The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Care-O-bot® 3 - Creating a product vision for service robot applications by integrating design and technology

Ulrich Reiser, Christian Connette, Jan Fischer, Jens Kubacki, Alexander Bubeck,
Florian Weisshardt, Theo Jacobs, Christopher Parlitz, Martin Hägele, Alexander Verl

ABSTRACT

This paper introduces Care-O-bot® 3, a highly integrated and compact service robot with manipulation, navigation and vision capabilities. In particular, Care-O-bot® 3 combines the best of available technology including a 7 DOF light-weight arm, an omnidirectional platform and many high-end sensors along with a sustainable, end user oriented design concept enabling many interaction possibilities.

I. INTRODUCTION

In the last decades a lot of robotic platforms have been developed, most of which include mobility, some sort of autonomous navigation and - more recently - also manipulation capabilities. The list of service robots depicted in Fig. 1 is far from being complete, showing only the ones with complexer functionality. It is hard, however, to find service robot platforms which comprise in addition capable vision with object recognition, gesture recognition and scene modeling, comprehensive reasoning and planning components and elaborate user interaction concepts.

Most of these robots have in common: they are pure development platforms with little emphasis on end user related issues like design or usability. The target is to develop an overall concept suitable for a product vision, combining the above mentioned technological aspects with a compact and user friendly design. The result of these considerations, Care-O-bot® 3, is introduced in this paper.

This article therefore starts with design considerations in section II, followed by some insights into the hardware set-up in section III. Sections IV, V and VI give short overviews of already integrated algorithms for reliable object recognition, autonomous navigation and safe manipulation, respectively. In section VII experimental results for the implemented algorithms are given. The performance of the complete systems is assessed in an object delivery use case. The conclusion is presented in section VIII.

II. ROBOT DESIGN

The goal was to create a unique and iconic design for a service robot depicting an innovative product perception away from a humanoid approach. The design intends to convey a future product vision that is very different from existing humanoid robots, and that will create fascination and acceptance for service robots.

This work was conducted in the department for Robotic Systems at the Fraunhofer Institute for Manufacturing Engineering and Automation (IPA), 70504 Stuttgart, Germany - The work was partially conducted within the EU Project COGNIRON ("The Cognitive Robot Companion" - www.cogniron.org) under Contract FP6-IST-FET-002020 and by the research project DESIRE by the German Federal Ministry of Education and Research (BMBF) grant no. 01IME01.

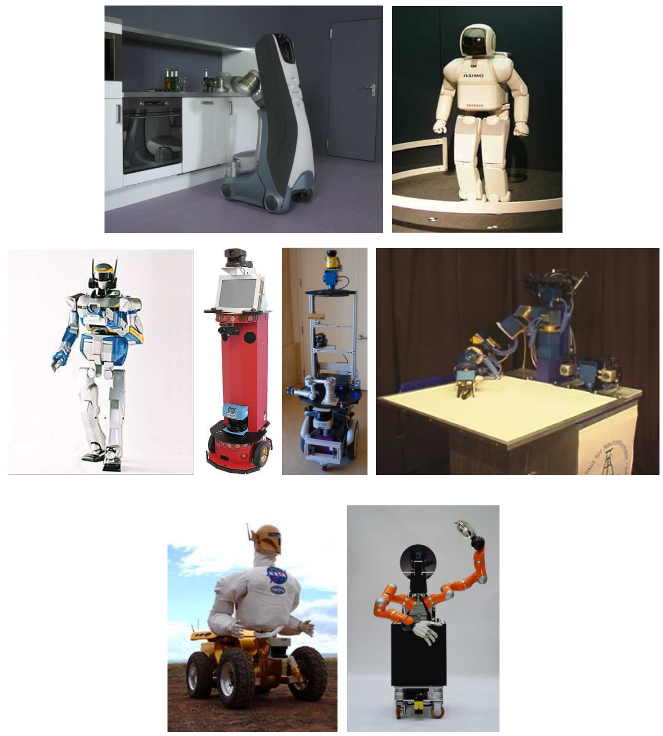


Fig. 1. State-of-the-art service robots: From left to right: Care-O-bot® 3, Asimo, HRP-2, Biron, STAIR, CoRA, Robonaut, DESIRE

