Dependable Interaction with an Intelligent Home Care Robot

Birgit Graf, Martin Hägele
Fraunhofer Institute for Manufacturing Engineering and Automation (IPA)
Nobelstrasse 12, 70569 Stuttgart, Germany
Email: {btg, mmh}@ipa.fhg.de

Abstract

Care-O-bot is the prototype of a multi functional home assistant, to be used by elderly people in order to live independently in their homes for a longer time. Easy, intuitive, and dependable operation of the home care system is essential. The interface to the robot consists of a graphical I/O display, speech in/output and various safety and navigation sensors. A three level safety system has been installed on the robot for dependable collision avoidance. To improve fault tolerance, a sensor library has been developed which enables to load drivers for newly detected devices on the fly. Standardized I/O- and diagnosis methods are provided together with the driver so that correct operation of the sensors can constantly be monitored by the application.

1. Introduction

Figure 1. Intelligent Home Care System Care-O-bot

Medical expenses are increasing all over the world. More than 30% of these expenses can be related to elderly people. By the year 2030 the number of 60-year-old people will have doubled [14]. Technical aids are required allowing people to live independently and supported in their private homes as long as they wish. As a contribution to these required technological solutions, a demonstrator platform for a mobile home care system – called Care-O-bot (Figure 1) – was designed and realized by Fraunhofer IPA [15].

Safety is the most essential issue in order to get people to trust and live with the assistant without worries, especially for users who might not have any detailed technical knowledge.

2. Safety Issues

Despite several considerations no standard procedures or solution catalogues have been provided for the safety conformable design of robot assistants as is the case with industrial robots [1]. In order to make service robots safe enough to operate in public environments the following process given for industrial robots must be modified according to a given application [7]:

- The starting point should be a risk assessment to determine the probability and the consequences of a system failure. This process is standardized and well documented [4]. Based on the determined safety category adequate components can be selected, procedures and precautions be designed.
- When possible, simple means such as the reduction of forces or moments exerted by moving parts should be pursued.
- Redundancy should be employed in safety critical systems (sensors).

A safety conformable design for robot assistants can be achieved. The following solutions have already been accepted for recent projects:

- Mobile robots exposed to public environments can be made safety conformable with limited effort (use of “category four” laser scanner or “bumper”).
- Service robots such as in the case of automatic refueling comply with safety regulations in fully automatic mode by limiting forces and velocities.

Additional experience such as the prototype operation of service robots, e.g. in private home environments, are crucial for further safety evaluation.

3. The Care-O-bot Mobile Robot Platform

Care-O-bot is the prototype of a robotic home care system developed at Fraunhofer IPA. A first mobile platform has been built in 1998. Care-O-bot has already proved its ability to operate safely and reliably in public environments. Three robots based on the same hardware platform have been installed in March 2000 for constant operation in the "Museum für Kommunikation Berlin" where they autonomously move among the visitors, communicate to and interact with them [6] [16].

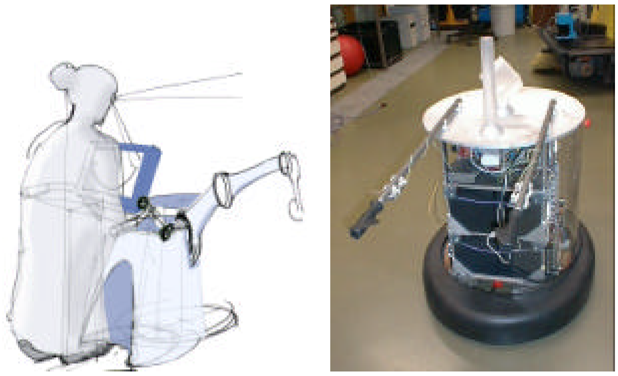

Figure 2. Care-O-bot II design and prototype in its current version (Jan. 2001)

Care-O-bot II (Figure 2) will be equipped with a manipulator arm and adjustable walking supporters. The basic platform has already been built. In order to provide support for walking and standing up, two supporting arms have been attached to the mobile platform. A manipulator arm to be used for fetch/carry and handling tasks will soon be added to the platform.

