in: IEEE Int. Conf. on Systems, Man, and Cybernetics (IEEE-SMC 2011), Anchorage, USA, pp. 2481-2488, IEEE 2011

I'll keep an Eye on You: Home Robot Companion for Elderly People with Cognitive Impairment

H.-M. Gross, Ch. Schroeter, S. Mueller
M. Volkhardt, E. Einhorn
Neuroinformatics and Cognitive Robotics Lab
Ilmenau University of Technology
98684 Ilmenau, Germany
Horst-Michael.Gross@tu-ilmenau.de

A. Bley, T. Langner
Ch. Martin, M. Merten
MetraLabs GmbH
Am Vogelherd 22
98693 Ilmenau, Germany
Andreas.Bley@Metralabs.com

***Abstract*—The paper gives an overview of the progress in developing a socially assistive home robot companion for elderly people with mild cognitive impairment (MCI) living alone at home. The spectrum of required assistive functionalities of such a robot companion is broad and reaches from reminding functions (e.g. taking medication or drinking) and cognitive stimulation exercises, via mobile videophony with relatives or caregivers, up to the detection and evaluation of critical situations, like falls. The paper is addressing several aspects of our work as part of the European FP7 project "CompanionAble", as for example the developed robot hardware and its software and control architecture, the implemented skills for robust user detection and tracking and user-centered navigation in the home environment, and reports on already conducted and still ongoing functionality testings and pending usability studies with the end-user target groups (the elderly, relatives, caregivers).**

***Index Terms*—assistive robotics; robot control architecture; user observation; robot navigation**

I. INTRODUCTION

The European FP7 project CompanionAble [1] (project duration 2008-12) aims at enabling elderly people suffering from mild cognitive impairment (MCI) to remain in their accustomed home environment as long as possible by using assistive technologies. An important unique selling point of the project lies in the synergetic combination of the advantages of a mobile robot companion with the strengths of a stationary "smart home" solution. The fact that a mobile robot companion is able to navigate autonomously or, if necessary, by means of tele-operation in the elderly's home has proved as an convincing argument in accepting assistive technologies by the potential end-users of such technologies - the care recipients, relatives, and caregivers. For them it is well comprehensible that mobility of an assistive system significantly simplifies the interaction with the care recipient and makes a spectrum of assistive functions possible at all, as for example greeting the care recipient upon his/her returning home, facilitating contact (by videophony) with relatives or caregivers, detecting dangerous situations, like falls anywhere in the home, and evaluating them by authorized persons via tele-operated robot control. To this end, within the CompanionAble project a socially assistive, mobile robot companion is being developed (see Fig. 1), which aims to assist the elderly in their daily life. The specifics of socially assistive robotics is that it is focused on helping human users through social rather than physical interaction [2]. So, manipulation skills are explicitly not subject of this project. Nonetheless, the spectrum of assistive functionalities of such a robot companion is broad and reaches from cognitive stimulation exercises and reminding functions (e.g. taking medication or drinking), via mobile videophony with relatives or caregivers, up to the above mentioned evaluation of critical situations, like falls. The required functionalities of a mobile home robot companion which were identified in project-internal studies with the end-user target groups can be grouped into the following categories:

- *Communication:* any function that allows keeping in touch with the care recipient's (CR) family, friends or care professionals, e.g. by videophony
- *Safety:* any function that directly increases the safety of the CR, e.g. reminding taking medication, user monitoring and fall detection including emergency call functionality, tele-operated robot navigation
- *Services and assistive functions:* any function that improves the CR's daily life, as e.g. daytime management (daily routine, agenda updating), keeping and dispensing personal items of the CR, entertainment and interactive media access, etc.
- *Therapy:* all functions that are intended to improve the CR's state of health, e.g. cognitive training exercises or stimulating games
- *Smart situation awareness:* information the robot companion can collect to allow for adapting its behavior to the CR's specifics, e.g. her typical daily activity course, or her preferences in communication with the robot.

This paper is giving an overview of the progress in developing such a mobile home robot companion for the elderly suffering from MCI. For this purpose, it is addressing several aspects beginning with related work in the field of socially assistive robotics for domestic use (Section II), continuing with the robot hardware (Section III) and its control architecture (Section IV), via developed technologies for robust

This work has received funding from the European Community's 7th Framework Program (FP7/2007-2013) under grant agreement no. 216487.

Fig. 1. 89 years old lady in interaction with the mobile home robot companion developed in the CompanionAble project [1].

user detection and tracking and user-centered navigation in home environments (Section V), up to the results of already conducted and ongoing functionality testings and upcoming usability evaluations of the companion robot (Section VI).

II. RELATED WORK

In recent years, socially assistive robotics for domestic use has been a rapidly increasing field of research and development [2]. Therefore, this section can only give a fragmentary overview of the state-of-the-art in this field with a focus on socially assistive home robot companions. Developments on mobile robots with manipulation skills, as for example the well-known MOVAID, Hermes, ARMAR, or the probably most advanced wheel-based robotic home assistant Care-O-Bot, are not covered here. A very good overview of this branch of assistive robotics is given in [3].

