Coimbra, Coimbra, Portugal.

He is an Associate Professor with the University of Coimbra, where he holds his research activities with the Institute of Systems and Robotics. His current research interests include robot vision, computer vision, human–robot interfaces, and social robots. He has been performing activities and making contributions in the above areas since 1984. He has been the Principal Researcher in several European research projects.