4. Software Implementation

A basic software architecture is currently being integrated. It will meet the following central requirements:

- Arbitrary and simultaneous use of various input channels (touch screen or speech) and situation-dependent use of appropriate output-channel.
- Automatic selection among alternative plans according to optimization criteria, possibility of interactive or autonomous plan modification after changing goals or sub-goals.
- Translation of goals into actions, motion commands and skills (an encapsulation of motion and sensing tasks).
- Dependable navigation system including path planning, obstacle avoidance and sensor based motion.
- Automated diagnosis and control of sensors and actuators.

4.1. Primary User Interface and Task Planner

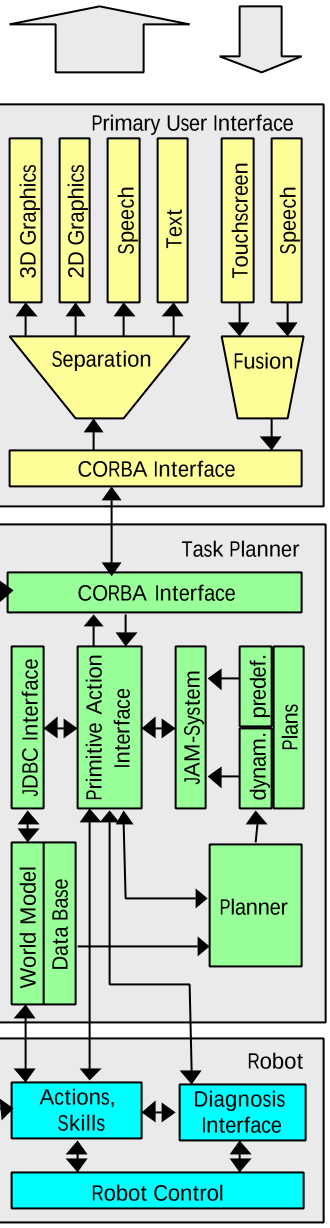

Figure 3. High Level Software Architecture

The primary user interface and task planner (Figure 3) have to play together properly for the robot to act correctly according to its user's inputs. After an input has been given through the user interface, a plan to solve the user's request must be generated. This plan must be presented to the user for possible corrections. Only when it is clear that a correct plan has been chosen, the robot can start executing the selected actions.