The CMU Pittsburgh NurseBot project [4] (1998-2005) was one of the first research endeavors that dealt with robotic assistance for the elderly. The robotic platform was used to test several ideas for such a companion, including, but not limited to, intelligent reminding functions, tele-presence applications, surveillance, simple social interaction and help for physically impaired people. The robot learned patterns of typical movements from the people it cares for by observing them, tracking them, and inferring certain behaviors.

The European FP6 project COGNIRON [5] (2004-2007) was also focussed on researching enabling technologies for a socially assistive robot whose ultimate task was to serve humans (not primarily elderly or cognitively impaired persons) as a companion in their daily life. To this end, the focus was on studying the perceptual, representational, reasoning and learning capabilities of embodied robots in human-centered environments. The project also aimed to develop methods and technologies for the construction of cognitive robots able to evolve and grow their capabilities in close interaction with humans in an open-ended fashion.

The European FP6 project Robots@Home [6] (2007-2010) aimed to create an open mobile platform that should pave the way for introducing robots in homes. This involved not only the development of the hardware, but also the creation of an embedded perception system for learning rooms and maps which should facilitate safe and robust navigation in homes. Several application domains were identified in scenarios which defined the task space of the mobile robot, e.g. security tasks (checking rooms), or delivering food.

LIREC (Living with Robots and Interactive Companions) [7] is a European FP7 project (2008-2012) that addresses a number of key scientific and technical areas required for long-term companion relationships with a focus on perception, memory, emotions, communication, and learning. This also includes an empirical evaluation in real social settings to validate the theory developed. Both physical and virtual companions are in the focus of this project.

Most of the approaches introduced here are either research-oriented systems or proof-of-concept demonstrators not yet ready for autonomous operation in real home environments. Typically they show limited functionality regarding a completely autonomous navigation behavior and require instructed or even technically equipped users and the presence of roboticists during their operation to guarantee the safety of the users and to handle unexpected situations. Having true autonomy in mind, none of the aforementioned systems can already be considered as really autonomous home robot companion suitable for everyday use. Moreover, most of them haven't been involved in assistive tasks over longer periods of time, or were subject of long-term field trials in real homes or care centers. Both aspects are, however, essential prerequisites for a lasting acceptance of robot companions by the end-user target groups - the elderly as the care recipients, their relatives, and the caregivers.

Taking this into consideration, from the beginning the work in CompanionAble has been focused on i) only the doable and useful robot functionalities selected from a larger ”wish-list” specified by the different end-user target groups, ii) on the marketability of the robot companion to be developed with the consequence of finding a low-cost solution for the robot hardware with a friendly appearance and an adequate size and weight, iii) on developing a robot software that fulfills the main requirements to a modern robot control architecture (modularity, extensibility, efficiency, customizability, reusability) allowing for a rapid application development, and iv) the necessary navigation and interaction autonomy of the robot that is required for such ”simple” tasks, like an autonomous re-charging or a user-search.

III. THE ROBOT COMPANION

To ensure a high acceptance of the robot companion by its potential end-users, their specific requirements regarding the functionality, usability, and appearance of the robot have been taken into account from the beginning of the design process [8].

With a height of 120 cm, the robot companion is comparable to the size of an 8 years old child. Its size is optimized for a friendly appearance and an ergonomic operability by a standing or sitting user (Fig. 1). The drive system of the robot consists of a differential drive and a castor on the rear. This gives it a good maneuverability and stability in spite of its height and weight of about 50 kg. For navigation, user monitoring, HRI and safety, the robot is equipped with various interaction devices and sensor systems (Fig. 2) subsequently described.

Tiltable touch-Screen: The touch-screen is the central unit for graphical communication with the robot. We decided to use a vertically mounted 15,4 inch wide screen display, transferring commands from the user via the touch screen to the embedded PC, and to provide video based information concerning the user. To ensure the best usability, the display unit can be tilted up and down in order to allow for an optimal interaction with a standing and sitting user.

Robot face: As additional interaction channel and imitating a simple technical face, the robot has two OLED displays as eyes, which can be used for expressing robotic emotions, following a moving user with the eyes, or executing other facial gestures that could be helpful for HRI.

Personal Storage Box: As suggested and requested by all end user target groups, the robot companion is equipped with a storage box, where the elderly person can place personal items, which s/he tends to mislay more or less often, like glasses, wallet, keys, mobile phone, etc. The box is located below the display. As an important feature, the storage box is equipped with RFID technology. This way, the robot is able to detect and automatically register any changes of items equipped with RFID tags in the box. Meanwhile, specialized RFID tags are available in sizes as small as 2 mm by 2 mm. Hence, nearly all items with a tendency to be misplaced can be fitted unobtrusively with these tags. Using this knowledge, the robot is able to remind the user to move something in or out of this box, if needed.

