Journal of Intelligent and Robotic Systems **22:** 191–209, 1998.

Design Issues of a Semi-Autonomous Robotic Assistant for the Health Care Environment

E. ETTELT, R. FURTWÄNGLER, U. D. HANEBECK and G. SCHMIDT
Institute of Automatic Control Engineering (LSR), Technische Universität München, 80290 München, Germany; e-mail: roman@lsr.e-technik.tu-muenchen.de

(Received: 15 May 1997; in final form: 16 March 1998)

Abstract. This paper discusses design issues of a mobile robotic assistant for health or home care applications with the objective of relieving hospital personnel or domestic users of time-consuming routine tasks. These tasks are delegated to the robot via natural language and in turn autonomously executed. With respect to the execution of typical fetch-and-carry tasks, key components are surveyed and a system architecture for integration of the individual hardware and software modules into a service robot is presented. A prototype implementation is described and used for demonstrating the performance of the proposed design approach in real-world service scenarios.

Key words: robotic assistant, mobile service robot, health care, fetch-and-carry task, mobile manipulation, human-robot interface, augmented virtual workspace.

1. Introduction

In recent years mobile service robots have been introduced into various non-industrial application areas such as entertainment, building surveillance, and hospitals. They are relieving humans of tedious work with the prospect of 24-hour availability, fast task execution, and cost effectiveness. A well-known example of a service robot which is successfully employed in more than 70 hospitals is the HelpMate [13]. This mobile robot is a self-navigating transport vehicle without any manipulation capabilities. When distributing meals, for example, it is manually loaded, travels towards the ward room, and finally awaits to be unloaded in front of the door.

This paper is concerned with major design issues of a more advanced robotic assistant for application in health or home care scenarios, providing manipulation capabilities and an adequate interface for high-level human-robot communication. Its objective is to assist nurses or other users in simple routine tasks such as fetch-and-carry of laboratory specimens, meals, medicine or documents. These tasks are time-consuming and keep experienced hospital staff from pursuing their primary job function: caring for patients. Of course, to successfully execute even simple fetch-and-carry tasks, a service robot needs to be capable of performing a variety of subtasks resulting from the primary task, e.g., recognizing and grasping objects,

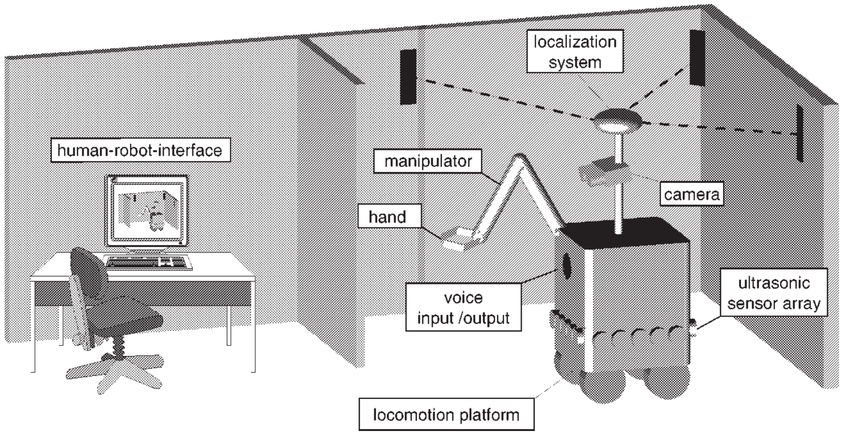


Figure 1. Robotic assistant with on-board components and human-robot interface.

opening or closing drawers and doors, traveling along corridors, using elevators, and delivering objects to the desired goal location. The proposed robotic assistant comprises the following components (Figure 1):

Human-Robot Interface (*HRI*). An adequate interface for natural speech based, dialogue-oriented human-robot communication allows non-expert robot users such as nurses or patients to easily specify the tasks to be performed. Furthermore, the robot is embedded within the existing information infrastructure of the hospital and can thus be remotely accessed.

Intelligent Locomotion Platform. A wheel-based omnidirectional platform has been developed to achieve high maneuverability. The platform is equipped with a reliable multi sensor system, which is important for self-localization and obstacle detection. This component ensures safe operation because the robot, patients, and other humans are sharing the same space.

Anthropomorphic Manipulation. For performing manipulation tasks, the robot is equipped with a human-like arm. Appropriate coordination of arm and platform motion yields smooth transitions between motion subtasks and sufficiently rapid task execution. The resulting human-like motion behavior is a prerequisite for gaining acceptance by hospital staff and patients.

Vision-Based Object Recognition. To handle the variety of objects encountered in health care applications, two object recognition methods have been developed. A feature-based approach is used for recognizing *extended objects* for which explicit 3D-models are available. For reliable recognition of *small objects* with cluttered background, an appearance-based approach is proposed.

Task Planner and Coordinator. The robot employs a task planner to autonomously perform tasks of typical complexity. This includes finding its way to the goal location, opening of doors blocking the way or handling the specified objects. During task execution, the task planner coordinates the individual robot components in such a way, that the resulting robot operations are performed correctly, smoothly, and safely.

The components of the proposed robotic assistant are discussed in detail in Section 2. Experimental results obtained with a prototype implementation are presented in Section 3. Section 4 concludes with an outlook onto future activities.

2. Components of the Robotic Assistant

2.1. USER-FRIENDLY HUMAN-ROBOT INTERFACE

To make easy use of the robotic assistant's capabilities, a human-robot interface is provided, which allows local users a high-level dialogue-oriented communication with the robot based on natural language. Alternatively, a remote interface serves to command, supervise, and diagnose a service robot through acoustic and visual channels. This kind of more advanced user interface is especially useful to coordinate several robots operating in the same building.

2.1.1. *Local Robot Commanding*

For robot commanding in the immediate vicinity of the robot, mainly acoustic communication is used (Figure 2). This is achieved by interfaces for

- *dialogue-oriented naturally spoken command input* for fast and comprehensive commanding on a high level of abstraction, and
- *voice output* by means of a speech synthesizer for bidirectional communication.

Hence, a nurse as a typical user can work in her customary fashion. Commands similar to those she would give to a colleague such as "Please carry that tray into the ward room!" are understood by the robot.

