

# A multi-modal perception based assistive robotic system for the elderly

Christophe Mollaret, Alhayat Ali Mekonnen, Frédéric Lerasle, Isabelle Ferrané, Julien Pinquier, Blandine Boudet, Pierre Rumeau

▶ **To cite this version:**

Christophe Mollaret, Alhayat Ali Mekonnen, Frédéric Lerasle, Isabelle Ferrané, Julien Pinquier, et al.. A multi-modal perception based assistive robotic system for the elderly. Computer Vision and Image Understanding, 2016, Special issue on Assistive Computer Vision and Robotics : Assistive Solutions for Mobility, Communication and HMI, 149, pp.78-97. 10.1016/j.cviu.2016.03.003 . hal-01300463

**HAL Id: hal-01300463**
**https://hal.science/hal-01300463**

Submitted on 11 Apr 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A Multi-modal Perception based Assistive Robotic System for the Elderly

C. Mollaret[a,b,c], A. A. Mekonnen[a,b], F. Lerasle[b,c], I. Ferrané[a], J. Pinquier[a], B. Boudet[d], P. Rumeau[d]

[a] *IRIT, Univ de Toulouse, 118 route de Narbonne, 31062 Toulouse Cedex 9, France*
[b] *CNRS, LAAS, 7, Avenue du Colonel Roche, F-31400 Toulouse, France*
[c] *Univ de Toulouse, UPS, LAAS, F-31400 Toulouse, France*
[d] *UMR 1027 Inserm, Gérontopôle, laboratoire de gérontechnologie La Grave, CHU Toulouse, Université Toulouse-3, place Lange, TSA 60033, 31059 Toulouse Cedex 9, France*

**Abstract**

In this paper, we present a multi-modal perception based framework to realize a non-intrusive domestic assistive robotic system. It is non-intrusive in that it only starts interaction with a user when it detects the user's intention to do so. All the robot's actions are based on multi-modal perceptions which include user detection based on RGB-D data, user's *intention-for-interaction* detection with RGB-D and audio data, and communication via user distance mediated speech recognition. The utilization of multi-modal cues in different parts of the robotic activity paves the way to successful robotic runs (94% success rate). Each presented perceptual component is systematically evaluated using appropriate dataset and evaluation metrics. Finally the complete system is fully integrated on the PR2 robotic platform and validated through system sanity check runs and user studies with the help of 17 volunteer elderly participants.

*Keywords:* Assistive Technology, Elderly Care, Intention Detection, Multi-modal Data Fusion, Human-Robot Interaction, Robotic Perception

## 1. Introduction

As living conditions and health care facilities improve, the average life expectancy increases leading to a growing elderly population. For example, in France, in 2005, there were five young and adult people for one senior, in 2050, it is expected that there will be ten young and adult people for every seven seniors [1]. Though most people age well, some become frail, at risk of disease and costly dependence. Therefore, the financial and organizational burden on the society is likely to rise. How can we provide quality care to people requiring constant assistance in various aspects, including those suffering

*Email addresses:* cmollare@laas.fr (C. Mollaret), aamekonn@laas.fr (A. A. Mekonnen), lerasle@laas.fr (F. Lerasle), ferrane@irit.fr (I. Ferrané), pinquier@irit.fr (J. Pinquier), rumeau.p@chu-toulouse.fr (P. Rumeau)

from deterioration in cognitive capabilities due to aging, head trauma, and Alzheimer's disease (AD)? This enlightenment has led to a growing necessity for new technologies that can assist the elderly in their daily living. One such technology is the deployment of assistive robots for the elderly. In fact, such kind of robotic systems serving various tasks and purposes in the social care and medical/health sectors beyond the traditional scope of surgical and rehabilitation robots are poised to become one of the most important technological innovations of the $21^{st}$ century [2]. But, when robots leave industrial mass production to help with daily living activities, i.e., household chores, the requirements for robot platforms change. While industrial production requires strength, precision, speed, and endurance, domestic service requires a robust navigation in indoor environments, a dexterous object manipulation, and an intuitive way of communicating (speech, gestures, and body language) with users. In this perspective, many issues are still to be solved, such as perception and system integration.

Making a robot a socially competent service provider in all the daily life areas is very challenging. Hence, we focus on the conception of a robotic system that provides mild memory assistance to the elderly, a requirement highly coveted for the elderly whom might exhibit Mild Cognitive Impairment (MCI) or Alzheimer's disease (AD) [3, 4]. A serious issue for the elderly with MCI is forgetting where they have put objects that they use everyday, for example, keys, remote control, glasses, etc. This leads them to experience stress, loss of confidence, and become irritable, putting them at health risk especially considering their frailty [4]. Consequently, we consider deploying a robotic system that helps the elderly with MCI in locating everyday objects which are hidden (out of the user's sight), or put in unusual places. The work presented in this paper is part of a French National Research Agency (ANR) funded research project called RIDDLE[1], an acronym for *Robots for perceptual Interactions Dedicated to Daily Life Environment*, which aims to make a step forward in these directions by combining the underlying multiple and uncertain perceptual analyses related to, (1) objects and space regarding the robot's spatial intelligence, and (2) multi-modal communication regarding the robot's transactional intelligence.

To paint a clear picture, let us consider the following exemplar scenario. A person suffering from MCI, which we henceforth refer as the user, is carrying his/her normal everyday activity. Then, let us say the user wants to change the channel on the TV but realizes he/she could not remember where the remote control is. The user will then have to pose the question to the robot which is monitoring him/her stowed away in a non-interfering position. The robot will then answer the user's questions/*riddles* about the object utilizing appropriate actions (speech, displacement, pointing action). Based on this scenario, we identify three main key functional requirements for the robot: (1) detecting the user at all times, (2) detecting when the user wants to interact with the robot, i.e., when the user wants to pose a question to the robot or needs its attention – called user's *intention-for-interaction*, and (3) interaction via speech based communication. In this

[1] `http://projects.laas.fr/riddle/`

work, we assume the type and position of the objects are known a priori and we focus only on the highlighted three requirements. The considered objects are specifically eyeglasses, keys, mobile phone, wallet, remote control, and medication – frequently lost objects by the elderly as identified through a pilot study [4].

The entire behavior of the proposed assistive system is based on a widely accepted reactive behavior which cycles through "monitor" and "interact" phases, e.g., [5, 6, 7]. In this behavior, the robot monitors the user until he/she demands it to do something or shows an interest to interact with it. Then the robot continues through the interaction phase where it interacts with the user to provide requested service or assistance. In line with this, we propose a domestic assistive robot system that incorporates a novel *user's intention* detection mechanism to transition from monitoring phase to interaction phase. Therefore, we propose a scenario where the robot comes into a room, checks the presence of a user in this room, and then stows away at an observation place to monitor the user discretely. When the user expresses his/her intent to interact with the robot – either by looking at it, calling it, or a combination of both – the robot approaches the user and starts the close interaction phase. We refer to this as a *non-intrusive* behavior – the robot is not moving to stalk the user – with three distinct phases: user detection, user monitoring, and interaction phases (see Fig. 1 in Section 3). Initially when the robot correctly detects the user, it goes to the monitoring stage, actively reading the user's intention. Then, when it detects the user's *intention-for-interaction*, it makes a transition into the interaction phase which is carried out via speech modalities. During the speech based communication, depending on the utilized sensor, recognition tool, and Human/Robot (H/R) situation (distance variations), the communication quality can be affected [8, 9]. Whenever possible an adaptive mechanism should be put in place to maximize the chances of having an ideal communication given the available resources.

In this work, the robot considers that there is only one user to communicate with. The used language is French; it could be, nevertheless, generalized to any language. The person can freely move in his/her environment. However, when an interaction starts, after the *intention-for-interaction* step, the person's position is fixed during the interaction process. The goal of the interaction phase is for the robot to find a lost object upon request. The robot has to indicate the direction of the object by pointing its head towards it and giving some verbal precisions about its location – at the moment no object displacement is managed by the robot. For example: "the remote control is under the table". The objects are stored in a user-defined semantic map since the focus of the paper is not on object detection. All the algorithms presented in this work are embedded on the PR2[2] robotic platform using the Robot Operating System (ROS) middleware framework [10]. ROS is a collection of software frameworks for robot software development with a very active community and numerous publicly available packages that provide an operating system-like functionality on a heterogeneous computer cluster. Hence, we base all our

[2]PR2 (Personal Robot 2) is a robotic platform developed by Willow Garage: `http://www.willowgarage.com`

implementations and associate robotic integration on ROS (based on C++ and Python programming languages). Furthermore, all essential algorithms are integrated on the robot, while data visualization modules are seamlessly integrated on an external computer without overloading the PR2 system. All sensors are embedded on the robot. The only exception is an Android smartphone, which is located within 2 meters from the user (in his/her hand or on a table near him/her), that is used to capture audio signal.

This paper makes the following four core contributions:

1. A complete multi-modal perception driven non-intrusive domestic robotic system;
2. A novel multi-modal user's *intention-for-interaction* detection modality;
3. A fusion method to improve the speech recognition given the user's position, available sensors, and recognition tools;
4. Details of the complete implemented system along with relevant evaluations that demonstrate the soundness of the framework via an exemplar application. The application is an assistive scenario whereby the robot helps the user find hidden or misplaced objects in his/her living place.

The proposed framework is further investigated by conducting relevant user studies involving 17 elderly participants.

This paper is organized as follows: Section 2 discusses related work briefly, Section 3 presents an overview of the adopted system. Sections 4, 5, and 6 describe the user detection, the user's intention detection, and close HRI (with speech modality) part, respectively. Then, Section 7 explains how the task-level coordination is realized along with relevant implementation details. Experiments and results on PR2 are detailed in Section 8, and finally, the paper finishes with concluding remarks in Section 9.

## 2. Related Work

As highlighted in the introduction section, we consider to endow the robot with monitoring and interacting phases similar to most assistive robotic systems, e.g., [5, 6, 7]. During the monitoring phase, the robot remains static and observes the scene until a triggering action leads it to transition to the interacting phase, making it begin an interaction with the user. In the literature, the interaction and monitoring phases adopted by various assistive robotic system are very similar – Broekens *et al.* [3] provide a review of relevant assistive social robots in elderly care. The main difference comes from what triggers the transition from monitoring to interaction. Different approaches have been investigated: a user interface (on screen) [7], an explicit vocal demand [5], a recognized gesture [11], or a scheduled triggering (e.g., take medicine at 8 am) [6], are some examples. Even though all these triggering mechanisms have led to a successful assistive robotic scenario in their specific context, we argue that further improved system can be reached by triggering mechanism on user's intention, i.e., start interaction phase whenever the user expresses intent to interact

with the robot. We call this notion the user's *intention-for-interaction*. Indeed, endowing robots with the ability to understand humans' intentions opens up the possibility to create robots that can successfully interact with people in a social setting as humans, stepping towards proactivity [5]. We use the term *non-intrusive* to describe the behaviour of this assistive robotic system. It is non-intrusive in that it only starts interaction with a user when it detects the user's intention to do so, and before that the robot monitors the person with an RGB-D camera and audio sensor. This is inline with several work in the literature that identify with the term by using cameras for observation without interrupting and/or intruding into the target user [12, 13, 14, 15].

The rest of this section presents related work that pertains to each considered perceptual component, i.e., user detection, intention perception, and speech recognition.

### *2.1. User Detection*

User detection in our context falls in the generic research area of automated people detection. Most of the promising approaches rely on visual sensors, primarily classical RGB cameras and RGB-D (D stands for depth) cameras/stereo-heads providing 3D data. To date, several visual (visual camera based) people detectors have been proposed (see comprehensive surveys in [16, 17]). When considering a camera on a moving vehicle, as in a mobile robot, the detector has to rely on information per frame and cannot rely on stationary or slowly changing background assumptions/models. In RGB based sensors, the most relevant works are that of Dalal and Triggs's Histogram of Oriented Gradients (HOG) based detector [18], Felzenszwalb *et al.*'s Deformable Parts Model (DPM) based detector [19], and the Aggregate Channel Feature (ACF) based detector of Dollar *et al.* [20]. In this vein, further improvements have been achieved by utilizing various heterogeneous pool of features [21, 22]. With the advent of efficient and easy to use RGB-D cameras, e.g., Microsoft's Kinect sensor, more improved and robust people detectors have been proposed [23, 24]. These RGB-D based propositions have a remarkable impact in the robotics community; in addition to being compelling alternatives to laser range finder based detectors [25], they have various useful qualities: accurate distance to user, better occlusion reasoning, robustness to appearance clutter, etc. As a result, they have recently been a popular choice whenever an RGB-D sensor is involved [23]. Consequently, in our framework, we primarily use an RGB-D based detector and couple it with an RGB based detector for further improvement.

### *2.2. Intention Perception*

Detecting user's intention has, in recent years, gained significant attention in Human-Robot Interaction (HRI) research. Generally speaking, understanding user's intentions plays a fundamental role in establishing coherent communication amongst people [26]. Inspired by this, different researchers have been working on detecting user's intention for improved human-machine interaction in general, e.g., [27, 28, 29, 30]. Knowing a user's true intention opens up several possibilities: (1) understand his/her activity at the earliest (before the activity is even complete); (2) constrain the space of possible future actions and provide context [29]; and (3) correctly understand his/her action, for example, in the

event of a motor neuron disorder where actions might not reflect true user's intention [31]. In particular, a Parkinson user could emit incoherent gestures for the robot, which could be difficult to interpret.

Intention has previously been synthesized in the literature as "robot-awareness" by Drury *et al.* [32]. We can also find a similar concept defined as "attention estimation" in Hommel *et al.* [33]. However, these terms refer to lots of notions in robotics, such as "context-awareness", "user-awareness", etc. Even "user-awareness" can have more than one meaning. For example, it could mean that the robot needs only to know where the user is, for an obstacle avoidance context, or it could mean that the robot needs to interpret the user's emotions in the context of human-robot interaction [34]. In this work, we will focus on detecting the intention of a user to interact with the robot, which we call *intention-for-interaction*, often simply stated as user intention detection. The usage of the term intention detection is in line with the terminology used in relevant literatures that address similar problems, e.g., [35, 36, 37].

Recently, various work revolving around user intention perception has been burgeoning in the HRI community [31, 29, 38, 39]. The need to understand people's intention mostly stems from early activity detection [30, 38, 40], context establishment [29, 30], and true intention understanding in case of confusing actions [31]. Intention can be described in several aspects, such as the nature of data (mono-modal, multi-modal, discrete, continuous, etc.), the fusion strategy, and finally the application context. Focusing on the inputs for the intention perception detector, several data channels can be distinguished. First, the most obvious should be the head pose and eye gaze estimation as demonstrated in Martinez *et al.* [31]. A second cue comes straightforward with the context awareness in Clair *et al.* [29]. Bascetta *et al.* [40] used an online prediction of user's trajectory, which can be associated with a user's habit. More cues are related with user's body part orientation. Huber [28] based his work on user's feet position and orientation, Kuan *et al.* [38] used elbow angles and force signals. In order to extract all these features, RGB-D cameras and classical cameras are dominantly chosen for tracking, head pose and eye gaze estimation, but sometimes physiological sensors are used such as muscular electromyography (EMG) and force sensors. Surprisingly, contrary to its pervasive presence, audio sensors/signals have been rarely utilized for intention detection, but rather for user engagement detection in few occasions, e.g., [41].

Evidently, fusing different heterogeneous cues robustifies the estimation step further. When considering multi-modal/multi-cue based intention estimation, the considered fusion/inference module plays an important role in robustness. In the literature, the most promising work utilizes probabilistic frameworks for fusion and inference. For instance Dynamic Bayesian Networks [42], Hidden Markov Models [40, 30], and generic recursive Bayesian filters [31]. Generally, all intention perception modules relate to safety considerations and improved communication. They are used in a large variety of applications: action prediction [29], electric wheelchair's navigation [31], and guiding or resisting a user as part of a rehabilitation process [38]. These types of perception modules are even used in smart public display in order to determine the intention to read an advertisement [28]. Based on these insights, the presented multi-modal *intention-for-interaction* detection scheme

fuses user head orientation, user anterior body orientation, and audio activity – heterogeneous cues that have not been considered altogether before for detecting intention – in a probabilistic framework.