To extract necessary functionality, first of all the roles (Butler, Info-Terminal, Attraction, ...) and typical tasks (Lay a table, Serve drinks, Fetch-and-Carry tasks) of the robot were defined. Simultaneously, available state-of-the-art robot technology was evaluated. Constraints concerning size and weight set by a typical house-hold environment had to be considered. Finally, the experiences made with former robot developments [1], [2] delivered valuable input. The fundamental concept developed was to define *two sides* of the robot. One side is called the 'working side' and is located at the back of the robot away from the user. This is where all technical devices like manipulators and sensors which can not be hidden and need direct access to the environment are mounted. The other side is called the 'serving side' and is intended to reduce possible users' fears of mechanical parts by having smooth surfaces and a likable appearance. This is the side where all physical human-robot interaction takes place. One of the first design sketches can be seen in Fig. 2 (left). After several steps of design-technology convergence a simplified rendering can be seen in Fig. 2 (right). Based on these images the underlying technology was integrated into this shape.

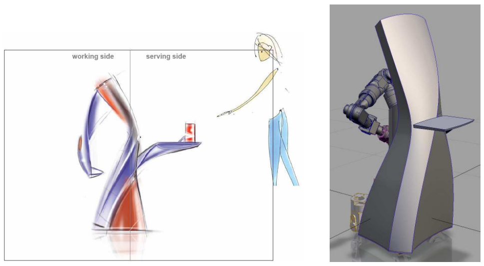


Fig. 2. Left: First design sketch. Right: First technical rendering.

Experiences with former robots showed that the passing of objects directly from human to robot via a robot's gripper was not satisfying. The crucial timing of when the object can be released can not be easily detected by the robot. Between humans it is something which is done very unconsciously and automatically. We have therefore developed a tray concept as an interface between robot and human for the passing of objects. Furthermore, a touchscreen is integrated into the tray for traditional human-computer interaction. If the tray is not used it can be retracted so that the robot is as compact as possible in stand-by.

The torso is designed to be flexible, such that simple gestures like bowing or nodding can be performed. This allows for a more natural, the style of butler conform way of communication.

The convergence of the original design idea and the underlying technology can be seen in Fig. 1 showing the robots final appearance.

III. Hardware Architecture

One special issue with Care-O-bot® 3 is that design and hardware setup were developed in parallel from the very beginning of the project. The benefits of the close cooperation of engineers and designers are visible in the compactness and high level of mechatronical integration of the final hardware-setup (see Fig. 3).

The Care-O-bot® 3 hardware setup includes altogether 28 DOF. The main components of the robot are mobility base, torso, manipulator, tray and sensor carrier with sensors.

The robot is driven by four wheels. Each wheel's orientation and rotational speed can be set individually. This gives the robot an omnidirectional drive enabling advanced movements and simple complete kinematic chain (platform-manipulator-gripper) control (see section V). The wheeled drive was chosen over leg drive because of safety (no risk of falling) and stability during manipulation. The base also includes the Li-Io battery pack (50 V, 60 Ah), laser scanners and a PC for navigation tasks. The size of the base is mainly defined by the required battery space. Nevertheless, the maximal footprint of the robot is approx. 600 mm x 600 mm and the height of the base is approx. 340 mm.

The torso sits on the base and supports the sensor carrier, manipulator and tray. It contains most of the electronics and PCs necessary for robot control. The base and torso together have a height of 770 mm.