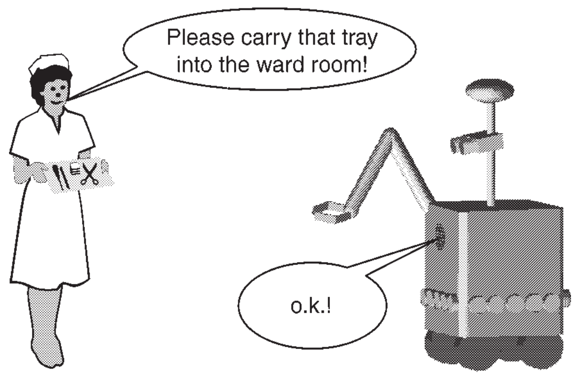


Figure 2. Local task specification by natural spoken language.

If the command is incomplete, ambiguous, inconsistent, or simply not understood, the robot checks back by means of speech output. Furthermore, he reports his current state, for example, successful task completion.

2.1.2. *Remote Robot Commanding and Supervision*

A service robot must also be capable of receiving commands from a remote operator. In addition, supervision and support of the robot during task execution must be possible (Figure 3). This becomes especially important, when several robots are used within the same building. In that case, problems such as coordination and resource management are becoming major issues. Resulting implementation requirements are

- a *bidirectional acoustic interface* using standard information channels such as telephone lines or Intranet/Internet facilities,
- a *video interface* for obtaining detailed and current information about the environment, and
- a *virtual workspace* display for supervision and support of the robot.

Besides the acoustic communication, which already proved to be useful for local robot commanding, the video interface is a major issue for remote operation. Visual monitoring is performed by means of online images captured by cameras either located in the environment or mounted on board the robot.

Additional synthetic visual information about the robot and its environment is provided by a virtual workspace, which displays a three-dimensional view of the environment including fixed furniture, the animated robot, and objects detected on the basis of current sensor data. Compared to video images, a virtual workspace requires less bandwidth for storing and transmitting the actual scene and the viewpoint can be arbitrarily chosen. Furthermore, the virtual workspace is used for robot support. For example, if a command turns out to be ambiguous with respect to the

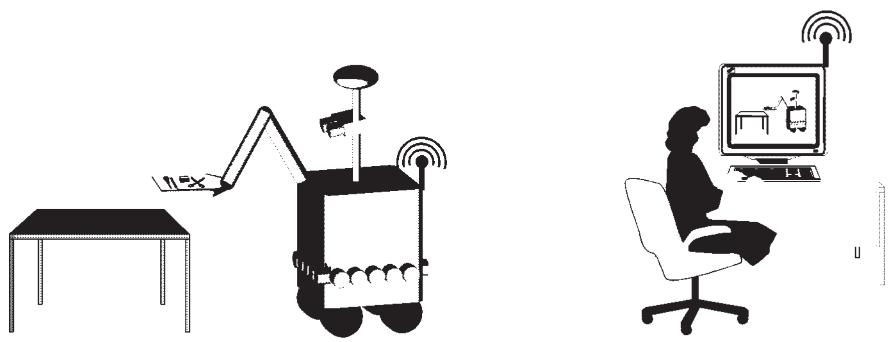


Figure 3. Commanding and supervision via remote HRI.

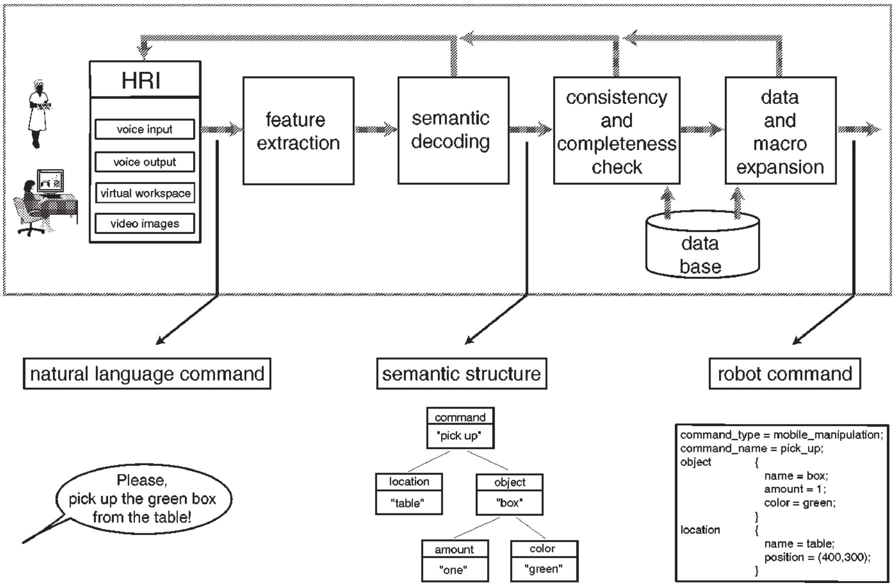


Figure 4. Conversion of natural language into a robot command.

object to be manipulated, the operator specifies the desired object in the virtual workspace.

2.1.3. *Robot Command Generation*

A natural language command has to be converted into an appropriate robot command before being executed (Figure 4). This is done in several steps: 1) acoustic feature extraction, 2) semantic decoding, 3) consistency and completeness check, 4) data and macro expansion. The latter two steps are performed with help of an internal robot data base. Symbolic information within the semantic structure, such as “table”, is replaced by its geometrical and numerical counterpart.

If robot command generation fails at any step, a dialogue is initiated and the user is asked to repeat, cancel or complement the command via the different multimodal HRI resources.

2.2. INTELLIGENT LOCOMOTION PLATFORM

A self-contained subunit for locomotion is presented, which is based on a highly maneuverable omnidirectional platform. For navigation, the platform is equipped with a localization system, which provides posture estimates, i.e., platform position and orientation data, in real-time. Furthermore, an advanced ultrasonic sensor array for obstacle detection is closely coupled with the platform motion controller to prevent collisions with unexpected obstacles.