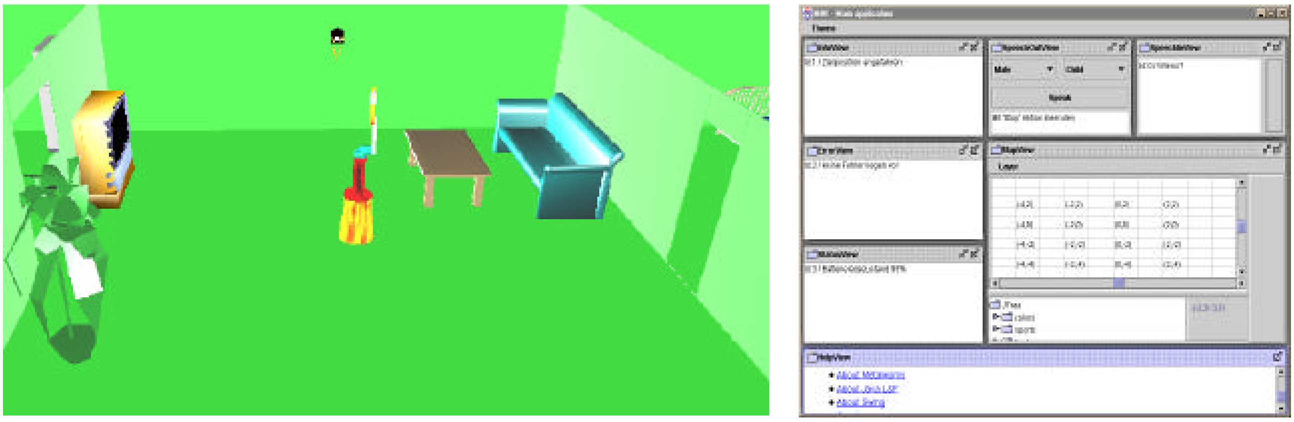

Figure 4. 3D- Visualization of the environment (VRML-simulation) and user control panel

The primary user interface includes the graphical interface (Figure 4), used for visualizing the robot and user's environment as well as a speech in/output system. The user control panel displays all current variables of the robot, as e.g. the motion state, the position, or planned actions (e.g. planned path) of the robot, state of the speech input system, and possible error messages. , An online help system is further provided. Central components of the user interface are the fusion module which merges all information given through the communication channels and the separation module which relates the output on the adequate channels.

As the basis for the planner JAM-software [8] was chosen. The package supports both top-down, goal-based reasoning and bottom-up data-driven reasoning. JAM selects goals and plans based on maximal priority. The planner is composed of five components. A world model saved in a database represents the environment and abilities of the robot. The plan library is a collection of plans that the agent can use to achieve its goals. The interpreter is the planner's "brain" that reasons about what and when the agent should do it. The intention structure is an internal model of the goals and activities the agent currently has and keeps track of progress the agent has made toward accomplishing those goals.

4.2. Dependable Navigation of the Robot as an Intelligent Walking Aid

Secondary input components comprise sensors which reflect the intention of the user, like e.g. force or force/torque sensors. According to the data read by the sensors, actions in the robot control program (navigation of the vehicle) might be triggered. Changes of action caused by input read from the secondary interface are documented on the primary user interface for better understanding and to provide the possibility to make changes. For example, if the robot is being pushed forward by the user it will interpret this action as an order to move. The assistant will document its planned movement on the graphical interface. If the intention of the user has been misinterpreted, the user can correct or cancel the action planned by the robot.

Figure 5. Walking aid handle with sensor (prototype realization)

One application of the secondary user interface is navigation while using the walking supporters of the robot [5]. In order to enable easy manipulation of the robot platform, the way to use it as a walking aid has been adapted to conventional walking aid systems. The robot drives in reaction to input forces given by the user. These forces are measured by sensors integrated in the walking supporters (Figure 5). For example, if the user "pushes" the robot forward, the vehicle will start moving in the required direction. As an improvement to conventional walking aid systems, intelligent behaviors are included.

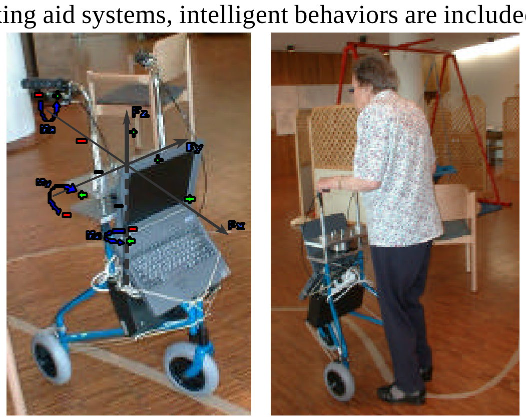

Figure 6. Test platform for force and torque measurements

In order to measure the forces and torques which are applied to a conventional walking aid system during motion, a test model consisting of a three wheeled walker and a force torque sensor mounted between the handles and the base of the walker (Figure 6) has been built. A field test was done with people from a nursery home that walked the vehicle along an S-shaped test route.

For the autonomous system, two major modes of operation have been implemented: (1) direct user control, enabling the user to "push" the robot to a certain direction, and (2) target mode, with the user following the robot to a specified goal along a preplanned path.