### *2.3. Speech Recognition*

For many years, researchers have been proposing methods to improve speech recognition accuracy by combining automatic speech recognition (ASR) systems. Different fusion techniques are described by Lecouteux *et al.* in [43], some relying on the posterior merging of the recognition hypotheses, others on acoustic cross-adaptation. These authors have also proposed a driven decoding algorithm integrating the outputs of secondary systems in the search algorithm of a primary one, improving their results by 14.5% relative Word Error Rate (WER). In these different cases, each system processes the same audio signal. However, in our context, we consider two different inputs: embedded and external microphones. We also consider that the user can speak from different locations, being close from one microphone and far from another. Depending on the user position, the speech recognition could be more efficient if the input was sometimes taken from one microphone and sometimes taken from another. In many cases, audio and computer vision are used separately. However, approaches based on the user/microphone distance have already been investigated in different contexts. With meeting data, like in Maganti *et al.* [44], where the user is tracked in order to extract his position which is then sent to a beam forming algorithm. The beam is created in the direction of the user in order to remove any external audio interference and retrieve the user's speech only. In this vein, the work by Onuma *et al.* in [8] presents a speech denoising algorithm working with a bank of filters and using the position of the user around two microphones. In both cases, the work is done at the signal level while our fusion approach is carried out at the hypothesis level. In Stiefelhagen *et al.* [9], a study on remote microphones and variable distances was carried out. The data was recorded from different "user-to-microphone" distances and the speech recognition system was adapted, with an adaptive algorithm, to use the correct model knowing the distance. In this work a 10 to 15% of WER reduction was reached. In our work we want to take benefit from the specificity and the diversity of existing sensors, and ASR systems in order to exploit the link between multiple Microphone/Speech API combinations, we also want to use the current distance between the user and the sensor capturing the audio input – User/Microphone distance. These distances are inferred by computer vision with the skeleton fitting algorithm [45]. Since there is no one best global combination for each distance, we propose a framework to fuse information from simultaneously running combinations and dynamically select the most adapted or appropriate configuration as a function of the user/microphone distance. The aim is to obtain a reduced Word Error Rate (WER) during an interaction session. This fusion framework, inspired from the work of Erranz *et al.* in [46] comes to enhance our interaction module and adapt it to more spatial variations in interaction.

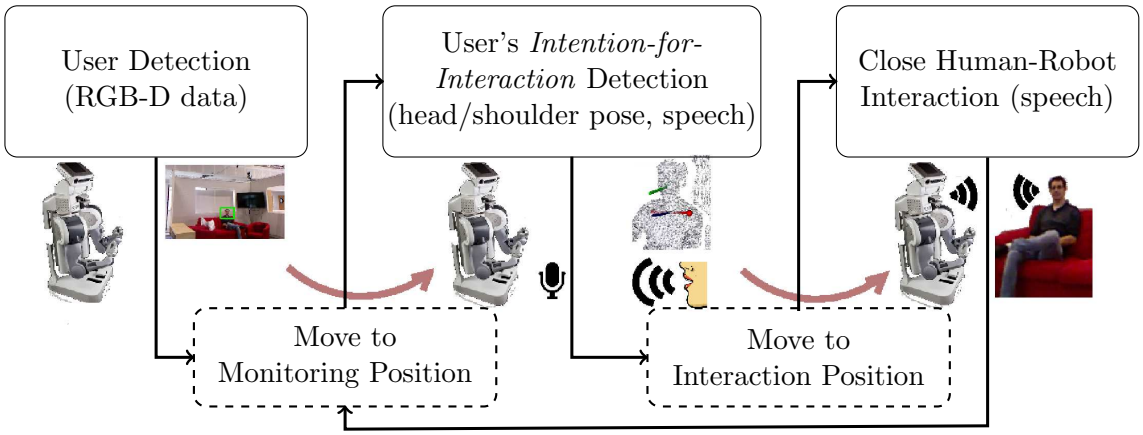



Fig. 1: Framework adopted to realize the complete non-intrusive autonomous assistive robotic system.

## 3. Complete System Overview

This work is based on the overall framework illustrated in Fig. 1. This framework is generic and can be applied to a whole range of scenarios involving HRI. It can be summarized as follows: (1) Find the person in the environment; (2) Go to a predefined monitoring (garage) position and start monitoring the person; (3) Detect the person's intention-for-interaction (desire to interact with the robot), e.g., name calling, directing gaze, etc.; (4) Approach this person, who becomes the user once intention has been detected, by moving to a convenient position for interaction; And (5) begin close Human-Robot interaction. To elaborate further, first, similar to any system involving HRI, our framework begins by finding the location of the *user-to-be* person. This is accomplished with the user detection modality presented in Section 4. Once the robot has detected the person in the room, it does not start interaction directly as we are interested in a non-intrusive robotic behavior (see Section 1). It rather positions itself in a waiting area monitoring the activity of this person. In our framework, the monitoring step aims to detect the user's *intention-for-interaction*, which is presented in Section 5. When detected, the robot approaches the person and starts the envisaged interaction routine described in Section 6. In Fig. 1, the steps involving robot motion (physical transition) are shown in dashed rectangles; the other blocks, which are also pictorially illustrated, involve some form of sensor driven multi-modal perceptual activity. It goes without saying that the close HRI block is self looping unless explicitly stopped by the user, in which case the robot goes back to the monitoring state. As an instance of this framework, we illustrate a demonstrative application in Section 8 where this framework is used to help a user find various objects in the vicinity of the robot through primarily speech based interaction and knowledge about the user environment.

Throughout this work, we use the PR2 robot from Willow Garage Inc. shown in Fig. 2 as the target robotic platform. PR2 is a popular robot that has been used as a test bench by many robotic researchers all over the world. It is composed of an omnidirectional mobile base enabling its movements; two articulated arms; a telescopic spine; two laser sensors,

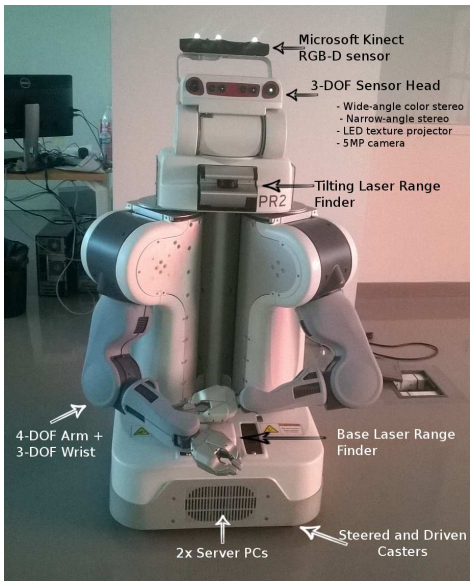



Fig. 2: The PR2 robot with part of its sensors and hardware information.

one tilting on the torso and another fixed on the base; a pan/tilt capable head; and many more components (which are not listed here as they are not used in this work). The head features an RGB-D sensor (Kinect), two wide-angle color cameras, and two narrow-angle monochrome cameras. Its computing power relies primarily on two Quad-Core i7 Xeon Processors (8 cores) with 24 GB RAM. The robot is equipped with a 1.3kWh lithium-ion embedded battery pack that provides 2 hrs approximate runtime – this made cord-less robotic runs possible during experimental sessions. We primarily use the Kinect RGB-D sensor mounted on it for user related perception. Audio data, in addition to the Kinect microphones, is captured using a Samsung Galaxy Note 2 smartphone (running Android 4.2). The smartphone communicates with PR2 via a common Wi-Fi network. Software-wise, our PR2 uses the Groovy instance of the Robotic Operating System (ROS) software architecture [10]. Fig. 3 illustrates a ROS node based architecture of our perception driven assistive robotic system implementation.

Fig. 3 is presented here to provide a general idea of the overall system structure. Each rounded rectangle represents an individual ROS node with the arrows indicating the data flow between the different nodes. The shaded nodes correspond to the nodes we implemented, either from scratch or as a wrapper on top of an existing classical implementation, and the rest denote publicly available implementations. The main perceptual modalities presented in this work rely on the Kinect sensor – RGB-D data for vision related perceptions and audio data for speech related perceptions. The intention-for-interaction detector node, labeled "fused_intention", takes shoulder pose estimation, head pose estimation, and the result of Voice Activity Detection (VAD) to measure the intention of the user. On the other hand, the dialogue module, "audio_interpret" node, takes speech recognition results and gives the sought objects' location information (with the help of the speech synthesizer module). The task coordinator, dubbed "demo_smach_supervisor" node, is the

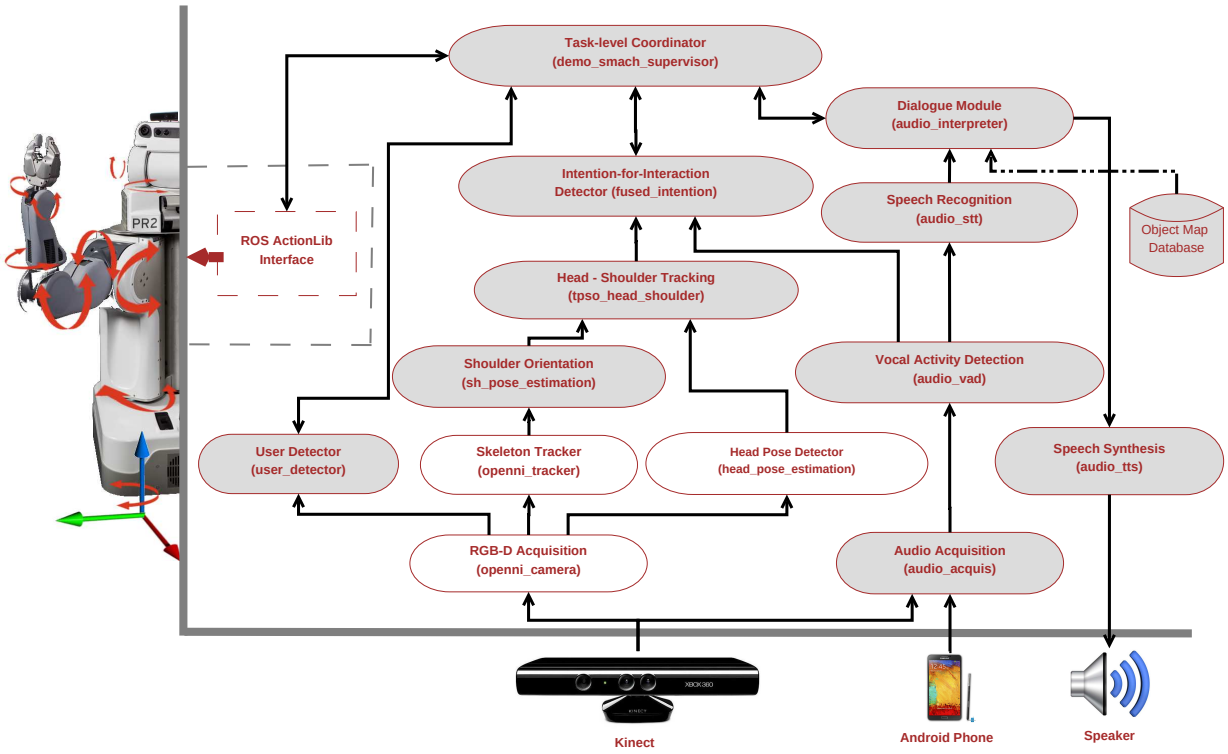



Fig. 3: Illustration of the complete implemented system based on the ROS framework. Each rounded rectangle represents a standalone ROS node and the arrows indicate the message (data) passing pipeline. The shaded nodes correspond to the nodes we implemented or adapted and the rest denote publicly available implementations we utilized.

highest decision maker. It controls the execution of the different steps in the scenario along with their transitions coherently. It acts as the main interface between the robot's native software that controls its actuators and the developed modalities that leads to consistent coordination and execution of the envisaged scenario. It is worth mentioning here that the robot actually has native software architecture that carries the basic functionalities of an autonomous mobile robot. Some of these functionalities include accurate robot localization within a given map, navigation, obstacle avoidance, system diagnosis, etc. Further details are provided in Section 7.

## 4. User Detection

As highlighted in Section 3, one requirement of the presented framework is the correct detection and localization of the user in the room. The user detection module should be able to detect the user whether he/she is sitting on a chair or standing in an upright position. It should also be robust to partial body occlusions as much as possible. Even though during close interaction phases the distance between the robot and the user is less than $5m$ (our depth sensor's range), the adopted use case entails detecting a person at a

farther distance than that, especially when the robot is initially looking for the user in a room.

Primarily relying on the RGB-D sensor mounted on the robot, we have chosen to use a state-of-the-art upper body detector (head and shoulder) recently proposed by Jafari *et al.* [23] with two main improvements. The original detector proposed in [23] has two components: an RGB-D data based upper body detector, and an RGB only full body detector called groundHOG. The upper body detector detects close-by persons upto $5m$ – the sensor's depth operating range, but fails to detect people that are beyond this range. On the other hand, the complementary groundHOG, which is based on Dalal and Triggs HOG detector [18], detects people that are farther effectively – as a person's full body will be in the camera Field Of View (FOV) – but fails to detect close-by people due to possible cropped out (out of camera FOV) body parts (e.g., legs). Both detections are then directly combined and non-maximal suppression is applied to discard overlapping bounding boxes on the image plane. In this work, we use this combined detector by further making two important modifications: (1) we replace groundHOG with our optimized Binary Integer Programming (BIP) based full body detector (BIP-HOG) [47], and (2) we employ a greedy data association algorithm [48] to combine the upper body and full body detections with real world spatial consistency. These modifications improve the overall detection performance as demonstrated in Section 8.3.1.

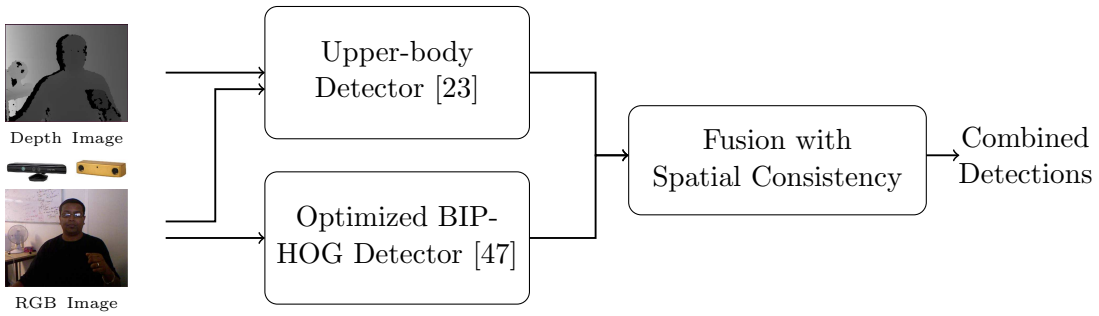



Fig. 4: Block diagram illustrating the utilized user detector.

Fig. 4 shows a block diagram of the combined detector. The upper body detector and the optimized BIP-HOG based detector are used to detect people in the environment independently. Their detections are then combined to try to maximize the chance of detecting all people in the scene. The upper body detector [23] is based on a learned upper body template which is applied on incoming depth images in a sliding window mode. It computes a distance matrix consisting of the Euclidean distance between the template and each overlayed normalized depth image segment, labeling each window whose normalized exponential distance score from the template is above a threshold as a positive instance. Rather than applying the template on all positions, the authors use different techniques to extract Region Of Interest (ROI) based on ground plane estimation and color image segmentation. This step filters out the majority of the image background reducing possible false alarms and speeding up detection rates. The optimized BIP-HOG

detector is a Graphical Processing Unit (GPU) implementation of our detector presented in Mekonnen *et al.* [47]. This detector uses discrete optimization to select a subset of HOG features in a cascade framework that leads to improved frame rate while maintaining comparable detection to the classical Dalal and Triggs HOG detector.