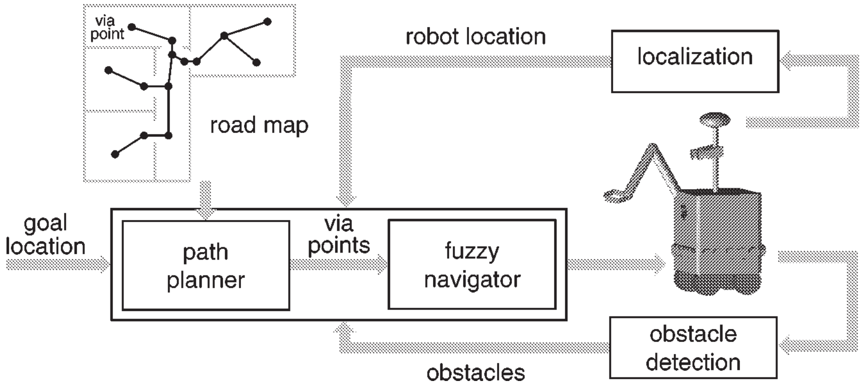


Figure 5. Performing long distance motion on the basis of a predefined road map.

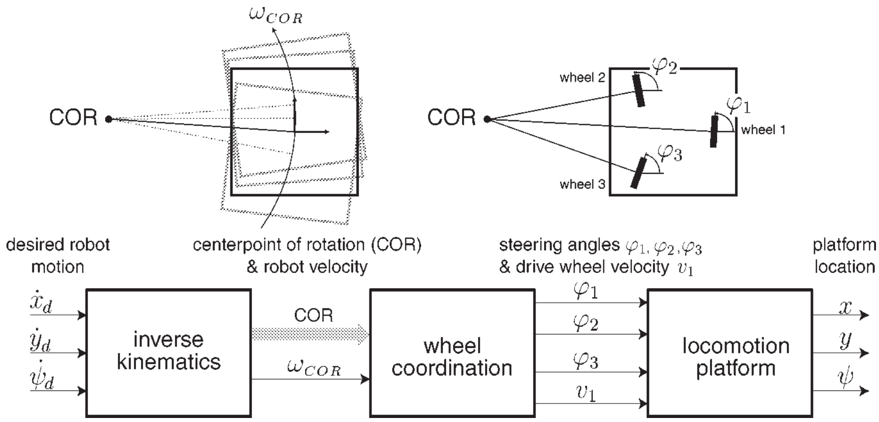


Figure 6. Inverse kinematic module and wheel coordination mechanism.

2.2.1. *Omnidirectional Locomotion Platform*

During service task execution, the locomotion platform performs two distinctive types of motion: *long distance motion* and *local motion*. *Long distance motion* means, for example, traveling from one room to another. This type of motion is based on a predefined road map, i.e., a topology graph, which connects typical robot workspaces in the environment (Figure 5). The shortest feasible path between two robot workspaces is determined as a sequence of via-points by Dijkstra's algorithm [14]. Appropriate maneuvers between the via-points are generated by a fuzzy navigator. In contrast, *local motion* is used to extend the manipulator workspace during mobile manipulation, see Section 2.3.

To achieve the desired high maneuverability, an omnidirectional locomotion platform is employed. It is equipped with three actively steered wheels; only one wheel is driven. A control interface is provided, that makes use of an inverse kinematic platform model (Figure 6). Desired translational velocities $\dot{x}_d$, $\dot{y}_d$ plus the desired rotational velocity $\dot{\psi}_d$ are converted into a representation suitable for the

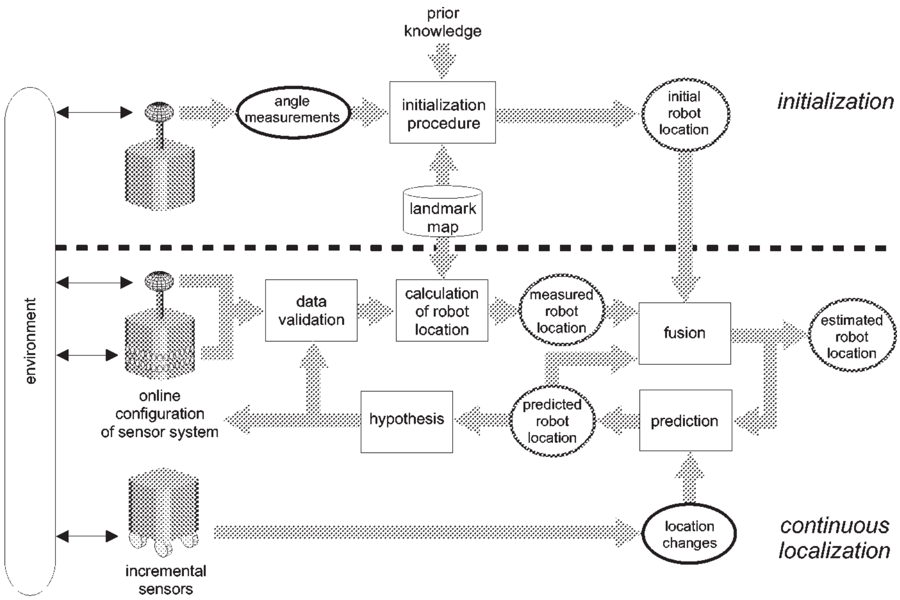


Figure 7. Scheme of the proposed robot localization system.

control of the considered omnidirectional vehicle type: robot velocity and instantaneous center point of rotation (COR). Subsequently, the corresponding steering angles ϕ_1, ϕ_2, ϕ_3 and the drive wheel velocity v_1 are calculated on the basis of the physical vehicle layout.

2.2.2. *Platform Localization and Obstacle Detection*

Localization of the locomotion platform could be performed by dead-reckoning, i.e., by propagating a given initial location with data from incremental sensors like an odometer or a gyroscope. However, due to systematic and correlated uncertainties of sensor data and the required velocity integration, dead-reckoning suffers heavily from error accumulation. Its accuracy is only sufficient for small path lengths.

To keep localization accuracy within specified bounds though, additional geometry sensors are required for repeatedly updating the robot location with respect to known environmental features. Two types of inexpensive and simple on-board sensors are considered: an angle measurement system and a sonar sensor array. The limited perception capabilities of these sensors are compensated by the introduction of intelligent sensing strategies.

A scheme of the proposed localization system is shown in Figure 7. Its operation comprises two phases: 1) Initialization, i.e., determination of the initial platform location with a minimum amount of a priori knowledge. 2) Continuous localization, i.e., cyclic updating of the robot location during motion.