In the first operation mode, the robot moves exclusively in reaction to forward pressure applied to the walking supporters. Due to fluctuations of the applied forces, values read from the sensors need to be filtered. The approach applied uses the filtered input values to directly influence the speed of the robot.

In case of obstacles in the assigned moving direction of the robot, the calculated velocity vector must be modified to lead the robot along a collision free path. It is further important that the robot can be moved close enough to objects for the user to be able to touch them, e.g. pick up things from a table. However, it must not get too close to an object in its environment so as getting away from it might become impossible, given the motion constraints for being followed by a user. To enable this, parameters for obstacle avoidance must be chosen carefully and the heading of the robot towards an obstacle must be considered.

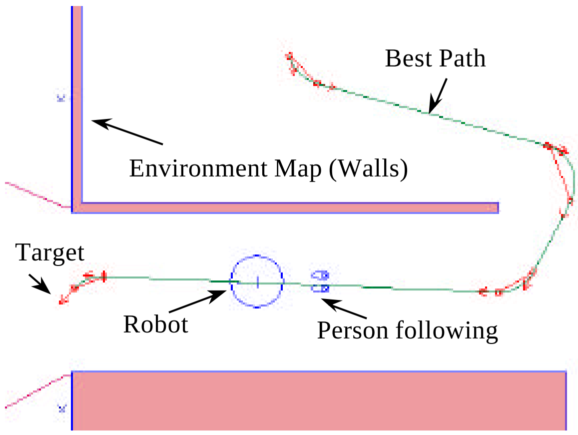

Figure 7. Collision free path planned for the robot while being used as an intelligent walking aid

If the user specifies a target (through primary interface) within the known operation area of the robot, the latter will find and follow the best path (Figure 7) leading to this position. Hereby "best" does not in any case mean shortest, safety issues require the path to e.g. not include sharp turns. Furthermore, restricted operation areas must be considered. A path planner has been developed, based on the algorithm presented in [10]. It employs a global-local strategy, and solves the problem in the 2D workspace of the robot, without generating the configuration space.

Dynamic path modifications due to obstacles which have not been enlisted in the map or in reaction to input via the secondary user interface are applied based on [9].

Safety precautions have to ensure that a person using the robot for support can rely upon the assistant and will never be exposed to any safety hazards. For example, the robot must continuously check whether its user is still behind and it must never execute movements which its user is unable to follow.

4.3. Three Level Safety System

One of the most common accidents caused through industrial robots is a person being hit by the robot [2]. For stationary robots the responsibility lies partly with the user – safety measures, as e.g. keeping a certain distance to the robot, must be obeyed. For mobile robots, however, all responsibility lies by the vehicle, therefore the major goal for safe operation should be to prevent a mobile robot from driving into people or from leaving its operation area which might lead to additional incidents as e.g. by a fall down stairs onto people.

For maximum safety a redundant three level safety system has been implemented on Fraunhofer IPA's mobile platforms.

Level one is the laser scanner based collision detection (Sick LMS [18]). Whenever an obstacle is detected in the robot's vicinity, the speed of the vehicle is reduced at a degree depending on the distance to the obstacle. If an obstacle or a person gets too close to the vehicle, the robot will stop and wait until the area is clear again.