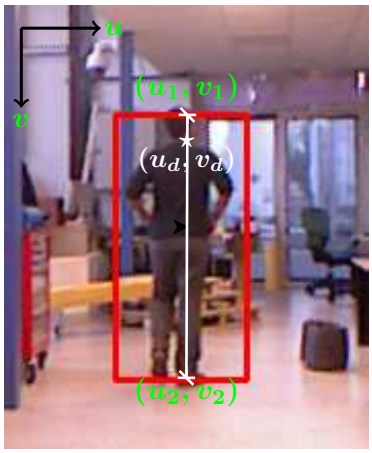



Fig. 5: Illustration of pixel coordinates used to determine an approximate 3D position of a BIP-HOG detection.

To clearly present the adopted fusion strategy in detail, let us denote all the detections obtained from each detector as $\mathcal{D}_{bip} = \left\{\mathcal{D}_{bip}^{(k)}\right\}_{k=1}^{N_{bip}}$ and $\mathcal{D}_{ub} = \left\{\mathcal{D}_{ub}^{(k)}\right\}_{k=1}^{N_{ub}}$, where the suffix *bip* refers to BIP-HOG and *ub* for upper body detections. $N_{bip}$ and $N_{ub}$ stand for the number of detections from BIP-HOG and upper body respectively. Each detection $\mathcal{D}$ is represented as $\{\mathbf{r}, \mathbf{p}, \vartheta\}$, where $\mathbf{r}$ specifies the rectangular bounding box, $\mathbf{p}$ denotes the 3D position, and $\vartheta$ denotes the detection confidence score determined by the detector, of the target. For example, for the $k^{th}$ detection $\mathcal{D}_{bip}^{(k)}$ obtained using the BIP-HOG detector, it is expressed as $\{\mathbf{r}_{bip}^{(k)}, \mathbf{p}_{bip}^{(k)}, \vartheta_{bip}^{(k)}\}$. The $\mathbf{r}, \mathbf{p}$, and $\vartheta$ for the upper body detector are provided by the detector ($\mathbf{p}$ corresponds to the 3D coordinate of the approximate head position determined directly from the depth data). For BIP-HOG, $\mathbf{r}$ and $\vartheta$ are also provided by the detector directly; to determine the corresponding values for $\mathbf{p}$, we rely on the standard camera calibration parameters ($\alpha_u, \alpha_v, u_o, v_o$ [49]) and a $h_p = 1.75m$ average human height assumption. First, given a detection bounding box $\mathbf{r}$, we determine an approximate head position in pixel corrdinate $(u_d, v_d)$ by taking a fixed offset from the top bounding box margin as shown in Fig. 5. Then, the approximate 3D position of the $k$th detection $\mathbf{p}_{bip}^{(k)}$ (corresponding to the pixel $(u_d^{(k)}, v_d^{(k)})$) is determined with the help of the assumed $h_p$ using Eq. (1) (please cross reference relevant variables with Fig. 5). Finally, the fused set of detections $\mathcal{D}_{fus} = \left\{\mathcal{D}_{fus}^{(k)}\right\}_{k=1}^{N_{fus}}$ are determined by combining $\mathcal{D}_{bip}$ and $\mathcal{D}_{up}$ using the greedy data association algorithm outlined in Algorithm 1. The algorithm merges all detections arising from the same spatial position and separately adds the rest if they seem reliable.

$$\mathbf{p}_{bip}^{(k)} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}_{bip}^{(k)} = \frac{h_p \cdot \alpha_v}{\left(v_1^{(k)} - v_2^{(k)}\right)} \cdot \begin{bmatrix} \frac{u_d^{(k)} - u_o}{\alpha_u} \\ \frac{v_d^{(k)} - v_o}{\alpha_v} \\ 1 \end{bmatrix} \tag{1}$$

**Data**: $\mathcal{D}_{bip}, \mathcal{D}_{ub}$
**Result**: $\mathcal{D}_{fus}$
1 $\mathcal{D}_{fus} = \emptyset$, $m \in \{1, ..., |\mathcal{D}_{bip}|\}$, $n \in \{1, ..., |\mathcal{D}_{ub}|\}$
2 $\mathcal{S}(m, n)$: scores for each detection-detection $(\mathcal{D}_{bip}^{(m)}, \mathcal{D}_{ub}^{(n)})$ pair
3 **while** $\mathcal{D}_{bip} \neq \emptyset$ *and* $\mathcal{D}_{up} \neq \emptyset$ **do**
4 $\quad (m^*, n^*) = \arg\min_{m,n} \mathcal{S}(m, n)$
5 $\quad$ **if** $\mathcal{S}(m^*, n^*) < \rho_d$ **then**
6 $\quad\quad d^* \leftarrow \{\mathbf{r}_{ub}^{(n^*)}, \mathbf{p}_{ub}^{(n^*)}, \vartheta_{bip}^{(m^*)} + \vartheta_{ub}^{(n^*)}\}$
7 $\quad\quad \mathcal{D}_{fus} \leftarrow \{\mathcal{D}_{fus} \cup d^*\}$
8 $\quad$ **else**
9 $\quad\quad \mathcal{D}_{fus} \leftarrow \{\mathcal{D}_{fus} \cup \mathcal{D}_{ub}^{(n^*)}\}$
10 $\quad\quad$ **if** $\mathbf{p}_{bip,z}^{(m^*)} > 5m$ **then** $\mathcal{D}_{fus} \leftarrow \{\mathcal{D}_{fus} \cup \mathcal{D}_{bip}^{(m^*)}\}$
11 $\quad$ **end**
12 $\quad \mathcal{D}_{bip} \leftarrow \{\mathcal{D}_{bip} \backslash \mathcal{D}_{bip}^{(m^*)}\}$, $\mathcal{D}_{ub} \leftarrow \{\mathcal{D}_{ub} \backslash \mathcal{D}_{ub}^{(n^*)}\}$
13 **end**
14 **while** $\mathcal{D}_{bip} \neq \emptyset$ **do**
15 $\quad$ **if** $\mathbf{p}_{bip,z}^{(m)} > 5m$ **then** $\mathcal{D}_{fus} \leftarrow \{\mathcal{D}_{fus} \cup \mathcal{D}_{bip}^{(m)}\}$
16 $\quad \mathcal{D}_{bip} \leftarrow \{\mathcal{D}_{bip} \backslash \mathcal{D}_{bip}^{(m)}\}$
17 **end**
18 **while** $\mathcal{D}_{up} \neq \emptyset$ **do**
19 $\quad \mathcal{D}_{fus} \leftarrow \{\mathcal{D}_{fus} \cup \mathcal{D}_{ub}^{(n)}\}$, $\mathcal{D}_{ub} \leftarrow \{\mathcal{D}_{ub} \backslash \mathcal{D}_{ub}^{(n)}\}$
20 **end**
21 **return** $\mathcal{D}_{fus}$

**Algorithm 1:** Algorithm for people detection fusion with spatial consistency.

Referring to Algorithm 1, the fusion algorithm begins by computing a distance score matrix $\mathcal{S}(m, n)$ for each detection-detection $(\mathcal{D}_{bip}^{(m)}, \mathcal{D}_{ub}^{(n)})$ pair as $\mathcal{S}(m, n) = \|\mathbf{p}_{bip}^{(m)} - \mathbf{p}_{ub}^{(n)}\|_2$ (line 2). Then two detections $m^*$ and $n^*$ with the least score are associated if their distance score is less than the threshold $\rho_d$ (lines 4-8). This associated detection is added to the set of fused detections with $\mathbf{r}$ and $\mathbf{p}$ taken from the upper body detector and a detection score set to the sum of the coupled detection scores. The corresponding detections are removed from the *bip* and *ub* detection sets. If associated detections have scores higher than the threshold (lines 8-12), the upper body detection is directly added to the fused detection set, whereas the BIP-HOG detection is added only if it is farther than $5m$ – upper body is more reliable in close range while BIP-HOG is not. Finally, all unassociated upper body detections are directly added to fused detection, whereas unassociated BIP-HOG detections are added

only if they are farther than $5m$ (lines 14-20). Comparative experimental evaluation of this fused detector and its constituents is detailed in Section 8.3.1.

## 5. *Intention-for-Interaction* Detection

Fig. 6 shows a block diagram of the framework used to estimate user's *intention-for-interaction* using an RGB-D camera (e.g., Kinect, stereo rig) and an audio sensor (e.g., smartphone, tablet, microphone, etc.). It estimates the user's intention based on three important cues: user's line of sight – inferred from head pose; user's anterior body direction – determined from shoulder orientation; and speech used to draw attention – captured via Voice Activity Detection (VAD). The head pose detection and shoulder orientation detection modules rely on depth images. The detection outputs are further filtered using a Particle Swarm Optimization inspired Tracker (PSOT) presented in Mollaret *et al.* [50]. Both the VAD and tracker output are considered as observation inputs and are fused to provide a probabilistic intention estimate using a Hidden Markov Model (HMM).

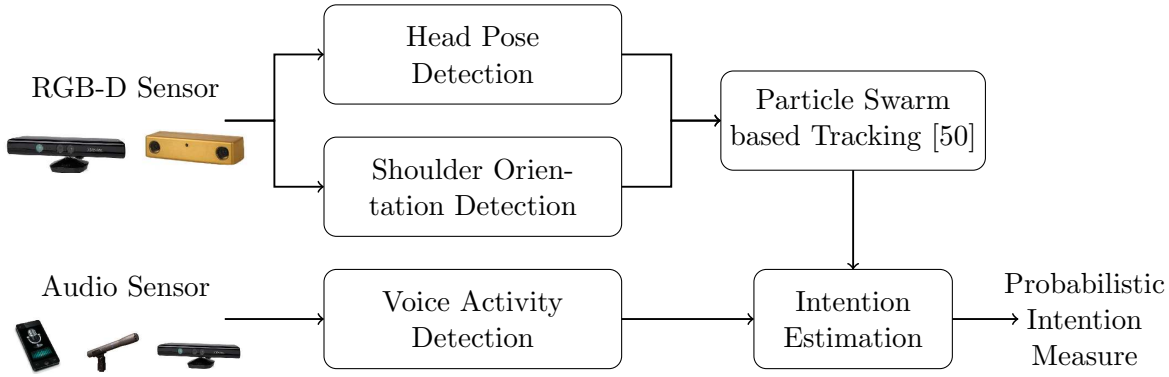



Fig. 6: Block diagram depicting the *intention-for-interaction* detection component.

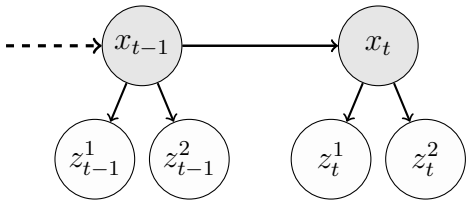



Fig. 7: Probabilistic graphical model used for intention estimation.

The probabilistic graphical model depicted in Fig. 7 illustrates the relationship between the hidden variables, $x_t$ and $x_{t-1}$, which are the intention indicators at time $t$ and $t-1$ respectively, and the observation variables $z^1_t$, $z^2_t$, $z^1_{t-1}$, and $z^2_{t-1}$. The intention indicator $x_t$ is a discrete random variable that takes on values $\{intent, \neg intent\}$ at time $t$. The

observation variables $z_t^1$, and $z_t^2$ are defined according to Eq. (2).

$$z_t^1 = \begin{bmatrix} \theta_h \\ \phi_h \\ \theta_{sh} \end{bmatrix}; \qquad z_t^2 \in \{vad, \neg vad\} \tag{2}$$

$z_t^1$ is a continuous vector-valued random variable that represents the head orientation (head yaw $\theta_h$ and pitch $\phi_h$) and shoulder orientation (shoulder yaw $\theta_{sh}$) observations from the particle swarm based tracker that provides estimated head pose (position and orientation) in space, and shoulder orientation with respect to the vertical plane of the camera optical frame in space (yaw). $z_t^2$, on the other hand, is a discrete random variable that takes on either $vad$ or $\neg vad$ to represent the observation from the voice activity detection module. Further description of these two observation variables along with their associated probability distributions are provided in Sections 5.1 and 5.2 respectively.

With the assumption that the observations are conditionally independent given the state (encoded in the graphical model in Fig. 7), and making use of Bayes rule, the posterior probability distribution over the state $P\left(x_t|Z_{1:t}\right)$ given all measurements upto time $t$, $Z_{1:t}$, can be expressed with Eq. (3).

$$P\left(x_t|Z_{1:t}\right) = \eta P\left(z_t^1|x_t\right) P\left(z_t^2|x_t\right) \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}|Z_{1:t-1}), \tag{3}$$

where $P(x_t|x_{t-1})$ is the state transition (dynamics) distribution and $\eta$ is a normalization factor. Again here both $x$ and $z^2$ are discrete random variables that take on values $\{intent, \neg intent\}$ and $\{vad, \neg vad\}$ respectively, whereas $z^1$ takes on continuous values given the PSOT tracker output.

### *5.1. Head and Shoulder Pose Estimation*

This observation is based on head pose estimation, shoulder orientation estimation, and particle swarm optimization based tracking for filtered estimates as explained in the previous section.

*Head Pose Estimation.* This module is based on the work of Fanelli *et al.* [51] which formulates the pose estimation as a regression problem and uses random regression forests on depth images from an RGB-D sensor. This choice is motivated by regression forests capability to handle large training datasets. The regression is based on difference of rectangular patches resembling the generalized Haar-like features in [52]. The training is done using the Biwi Kinect Head Pose Dataset [51]. The 6D state vector $[\mathrm{x'}_h, \mathrm{y'}_h, \mathrm{z'}_h, \theta'_h, \phi'_h, \psi'_h]^\mathsf{T}$ contains the 3D head position and the three orientation angles relative to the sensor. The claimed precision in the paper is $5.7^o$ mean error in yaw estimation with $15.2^o$ standard deviation, and $5.1^o$ mean error in pitch estimation with $4.9^o$ standard deviation. Additionally, head pose is detected with a mean error of $13.4mm$ ($\pm 21.1mm$).This mode works best with close-by subjects, subjects placed at a distance of $1.5m$ to $2.0m$.

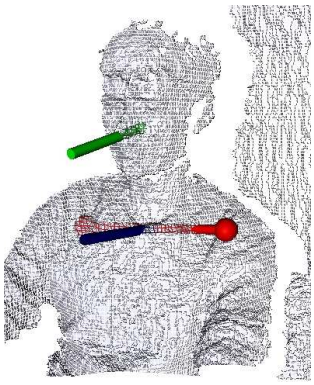


Fig. 8: The head pose is displayed with the green cylinder (head) on the point cloud, while shoulders are displayed in red and their orientation in dark blue (below the neck).

*Shoulder Orientation Estimation.* For this we primarily rely on OpenNI library [45] which provides a fitted skeleton model of the user based on the depth data. Then, using simple geometry, the user's shoulder orientation is obtained by computing the vector between the left and right shoulder joint pose determined from the fitted skeleton. The shoulder orientation is expressed with respect to the RGB-D sensor parallel optical plane providing a yaw angle $\theta'_{sh}$ for the tracking step. Illustrative estimated shoulder and head orientations are displayed in Fig. 8. Following the Kinect – the specific RGB-D camera used in this work – characteristics and OpenNI library, the skeleton tracking algorithm works up to a range of $4.0m$.