For phase 1, a simple and efficient initialization procedure has been developed [10], which is based on an interpretation tree approach [7] and a new linear solution scheme for converting angle measurements to known landmarks into robot locations. The procedure makes use of a priori knowledge and copes with indistinguishable landmarks, faulty measurements, and landmark occlusions.

To ensure accurate and stable in-motion robot localization, despite the limited perception capabilities of the sensors considered, both spatial and temporal continuity conditions are exploited. For that purpose, the incremental sensor data are used to predict the robot location and to generate landmark hypotheses. Based on the hypotheses, the sensor system is configured online, i.e., both sensing and processing power are focussed on currently relevant landmarks at a very early stage. In addition, measurement hypotheses are derived for validating actual measurements. As a result, environmental features can be reliably sensed with high sampling rates [8]. This is important for high-speed, high-accuracy maneuvers.

To increase localization accuracy, efficient closed-form solutions have been derived for calculating the robot location based on both angle and distance measurements. These solutions consider as an integral part measurement errors and uncertainties of landmark positions [11, 12]. The robot location calculated on the basis of these measurements is fused with the predicted location by use of a set-theoretic approach. The fusion result serves as the starting point for the next prediction step.

Obstacle detection is performed by use of a sonar sensor array. For that purpose, an algorithm for environmental sensing has been developed, which provides the Cartesian positions of surrounding obstacles at a high sampling rate. A detailed description is given in [9].

One of the unique features of the localization and obstacle detection algorithms is the rigorous modeling of uncertainties by just specifying error bounds. This error model is appropriate when major systematic uncertainties are present in landmark positions, sensor locations, and measurement noise. Rather than propagating point estimates, the algorithms propagate all feasible states that are compatible with the a priori error bounds. The inherently high complexity of this approach is reduced by approximating the sets of feasible states with ellipsoidal sets. A new formalism for ellipsoid calculus yields simple and computationally efficient algorithms for prediction and fusion.

2.3. ANTHROPOMORPHIC MANIPULATION

An appropriate manipulator for handling of small, light-weight, and geometrically simple objects is a major prerequisite for performing fetch-and-carry tasks in the considered application area. Example objects are laboratory specimens, dishes, bottles or journals. Since these objects are usually handled by hospital staff, the manipulator's size should be comparable to a human arm.

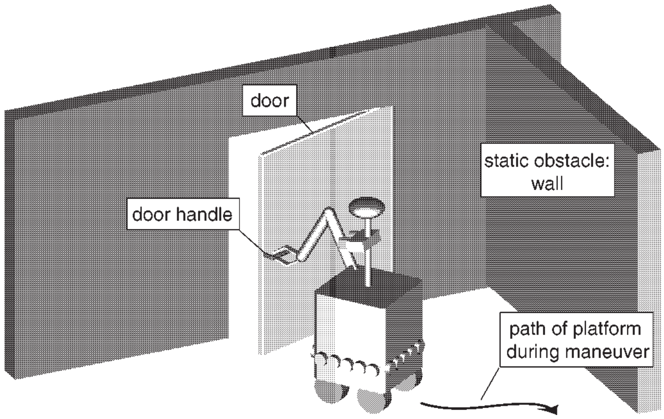


Figure 8. Door opening maneuver.

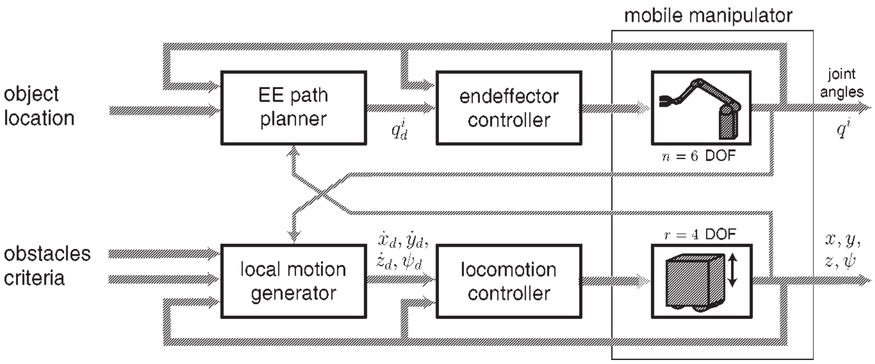


Figure 9. Control strategy for motion coordination of a 6 DOF manipulator mounted on a vertical linear axis and a 3 DOF omnidirectional platform.

Grasping general objects requires at least 6 degrees of freedom ($n \geqslant 6$ DOF) for attaining any end-effector position and orientation in the local workspace. However, for safe grasping in a moderately structured and often cluttered hospital environment, fixed or moving obstacles have to be considered, which calls for additional DOF to increase manipulability.

2.3.1. *Platform-Supported Mobile Manipulation*

Additional DOF are also required to extend the manipulator workspace, which is often not sufficient for performing complex subtasks like for example door opening (Figure 8). Hence, a 6 DOF arm is used and supported by appropriate motion of the 3 DOF locomotion platform and a 1 DOF vertical linear axis. This leads to a mobile manipulator with a total of 10 DOF. The corresponding coordination strategy based on the natural separation of two motion subsystems is shown in Figure 9. In addition to improved manipulability and a larger workspace, platform support also results in more rapid task execution. Nevertheless, it requires controlling all DOF simultaneously.

In order to simplify this task, mobile manipulation is decomposed into a *preparation phase* to unfold the arm, a *path following phase* where the arm and the end-effector approach the object and perform the required manipulation, and the *completion phase* where the arm is folded into a transport configuration. Only during the path following phase, locomotion and manipulation are tightly coupled.

2.3.2. *End-effector Control During Mobile Manipulation*

For manipulation purposes, the approximate object location is derived from the user command with the help of an internal data base. However, the exact Cartesian object position is determined online by the object recognition module described in Section 2.4. Based on the object position, an appropriate end-effector path is generated. The end-effector is then controlled by use of a 6 DOF differential arm model via well-known inverse Jacobian techniques [15].

2.3.3. *Locomotion Control During Mobile Manipulation*

The redundant 4 DOF, Figure 9, of the mobile manipulator are bound by a corresponding number of configuration criteria. For that purpose, the set of orthogonal criteria shown in Figure 10 has been selected: elbow angle, shoulder angle, wrist angle, and shoulder height. They are prescribed in such a way, that both ill-posed arm configurations are avoided and environmental constraints are considered. During task execution, a fuzzy penalty mechanism initiates appropriate platform motions in order to satisfy the criteria.