Cameras: For HRI and navigation, the robot is equipped with three color cameras. One high resolution camera, an eco274CVGE produced by SVS-VISTEK, with a 180^o fish-eye lens is mounted in the robot head. It is connected via Gigabit Ethernet to the embedded PC and is used for people tracking and obstacle detection. The camera has an image resolution of 1600×1200 pixels and is able to record up to 25 pictures per second. The second camera, a low-cost USB camera with a maximum resolution of 1.3 megapixels, is placed within the tiltable display unit immediately above the touch screen. This way, the camera can continuously face the user during interaction with the robot which is a prerequisite for such functionalities like videophony or facial expression analysis during training exercises. A similar USB camera module is integrated in the rear and used for the autonomous docking to the charging station (see below).

Microphones: The robot is equipped with a microphone array using the coincidence microphone technology (CMT) developed by project partner AKG in Austria (www.akg.com). The CMT array is utilized for localizing a talking person and sound signal enhancement based on beam forming which is important for videophony and speech-based people tracking. For an optimal acoustic performance, the CMT microphone array is placed at the top of the robot.

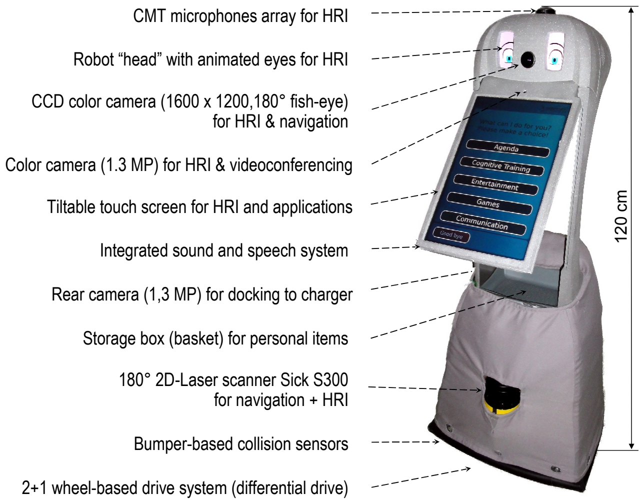

Fig. 2. Robot companion developed in the CompanionAble project [1] with its main equipment for environment perception, navigation, and HRI.

Laser range finder: For obstacle detection, map building, localization, and person tracking, the robot is equipped with a SICK S300 2D-laser range finder with a scanning range of 270^o. The laser is placed in the main driving direction of the robot just above the wheel case to reach a scanning range of at least 180^o (see Fig. 2). The selected model conforms to the security class SIL 2 that could - in combination with an adequate safety concept of the robot - help to fulfil the requirements of the German TUV (Technical Inspection Agency) or other certification institutions.

Collision sensors: The collision sensor of a robot system is needed to detect the collision of the robot with obstacles that are invisible for other sensor systems or moving faster than the reaction time of the robot system. These could include, for example, very low objects, like a book on the floor, or fast objects, like a person walking straight in the way of the robot. We use a closed rubber-based security collision sensor that is mounted around the base plate.

On-board PC: The robot is controlled by an embedded PC with an Intel i7-620M quad core processor and a multitude of small hardware units which control and monitor several functions of the robot.

Battery and charging technology: The battery technology of the robot is based on Lithium-Polymer accumulator cells. The chosen battery capacity of 40 Ah is a compromise allowing for a working time of about ten hours but only requiring a charging time of less than four hours. For completely autonomous re-charging of the battery, a new docking concept had to be developed. One of the main challenges was to find a technological solution that allowed for a direct connection of the robot to the line voltage of the home, because it should be possible to connect and disconnect the charging connector by the robot platform itself without expansive manipulators or actuators. Inspired by the connector concept known from water boilers used in millions of households, a first version of a safe charging connector has been developed. For test purposes, we have integrated this kind of socket on the robot and designed a spring loaded plug for the wall. Above the socket there is a small camera to detect synthetic markers that are used for the autonomous docking to the socket. In functional testings of this technology, we achieved a very good and robustly repeatable docking behavior which is an important prerequisite for autonomous re-charging and therewith the long-term applicability of the robot companion.

IV. ROBOT CONTROL ARCHITECTURE

To guarantee the main requirements to a modern robot control architecture, like modularity, extensibility, efficiency, customizability, reusability, and rapid application development, in continuation of our approach presented in [9] we have separated the robot-specific methods and skills from the application itself resulting in a flexible layered system architecture (Fig. 3). In this architecture, the bottommost layer, the *Hardware Layer*, encloses the robot hardware (sensors and actuators), the operating system, and the low-level interfaces to the hardware. The low-level sensor information is processed in the next higher level, the *Skill Layer*, which provides a set of necessary, robotic-specific basic skills that are executed in the Hardware Layer. These skills are only weakly coupled and do not need any coordination due to non-conflicting hardware resources. The *Skill Layer* covers the whole spectrum of navigation skills including map building and localization. Furthermore, specific perception skills, like person detection and tracking as well as face analysis, are implemented in that layer. Speech synthesis and the GUI are also placed in this layer and provide their services to the higher application levels.