The head and shoulder poses provided by the above two modules are computed frame by frame without any temporal link. It is also possible to have missing estimates from any of the modules at any times. To alleviate that and provide a smoothed continuous estimate, hence filtering, we make use of our PSOT tracker. Given head pose and shoulder orientation in the form of $s_d = [\mathrm{x'}_h, \mathrm{y'}_h, \mathrm{z'}_h, \theta'_h, \phi'_h, \psi'_h, \theta'_{sh}]^\mathsf{T}$ from the head pose and shoulder orientation estimation modules at time frame $t$, PSOT is used to determine spatio-temporal posterior point estimates (filtered estimates) of head pose and shoulder orientation at time frame $t$ with the state vector of the PSOT tracker represented as $s = [\mathrm{x}_h, \mathrm{y}_h, \mathrm{z}_h, \theta_h, \phi_h, \psi_h, \theta_{sh}]^\mathsf{T}$. The PSOT is a filter adapted from Particle Swarm Optimization (PSO) [53] that combines interesting amenities of PSO with Particle Filter [54], namely state dynamic model for improved tracking performance. Contrary to PSO, there is only *one* iteration of the PSOT at each time frame (there is no adaptive learning). It has a linear complexity with the number of particles used (which is 150 in this work), and it does not need an expensive computing resource to work in real time. Also contrary to other particle based algorithms, particles interact with each other with a *social* and *cognitive* component in the update step. This behavior leads to a more efficient estimation of state as there is no particle degeneration phenomenon. For the head pose and shoulder orientation estimation, the adopted tracker uses a random walk dynamic model and a multivariate Gaussian model with a diagonal covariance matrix as the observation model in the fitness evaluation (please refer to [50] for details).

The distribution $P(z_t^1|x_t)$ is derived based on the tracker output, i.e., based on $\theta_h, \phi_h, \theta_{sh}$ angles. These angles are represented in such a way that when the user is looking right into the optical frame with their anterior body oriented parallel to the image plane, all angles are 0. With this in mind, $P(z_t^1|x_t)$ is represented as a multivariate normal distribution, i.e., $P(z_t^1|x_t) = \mathcal{N}(z_t^1; 0, \Sigma)$ with $z_t^1 = [\theta_h, \phi_h, \theta_{sh}]^\intercal$. The covariance matrix $\Sigma$ is a diagonal matrix; though not applied here, its values can be varied using the tracked head position.

### *5.2. Voice Activity Detection (VAD)*

Audio signal is used for intention detection based on user voice activity detection. Users have the tendency to talk to a robot when they want its attention. Taking advantage of this, we denote the onset of a voice activity as one indicator for user's *intention-for-interaction*. In this work we rely on the voice activity detection module from PocketSphinx C library[3]. This algorithm is based on signal energy. It flags the given audio frame as containing speech elements if the signal energy is above a predefined threshold. Since signal energy is affected by the noise in the environment, the implementation in PocketSphinx does an initial calibration stage so as to best separate signal from stationary noise using a statistical-based noise removal method. Depending on the environment ambient noise (robot noise and room noise), the VAD is estimated properly up to $2m$ from the audio sensor.

The observation from the VAD module is represented by the random variable $z_t^2$ at time $t$ taking discrete values $\{vad, \neg vad\}$. Since both $z_t^2$ and $x_t$ take binary discrete values, the associated likelihood distribution $P(z_t^2|x_t)$ is simply represented by four probability values.

## 6. Close Human-Robot Interaction

During this phase, the human and the robot are engaged in basic interaction. The person asks for assistance, and the robot answers by providing a useful response or assistance. This phase is implemented by a static state machine dialogue module that manages the interaction. Each specific question/request coming from the user can trigger transitions to different states leading to robotic service provision.

The close interaction is started when the user's *intention-for-interaction* has been detected. The interaction stops when the user explicitly tells the robot that he/she does not need it anymore, for example, by saying the French equivalents of, "thank you, goodbye!", "thank you, all is ok now", "goodbye", and the like. The robot then goes back to its garage position and resumes the user's monitoring state until the next user's *intention-for-interaction* is detected.

As an exemplar instance, we implement a scenario where the robot, upon specific inquiry, helps to find objects that the user has forgotten. Hence, in this specific case the goal of the dialogue is to retrieve targeted object. The basic semantic analysis is done by using keywords in the speech recognition hypothesis and a vote to find the preponderant meaning between the N-Best hypothesis from the interpretation. For example, if the interpretation set is $I_s = [FIND, FIND, GREET, UNKNOWN, UNKOWN, FIND]$, the

[3] `http://cmusphinx.sourceforge.net/`

interpretation result will be equal to $I = FIND$, moving the interaction to the next state accordingly. The verbal answer will be given to the user by the Google Text To Speech API which contains a set of predefined sentences.

Our contributions in this part do not lie in the interaction per se, but in the improvement of the user's perception by using multiple input audio streams and multiple ASR systems. In this work, we use the CMU API PocketSphinx and the Google Speech API as ASR systems. We also use two microphones: the embedded microphone of the Kinect sensor, and the microphone from a Samsung Galaxy Note 2 smartphone. We create four combinations of microphone/ASR system, but the global framework could easily be generalized to more combinations with the addition of more microphones or ASR systems. Based on the work of [46], a fusion framework is created as the Bayesian network model illustrated in Fig. 9. The idea is that, for each set of distances $d$ between the user and the microphones, and for each set of hypothesis $U$ coming from each combination, we want to find the best combination $S$. This is illustrated by Eq. (4) where $P(S|d, U)$ represents the probability of the combination $S$ being the most reliable knowing the distances and the hypothesis. $P(S|d)$ stands for the probability of the combination $S$ being the most reliable knowing the distances alone, and $[U_S \neq \emptyset]$ represents the presence of a hypothesis for the combination $S$. $[U_S \neq \emptyset] = 1$ if there is a hypothesis or $[U_S \neq \emptyset] = 0$ otherwise.

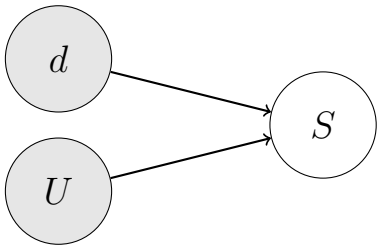



Fig. 9: Bayesian network for homogeneous fusion.

$$\underset{S}{argmax}(P(S|d, U)) \propto \underset{S}{argmax}(P(S|d)[U_S \neq \emptyset]) \tag{4}$$

In order to dynamically select the best combination $S$ given the user distances and the hypothesis, we first learn the density $P(S|d)$ which is inversely proportional to the Word Error Rate (WER) function of the distance. This measure will be further explained in Section 8.2. The density is learned with the dataset presented in Section 8.1 and the WER values are interpolated using a $3^{rd}$ degree polynomial. This density is then used in Algorithm 2, which is an implementation of the Bayesian network previously described.

## 7. Task-level Coordination and Implementation Details

Deploying the complete system described in this paper on a robotic platform is a complex task which needs a coordinating tool to start and stop services whenever required and to make transitions between different services smoothly with proper exception propagation. These different services could be thought of as individual robotic tasks that can

**Result:** $\underset{S}{argmax}(P(S|d,U))$=Fusion($d$, $U$, $P(S|d)$)
$P_{max} = 0$
**for** *each system* $S$ **do**
    Compute $[U_S \neq \emptyset]$
    Compute $P(S|d,U) \propto P(S|d)[U_S \neq \emptyset]$
    **if** $P(S|d,U) > P_{max}$ **then**
        $P_{max} = P(S|d,U)$
    **end**
**end**
Compute $S = \underset{S}{argmax}(P(S|d,U))$
**return** $S$

**Algorithm 2:** Fusion algorithm based on a Bayesian network for homogeneous fusion.

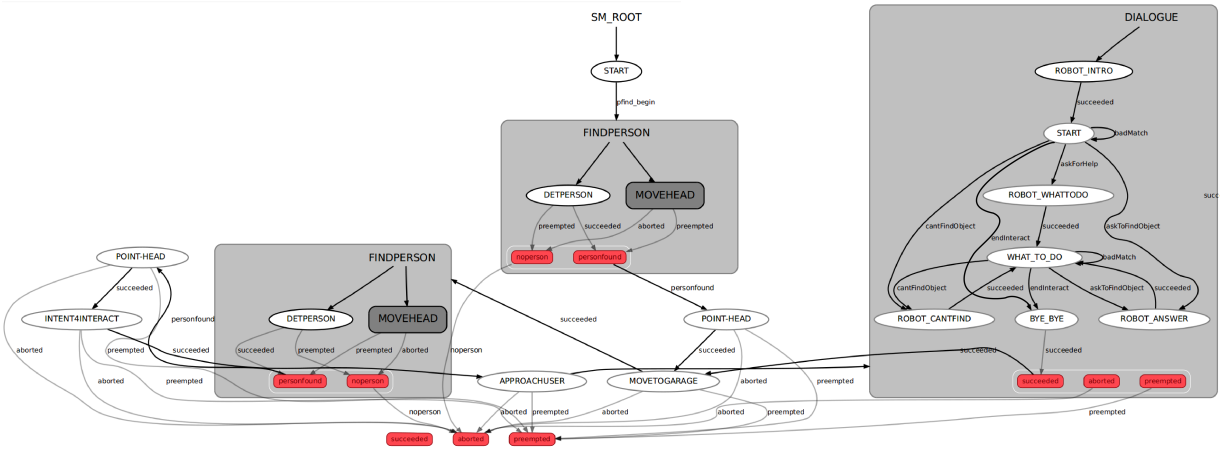



Fig. 10: Visualization of the task-level coordination state machine constructed using *smach* [55]. Each shaded state machine is a container by itself (consisting of a sub state machine), the outputs of which are shown in red rectangular blocks.

be coordinated to create and launch a complete working robotic demo. For example, in this work, user detection, user's intention detection, human-robot interaction, and robot control (robot movement or specific joint control) can make up individual tasks. A popular tool, especially for ROS based systems, adopted here is *smach* [55]. *Smach* is a task-level architecture for rapidly creating complex robot behavior that has been successfully used in many robotic applications, e.g., [58, 59]. The complete system spanning from user detection to human-robot interaction is coordinated via different state containers including finite state machines, concurrent state machines, and action state machines. The concurrent state machine, for example, is used to launch the user detector action in parallel with a robot head pan action to scan the room for possible presence of a person, assuming the person is not obscured by any furniture in the environment. Fig. 10 shows a trimmed down

Table 1: List of the main tasks and associated sub-tasks deployed to realize the assistive robotic system presented. Boldface nodes indicate the ones developed in this work.

| Main task | Main *smach* state machine | Sub-tasks | Corresponding ROS node (cf. Fig. 3) | Executing machine |
|---|---|---|---|---|
| User Detection | FINDPERSON | User detection | **user_detector** | PR2_c1 |
| | | RGB-D Kinect streaming | openni_camera [56] | PR2_c1 |
| | | Move Robot head | ROS actionlib interface | PR2_c1 |
| Move to Monitoring Position | MOVETOGARAGE | Robot localization | pr2_2dnav [57] | PR2_c2 |
| | | Navigation to goal | ROS actionlib interface | PR2_c2 |
| | | Obstace avoidance | ROS actionlib interface | PR2_c2 |
| *Intention-for-Interaction* Detection | INTENT4INTERACT | RGB-D Kinect streaming | openni_camera [56] | PR2_c1 |
| | | Head orientation | head_pose_estimation [51] | PR2_c1 |
| | | Shoulder orientation | **sh_pose_estimation** | PR2_c1 |
| | | PSOT tracking | **tpso_head_shoulder** | PR2_c1 |
| | | Audio (Android/Kinect) streaming | **audio_acquis** | Smartphone/PR2_c1 |
| | | VAD detection | **audio_vad** | PR2_c1 |
| Move to Interaction Position | APPROACHUSER | Robot localization | pr2_2dnav [57] | PR2_c2 |
| | | Navigation to goal | ROS actionlib interface | PR2_c2 |
| | | Obstace avoidance | ROS actionlib interface | PR2_c2 |
| Close Human-Robot Interaction | DIALOGUE | Audio (Android/Kinect) streaming | **audio_acquis** | Smartphone/PR2_c1 |
| | | VAD detection | **audio_vad** | PR2_c1 |
| | | Speech recognition | **audio_stt** | PR2_c1 |
| | | Speech synthesis | **audio_tts** | PR2_c1 |
| | | Speech interpretation | **audio_interpreter** | PR2_c1 |
| Point robot head | POINT-HEAD | Move Robot head | ROS actionlib interface | PR2_c2 |
| Task-level coordination | All Fig. 10 | – | **demo_smach_supervisor** | PR2_c2 |
| Data visualization | – | – | ROS tools (rviz,rqt_plot,etc) | Workstation |

version (without subtle intermediary states that make adjustment for detected user location) of the *smach* based state machine for this specific application. The 'FINDPERSON' concurrent machine handles the search for user in the room, and the 'DIALOGUE' state machine shows a representation of the different states in the close HRI interaction geared by a to-and-fro dialogue.

The main tasks and associated sub-tasks are categorized and listed in Table 1. Each main task has an associated *smach* state machine shown in Fig. 10. The sub-tasks represented as ROS node implementations are also shown (cross reference with Fig. 3). The boldface nodes indicate the ones developed in line with this work, while the rest are publicly available implementations. The main components of our framework, as presented in Section 3, have corresponding *smach* state machines labeled as FINDPERSON, MOVETOGARAGE, INTENT4INTERACT, APPROACHUSER, and DIALOGUE respectively. The POINT-HEAD state machine is an intermediary state used to point the head of the robot towards the person before the INTENT4INTERACT and DIALOGUE phases. The corresponding sub-tasks are also detailed clearly showing the link with the corresponding ROS node. For all the robot actions that involve an actuator (robot motion, head movement), the ROS actionlib interface specifically developed for the PR2 is used [60]. All the modules presented in Table 1 execute across four heterogeneous machines: two Intel(R) Quad-Core i7 Xeon machines on the PR2 (PR2_c1 and PR2_c2), an Intel(R) Core(TM) i7-2720QM workstation, and a Samsung Galaxy Note 2 smartphone, thanks to ROS's abstraction that enables seamless TCP/IP communication between these machines. The robot and the smartphone are connected to the network via Wi-Fi connections, whereas

the workstation is connected via a fixed network cable. In the current implementation, the position of the objects are assumed to be known a priori. This information is stored in the Object Map Database. It contains the position of the various objects in the environment, for example, as "The remote is under the table". For improved and more realistic scenarios, it will suffice just to update this database with information of detected objects should an automated object detection module become available in the future.

Regarding the audio data from the smartphone, the real-time audio signal used by the ASR system is streamed to the robot from an Android application. This application is developed in the context of this work using the *rosjava* Android build tool [61] to both use the phone as an input device and a debugging tool when operating the robot. Therefore, the smartphone can be used to visualize the speech recognition hypothesis and the current state of the robot. The audio is streamed to the network with 512 sample buffers of 16bits quantization level. A lock button is set to prevent any unwanted touches on the screen (e.g., from stopping the audio stream accidentally). A screen snapshot of the Android application user interface is shown in Fig. 11.

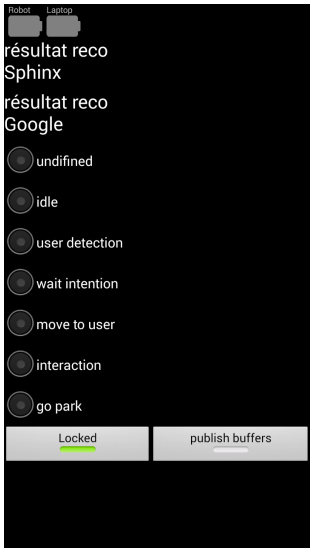



Fig. 11: A screen snapshot of the developed Android application user interface.

## 8. Experiments and Results

This section presents the different experimental evaluations carried out to validate and demonstrate the proposed multi-modal perception driven assistive robotic framework along with the obtained results. It is categorized under four headings: (1) the various datasets used for evaluation in Section 8.1; (2) the adopted standard evaluation metrics in Section 8.2; (3) details of the obtained results in Section 8.3; and (4) the conducted user study along with analysis of observations made therein in Section 8.4.