The proposed coordination scheme leads to a slaving behaviour of the platform: While the end-effector follows a given path to reach its goal, the platform smoothly supports the manipulator during operation and takes into account the active constraints. As a result, the total system shows a human-like motion behaviour and keeps a safe distance to surrounding obstacles. Furthermore, compared to a strategy employing a single kinematic model for control of the total mobile manipulator system, the complexity and the computational power needed are reduced.

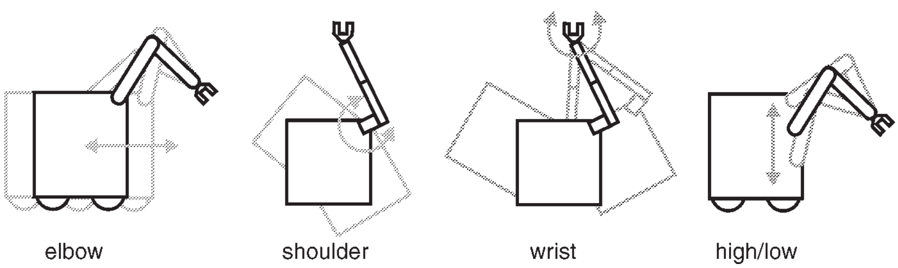


Figure 10. Visualisation of arm configuration criteria.

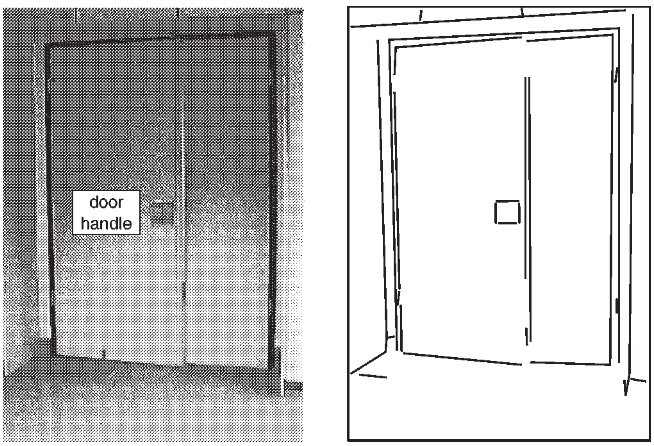
2.4. VISION-BASED OBJECT RECOGNITION

During the execution of service tasks, a collection of a priori known objects needs to be manipulated. These objects are recognized and precisely localized by means of vision techniques. On the one hand, extended objects like doors, cupboards, and drawer boxes are considered, which are defined by simple features like straight lines or planes. On the other hand, fetch-and-carry tasks often require grasping of small objects with cluttered background. Because of these diverse requirements, two approaches for object recognition are used.

2.4.1. *Feature-Based Approach*

Extended objects like doors, cupboards, and drawer boxes can be explicitly modeled by a few features, for example lines. However, approaches for recognizing arbitrary objects based on explicit models prove to be too slow for real-time applications. Hence, a fast three-step feature-based recognition procedure has been developed [1], which makes use of a priori information usually available in the considered application domain.

In a first step, standard techniques are used to extract straight lines from the image. For example, the lines extracted from the image of a typical door are depicted in Figure 11. These line features are then transformed in order to remove perspective distortions caused by the (known) camera tilt angle. Finally, the transformed features are matched to a model representation. In case of a door, the model comprises geometric informations like width, height, and location of the handle, but also information about the door's absolute location and its kinematic. Exploiting this wealth of a priori information leads to a rather robust and precise location estimate.



(a) (b)

Figure 11. (a) Video image of a typical door. (b) Corresponding extracted line features.

Figure 12. Typical object templates for a medicine pack, a bottle, and a glass.

2.4.2. *Appearance-Based Approach*

An appearance-based approach [2] is used for recognizing small objects with cluttered background like medicine packs, bottles or glasses (Figure 12). For each object, a collection of *typical* views is used for generating a set of object templates offline. In addition, a set of background templates is extracted. The advantage of this approach is the implicit modeling of objects on the basis of brightness patterns. No explicit features are needed. However, due to varying illumination conditions, different view points, and different backgrounds, a huge number of templates is required for reliable recognition. Hence, comparing the current image with all the templates is not feasible.

Therefore, our approach is to store the huge number of templates in a binary tree structure, called template tree. Online, a search window is then shifted through the image under consideration on a pixel-by-pixel basis. After each step, the search window is sent down the template tree and a binary decision is made at every level. As a result, the image is typically classified after a maximum of 10 to 20 comparisons. Hence, in contrast to considering every template, the number of comparisons required is reduced by several orders of magnitude.

2.4.3. *Continuous Object Recognition*

So far, object recognition based on a single image was considered, which may not always be successful in real world applications. Hence, when recognizing objects for mobile manipulation purposes, a sequence of several images from different viewpoints is used to enhance robustness. These images are collected while approaching the anticipated object location and used to recursively update the estimated object location. To ensure precise grasping, the end-effector path is continuously adjusted on the basis of the current location estimate.

2.5. TASK PLANNER AND COORDINATOR

The robot components described in Sections 2.1–2.4 have been implemented as encapsulated expert modules, which exchange high-level data via dedicated interfaces. These expert modules are administered by a task planner as shown in Figure 13.