Above the independent skills there are diverse modules representing the application control which make use of the basic features provided by the skills. The simplest controllers using the navigation and people detection skills are the *Navigation Behaviors*. These are exclusive units each representing an individual control loop for accomplishing the different task-oriented navigation behaviors of the robot. Here, for example, a ”Search user” behavior is realized that uses the history of person detections and a set of navigation points, which are checked one after the other in order to cover the whole apartment, while searching for the user using the ”Person detection and tracking” and ”Navigation” skills. Other behaviors are ”Approach user” and ”Follow user” which are necessary for direct interaction as well as passive user observation, which is active most of the time when no direct interaction takes place. If this navigation behavior is active, the robot looks for an unobtrusive position, from where it has a direct view to the user in order to recognize critical situations, like fall events.

The exclusive navigation behaviors as well as the input and output skills are utilized by the modules of the *Control layer*, which form the ”Dialog Manager” and the ”Global flow control”. The ”Dialog Manager” consisting of an input interpreter, a frame manager, and an output generation module is responsible for organizing the direct interaction with the user. Here, the infrastructure for multi-modal input fusion, coordination of the dialog progress, and generation of multi-modal outputs using speech, display, and facial expressions is provided. The ”Dialog Manager” follows a frame-based approach, where the content of the dialog is defined in a set of scripts residing in the highest layer, the *Task Layer* where the content of the application is defined.

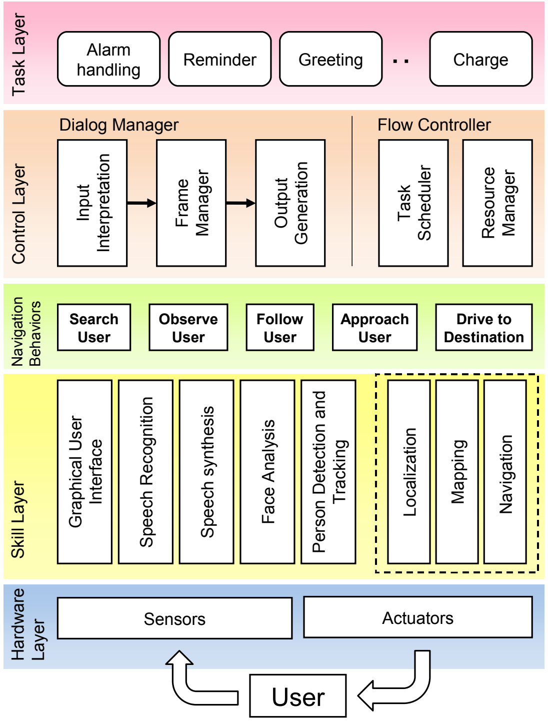

Fig. 3. Main components of the robot’s layered system architecture.

Since the ”Dialog Manager” is more a reactive subsystem handling interaction with human once the dialog has been initiated, the proactive part consists of a ”Task Scheduler” and a ”Resource Manager”, which are responsible for coordination of the partially competing task controllers implemented in the *Task layer*. These controllers are realizing different primitive control loops, as e.g. adjusting the display to the user position or coordinating the charging of the robot, which incorporates a docking to the charging station as well as an undocking if any other task needs to drive around. Also hierarchical dependencies among these tasks can occur.

By means of this decomposition, we could avoid instantiating one complex state machine for application control and preserved extensibility and reusability. Coordination among these controllers of the *Task layer* is realized by a set of system resources, which are granted according to a dynamic prioritization by means of the ”Resource Manager”. This prioritization is also represented in the ”Task Scheduler”, which is responsible for the activation of event- and time-triggered tasks, like reminding to take medication, giving advice in daily activities, or offering the cognitive training exercises. All these tasks have a priority value allowing for a coordination with the user's current activity, in order to be situation-adaptive and not disturbing.

Example for activation flow: For better explaining the internal flow of activation, a typical use case has been selected - the delivery of a reminder (e.g. to take medication), which is defined by a period in time when it has to be delivered, and a priority to prevent an immediate interruption of other ongoing and possibly more important activities. In that case, the robot normally is in the passive observation behavior, and the "Person detection and tracking" skill tries to recognize the user's current activity, which is written to a user state model. The "Task Scheduler" now waits for an adequate moment in time when the user has finished the current activity and is available for a reminder dialog. In this case, the "Task Scheduler" will indicate the need for interaction to a controller in the *Task layer*, which will request the drive resource in order to activate the "Approach user" behavior. Hence, the other controllers have to reach a safe state and have to release the drive resource first. Once the "Approach user" behavior has been activated, the "Navigation" skills are utilized for driving to the position determined by the "Person Detection" skill. In case of a user found in interaction distance to the robot, the "Task Scheduler" finally will execute the task and trigger the respective frame in the "Dialog Manager". This reminder frame will use the "Dialog output" generation in order to express the message verbally as well as on the screen. A possible answer given by the user will be recognized by the "Speech recognition" skill or the touch display. Also a head nodding is possible and will be interpreted by the input interpretation module as a confirmation that the message has been understood by the user. In case of a missing confirmation or a request for presenting the reminder later, the frame can reschedule the reminder at the task scheduler. If the message has been delivered successfully, the frame will be finished and the dialog will be finalized.