### 8.1. Datasets

The various components of the presented system rely on multi-modal data acquired using a Kinect sensor (RGB-D and audio) and an Android mobile device (audio). Due to the multi-modal nature and hence the difficulty in finding a public dataset tailored for our application, we have used several proprietary datasets. The datasets include RGB-D image frames for user detection, combined RGB-D and audio for intention detection, and audio only for the speech recognition part. All the datasets described in this section are purposely collected by us for consequent experimental evaluations. Table 2 summarizes the ones used for evaluating the user detection, and intention detection components.

Table 2: Summary of the different datasets used for evaluating user detection, tracker precision, and intention detection components. A partial view of the cluttered robotic lab-1 is shown in Fig. 14, and that of the cluttered robotics lab-2 in Figs. 15 (bottom row), 19, and 20. The cluttered office scene is shown in Fig. 15 (top row).

| Name | Mode | Image frames | Duration | Max persons | Ground truth | Environment |
|---|---|---|---|---|---|---|
| UserDet-DT | RGB-D | 235 | — | 4 | Manual annotation | Cluttered robotic lab-1 |
| Intent-DT1 | RGB-D + Audio | 4230 | 141s | 1 | Manual annotation | Cluttered office |
| Intent-DT2 | RGB-D + Audio | 3930 | 131s | 1 | Manual annotation | Cluttered robotic lab-2 |
| Intent-DT3 | RGB-D + Audio | 3180 | 106s | 1 | Manual annotation | Cluttered robotic lab-2 |

*User Detection Dataset (UserDet-DT).* To evaluate the user detection module, we use a proprietary RGB-D dataset consisting of 235 image frames intermittently acquired using a Kinect sensor mounted on our mobile robot in our robotic lab (cluttered robotic lab-1 in Table 2). Each frame of the dataset contains at least one person, the majority contain two persons, and a few image frames feature four persons (the maximum number of persons per image frame). In terms of detection targets, there are a total of 521 target occurrences out of which 182 are situated farther than $5m$ from the Kinect sensor. Even though the application context in this work is single user detection, we evaluate the detection module with this dataset containing multiple persons per image frame to characterize its detection capability thoroughly as is done in standard people detection literature [16]. This will increase the chance of testing the detector under broadly varying conditions, e.g., inter-person occlusions, deformations due to articulations, and different person postures (walking, sitting, standing, etc.). Additionally, it will help demonstrate its capability and potential to be used in multi-user scenarios. The dataset is manually annotated to create a complete ground truth by delineating each person in each image frame with rectangular bounding boxes.

*Intention Evaluation Dataset (Intent-DT).* For user's intention detection evaluation, we acquire three separate datasets: Intent-DT1, Intent-DT2, and Intent-DT3. Intent-DT1 is acquired merely in an office setting using a standalone Kinect and smartphone, whereas the other two, Intent-DT2 and Intent-DT3, are acquired using the PR2 in a robotic experimental area. Their lengths vary between 3180 and 4230 image frames (acquired at 30 fps).

The datasets constitute of RGB-D and audio streams. In all cases, the user seats at an approximate distance of $1.5m$ to $2m$ from the RGB-D sensor and demonstrates *intention-for-interaction* by facing the Kinect sensor and/or speaking. The datasets are manually annotated to mark intention active regions with the help of the user.

*Speech Recognition Evaluation Dataset.* To evaluate the microphone/speech recognition API fusion framework, a proprietary dataset (corpus) was collected dedicated to this study involving four speakers. In this corpus, each participant utters 17 French sentences that have been selected to fit out HRI context repeatedly. Each sentence is repeated three times by each of the four speakers at four different distances: $10cm$, $50cm$, $1m$, and $2m$. To further clarify, during the acquisition session, the same speaker repeats the same sentence a total of 12 times, leading to 204 sentences per speaker. When the user/microphone distance is greater than two meters, the Word Error Rate (WER), see Section 8.2, usually reaches a maximum and no hypothesis is produced by any combination. The acquisition is iterated with a total of four speakers. All in all, this dataset contains 17 sentences uttered 12 times by four speakers – resulting in 816 total number of recorded sentences spanning a total time of 34'03".

### *8.2. Evaluation Metrics*

To quantify the performance of the different perceptual blocks utilized in this work, we make use of various well established metrics.

*User Detector Performance.* To evaluate the performance of the user detector, we use Miss Rate ($\mathsf{MR}$) and average False Positives Per Image ($\mathsf{FPPI}$) metrics as defined in Eqs. (5) and (6) respectively. Both the $\mathsf{MR}$ and $\mathsf{FPPI}$ are the most widely established evaluation metrics in people detection [16]. $\mathsf{MR}$ indicates the proportion of people not detected in the entire dataset (it is "1−True Positive Rate"). $\mathsf{FPPI}$ on the other hand indicates the false positives averaged over the total number of image frames – it signifies how many false positives are likely to occur when applied on a new image frame.

$$\mathsf{MR} = 1 - \sum_{i=1}^{N_f} \frac{\mathsf{TP}_i}{\mathsf{TP}_i + \mathsf{FN}_i} \tag{5}$$

$$\mathsf{FPPI} = \frac{1}{N_f} \sum_{i=1}^{N_f} \mathsf{FP}_i \tag{6}$$

In these Eqs. (5) and (6), $i$ represents the image frame, $N_f$ the total number of image frames in the dataset, and $\mathsf{TP}_i, \mathsf{FP}_i, \mathsf{FN}_i$ denote True Positives, False Positives, and False Negatives at the $i$th test image frame respectively. The evaluation generally results in an $\mathsf{MR}$ – $\mathsf{FPPI}$ plot in log-log scale that is generated by varying the threshold ($\vartheta_o$) on the final detector score $\vartheta$. For example, increasing the threshold $\vartheta_o$ will increase the $\mathsf{MR}$, less windows will be detected, but it also reduces the number of false positives (hence the $\mathsf{FPPI}$), and vice-versa. $\vartheta_o$ is a tunable parameter that defines the operating point of the

detector. To summarize the performance of the detector, the log-average miss rate is used. It is computed by averaging miss rate at nine FPPI rates evenly spaced in log-space in the range $10^{-2}$ to $10^{0}$ (for curves that end before reaching a given FPPI rate, the minimum miss rate achieved is used) [16].

*Intention Evaluation.* The final inference engine of the *intention-for-interaction* is based on an HMM. Hence, to quantify its detection performance, we make use of various metrics mostly used in HMM applications in the literature (e.g., [62]) listed below. For easier mathematical notation, let us represent the different measures as follows: let us define the indicator function $\mathcal{I}(\mathtt{x},\mathtt{y})$ in Eq. (7) assuming $\mathtt{x},\mathtt{y} \in \{intent, \neg intent\}$ to be used to flag a true positive, a false positive, and a true negative. Let $\mathtt{I}_t, \mathtt{G}_t \in \{intent, \neg intent\}$ represent the intention label assigned by the intention detection module and the ground truth intention label at time frame $t$ respectively. Let $\mathtt{T}$ represent the entire length of the evaluation dataset which consists of $\mathtt{T}_{intent} = \{\mathtt{T}_{intent,j}\}_{j=1}^{\mathtt{N_{IT}}}$ and $\mathtt{T}_{\neg intent} = \{\mathtt{T}_{\neg intent,j}\}_{j=1}^{\mathtt{N_{NT}}}$ disjoint spans where there is and there is no user intention in the ground truth annotation respectively. $\mathtt{N_{IT}}$ is the total number of such spans where there is user intention, and $\mathtt{N_{NT}}$ is the total number of such spans where is no intention. The $|.|$ indicates the duration of a time span, e.g., $|\mathtt{T}_{intent,j}|$ is the duration of the $j$th intention marked time span. Consequently, $\mathtt{T} = \mathtt{T}_{intent} \cup \mathtt{T}_{\neg intent}$ is satisfied. Finally, let the set $\mathtt{J}^*$ stand to represent the set of indexes of such time spans where an intention is correctly detected at some point in the span, i.e., $\mathtt{J}^* = \{j | \sum_{t \in \mathtt{T}_{intent,j}} \mathcal{I}(\mathtt{I}_t, \mathtt{G}_t) > 0\}$.

$$\mathcal{I}(\mathtt{x},\mathtt{y}) = \begin{cases} 1, & \text{if } \mathtt{x} = intent \text{ and } \mathtt{y} = intent \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

- True Positive Rate (TPR): It is defined as the ratio of correct intention detection (in accordance with the ground truth) to that of total intention tagged ($\mathtt{G}_t = intent$) data frames. It is formally expressed using Eq. (8).

$$\mathsf{TPR} = \frac{1}{\sum_{t \in \mathtt{T}} \mathcal{I}(\mathtt{G}_t, \mathtt{G}_t)} \sum_{t \in \mathtt{T}} \mathcal{I}(\mathtt{I}_t, \mathtt{G}_t) \tag{8}$$

- False Alarm Rate (FAR): It is the ratio of the number of observation data frames for which the detection output flags an intention where there is none in the ground truth, to that of the total number of no intention data frames as described with Eq. (9).

$$\mathsf{FAR} = \frac{1}{\sum_{t \in \mathtt{T}} \mathcal{I}(\neg\mathtt{G}_t, \neg\mathtt{G}_t)} \sum_{t \in \mathtt{T}} \mathcal{I}(\mathtt{I}_t, \neg\mathtt{G}_t) \tag{9}$$

- Average Early Detection (AED): Given an observation span $j$ labeled with *intent*, $\mathtt{T}_{intent,j}$, the early detection time is the discrete time $t_{d,j}$ the system took to correctly detection an intention. The AED, then, is computed by averaging the normalized

early detection time over all correctly detected intentions. Using $\mathtt{J}^*$, the AED can be expressed as in Eq. (10).

$$\mathsf{AED} = \sum_{j \in \mathtt{J}^*} \frac{t_{d,j}}{|\mathtt{T}_{intent,j}|} \tag{10}$$

- Average Correct Duration (ACD): It is defined in a similar fashion as AED, but instead considers the correctly detected intention duration. If $c_{d,j}$ represents the discrete time span (duration) through which an intention is correctly detected, the ACD is determined by averaging over all correctly detected intentions as in Eq. (11).

$$\mathsf{ACD} = \sum_{j \in \mathtt{J}^*} \frac{c_{d,j}}{|\mathtt{T}_{intent,j}|} \tag{11}$$

*Speech Recognition.* The evaluation of the speech recognition module is based on the Word Error Rate (WER) metric. The classic Word Error Rate (WER) is defined by: $WER = \frac{S+D+I}{N}$ where S is the number of word substitutions, D is the number of word deletions, I is the number of word insertions and N is the number of words in the reference.

In order to learn $P(S|d)$ that is used by the fusion Algorithm 2, the WER of each combination function of the distance is first learned. Two sets of metrics are defined and computed on the dataset presented in Section 8.1 (predefined sentences uttered at different distances) used for the $P(S|d)$ estimation. These metrics are presented below.

- total WER (T-WER): in this case, the WER is computed for all utterances, even if there is no hypothesis found by a speech recognition API. This is the classic mean WER. This metric is used to compare different systems.

- utterance WER (U-WER): in this case, the WER is computed for all utterances, only if there are hypotheses found, which means that when nothing has been recognized, the hypothesis is discarded. This can be seen as the precision of the recognition of a speech recognition API. This metric is used to learn the mapping matrix.

Considering the N-Best results of each combination, we defined 3 sub-categories of metrics for T-WER and U-WER.

- "best WER": represents the mean WER computed considering hypotheses with the smallest WER of each spoken sentence.

- "worst WER": represents the mean WER computed considering hypotheses with the highest WER of each spoken sentence.

- "likely WER": represents the mean WER computed considering hypotheses with the highest likelihood from each spoken sentence, *i.e.*, the first of the N-Best hypothesis.

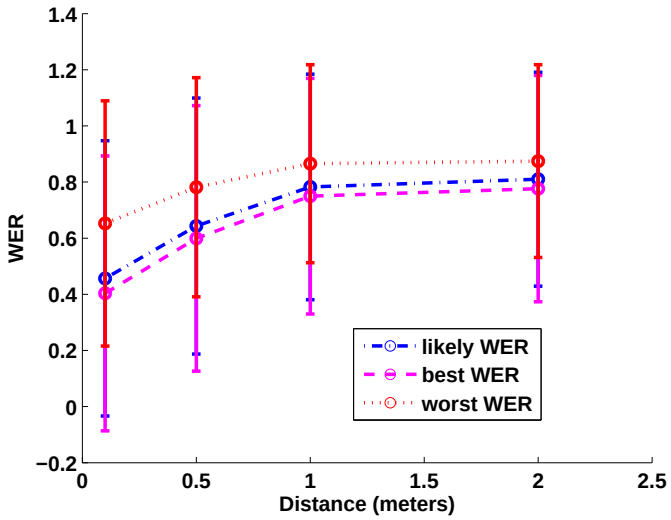



Fig. 12: Average performances of all combinations illustrating the differences between WER measures.

The difference between these categories is shown in Fig. 12, where the likely WER is bounded by the worst and the best WER. The above mentioned metrics have been computed on the basis of four combinations including the Android mobile device or Kinect microphones with the Google or PocketSphinx API. Since they all have the same behavior, and for display clarity, only the *"best WER"* curves are shown for U-WER and T-WER.

### *8.3. Experimental Results*

#### *8.3.1. User Detection*

To demonstrate the improvements brought by the combined upper body + BIP-HOG (fused) detector, we have carried out five detector evaluations: (1) the upper body detector only (without groundHOG), (2) the groundHOG detector only, (3) the BIP-HOG detector only, (4) the upper body + groundHOG detector (exactly as used in [23]), and (5) the proposed upper body + BIP-HOG detector – all discussed in Section 4. The evaluation is carried out on the UserDet-DT dataset (see Section 8.1) using the MR – FPPI evaluation metrics (see Section 8.2). Fig. 13 shows the results obtained for the different detectors. Based on the log-average miss rate, which characterizes the detector performance on the operating spectrum, the following observations can be made. (1) BIP-HOG shows better performance than groundHOG with a 5.09% log-average miss rate improvement. (2) The upper body detector, which is based on RGB-D data, does significantly better by itself, more than 18% improvement, than the BIP-HOG and groundHOG detectors which use RGB only data. (3) The combined upper body + BIP-HOG (fused) detector does better than all the others with a 27.59% log average miss rate – a 3.15% improvement over the upper body + groundHOG detector. Clearly, these percentage improvements might seem small, but depending on the detector operating point set, they can lead to significant MR

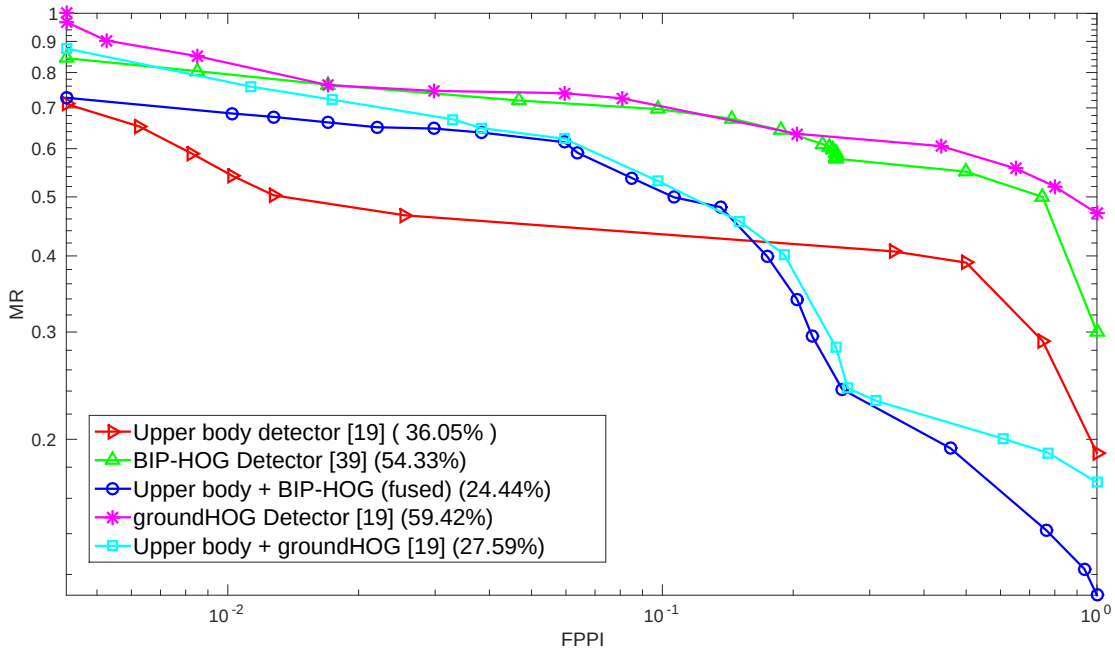



Fig. 13: User detector evaluations based on MR – FPPI metrics. The log-average miss rate percentage in bracket summarizes the performace of each detector (lower better).