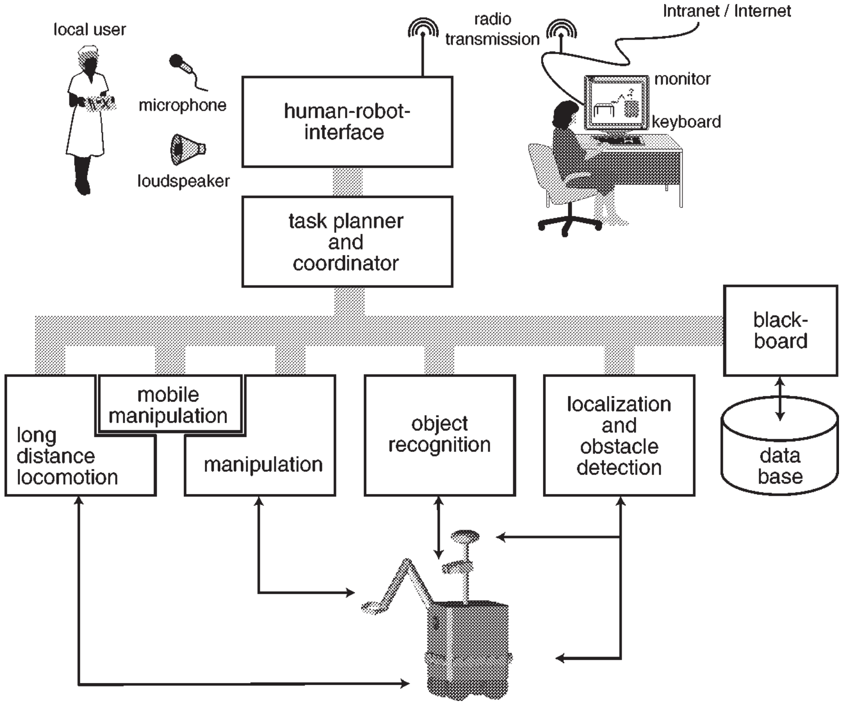


Figure 13. Information processing architecture.

Upon receiving a natural language command, the human-robot interface converts the acoustic signal into an executable robot command, as described in Section 2.1. Subsequently, the robot command is split up by the task planner into a sequence of subtasks such as door opening, door passing, or traveling along corridors. These subtasks, which include stereotypic operations, are then executed by either the expert for long distance locomotion or the mobile manipulation expert. Both experts are continuously supported by current sensor information.

The task planner also coordinates the experts during task execution. This includes connecting the experts depending on the current action. During door opening maneuvers, for example, the object recognition expert is connected with the expert for mobile manipulation, to determine the position of the door handle. In addition, a blackboard is used for posting data of general interest, such as the current robot posture, which is supplied by the localization expert.

3. Experimental Validation

The key components of a health care robot described so far have been integrated into the robotic assistant ROMAN. With this prototype robot, a variety of experiments has been conducted, which clearly underline the feasibility of supporting

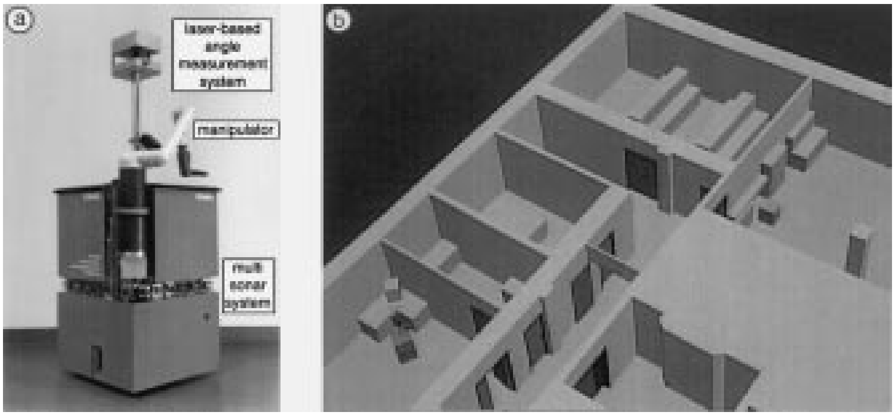


Figure 14. (a) Mobile service robot ROMAN and (b) corresponding virtual workspace.

human personnel by a service robot. After introducing some technical details of the robotic assistant in the next section, a sample experiment will be discussed in some detail.

3.1. ROBOT IMPLEMENTATION DETAILS

ROMAN's size is adapted to indoor environments: 0.63 m width $\times$ 0.64 m depth $\times$ $\times$ 1.85 m height and a weight of about 250 kg including batteries. ROMAN is based on an omnidirectional locomotion platform with three independently steerable wheels with a diameter of 0.2 m, one of which is driven. This results in smooth motion, even when traveling across rough surfaces. The maximum travel speed is 2 m/sec.

For commanding and for monitoring task execution, the multi-modal human-robot interface described in Section 2.1 has been implemented. A sophisticated speech recognition system is used, which has been developed at the Institute of Human-Machine-Communication of the Technische Universität München [16]. Currently, this system is limited to natural spoken sentences without subordinate clauses.

For manipulation purposes, ROMAN is equipped with a commercial anthropomorphic 6 DOF arm with a maximum range of 0.8 m and a maximum payload of 1.5 kg. To extend the manipulator's workspace in the vertical direction, the arm is mounted at one corner of the platform on a high-low linear axis. This allows manipulation of objects located on the floor as well as objects positioned on a shelve.

For self-localization, ROMAN uses an on-board laser-based angle measurement system. An eye-safe laser beam scans the environment in a horizontal plane and determines the azimuth angles to known artificial landmarks, i.e., retro-reflecting tape strips attached to the walls. In addition, ROMAN is equipped with a multi-sonar system for self-localization and obstacle detection.

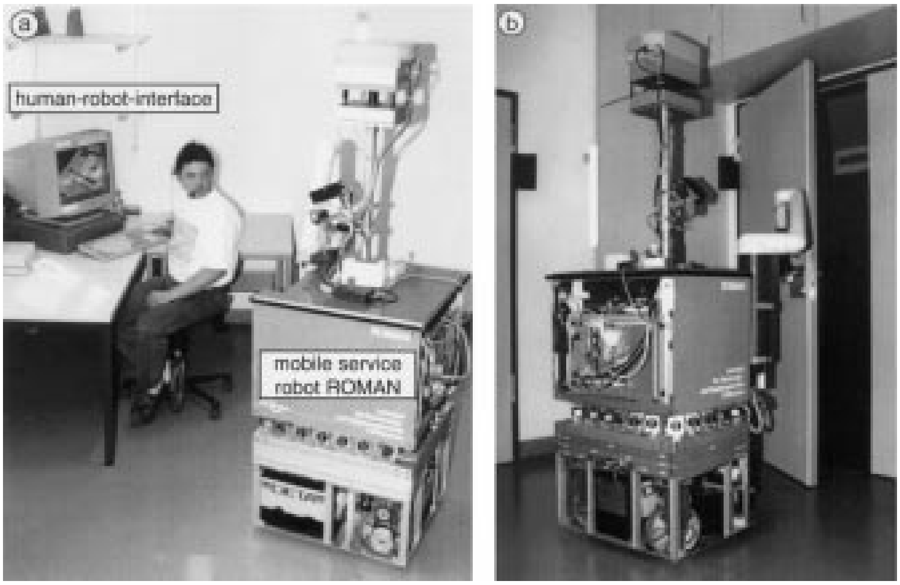


Figure 15. (a) Mobile robotic assistant ROMAN and the human-robot interface, (b) ROMAN opening a door.