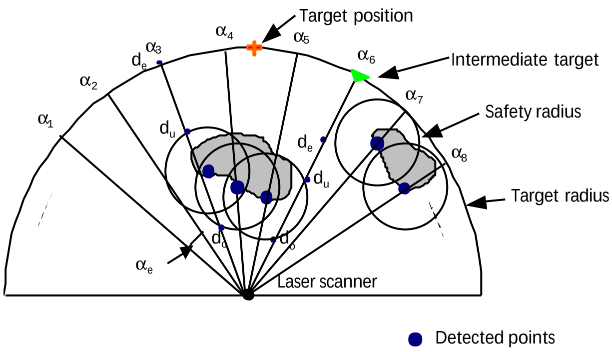

Figure 8. Reactive obstacle avoidance using the "PolarBug" algorithm

The safety module "obstacle detection and surrounding" (Figure 8) is applied in order to avoid unnecessary acceleration and deceleration caused by the collision avoidance. Obstacles detected by the laser scanner are surrounded in advance. The reactive obstacle avoidance algorithm PolarBug [17], based on the VisBug method [12] is being used. This algorithm has been developed especially for obstacle detection with a laser scanner, as well as for fast reaction and navigation in unsteady environments. The major difference to common obstacle avoidance algorithms is the direct processing of the laser scanner data (polar coordinates) which enables a very high efficiency of the algorithm.

Data not only in the planned path of the robot, but all measurements of the laser scanner are evaluated. In case obstacles have been detected between the current position of the robot and a given target, an intermediate position is being calculated which brings the robot around the obstacles as fast as possible. The best free passage is found considering several parameters like e.g. width and depth of passage, deviation of passage from direct line to target and distance of intermediate position to robot and final target position. All relevant factors are joined using a fuzzy logic approach.

The laser scanner is further used for robot localization by matching the robot's surroundings with a given local map. Allowed operation areas can be defined in the map.

Apart from the laser scanner the robot is equipped with a rubber bumper all around the vehicle. Activating the bumper results in an immediate stop. The operation speed of the robot is initially restricted depending on the size of the bumper – so that it can always stop before the bumper is crushed completely. In order to secure the area above the laser scanner, several infrared sensors have been integrated in the bumper facing upwards.

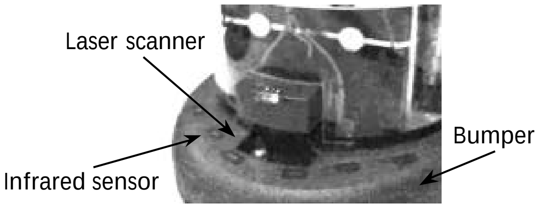

Figure 9. Safety sensors

Thirdly, each robot is equipped with magnetic sensors facing to the ground. They are used as a secondary system to ensure no robot ever leaves its operation area. In the unlikely case of a software failure, by leaving the given operation area and therefore crossing a magnetic band lowered in the ground, an emergency stop will be activated. In addition, each robot is equipped with two emergency stop buttons to deactivate the robots manually.

For applications where the mobile robots move among people in public environments, this safety system has been accepted by the responsible professional association. Furthermore, a CE certification could be acquired for the robots.

4.4. Runtime Diagnosis System

Conventional safety sensors provide different diagnosis tools for each sensor. Simple software interfaces to integrate the diagnosis of the sensors into existing programs are often not available. Different conventions concerning error codes can further be observed. Due to this diversity, during the development process, the integration of diagnosis and error check functionalities is often given a too low priority. This results in complex and error-prone diagnosis and monitoring software. Safety hazards caused by sensor failure might not be noticed.

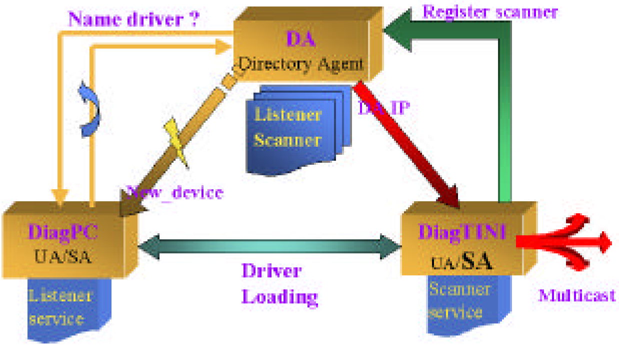

Figure 10. Automatic component detection

The following scenario (based on the "Service location protocol" SLP [11], Figure 10) has been created as a basis for Fraunhofer IPA's implementation: *A new component (service) has been attached to the control network. The service agent running on this component automatically registers with the directory agent in the network. Other agents (e.g. the user agent of an application, registered as a "listener") can detect available services registered within the directory agent. A software driver (class) can be loaded from the desired component enabling the user agent to access functionalities of hardware modules attached to the component (user interface, function calls, diagnosis).*