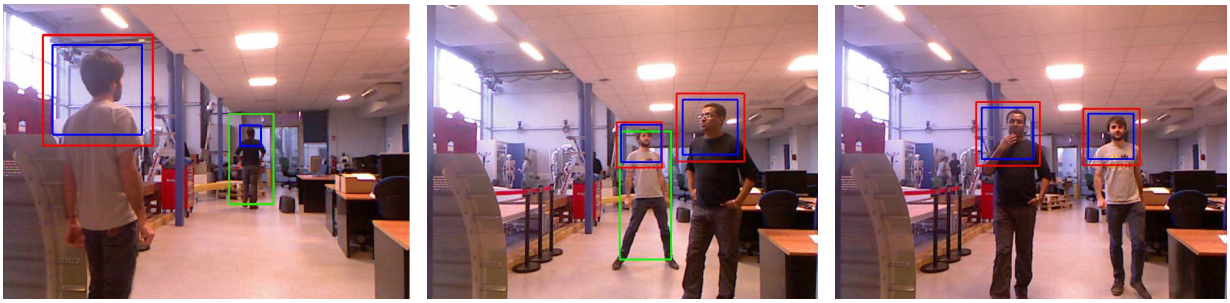


Fig. 14: Sample detections on the UserDet-DT dataset. Detections for upper body detector are shown in red, for BIP-HOG detector are shown in green (full body bounding boxes), and for the combined detector are shown in blue (smaller inner rectangles).

variations. For example, setting the operating point to FPPI = 1 leads to an MR of 11%, 17%, 19%, 30%, and 43% for upper body + BIP-HOG, upper body + groundHOG, upper body only, BIP-HOG, and groundHOG respectively. This means the fused upper body + BIP-HOG detector has a true detection rate of 89% with only 1 average FPPI. Even though direct comparison as a performance indicator is not valid, this is comparable with the best results reported on the INRIA public dataset in [16]. Clearly, since some of the detectors reported in [16] do better than BIP-HOG, it is possible to further improve the performance of the combined upper body detector by replacing the BIP-HOG. But this requires the

arduous work of implementing the algorithms in a way that can be integrated in real-time robotic systems. For example, one of the best detector reported in [16], *ChnFeats*, exists as a Matlab implementation and will have to be re-implemented in C++ and/or GPU compatible languages. Finally, Fig. 14 shows sample results obtained using the upper body + BIP-HOG detector. In all experimental settings henceforth, balancing MR – FPPI performance trade-off, the detector's operating point is set to the point that leads to a 20% MR and $\approx 0.5$ FPPI (a corresponding detector threshold value of $\vartheta_o = 0.21$).

*8.3.2. Intention Detection*

This core modality is evaluated using two datasets acquired in robotic and casual office settings. The final test results presented are based on one dataset acquired using PR2 (Intent-DT2) and another dataset acquired in an office environment (Intent-DT1). The third dataset, Intent-DT3 acquired using PR2, is used to tune and learn the HMM parameters. These discrete HMM parameters, the discrete probability distributions involved, are learned via a frequentist approach [63] by counting the occurrences of events in the Intent-DT3 dataset – by counting the proportion of transitions made for $P\left(x_t|x_{t-1}\right)$ and proportion of $vad/\neg vad$ occurrences during the presence and absence of intention for $P\left(z_t^2|x_t\right)$. Accordingly, $P\left(z_t^2|x_t\right)=\begin{bmatrix}0.30 & 0.75\\ 0.70 & 0.25\end{bmatrix}$ rows represent $\{vad, \neg vad\}$ and columns $\{intent, \neg intent\}$. Similarly, the transition matrix, $P(x_t|x_{t-1})=\begin{bmatrix}0.990 & 0.017\\ 0.010 & 0.983\end{bmatrix}$. For $P(z_t^1|x_t)=\mathcal{N}\left(z_t^1;0,\Sigma\right)$, $\Sigma$ is a diagonal matrix with values of 100 (tuned empirically).

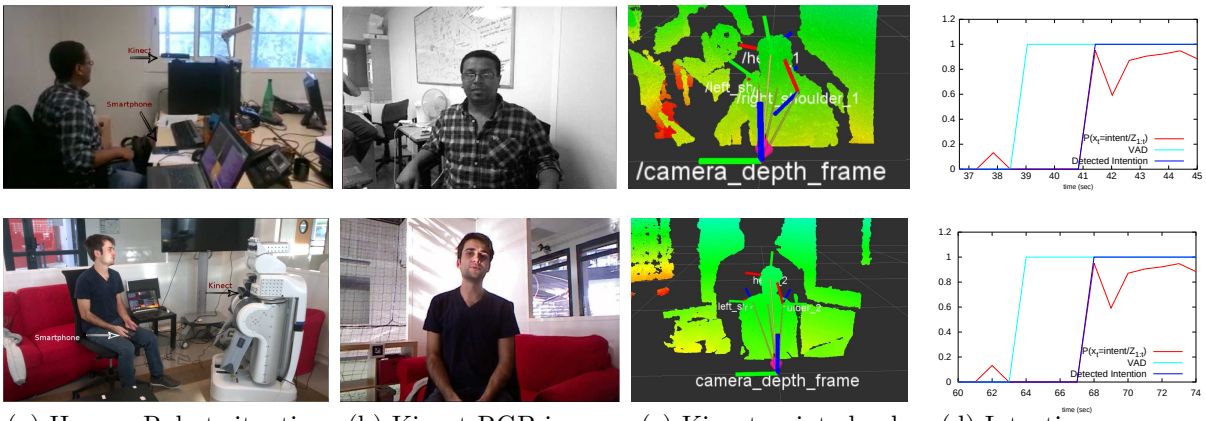
Table 3 shows the results obtained for the intention detection modality on the two datasets, Intent-DT1 (office environment) and Intent-DT2 (robotic environment). To see the improvement brought by each perceptual component, the evaluation is carried using VAD only as measurement, RGB-D data input only (PSOT tracker output) as measurement, and the combined Multi-modal system.

Table 3: User's intention detection evaluation results on datasets Intent-DT1 and Intent-DT2, reported as $\mu(\sigma)$ based on ten repeated runs. The best results in each metric are shown in boldface.

| | TPR (Eq. 8) | | FAR (Eq. 9) | | AED (Eq. 10) | | ACD (Eq. 11) | |
|---|---|---|---|---|---|---|---|---|
| | Intent-DT1 | Intent-DT2 | Intent-DT1 | Intent-DT2 | Intent-DT1 | Intent-DT2 | Intent-DT1 | Intent-DT2 |
| **VAD** | 0.56 (0.01) | 0.48 (0.02) | 0.50 (0.01) | 0.66 (0.04) | **0.03** (0.06) | **0.01** (0.00) | 0.56 (0.02) | 0.33 (0.02) |
| **RGB-D** | 0.72 (0.03) | 0.68 (0.05) | 0.12 (0.03) | 0.26 (0.03) | 0.10 (0.04) | 0.26 (0.06) | 0.73 (0.02) | 0.64 (0.12) |
| **Multi-modal** | **0.80** (0.02) | **0.72** (0.03) | **0.09** (0.04) | **0.14** (0.04) | 0.10 (0.04) | 0.20 (0.08) | **0.77** (0.03) | **0.74** (0.06) |

Clearly in all counts, except AED, the proposed multi-modal approach outperforms the others. In fact, it achieves to detect 80% and 72% of user's intentions correctly with low false alarm rate – 9% and 14% – on Intent-DT1 and Intent-DT2 respectively. In the robotic dataset, Intent-DT2, it detects with a 20% lag and manages to flag an intention correctly, on average, over 74% of its sustenance. It also demonstrates quite improved performance on Intent-DT1. The VAD based approach, though quite fast owing to the high audio frame rate, leads to significant false alarms and less than average TPR on the robotic dataset.

This arises because VAD only captures a speech signal without any know how about the intended listener. The RGB-D only approach shows quite promising achievements. The results clearly demonstrate, by fusing a very unreliable measurement like the VAD, which might be overlooked, with RGB-D further perceptual improvements can be gained – in our case a 4% to 8% gain in TPR, a significantly reduced FAR (almost by half in the robotic dataset), and improved correct coverage and early detection.



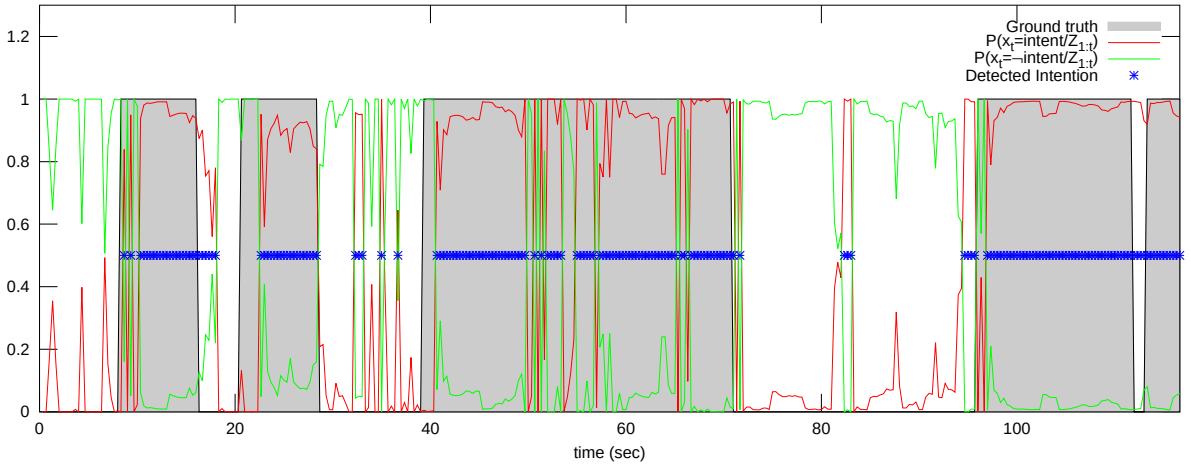

(a) Human Robot situation (b) Kinect RGB image (c) Kinect point cloud (d) Intention measures

Fig. 15: Illustrative scene for user *intention-for-interaction* detection. Top row corresponds to sample data from dataset Intent-DT1 while bottom row to that of Intent-DT2.

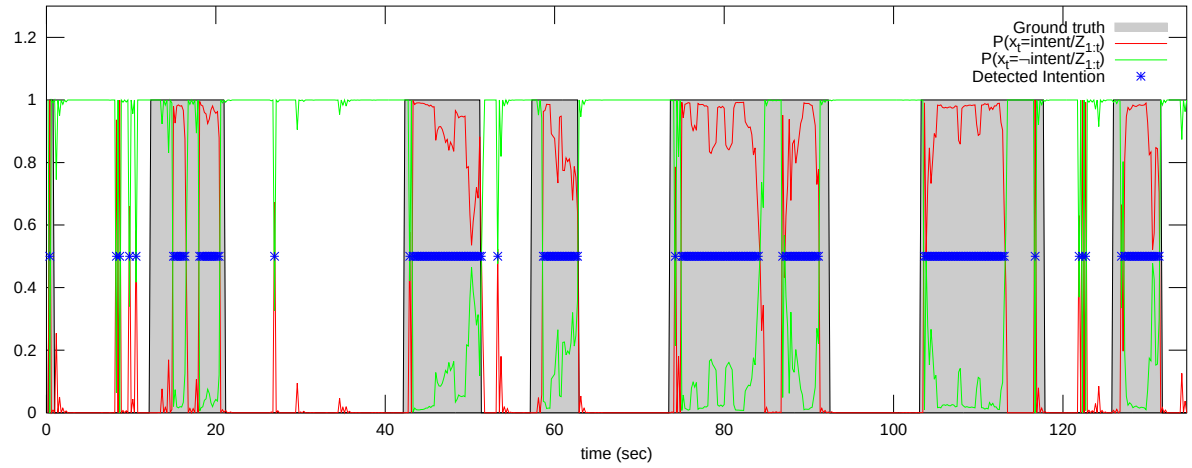
Fig. 15 illustrates instances taken from datasets Intent-DT1 and Intent-DT2. As the illustrated instances show, the user turns its attention to the Kinect sensor and starts talking. Figs. 15b and 15c show the data captured by the sensor. The tracked user head pose and shoulder poses are shown in the point cloud depth in Fig. 15c. The posterior on the user's intention increases in Fig. 15d flagging these instances as an *intention-for-interaction*. The output of the system for Intent-DT1 and Intent-DT2 for a duration of time is illustrated in Fig. 16, which shows the variation of the posterior over *intent* and $\neg$*intent*. Here, a visual correlation could be made between the ground truth annotation (gray shaded region) and detection output (blue asterisk). Both the ground truth and detection outputs take on binary values, but they are shown here as a gray shaded region (for ground truth) and halfway scaled asterisk markers (for detection) to enhance visibility. It is clear that the detection system does well producing results that coincide with the ground truth frequently. Further description of the used dataset and demonstration videos are made available at `http://homepages.laas.fr/aamekonn/cviu_riddle/`.