The manipulator used is based on the Schunk LWA3, a 7-DOF light-weight arm. It has been extended by 120 mm to increase the work area so that the gripper can reach the floor, but also a kitchen cupboard. Special attention was paid to the mounting of the arm on the robot torso. The result is based on simulations for finding the ideal work space covering the robot's tray, the floor and area directly behind the robot following the "two sides" concept (see section II). Since the manipulator has a hollow shaft no external cables are needed. A slim quick-change system allows to attach different grippers, robotic hands or other tools to the arm. The 7-DOF Schunk Dexterous-Hand used has tactile sensors in its fingers making advanced gripping possible.

The robot has a sensor carrier with high-resolution Firewire stereo-vison cameras and 3D-TOF-camera, enabling the robot to identify, to locate and to track objects and people in 3D (see section IV). These sensors are mounted on a 5-DOF positioning unit allowing the robot to direct his sensors in any area of interest, but also to gesture by body movement of the flexible torso.

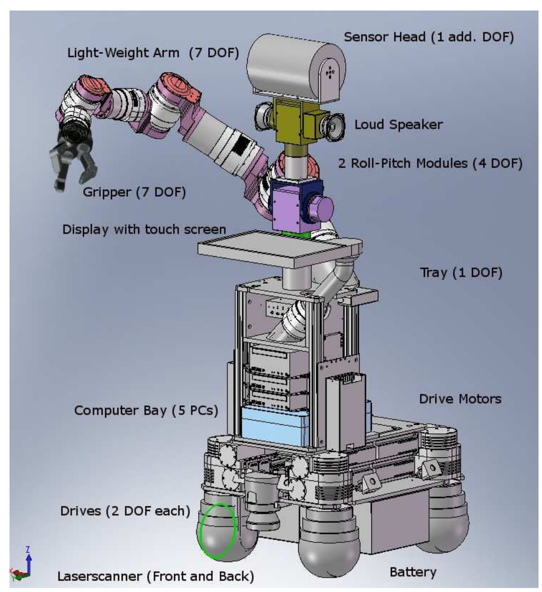


Fig. 3. Hardware Setup

IV. Vision

Object detection for Care-O-bot® 3 is based on the creation of a "shared image", that integrates range and color information. Through the determination of pixel-pair correspondences between range and color image, it is possible to calculate a mapping between image coordinates of range sensor and color sensors. The resulting information of 3D image data and color information is stored in the shared image.

The combined image data provides the foundation of the object detection procedure, which is divided in two fundamental steps: object training and object detection.

A. *Object training*

Before it is possible to detect objects within a foreign environment, the robot must be aware of concrete object representation through training.

In order to train an object, it is put into the robot's gripper, from where object training and abstract class generation is conducted through the robot itself. By rotating the gripper a series of images are taken, providing input data for the training procedure.

The training phase continues with a segmentation of relevant object data from the previously acquired images. As 3D image data and the approximate position of the object are known, a rather simple and fast segmentation strategy is applied. By defining a 3D sphere around the approximate object position in Cartesian space, only those points that reside within the specified sphere are considered as part of the object.

After segmentation, feature extraction is performed on the object relevant image data. The proposed method for feature extraction is based on the fast approximated SIFT [3] algorithm for online object detection. Fast approximated SIFT applies distinctive descriptors relatively invariant against influences of illumination, scale, rotation, distortions and change in viewpoint. By applying a Difference of Gaussian (DoG) filter on range and color image data, so-called "blob feature points" are extracted, that show high contrasts against the background (see Fig. 4).

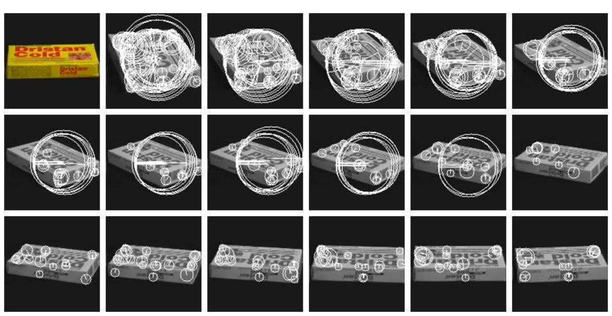


Fig. 4. View of typical blob features from different angles of a rotated object.