V. TECHNOLOGIES FOR USER-CENTERED NAVIGATION

A. The Person Detection and Tracking Skill

The robot companion offers several service functionalities and behaviors (see Fig. 3, top layer) that require knowledge about the care recipient's (CR) current position in the apartment. For example, the robot has to seek for the CR if a reminder has to be delivered or a video call comes in. In that case, the robot should also be aware if the CR is not at home. Another behavior is "Following" and "Approaching" the user if interaction is desired (see Fig. 3, middle layer). A prerequisite to these behaviors is the robust detection and tracking of the user in the apartment. Also a non-intrusive observation of the CR to provide services that require interaction or to react on critical situations needs knowledge about the current position of the CR.

In contrast to other interaction applications in public environments, people in home environments often do not face the robot in an up-right pose but sit on chairs or lie on sofas. Therefore, our system combines state-of-the-art methods for up-right pose people detection with a module to detect users independently of their pose at places, where they usually rest. The key idea of this module is to learn color-based models of the user's appearance and the appearance of predefined resting places beforehand. In the detection phase, the current visual impression is compared to both of these models to decide if the learned user is present. Given a user searching task, the robot system is optionally supported by infrared motion sensors of the smart home system. These sensors detect motion in the apartment and can be used as a hint where to search first.

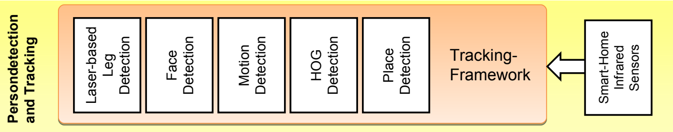

Fig. 4. Overview of our person detection and tracking system. Different observation modules are utilized in the tracking framework, while detection results of smart home sensors can be integrated optionally if available in the respective apartment.

1) User Detection and Tracking: Our approach comprises a multi-modal, multi-cue tracking framework based on Kalman Filter techniques presented in [10]. The key idea of our new approach is a set of independent 3D position hypotheses of persons, which are modeled by Gaussian probability distributions. Adding their movement velocity results in a 6D-state space $S = (x, y, z, v_x, v_y, v_z)$ for each hypothesis. This increases the prediction quality enormously and enables the system to distinguish hypotheses if their position coincides. New position observations provided by the detection modules are associated with the closest hypothesis in the system. If a distance threshold is exceeded, a new hypothesis is introduced. Once the system knows the associated source for that observation, the position of that hypothesis is updated using the Kalman Filter updating.

The tracking framework (Fig. 4) merges the output of different detection modules. We apply laser-based leg detection and multiple visual detection modules – namely faces, motion in the image, and two HOG person detectors [11]. Each of these detection modules produces Gaussian position hypotheses if it is activated by its specific input. We use the head of a potential user as the reference for alignment. The differing detection quality of the different detection modules is taken into account by selectively presetting the covariances of the 6D Gaussian distribution, i.e. for the laser-based detection with low variances in distance and direction but a high variance in the height, while a visual detection gets a high initial variance in the distance estimation.

a) Laser-based detection: The laser-based leg detection applies boosted classifiers to detect human legs in laser scans [12]. By applying the well-known AdaBoost algorithm, the system selects a set of weak classifiers, each of them working on a predefined feature, to distinguish leg from non-leg examples in a training phase. In the recognition phase, a weighted sum of the best weak learners' votes classifies new examples. By searching for paired legs, the system produces hypotheses of the user.

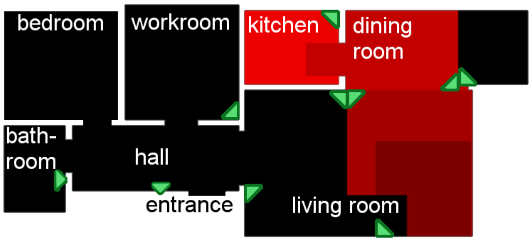

Fig. 5. User position likelihood map of the apartment used for tests: triangles indicate positions of motion sensors. Red color codes the time elapsed since the respective movement sensor was activated for the last time (the brighter the more recent).

b) Visual detection: The motion detection cue is only activated when the robot is standing still. Then we apply a fast and simple image difference approach. A noise and complexity reduction is achieved by working only on sums of image columns [10]. For face detection, we utilize the well-known face detector of Viola & Jones [13], while for non-face based person detection a modified version of Dalal's Histograms of Oriented Gradients (HOG) [11] is used. To cope with partial occlusion of the body, we are currently developing a combination of a part-based version of the HOG detector [14] and an upper body detector for frontal-view sitting people [15].

c) Detection at common resting places: The system described so far is able to detect and track people standing or moving in the local surroundings of the robot. To robustly detect sitting or even lying persons, a pre-training based approach is applied. The key idea of this approach is to learn the visual appearance of predefined places, where the user usually rests. We restrict the view-points of these places to those predefined observation positions, which are also used for the *Search user* and *Observe user* behaviors. For each place, a multi-modal color histogram is learned, given different times of the day and illumination conditions. An important condition is that the user is not occupying these places during the learning phase. Furthermore, we learn a multi-modal color model of the user when the tracker is very certain about the user's position and the robot is standing still. For this purpose, background subtraction [16] and GrabCut [17] are applied to segment the user from the background. Using this knowledge, the robot can search for the user at different places by comparing the current color model of the place with the learned model of the empty place and the model of the user. This comparison is done by a pre-trained linear Support Vector Machine (SVM) which receives the similarities to the user and place models as input. To be still more robust against illumination changes, a complementary HOG model is learned for each place which classifies empty places by another linear SVM.