### *8.3.3. User Distance Mediated Speech Recognition*

In this section, some experiments and results that focus on the multi-streams and multi-ASR fusion algorithm exposed in Section 6 are presented. This module is built using two audio inputs and two ASR systems tuned to a French grammar. A Kinect microphone embedded on the robot and an Android phone device are used as audio inputs. The two ASR systems are CMU's speech recognition PocketSphinx library, which is opensource,




(a)



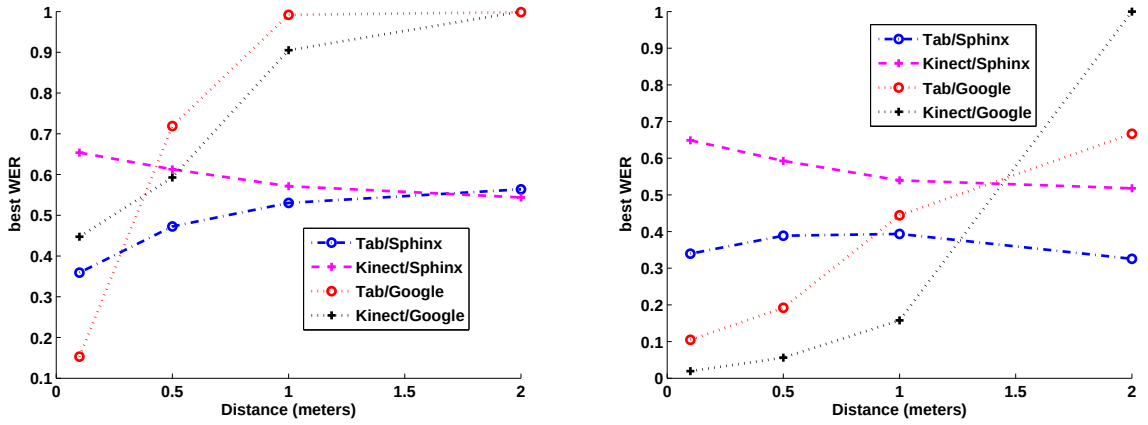

(b)

Fig. 16: User's intention detection system output on (a) Intent-DT1 dataset and (b) Intent-DT2 dataset, in time showing the posterior, ground truth annotation (gray shaded region), and detected intentions (in blue asterisk). The final detection is shown scaled halfway (at 0.5) to enhance visibility.

and Google's Speech API[4]. Each ASR system processes two audio streams which results in four combinations of recognition outputs giving N-Best hypothesis. Thus, the speech recognition module returns more than 20 hypotheses for one spoken utterance. The use of two speech recognition APIs is also motivated by the fact that they are not designed for the same kind of applications and can return very different results. The Google API is tuned to be a vocal assistant and built for a large vocabulary recognition. There is no real

[4] `https://www.google.com/intl/fr/chrome/demos/speech.html`

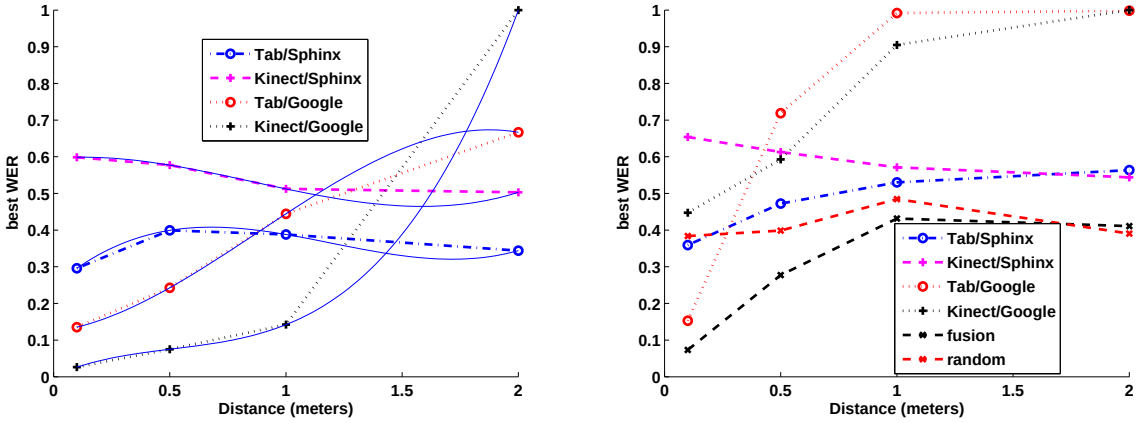
control on grammar and language models. With PocketSphinx, the recognition is done using Perceptual Linear Prediction (PLP) features and HMM, and a restrained grammar is built to focus on the "object search" topic, to limit the system to our robotic application and reduce the recognition error. An experiment is designed to evaluate the recognition module in terms of Word Error Rate (WER), and to demonstrate the improvement of the user perception by the homogeneous fusion framework.




(a) (b)

Fig. 17: (a) Performances of each combination considering T-WER. For (a) and (b) only the best WER is displayed for readability reasons. (b) Performances of each combination considering U-WER. Curve behaviors are the same for likely WER and worst WER.

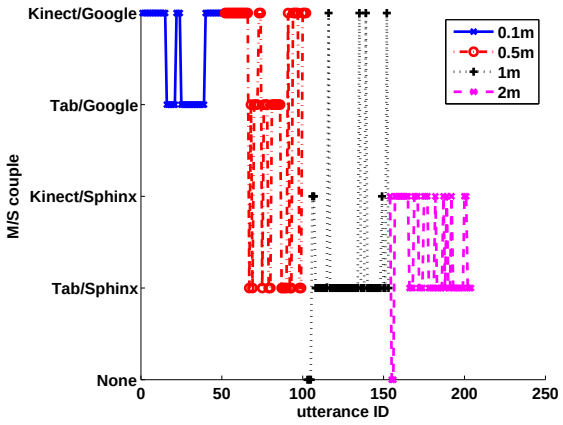
In Fig. 17a, results show the *"best T-WER"* previously defined, show that each combination do not have the same behavior regarding the distance. This is likely due to the fact that each combination is more adapted for one context of use. The combination of the smartphone and Google Speech API performs better during interactions. It recognizes more sentences from a short distance but does not work well beyond one meter. Meanwhile, the combination of the Kinect and PocketSphinx seems to be more efficient for distant interactions. This is likely due to the fact that the grammar is targeted for our application task, and the Kinect has more directive microphones. Thus, when the sentence is incorrectly recognized, the probability of emitting the correct hypothesis increases. As the best *"best U-WER"* shows in Fig. 17b, the curves do not evolve in the same way as every empty recognition results are not included in the mean WER computation (U-WER). These curves can be interpreted as the precision of each combination regarding the distance whenever a sentence is recognized modeling $P(S|d)$. This is contrary to Fig. 17a, where the results show WER if only one system is used at any time. U-WER results are used to learn the $P(S|d)$ density presented in Algorithm 2, since the estimation of the precision of each combination is a prerequisite to our fusion algorithm. T-WER results are used to evaluate our algorithm. The $P(S|d)$ density is estimated using a third degree polynomial regression and the four combinations previously mentioned. The extracted $3^{rd}$ degree polynomials are shown in Fig. 18a.




(a)




(b)




(c)

Fig. 18: Illustration of distance mediated speech recognition fusion algorithm regression functions and results. (a) The third degree polynomial regression of each curve is shown in blue. (b) The best T-WER is compared for each combination and the fusion framework. The random system stands for the fusion algorithm with randomly generated $P(S|d)$ density. (c) This figure represents the switch between each combination. When no hypothesis has been emitted by any combination, the "None" label is selected.

In order to validate the proposed fusion framework, the speech recognition evaluation dataset described in Section 8.1 is used. As four persons have been recorded, a leave-one-out cross-validation method is performed, i.e., three persons are used to learn the $P(S|d)$ density, and one person is used for the test. Since getting a non-empty hypothesis already greatly improves the T-WER, we also applied the same algorithm with a randomly generated density. This demonstrates the advantage of learning $P(S|d)$.

In Fig. 18b, our system outperforms the other combinations taken separately. Moreover, the learned density also outperforms the randomly generated $P(S|d)$. The relative gain in

WER is computed compared to the others combinations, taken alone, and compared to a random initialization of the density. These results are summed up in Table 4. The use of the density alone improves the WER to 11.6% since it is better than the randomly generated density. Therefore, the density and our fusion algorithm, selecting the next combination if nothing has been recognized by the former selected couple, improves the average WER by 29%. This confirms the interest to combine systems according to the current context in order to improve the reliability of speech recognition in variable situations.

Selecting the more appropriate combination during the same interaction session while the user is moving around (and the distance to the microphone is changing) is the underlying goal of our work. To demonstrate the ability of our framework to address this issue, we show in Fig. 18c the result of the application of our algorithm on all the utterances of a given speaker (51 sentences uttered at four different distances, representing 204 utterances). The $P(S|d)$ density is estimated on the data from the three other speakers. The figure shows how the algorithm switches from one combination to another as we go along the process of these 204 utterances, trying to use the more appropriate combination and lower the WER. Our framework selects preferentially and automatically the Google Speech API for small distances and PocketSphinx for long distances.

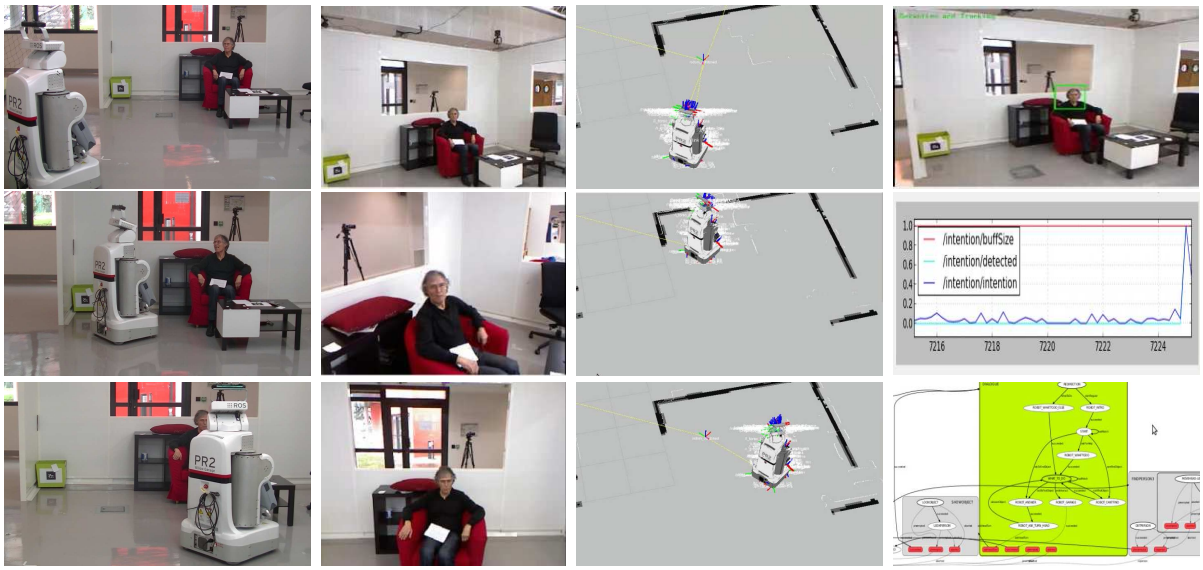
Table 4: Relative WER gain in %

| **Android device** | | **Kinect** | | **Random** | **Average** |
|---|---|---|---|---|---|
| **Sphinx** | **Google** | **Sphinx** | **Google** | | |
| 18.3% | 41.7% | 29.7% | 43.8% | 11.6% | 29.0% |

### *8.4. User Study*

Finally, to evaluate the developed complete system, a user study is carried out with 17 volunteer elderly participants. The volunteers were recruited from the *la Grave Gérontopôle* hospital and the *LAAS-CNRS* laboratory in Toulouse, France. They are all over 60 years of age, ranging from 61 to 84 with a median age of 71, 9 males and 8 females. 9 of them had a previous experience with robots (experts) and 8 had no experience with robots (naïve), there was a majority of males in the experts and of females in the naïve, Table 5 summarizes the demographic distribution of the users. No incentives were provided to the participants.

The objective of the user study was two fold: (1) To assess the soundness of the deployed system by analyzing the success or failure of the perceptual modalities during each experimental session with a user; And (2) to assess the reactions of actual elderly people towards the presented complete system, especially to assess any significant differences between experts and naïves. The user study was conducted in a two day period, mornings and afternoons, inside our robotic laboratory (a controlled environment) with one user at a time. The experimental sessions with a user and a robot lasted from 5 to 15 minutes. Each participant was individually briefed with minimal information possible about the




(a) H/R situation (b) Kinect view (c) Robot model with map (d) Relevant information

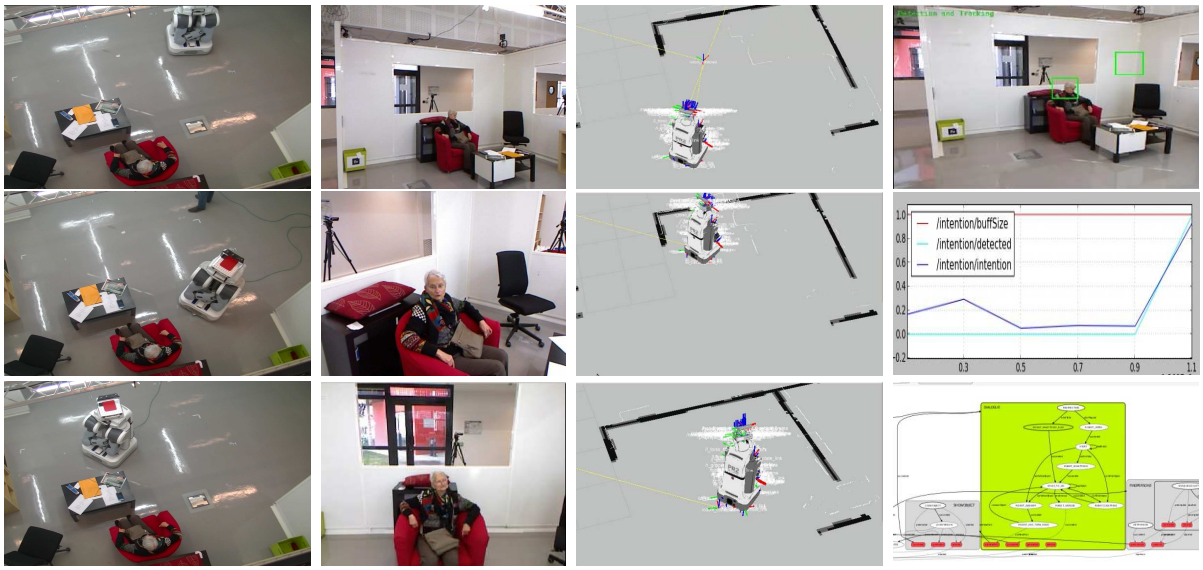
Fig. 19: Sample snapshots taken during the user study. Each row corresponds to representative illustrations taken during the three phases of our assistive system, i.e., user detection, intention detection, and interaction respectively. (a) Shows the H/R situation as captured from external camera, (b) the RGB feed from the onboard Kinect, (c) visualization of the robot model and its current localization within the environment map, and (d) various relevant information during each phase of the system. The last columns of the first, second, and third row depict the detected user, the instant a user's intention is detected, and state transitions during the interaction phase respectively.

capabilities of the deployed robotic system, basically that the robot will help him/her find the position of objects (they were informed the possible list of objects they could ask for), but he/she will have to first express an interest to interact with the robot. We applied a bottom up approach by observing the behavior of volunteers asking the PR2 robot to help them find an object. In all cases PR2 was operating in autonomous mode because a human-driven system (wizard of Oz) would have more reflected the behavior of the human controlling the robot than the autonomous functioning according to the actual command law implemented. The volunteers were filmed from five different simultaneous angles.

With regards to the soundness of the deployed robotic system, an experiment is labeled as successful, if the *smach* based state machine is traversed correctly leading to a

Table 5: Demography of the users who participated in the user study.

| | Male | Female | Age |
|---|---|---|---|
| Experts | 7 | 2 | $67 \pm 5.05$ |
| Naïve | 2 | 6 | $74.75 \pm 6.56$ |




(a) H/R situation (b) Kinect view (c) Robot model with map (d) Relevant information

Fig. 20: Another sample illustration taken during the user study. The caption descriptions of Fig. 19 apply.

"succeeded" output at the end, and it is considered a failure, if by any means, it resulted into an "aborted" or a "preempted" state. The dialogue module based on the Google ASR API using the Android phone microphone was used. The speech fusion framework was not used in this scenario since the user is always close to the smartphone and far from the Kinect sensor ($1.5m \sim 2m$). This is the determined configuration during interaction based on the closeness of the target to the audio sensors. In all but one case the robot managed to correctly detect the user, transition to its garage state, detect user's intention, and carry-out the close interaction phase as planned – a 94% mission success rate. In 68.75% of these cases (with 11 users), the robot detected the user's *intention-for-interaction* at the first correct user attempt, while in 18.75% of the time (with 3 users) it detected it at the second attempt, and the rest 12.5% (with 2 users) at the third attempt. In the one exceptional case, the robot failed to detect the user's intention as the user chose to sit far from the robot and the head and shoulder pose estimation modules failed to provide correct estimates. Nevertheless, the experiment continued to the interaction phase by manual triggering to provide further data for the second objective, user reaction assessment. All in all, the system meets expectations and reflects the results obtained during each perceptual component evaluation. Sample illustration taken from this experimental stage are shown in Figs. 19 and 20. Please visit `http://homepages.laas.fr/aamekonn/cviu_riddle/` for demonstrative videos.