Because standard SLP does not monitor the correct operation of its components regularly, a new functionality "Live check" has been added for the directory agent. All registered services are checked regularly. If one of the services does not answer, it will be deleted from the list of the directory agent and a message to all affected listeners will be sent.

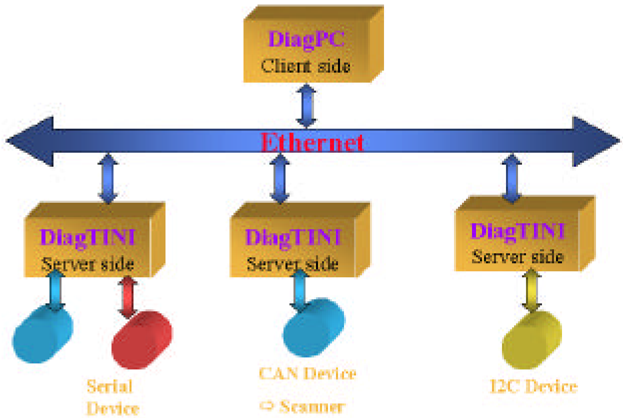

Figure 11. Local communication network

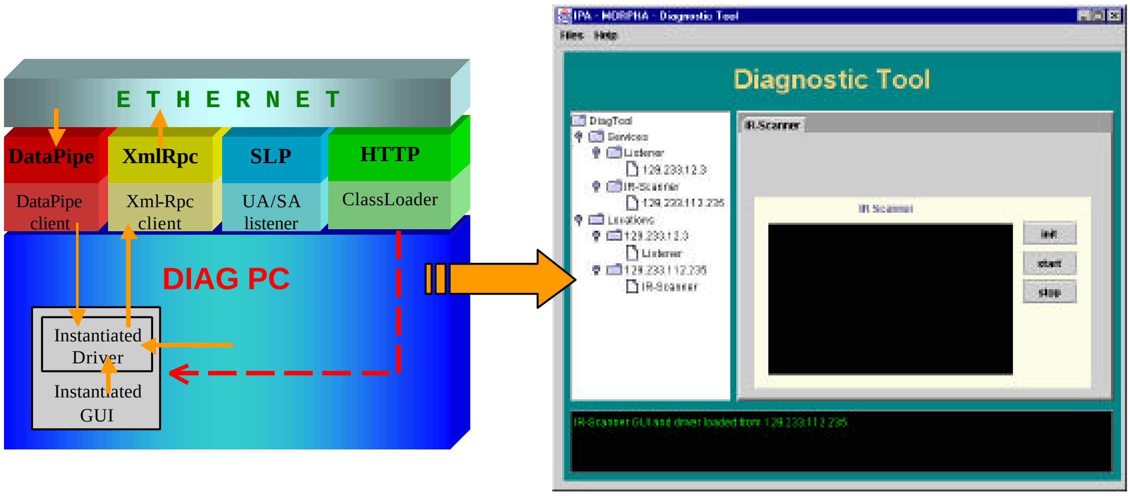

Figure 12. I/O connections of the diagnostic tool (control program side)