2) Integration of Smart Home Sensors: Every time the robot needs to find the CR in the apartment, a proper search strategy has to be activated in the "Search User" behavior. Then the robot checks each of the aforementioned observation positions one by one for the user's presence. By incorporating the detection capabilities of IR motion sensors installed in the smart home environment, a more sophisticated search strategy can be applied. Each of the installed motion sensors can detect a moving person in a 90^o area within a maximum range of about 4 meters. The achievable spatial accuracy of the sensors is not sufficient for fine user detection suitable for interaction, but more than enough to decide where to search first. The user position likelihood map shown in Fig. 5 is built up by the robot using the history of activations of the IR motion sensors. In the given example, the CR was sitting on the couch in the living room, before s/he moved through the dining room to the kitchen, where s/he is resting now. On a given "Search User" task, the robot now can start its search with the place with the highest (most recent) activation (the kitchen), before it begins to check the other places with weaker activations (dining room, living room).

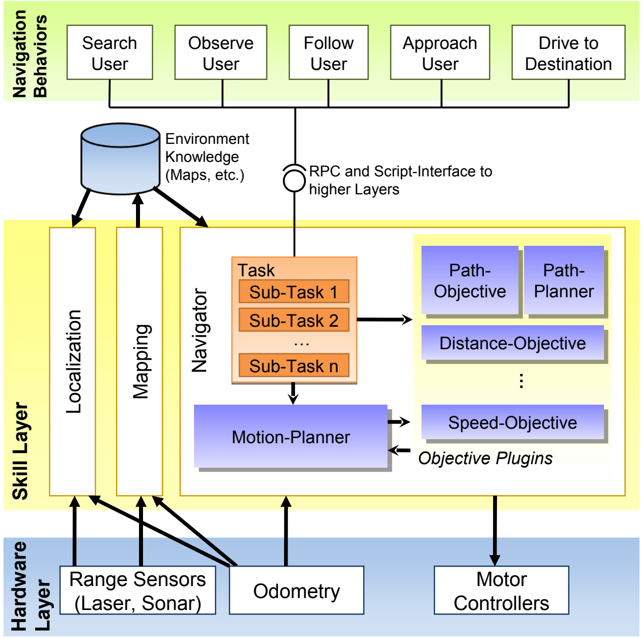

Fig. 6. More detailed view of the navigation skill (navigator) and its embedding into the overall system architecture shown in Fig. 3.

B. The Navigation Skill

The process of navigating in a home environment is not simply to move from point A to point B. It involves the aforementioned behaviors (see Fig. 3, middle, and Fig. 6, top) that have to be consistently integrated, as searching for the CR, observing and following the CR, driving to certain places within the home, interacting with the CR (e.g. let her put and retrieve items from the robot's storage box or ad items to the agenda), autonomous docking to the recharging station, or being remotely controlled from outside.

The core components for navigation are located within the "Skill layer" of the system architecture (see Fig. 3). They consist of mapping and localization algorithms and the navigator itself. Fig. 6 shows a more detailed view of these components within the system architecture. To represent the local surroundings of the robot, we use occupancy maps built using the laser scanner as well as a visual approach for monocular scene reconstruction [19]. The created maps are used by a localization approach to estimate the robot location in the global coordinate frame by using a local map of recent observations and the knowledge of the global map. Similar to the overall robotic architecture, we also use a modular software design for the navigator with several internal components as shown in Fig. 6. The core component of the navigator receives the current navigational task from overlying layers like "Navigation Behaviors" or the "Application Layer" via RPC (Remote Procedure Calls) or using a built-in scripting language. Both enable the control of the navigator from distributed applications.

After a task is received by the navigator, it is decomposed into distinct motion and velocity commands of the robot's wheels in order to process the desired task. For task processing, the motion planner and the objectives play a major role here. We decompose all navigational tasks and behaviors into objectives. Each objective is a separate software module specialized for certain tasks, like following a path or a person, driving at a certain speed or direction, etc. The objectives are realized as separate software plug-ins (see Fig. 7). This allows us to add new objectives easily, when new tasks and behaviors become necessary without changing existing parts of the navigator. The output of the objectives is then used by the motion planner to generate motion commands that are then sent to the robot's motor controllers. At the moment, we use the Dynamic Window Approach (DWA) [18] as motion planner. However, due to the modular concept arbitrary motion planning algorithms could be applied that operate in the robot's velocity space and that compute a cost- or fit-value based on the objectives. Some objectives require additional information from other navigational modules, such as localization and mapping algorithms or modules for user interaction, like person trackers. Therefore, the objectives can access this information directly from these modules.