The reactions of the elderly to the deployed robotic assistive system, the second objective of the user study, was assessed by analyzing the video films recorded during the experimental runs. The films were analyzed with a focus on facial expression, direction of look of the volunteer, vocal interaction, and body language. For a facial expression

the user exhibited during the experiment, a label of "smiling", "doubting", and/or "expectant" is assigned (multiple labels can be assigned depending on the manifested facial expression throughout the coarse of the experiment). The labels obtained were 11 "smiling", 7 "doubting", 3 "expectant", (4 were "smiling" and "doubting" and 1 was "smiling" and "expectant"), 2 manifested a different facial expression than the three categories. All volunteers would look at the robot during the interaction. 11 would look at where the robot says the lost objects are, with significantly more of the experts (9 out of 10, Fisher $p = 0.034$). 11 out of 17 volunteers would spontaneously bend towards the robot with no significant difference whether they were expert or naïve (Fisher $p = 0.10$). Regarding language, 3 people would speak slowly from the start, 1 of them would also mouth his words from the start, and another 1 would use sentence words. If the robot failed to understand them, 11 people would mouth the words, 7 would speed down their flow, 6 would use sentence words, 4 would try and help the understanding by a circumlocution and 4 by a reformulation. We did not observe any sign of fear in those experimental conditions. We could say that there was no difference regarding the position towards the robot between the experts and the naïves, the experts would be more readily accepting information from the robot and using it. Whatever their background is, the volunteers would use the same strategy as they would for a human or pet when observing the failure of a command: mouth the words, speak slowly, try and facilitate with the context or a reformulation. Sample snapshots for two of the participants are shown in Figs. 19 and 20. Further work on the intentionality will have to take into account that spontaneous behavior in the management of the vocal interaction. Ideally, some level of habituation to the robot, would be of interest if we want to include the validation of the success of the work-flow (i.e., the object is found) in the behavior of the robot.

## 9. Conclusions and Future Work

In conclusion, a multi-modal perception based architecture for non-intrusive domestic assistive robot has been described. The presented system exhibits non-intrusive characteristics as it only engages in a close HRI phase when the user expresses his/her intent. It relies on a multi-modal user detector, based on RGB-D data, to localize the user in the scene; a multi-modal user's *intention-for-interaction* detector, based on RGB-D data and VAD; and various ASR APIs for reliable communication. Each perceptual component has been evaluated separately: a user detector with low MR (24.4% log-average miss rate), a user intention detector with more than 72% TPR, and an ASR with less than 15% best WER (at the preferred configuration). All of these combined led to a non-intrusive robotic system that demonstrated a 94% success rate during experimental runs with 17 elderly volunteers. The user study carried out with these participants also revealed an overall pleasant interaction experience. In addition, the paper also presents relevant implementation details (ROS nodes, and *smach* based task-level coordinator) that would be pertinent for the scientific community in general. Even though the framework is presented in the context of helping a user find hidden and/or forgotten objects, it is fundamentally generic and can be easily extended to various assistive tasks.

In the near future, the presented system will be augmented with multi-modal action recognition modules to pave the way for more natural interactions and assistive contexts. It is also envisaged to deploy and test the overall system on a humanoid robotic system, specifically the new Romeo robot [64] from Aldebaran Inc. Additionally, several possible future pospects and research axes can be considered: (1) Integrating an automated object detection and recognition capability, possibly a vision and RFID based solution to handle small objects; (2) Further improving the intention detection module with context information, e.g., audio activity detection to identify when the user is watching TV, cooking, or the like; And (3) endow more navigation capability to the mobile robot to navigate to the location of the asked object and provide improved assistance.

## Acknowledgment

This work was supported by a grant from the French National Research Agency (ANR) under project RIDDLE with grant number ANR-12-CORD-0003.

## References

[1] I. Robert-Bobée, Projections de population 2005–2050 et vieillissement de la population en france métropolitaine, `http://www.insee.fr/fr/ffc/docs_ffc/ecostat_d.pdf` (2007).

[2] T.-S. Dahl, M.-N. K. Boulos, Robots in health and social care: A complementary technology to home care and telehealthcare?, Robotics 3 (1) (2013) 1–21.

[3] J. Broekens, M. Heerink, H. Rosendal, Assistive social robots in elderly care: A review, Gerontechnology 8 (2) (2009) 94–103.

[4] B. Boudet, T. Giacobini, I. Ferran'e, C. Fortin, C. Mollaret, F. Lerasle, P. Rumeau, Quels sont les objets égarés à domicile par les personnes âgées fragiles ? une étude pilote sur 60 personnes, {NPG} Neurologie - Psychiatrie - Gériatrie 14 (79) (2014) 38 – 42.

[5] A. Cesta, G. Cortellessa, V. Giuliani *et al.*, Proactive assistive technology: An empirical study, in: Human-Computer Interaction INTERACT'07, Rio de Janeiro, Brazil, 2007.

[6] A. Cesta, G. Cortellessa, M. Giuliani *et al.*, The RoboCare assistive home robot: environment, features and evaluation, Tech. rep., The ROBOCARE Technical Reports, RC-TR-0906-6 (2006).

[7] M. Vinagre, J. Aranda, A. Casals, An interactive robotic system for human assistance in domestic environments, in: Computers Helping People with Special Needs, 2014, pp. 152–155.

[8] Y. Onuma, N. Kamado, H. Saruwatari, K. Shikano, Real-time semi-blind speech extraction with speaker direction tracking on Kinect, in: 2012 Asia-Pacific Signal Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012, pp. 1–6.

[9] R. Stiefelhagen, C. Fugen, R. Gieselmann, H. Holzapfel, K. Nickel, A. Waibel, Natural human-robot interaction using speech, head pose and gestures, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'04), Sendai, Japan, 2004.

[10] M. Quigley, K. Conley, B. Gerkey *et al.*, ROS: an open-source robot operating system, in: ICRA Workshops, Kobe, Japan, 2009.

[11] B. Burger, I. Ferrané, F. Lerasle, G. Infantes, Two-handed gesture recognition and fusion with speech to command a robot, Autonomous Robots 32 (2) (2012) 129–147.

[12] T.-H. Pham, A. Kheddar, A. Qammaz, A. Argyros, Towards force sensing from vision: Observing hand-object interactions to infer manipulation forces, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15), Boston, MA, USA, 2015.

[13] Y. Gu, H. Do, Y. Ou, W. Sheng, Human gesture recognition through a kinect sensor, in: IEEE International Conference on Robotics and Biomimetics (ROBIO'12), Guangzhou,China, 2012.

[14] D. Cazzato, P. Mazzeo, P. Spagnolo, C. Distante, Automatic joint attention detection during interaction with a humanoid robot, in: A. Tapus, E. Andr, J.-C. Martin, F. Ferland, M. Ammi (Eds.), Social Robotics, Vol. 9388 of Lecture Notes in Computer Science, Springer International Publishing, 2015, pp. 124–134.

[15] M. Kleinehagenbrock, S. Lang, J. Fritsch, F. Lomker, G. Fink, G. Sagerer, Person tracking with a mobile robot based on multi-modal anchoring, in: IEEE International Workshop on Robot and Human Interactive Communication, Berlin, Germany, 2002.

[16] P. Dollar, C. Wojek, B. Schiele, P. Perona, Pedestrian detection: An evaluation of the state of the art, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (4) (2012) 743–761.

[17] D. Gerónimo, A. López, A. Sappa, T. Graf, Survey of pedestrian detection for advanced driver assistance systems, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (7) (2010) 1239–1258.

[18] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005.

[19] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (9) (2010) 1627–1645.

[20] P. Dollár, R. Appel, S. Belongie, P. Perona, Fast feature pyramids for object detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (8) (2014) 1532–1545.

[21] S. Walk, N. Majer, K. Schindler, B. Schiele, New features and insights for pedestrian detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10), San Francisco, CA, USA, 2010.

[22] W. Schwartz, A. Kembhavi, D. Harwood, L. Davis, Human detection using partial least squares analysis, in: IEEE International Conference on Computer Vision (ICCV'09), 2009, pp. Kyoto, Japan.

[23] O. H. Jafari, D. Mitzel, B. Leibek, Real-time RGB-D based people detection and tracking for mobile robots and head-worn cameras, in: International Conference on Robotics and Automation (ICRA'14), Hong Kong, China, 2014.

[24] L. Spinello, K. O. Arras, People detection in RGB-D data, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'11), San Francisco, CA, USA, 2011.

[25] B. Schiele, M. Andriluka, N. Majer, S. Roth, C. Wojek, Visual people detection: Different models, comparison and discussion, in: IEEE ICRA Workshop on People Detection and Tracking, Kobe, Japan, 2009.

[26] S.-J. Blakemore, J. Decety, From the perception of action to the understanding of intention, Nature Reviews Neuroscience 2 (8) (2001) 561–567.

[27] Y. Xiao, Z. Zhang, A. Beck, J. Yuan, D. Thalmann, Human-robot interaction by understanding upper body gestures, Presence 23 (2) (2014) 133–154.

[28] B. Huber, Foot position as indicator of spatial interest at public displays, in: ACM CHI'13 Extended Abstracts on Human Factors in Computing Systems, Paris, France, 2013.

[29] A. Clair, R. M. St, M. J. Matarić, Monitoring and guiding user attention and intention in human-robot interaction, in: ICRA-ICAIR Workshop, Anchorage, AK, USA, 2010.

[30] A. Tavakkoli, R. Kelley, C. King, et al., A vision-based architecture for intent recognition, in: International Symposium on Visual Computing, Lake Tahoe, CA, USA, 2007.

[31] J.-R. Martinez, A. Escobedo, A. Spalanzani, C. Laugier, Intention driven human aware navigation for assisted mobility, in: IROS-ASRHE Workshop, Vilamoura, Portugal, 2012.

[32] J. L. Drury, J. Scholtz, H. A. Yanco, Awareness in human-robot interactions, Washington, DC, USA, 2003.

[33] S. Hommel, U. Handmann, Realtime AAM based user attention estimation, in: IEEE International Symposium on Intelligent Systems and Informatics (SISY'11), Subotica, Serbia, 2011.

[34] D. Wang, A.-H. Tan, Mobile humanoid agent with mood awareness for elderly care, in: International Joint Conference on Neural Networks (IJCNN'14), Beijing, China, 2014.

[35] D. Novak, R. Riener, Enhancing patient freedom in rehabilitation robotics using gaze-based intention detection, in: IEEE International Conference on Rehabilitation Robotics (ICORR'13), Seattle, WA, USA, 2013.

[36] Y. Nakauchi, K. Noguchi, P. Somwong, T. Matsubara, A. Namatame, Vivid room: human intention detection and activity support environment for ubiquitous autonomy, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'03), Las Vegas, NV, USA, 2003.

[37] R. Kelley, M. Nicolescu, A. Tavakkoli, M. Nicolescu, C. King, G. Bebis, Understanding human intentions via hidden markov models in autonomous mobile robots, in: ACM/IEEE International Conference on Human-Robot Interaction (HRI'08), Amsterdam, The Netherlands, 2008.

[38] J.-Y. Kuan, T.-H. Huang, H.-P. Huang, Human intention estimation method for a new compliant rehabilitation and assistive robot, in: SICE Annual Conference, Taipei, Taiwan, 2010.

[39] E. A. Kulić, D. Croft, Estimating intent for human-robot interaction, in: International Conference on Advanced Robotics (ICAR'03), Coimbra, Portugal, 2003.

[40] L. Bascetta, G. Ferretti, P. Rocco *et al.*, Towards safe human-robot interaction in robotic cells: An approach based on visual tracking and intention estimation, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'11), San Francisco, CA, USA, 2011.

[41] R. Ooko, R. Ishii, Y. Nakano, Estimating a users conversational engagement based on head pose information, in: International Conference on Intelligent Virtual Agents (IVA'11), Reykjavik, Iceland, 2011.

[42] O. C. Schrempf, U. D. Hanebeck, A generic model for estimating user intentions in human-robot cooperation, in: International Conference on Informatics and Control, Automation, and Robotics (ICINCO'05), Barcelona, Spain, 2005.

[43] B. Lecouteux, G. Linares, Y. Esteve, G. Gravier, Dynamic combination of automatic speech recognition systems by driven decoding, IEEE Transactions on Audio, Speech, and Language Processing 21 (6) (2013) 1251–1260.

[44] H. K. Maganti, D. Gatica-Perez, I. McCowan, Speech enhancement and recognition in meetings with an audio-visual sensor array, IEEE Transactions on Audio, Speech, and Language Processing 15 (8) (2007) 2257–2269.

[45] T. Field, Openni tracker ROS package (groovy), `http://wiki.ros.org/openni_tracker/` (2013).

[46] L. Herranz, R. Xu, S. Jiang, A probabilistic model for food image recognition in restaurants, in: IEEE International Conference on Multimedia and Expo (ICME'15), Torino, Italy, 2015.

[47] A. A. Mekonnen, C. Briand, F. Lerasle, A. Herbulot, Fast HOG based person detection devoted to a mobile robot with a spherical camera, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'13), Tokyo, Japan, 2013.

[48] M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. Van Gool, Online multiperson tracking-by-detection from a single, uncalibrated camera, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (9) (2011) 1820–1833.

[49] J. Y. Bouguet, Camera calibration toolbox for Matlab (2008).
URL http://www.vision.caltech.edu/bouguetj/calib_doc/.

[50] C. Mollaret, F. Lerasle, I. Ferrané, J. Pinquier, A particle swarm optimization inspired tracker applied to visual tracking, in: IEEE International Conference on Image Processing (ICIP'14), Paris, France, 2014.

[51] G. Fanelli, M. Dantone, J. Gall, A. Fossati, L. V. Gool, Random forests for real time 3D face analysis, International Journal of Computer Vision 101 (3) (2013) 437–458.

[52] R. Lienhart, J. Maydt, An extended set of Haar-like features for rapid object detection, in: IEEE International Conference on Image Processing (ICIP'02), New York, USA, 2002.

[53] J. Kennedy, R. Eberhart, Particle swarm optimization, in: IEEE International Conference on Neural Networks, Perth, WA, USA, 1995.

[54] M. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, Transactions on Signal Processing 50 (2) (2002) 174–188.

[55] J. Bohren, Wiki: smach, http://wiki.ros.org/smach (2010).

[56] P. Mihelich, Openni launch ROS package (groovy), http://wiki.ros.org/openni_launch/ (2013).

[57] E. Marder-Eppstein, PR2 2D navigation (pr2_2dnav) ROS package (groovy), http://wiki.ros.org/pr2_2dnav (2013).

[58] R. Qiu, Z. Ji, A. Noyvirt *et al.*, Towards robust personal assistant robots: Experience gained in the srs project, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'12), Vilamoura, Portugal, 2012.

[59] J. Bohren, R. Rusu, E. Jones *et al.*, Towards autonomous robotic butlers: Lessons learned with the PR2, in: IEEE International Conference on Robotics and Automation (ICRA'11), Shanghai, China, 2011.

[60] S. Glaser, PR2 controllers ROS package (groovy), http://wiki.ros.org/pr2_controller (2013).

[61] D. Stonier, Rosjava Android build tools, http://wiki.ros.org/rosjava_build_tools (2013).

[62] N. Nguyen, D. Phung, S. Venkatesh, H. Bui, Learning and detecting activities from movement trajectories using the hierarchical hidden markov model, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005.

[63] B. Everitt, The Cambridge dictionary of statistics, 2nd Edition, Cambridge University Press Cambridge, U.K. ; New York, 2002.

[64] A. K. Pandey, R. Gelin, R. Alami, R. Viry *et al.*, Romeo2 Project: Humanoid robot assistant and companion for everyday life: I. situation assessment for social intelligence, in: International Workshop on Artificial Intelligence and Cognition (AIC'14), Torino, Italy, 2014.