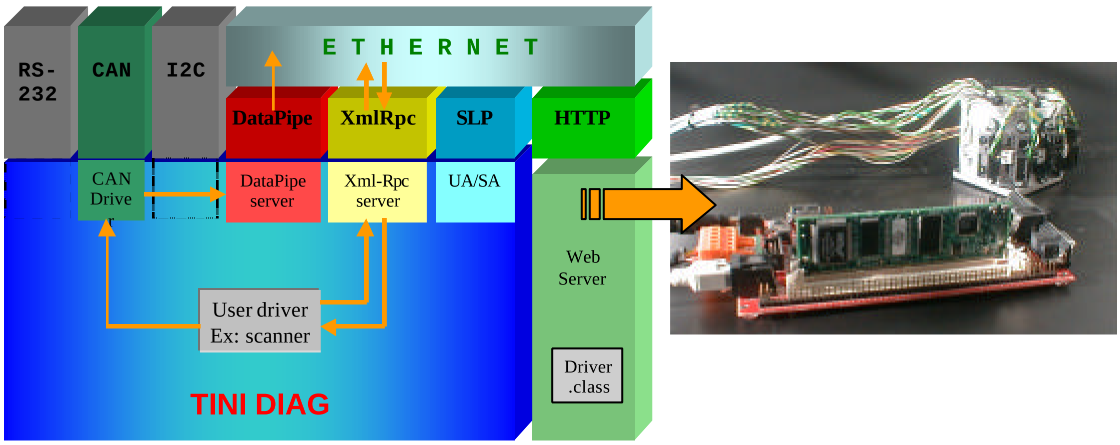

Figure 13. I/O connections (component side)

All components of the sensor network are connected via ethernet (Figure 11). TINI micro controllers [3] have been chosen as an interface for devices (sensors) which do not include their own ethernet interface. The chosen controllers meet all requirements: they support JAVA, offer several different interfaces (digital, analog in-/output etc.) to attach sensors and actuators, they can be connected to the ethernet and are available at a relatively low price.

Software drivers to be loaded from the components are implemented in JAVA. They are made available on each component through a HTTP server. Remote calls of component specific function calls are executed from the user agent using the XML-RPC [19] protocol. All I/O connections for the diagnostics tool (PC side) and each component (TINI board) are visualized in Figure 12 and Figure 13.

The described runtime diagnosis system comprises the following improvements to conventional sensor systems:

- Diagnosis of all system components by only one tool, without using additional (extern) diagnosis programs.
- Automatic detection of new components.
- Each detected component automatically provides a diagnosis service, the interface to load this service is identical for all components.
- Use of standard network protocols and hardware.

4.5. Summary and Outlook

Care-O-bot II, the second prototype of an intelligent mobile home care system developed at Fraunhofer IPA has been presented.

A software architecture has been proposed to give proper feedback of planned actions to the user and therefore avoid critical misunderstandings.

A person using the robot as an intelligent walking aid must be able to rely upon the assistant and must never be exposed to any safety hazards. Therefore, specific motion restrictions are laid upon the robot and the operation speed is adapted to the input of the user. The robot will never execute any movements that its user is unable to follow.

The three level safety system on the robot consists of a laser scanner, infrared sensors, a bumper and a magnetic sensor. This system prevents accidents caused by a person being hit by the robot.

A runtime diagnosis system which uses only one tool to check all system components has been presented. New components are detected automatically. Each detected component provides a diagnosis service, the interface to load the corresponding service is identical for all components. Standards network protocols and hardware are used.

The successful installation of a group of Fraunhofer IPA's mobile robots in a museum proves, that the robots are suited for safe autonomous operation in public environments.

For future developments new sensors will be integrated in the runtime diagnosis system. The system has already been tested for an infrared distance scanner and displacement sensors. The new sensors also aim to improve the three level safety system so that the mobile platforms can navigate safely in all kinds of hazardous environments.

For use of the robot in unstructured and outdoor environments algorithms for simultaneous localization and navigation are being developed.

Algorithms for movement prediction will further be added to the existing navigation and obstacle avoidance system, so that fast and secure motion can also be possible in crowded environments.