Tasks and Objectives: Based on this modular concept of objectives, we designed a suitable interface for the navigator that allows us to specify complex tasks. In order to develop a highly generic navigator, we introduce a task-based system which allows us to define jobs consisting of a list of several sub-tasks and their corresponding parameters. To meet the requirements of different applications, we have defined several sub-tasks that can be combined to create complex tasks, as for example the PositionTask (drive to a specified target position), the OrientationTask (adjust the orientation at the goal point), the LocalTargetTask (drive to a specified local target without using a global path planner), or the PreferredDirectionTask (drive in a preferred direction) (Fig. 7).

Objectives	**Supported Subtasks**	**Description**
Distance	*enabled on demand*	avoids collisions by calculating the distance between robot and obstacles on the predicted trajectory
Path	**PositionTask**	plans paths to a target pose; tries to follow the planned navigation function
Heading	**OrientationTask**	tries to achieve the specified orientation and alignment at the target position
ASTRoNAuT	**LocalTargetTask**	positions the robot very accurately at short-range w/o planning a global path, e.g. for docking at charger
Direction	**PreferredDirectionTask**	reinforces movement commands that move the robot into the specified preferred direction
Mileage	**MileageTask**	allows driving commands only until the specified distance is reached
No-Go	*enabled on demand*	avoids areas, the robot is not allowed to enter (specified using a No-Go map).
Explore	*enabled on demand*	moves robot in forward direction, prefers driving straight ahead; can be comb. with Distance/ Mileage Objectives to achieve collision free exploring behav.
PersonFollow	*enabled on demand*	follows a person that is detected and tracked by a person tracker; is enabled in the robot's Person Follow and Observation Behavior

Fig. 7. Subset of the already implemented navigator objectives

To fulfil the job, the navigator must process all sub-tasks in parallel by following the specified rules and options. Whenever a new complex task is set, the navigator asks each of its objectives if it is able to contribute to a sub-task, otherwise the objective is disabled to save resources. Additionally, the objectives can be enabled or disabled by *Navigation Behaviors* from the application depending on the robot's current active *Navigation Behavior*. Afterwards, the enabled objectives and the motion planner continuously compute motion commands until each sub-task has been achieved, and the complex task is completed. So far, the objectives shown in Fig. 7 have been implemented. This modular system of tasks and objectives allows the addition of new sub-tasks and objectives in the future, when new tasks and behaviors are required, without changing the interface of existing tasks or interfering with existing applications. This is an important advantage of this design since it reduces the effort necessary for software maintenance.

VI. FUNCTIONALITY TESTING AND USABILITY STUDIES

Functional testings are still being conducted to assess the functionality and the performance of the different subsystems and the robot companion as a whole before usability studies can take place. To that purpose, we defined a number of functional tests with respect to the navigation and HRI functionalities of the robot companion. With respect to the *navigation functionality*, a spectrum of capabilities has already been assessed in functional tests, as for example to build a map of the apartment while the robot is manually driven around, to self-localize within the apartment with an accuracy of 0.2 m or better, to avoid collisions with known and unknown obstacles, to efficiently pass through narrow doors and hallways, and to autonomously drive to and dock to the charging station. Most of these tests have already been completed successfully from qualitative point of view, only the autonomous driving and docking to the charging station is an open issue which needs to be finished until the upcoming long-term user studies. A quantitative analysis of the navigation capabilities regarding localization and positioning errors, velocity profile and smoothness of the movement trajectories, distances to obstacles, etc. is being conducted currently.

For *HRI*, diverse functionalities have been defined as mandatory for the companion. They are also currently being tested and evaluated, as for example to robustly detect and keep track of a moving/static person, to orient towards a user or drive in a position facing the user, to follow a user through the room and the apartment, to search a user in the apartment, to express simple emotions by using its facial capabilities, to notice when personal items of a user are placed in its storage box, etc. Most of these tests have also been successful from qualitative point of view. First quantitative studies have been finalized quite recently, and we expect the others to be finished by fall 2011. For example, the *User detection and Tracking* skill has been evaluated as follows: detection rates were determined for different places in the apartment (see Sec. V-A), and the average time required for successfully locating the user in his/her apartment using the *Search User* behavior was determined. First quantitative results of these hide-and-seek studies are presented in [8].

The focus of the still pending *usability studies* will be put on the evaluation of the user-robot interaction, and the way the system is perceived by the users as meeting their needs. This will reveal information about the perceived user comfort and quality-of-experience (QoE), as well as the level of conformance of the robot companion with all relevant guidelines. To that purpose, a number of scenarios was defined which build the basis for test cases for evaluating the usability of the robot companion. Already specified test cases are, for example, initiating a videophone call to relatives or a care service, reminding the user to do a cognitive stimulation exercise, reminding of a forgotten event (e.g. taking medication), greeting the user at the entrance of the apartment when returning home including storing personal items in the robot's storage box, and detecting a fallen user anywhere in the home and initiating an alarm call.