Object recognition is based upon a standard video camera. The camera is mounted on a tilt-unit to permit the recognition of objects within the entire manipulator workspace.

The information processing architecture introduced in Section 2.5 is implemented on a VMEbus-based multi-processor system. It communicates with the outside world via a wireless Ethernet link (10 Mbit/sec).

3.2. ROMAN EXECUTING A TYPICAL SERVICE TASK

The following experiment serves to demonstrate the capability and usefulness of the proposed service robot architecture.

If a user wishes support from ROMAN, the task to be executed is issued via natural language, e.g., "Clean the table", see Figure 15(a). After processing the acoustic signal, ROMAN checks his internal data base in order to derive appropriate actions. If ambiguities arise, ROMAN requests completion of the command via speech output. In this example, the data base includes more than one table and ROMAN therefore checks back, which table is supposed to be cleaned. After specification of the desired table by either natural language or by a mouse click in the virtual workspace, the command is ready to be executed.

Subsequently, the command is automatically decomposed into a sequence of subtasks. In a first step, ROMAN plans an appropriate path to the table and starts moving. When approaching a door on his way, ROMAN checks whether it is open or closed by means of the on-board video processing unit. If the door is closed,

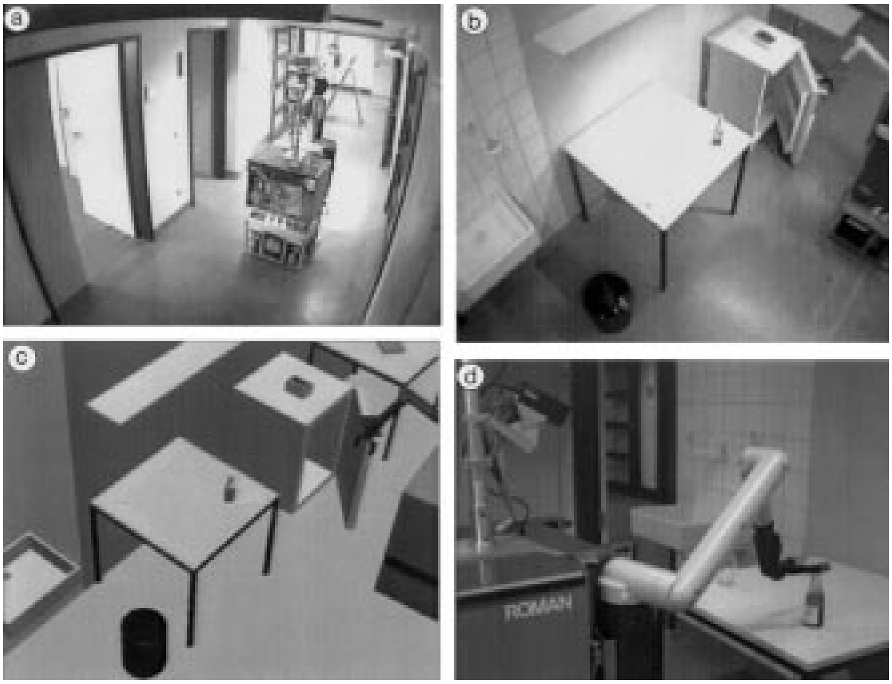


Figure 16. (a) ROMAN cruising down a corridor; (b) opening a refrigerator supervised via camera; (c) and virtual workspace; (d) ROMAN grasping a bottle.

ROMAN launches a stereotypic door opening maneuver (Figure 15(b)). This is, in fact, one of the most complex actions, as it requires continuous support of the manipulator by appropriate platform motion to adapt and to extend the manipulator workspace. Of course, the resulting motion takes into consideration environmental restrictions like walls, furniture, and persons.

After passing the door, ROMAN enters a corridor where high velocities of up to 1 m/sec are reached (Figure 16(a)). During fast motion, it is especially important to check the environment for obstacles at an adequate sampling rate. This is performed by the multi sonar system (see Section 2.2.2). When obstacles are encountered, the robot slows down and performs an evasive maneuver, if possible.

Upon reaching the specified table, ROMAN scans for specified objects. With every object an action is associated, which is performed when the object is recognized. In this experiment, one of the objects – a bottle – is put into the refrigerator. To achieve this task, the door of the refrigerator must be opened (Figure 16(b)) before fetching the bottle (Figure 16(d)), because ROMAN is equipped with only one arm.

During task execution, ROMAN's location is continuously estimated by the localization expert module. The corresponding location estimate is not only used for navigation purposes, but also for visualising ROMAN within the virtual workspace (Figure 16(c)).

When the table has been cleaned up, ROMAN reports successful task completion and is ready to execute the next command.

4. Conclusions and Future Activities

A prototype robotic assistant for health care applications has been introduced. To ensure the robot is capable of performing relevant tasks in a hospital environment, a set of advanced components has been developed and successfully tested:

- A custom designed human-robot interface allows untrained users a high-level dialogue-oriented communication with the robot based on natural language. For remote commanding, diagnosing, and supervising the robot, a cockpit-like user interface has been developed, including visual and acoustic feedback.
- For locomotion, a highly maneuverable omnidirectional wheeled platform has been developed, which includes a reliable self-localization and obstacle avoidance system. Furthermore, a control interface is provided for both local motion and global long distance maneuvers.
- A light-weight, human-like arm with 6 DOF has been integrated into the locomotion platform. For extending the manipulator workspace by support of the platform, a coordination strategy has been developed, that leads to a smooth, non-robot like motion.
- A fast and reliable vision-based object recognition system is used for precise end-effector guidance.

These components have been integrated into the full-size robot ROMAN and were evaluated in a number of typical service scenarios in realistic environments.