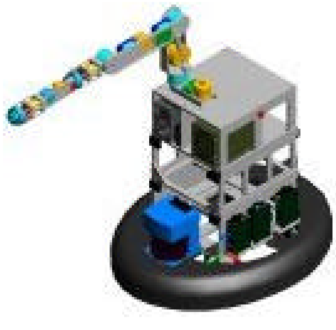

Figure 14. CAD model of mobile manipulator

A manipulator arm will further be added to the platform (Figure 14) which will pose new safety questions. A camera system will be integrated on the robot in order to track the working area of the robot arm and detect illegal interferences.

5. Acknowledgements

Part of this work has been supported by the MORPHA-project [13] funded by the German Ministry of Education and Research (bmb+f) under grant 01IL902G/9.

6. References

[1] ANSI/RIA R15.06 - 1999 Robot Safety Standard; Robotic Industries Association (RIA), Ann Arbor, USA, 1999.

[2] Corke, Peter I: "Safety of advanced robots in human environments. A discussion paper for IARP", 1999.

[3] Dallas Semiconductor: "TINI Developers Home Page", http://tini.dalsemi.com.

[4] Dhillon, B. S.; Fashandi, A. R. M.: "Safety and Reliability Assessment Techniques in Robotics". In Robotica , Vol. 15, 1997, pp. 701-708.

[5] Graf, B.: „Reactive Navigation of an Intelligent Robotic Walking Aid“. Submitted to IROS-2001.

[6] Graf, B.; Schraft, R.D.; Neugebauer, J.: "A Mobile Robot Platform for Assistance and Entertainment." In Proceedings of ISR-2000, Montreal, pp. 252-253.

[7] Hägele M., Schraft, R.D.; Neugebauer, J.: "From Robots to Robot Assistants". In Proceedings of ISR-2001, pp. 404-409.

[8] Huber, M. J.: "JAM Agents in a Nutshell", Version 0.61+0.79i.; July 4th, 1999.

[9] Jaouni, H.; Khatib, Maher; Laumond, J.P.: "Elastic Bands For Nonholonomic Car-Like Robots: Algorithms and Combinatorial Issues." In: 3rd International Workshop on the Algorithmic Foundations of Robotics (WAFR'98), Houston, 1998.

[10] Jiang, Kaichun; Seneviratne, Lakmal D.; Earles, PP. W. E.: "A shortest Path Based Path Planning Algorithm for Nonholonomic Mobile Robots." In: Journal of Intelligent and Robotic Systems, 24th Ed (1999), pp. 347-366.

[11] Kempf, James; St. Pierre, Pete: "Service Location Protocol for Enterprise Networks. Implementing and Deploying a Dynamic Service Finder". Wiley, 1999.

[12] Lumelsky, V. J.; Skewis, T.: Incorporating range sensing in the robot navigation function. IEEE Trans. Sys. Man Cybernet. 20(5), 1990, pp. 1058-1068.

[13] "MORPHA – intelligente antropomorphe Assistenzsysteme", http://www.morpha.de, 2000.

[14] Schaeffer, C.: "ECHOS – Elderly Care Home Systems. Technische Hilfen für unterstützungs- und pflegebedürftige Personen im häuslichen Bereich." Study at Fraunhofer IPA, Stuttgart, 1997.

[15] Schaeffer, C.; May, T: "Care-O-bot: A System for Assisting Elderly or Disabled Persons in Home Environments". In Proceedings of AAATE-99, Düsseldorf, 1999, pp. 340-345.

[16] Schraft, R.D.; Graf, B.; Traub, A.; John, D.: „A Mobile Robot Platform for Assistance and Entertainment". In Industrial Robot Journal, Vol. 28, 2001, pp. 29-34.

[17] Schraft, R.D; Graf, B.; Traub, A.; John, D.: „PolarBug – ein effizienter Algorithmus zur reaktiven Hindernisumfahrung“. In Proceedings of AMS 2000.

[18] SICK-Detect the difference, http://www.sick.de.

[19] XML-RPC Home Page, http://www.xmlrpc.com.