VII. SUMMARY AND OUTLOOK

In this paper, we have presented an overview of the progress in developing a mobile, interactive home robot companion for elderly people suffering from mild cognitive impairment (MCI) and living alone at home. The paper is addressing several aspects of our work beginning with related work in the field of socially assistive robotics for domestic use, continuing with the robot hardware and its software and system architecture, via required skills for a user-centered navigation, up to the already conducted and still ongoing functionality testings. Future work includes quantitative analyzing and benchmarking the navigation and HRI capabilities regarding localization and positioning errors, smoothness of the movement trajectories, success and error rates of the observation modules, and the person detection and tracking system as a whole.

The assessment of usability is going to be started in summer 2011 and will be done by *domain experts* which includes the observation of the test participants by recording usability problems, and *real end-users* - volunteers suffering from MCI. The evaluation will take place in two different settings. A first setting will be provided with the robot in a fully equipped smart house of the Dutch project partner Smart-Homes (www.smart-homes.nl) in Eindhoven, the Netherlands, which serves as a test house in general. In this case, the test users will temporarily stay in the house and perform the aforementioned test cases. Because it is important for persons with MCI to stay in their own home environment, a subset of the usability test will be conducted with the robot companion and the volunteers in their own accustomed apartments. Because of its strong interdisciplinary nature, partners from a variety of disciplines (for details, see [1]), namely the provider of the smart home in Eindhoven, consortium partners with strong geriatric background, and experts for usability engineering and ethical evaluation will be closely involved in the upcoming usability studies.

REFERENCES

[1] CompanionAble - Integrated Cognitive Assistive & Domotic Companion Robotic Systems for Ability & Security (EU-FP7 project; 2008-2012); www.companionable.net

[2] A. Tapus, M. Mataric, B. Scassellati, "Socially Assistive Robotics", in *IEEE Robotics & Automation Magazine*, 14 (2007) 1, 35-42

[3] B. Graf, H. Staab, "Service Robots and Automation for the Disabled/Limited", in *Springer Handbook of Automation*, 1485-1502, Springer 2009

[4] M. Montemerlo, J. Pineau, N. Roy, S. Thrun, and V. Verma, "Experiences with a Mobile Robotic Guide for the Elderly", in *Proc. AAAI Nat. Conf. on AI 2002*

[5] Project homepage: www.cogniron.org

[6] Project homepage: www.robots-at-home.org

[7] Project homepage: www.lirec.org

[8] H.-M. Gross, Ch. Schroeter, S. Mueller, M. Volkhardt, E. Einhorn, A. Bley, T. Langner, M. Merten, "Progress in Developing a Socially Assistive Mobile Home Robot Companion for the Elderly with Mild Cognitive Impairments", in *Proc. IROS'11*, to appear

[9] H.-M. Gross, H.-J. Boehme, C. Schroeter, S. Mueller, A. Koenig, E. Einhorn, Ch. Martin, M. Merten, A. Bley, "TOOMAS: Interactive Shopping Guide Robots in Everyday Use- Final Implementation and Experiences from Long-term Field Trials", in *Proc. IROS'09*, 2005-12

[10] H.-M. Gross, H.-J. Boehme, C. Schroeter, S. Mueller, A. Koenig, Ch. Martin, M. Merten, A. Bley, "ShopBot: Progress in Developing an Interactive Mobile Shopping Assistant for Everyday Use", in *Proc. SMC 2008*, 3471-78

[11] N. Dalal, B. Triggs, "Histograms of Oriented Gradients for Human Detection", in *Proc. CVPR 2005*, 886–893.

[12] K.O. Arras, Ó.M. Mozos and W. Burgard, "Using Boosted Features for the Detection of People in 2D Range Data", in *Proc. ICRA 2007*, 3402–3407.

[13] P. Viola, M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", in *Proc. CVPR 2001*

[14] P.F. Felzenszwalb et. al, "Object Detection with Discriminatively Trained Part Based Models", in *IEEE Trans. on PAMI*, 2009, 1–20.

[15] V. Ferrari, M. Marin-Jimenez, A. Zisserman, "Progressive search space reduction for human pose estimation", in *Proc. CVPR 2008*, 1–8

[16] C. Stauffer, W.E.L. Grimson, "Adaptive Background Mixture Models for Real-Time Tracking", in *Proc. CVPR 1999*, 2246–2253.

[17] C. Rother, K. Vladimir, A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts", in *Proc. ACM SIGGRAPH 2004*, 309–314.

[18] D. Fox, W. Burgard, S. Thrun, "The dynamic window approach to collision avoidance", in *IEEE Robotics & Automation Magazine*, 1997, 23–33

[19] E. Einhorn, Ch. Schroeter, H.-M. Gross, "Finding the Adequate Resolution for Grid Mapping-Cell Sizes Locally Adapting On-the-Fly", in *Proc. ICRA 2011*, pp. 1843-1848