The work presented in this paper focussed on the semi-autonomous execution of fetch-and-carry tasks, i.e., a human delegates routine tasks to the robot, which are in turn performed autonomously. Hence, the user simply specifies *what* to do, and need not bother with the details of the actual task execution. The semi-autonomous approach is closer to the current goals of potential end-users than a fully autonomous system. Furthermore, appropriate artificial intelligence techniques for achieving fully autonomous behaviour are not yet available.

For commercial health care robots, cost will not be a major issue, as dedicated cost-effective solutions can be derived for specific application domains based on the developed universal key components.

Nevertheless, there are many problems which remain unsolved. Before the new generation of robotic assistants can be launched commercially, they need to become more robust, more reliable, and most importantly *fail-safe*. More user-friendly operation is a further prerequisite for successful commercial applications. This includes self-diagnosis in the event of operation failures and the automatic derivation of appropriate counter-actions.

Besides automatic diagnosis and correction of operation failures, remote supervision proved to be essential for dealing with unexpected problems. For that purpose, the human-robot interface described in Section 2.1 needs to be embedded into the existing information infrastructure of a modern hospital, i.e., the hospital Intranet transfers data between human and robot. In addition, future hospital Intranets will also provide valuable information on the current location of staff members and other robots. Furthermore, the robot may gain access to the hospital database containing the corresponding ward room numbers of patients, the room locations, and the meals ordered. Thanks to the connection between Intra- and Internet, world-wide access of the robot will be possible, which will allow robot manufacturers to remotely perform robot maintenance.

The next phase of our research includes the utilization of dextrous hands to extend the class of graspable objects. Manipulation skills will also be increased by using a faster light-weight manipulator with a larger workspace. The robot's capabilities will also be further enhanced by exploiting advanced recent technologies. For example, CMOS cameras will reduce the dependency of object recognition on varying object illumination. A larger traveling range will be possible with advanced batteries together with low-power electronic components.

Automatic behavior adaptation will be investigated in order to equip robotic assistants with a higher level of autonomy. Instead of using knowledge delivered to the robot via engineering labour such as manual programming, the robot could collect relevant knowledge during task execution. Hence, situations different from those anticipated could be mastered by the robot without human intervention.

Acknowledgement

The work reported in this paper was supported by the Deutsche Forschungsgemeinschaft as part of an interdisciplinary research project on "Information Processing Techniques in Autonomous Mobile Robots" (SFB 331). The authors would like to thank their colleagues W. Daxwanger and C. Fischer for their invaluable contribution to the research reported in this paper. Furthermore, the authors are indebted to M. Lang, J. Müller, and H. Stahl for providing their sophisticated speech recognition system.

References

1. Ettelt, E. and Schmidt, G.: Vision based guidance and control of a mobile Forklift robot, in: *Proc. of the Int. Conf. on Recent Advances in Mechatronics*, Istanbul, Turkey, 1995, pp. 180–186.
2. Ettelt, E. and Schmidt, G.: Videobasierte Objekterkennung Mittels Musterbaumgestützter Kreuzkorrelation, in: *13. Fachgespräch Autonome Mobile Systeme (AMS)*, Stuttgart, Germany, 1997, pp. 72–83.
3. Fischer, C., Buss, M., and Schmidt, G.: Human-robot interface for intelligent service robot assistance, in: *Proc. of the IEEE Int. Workshop on Robot and Human Communication (ROMAN)*, Tsukuba, Japan, 1996, pp. 177–182.

4. Fischer, C., Buss, M., and Schmidt, G.: Soft control of an effector path for a mobile manipulator, in: *Proc. of the Int. Symp. on Robotics and Manufacturing (ISRAM)*, Montpellier, France, 1996, pp. 299–306.
5. Fischer, C., Buss, M., and Schmidt, G.: Hierarchical supervisory control of service robot using human-robot interface, in: *Proc. of the Int. Conf. on Intelligent Robots and Systems (IROS)*, Osaka, Japan, 1996, pp. 1408–1416.
6. Fischer, C. and Schmidt, G.: Multi-modal human-robot interface for interaction with a mobile service robot, in: *Proc. of the 6th Int. Workshop on Robotics in Alpe-Adria-Danube Region (RAAD)*, Cassino, Italy, 1997, pp. 559–564.
7. Grimson, W. E. L. and Lozano-Pérez, T.: Recognition and localization of overlapping parts from sparse data in one and two dimensions, in: *Proc. of the 1985 IEEE Int. Conf. on Robotics and Automation*, St. Louis, MO, 1985, pp. 61–66.
8. Hanebeck, U. D. and Schmidt, G.: A new high performance multisonar system for fast mobile robot applications, in: V. Gräfe (ed.), *Intelligent Robots and Systems 1994 (IROS)*, Elsevier Science, Amsterdam, 1995, pp. 1–14.
9. Hanebeck, U. D. and Schmidt, G.: Schnelle Objektdetektion mit Ultraschallsensor-Arrays, in: R. Dillmann, U. Rembold, and T. Lüth (eds), *11. Fachgespräch Autonome Mobile Systeme (AMS)*, Karlsruhe, Germany, Springer, Berlin, 1995, pp. 162–171.
10. Hanebeck, U. D. and Schmidt, G.: Localization of fast mobile robots using an advanced angle-measurement technique, *IFAC Control Engineering Practice* **4**(8) (1996), 1109–1118.
11. Hanebeck, U. D. and Schmidt, G.: Set-theoretic localization of fast mobile robots using an angle measurement technique, in: *Proc. of the 1996 IEEE Int. Conf. on Robotics and Automation (ICRA)*, Vol. 2, Minneapolis, MN, 1996, pp. 1387–1394.
12. Hanebeck, U. D. and Schmidt, G.: Closed-form elliptic location with an arbitrary array topology, in: *Proc. of the 1996 IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Vol. 6, Atlanta, GA, 1996, pp. 3070–3073.
13. HelpMate Robotics Incorporation, `http://www.helpmaterobotics.com`.
14. Noltemeier, H.: *Graphentheorie mit Algorithmen und Anwendungen*, de Gruyter, Berlin/New York, 1976.
15. Paul, R. P.: *Robot Manipulators: Mathematics, Programming and Control*, MIT Press, Cambridge, MA, 1981.
16. Stahl, H., Müller, J., and Lang, M.: An efficient top-down parsing algorithm for understanding speech by using stochastic and semantic models, in: *Proc. of the Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Atlanta, GA, 1996, pp. 397–400.