Once all features have been extracted, a final step of image processing associates 6D image data to represent position and orientation of the object. For each feature point, a Cartesian coordinate frame is established. Combining all frames of all object views, it is possible to create a 6D feature point cloud that gives an approximated representation of the object's shape. This information is especially important for dexterous object manipulations.

The basic steps of the model construction procedure are outlined in Fig. 5.

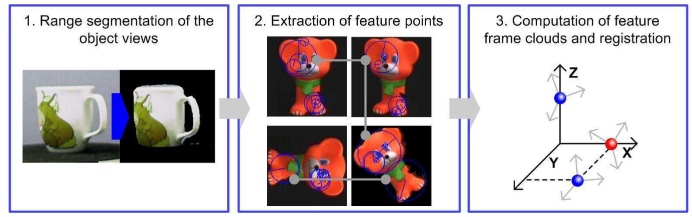


Fig. 5. The basic steps of the model construction procedure.

TABLE I
CLASSES OF WHEELED MOBILE ROBOTS

	carlike	diff drive	pseudo-omnidir. 1	pseudo-omnidir. 2	omnidir.
δ_m	1	2	1	2	3
δ_s	1	0	2	1	0

B. *Object detection*

Recognition of an object within a foreign environment is accomplished by applying a voting based approach. At first, blob features and the corresponding feature frame cloud are extracted from the whole image or a specified region of interest. Then, the descriptors are matched to existing descriptors from previously taught objects. Each correspondence increases a counter for the related object. As soon as a specific counter crosses a given threshold, the object is considered as detected.

V. MOBILITY AND NAVIGATION

Navigation and collision avoidance currently rely on two SICK S300 laserscanners, which are mounted in the middle of the robot's front and rear. For pose estimation an extended Kalman-Filter is used which correlates line segments extracted from the current scan with a map of the environment. The concept of the mobile base with its four actively driven and steered wheels grants the robot a high degree of maneuverability and flexibility needed for motion in narrow home environments [4]. The kinematic properties of Care-O-bot® 3's mobile base are highlighted in the following section. Then the overall control architecture is outlined, before the approach to the control of the robot's undercarriage, coordinating the eight actuators for steering and driving of the wheels, is addressed.

A. *Mobility Aspects*

According to the work by Campion, Bastin and D'Andréa-Novel [5] all wheeled mobile robots may be classified according to their degree of steerability

$$\delta_s = \text{rank}(M), \tag{1}$$

and their degree of mobility

$$\delta_m = \dim\left(N[M]\right) = 3 - \text{rank}(M). \tag{2}$$

Where M in equations (1,2) is the matrix-representation of the system's non-holonomic bindings and $N[M]$ indicates the Nullspace of M. Thus, δ_m resembles to the dimension of the system's instantaneously accessible velocity space, also called the differentiable degrees of freedom (DDOF) of the system. The degree of steerability δ_{S} resembles to the dimension of the system's configuration space, reduced to the submanifold of configurations that allow non-zero movements ($\delta_{\mathrm{m}} \geq 1$) of the robot in the plane. This is equivalent to the number of independently steerable wheels. According to [5] the following holds for all reasonable wheeled mobile robots

$$1 \leq \quad \delta_{\mathrm{m}} \quad \geq 3 \tag{3}$$

$$0 \leq \quad \delta_{\mathrm{S}} \quad \geq 2 \tag{4}$$

$$2 \leq \quad \delta_{\mathrm{S}} + \delta_{\mathrm{m}} \quad \geq 3 \tag{5}$$

and thus allows to classify all wheeled mobile robots into the five different categories listed in table I. Care-O-bot® 3's kinematics fall in the third category (pseudo-omnidirectional robot of class 1) of this scheme. This means that while Care-O-bot® 3 is moving, it can instantaneously realize only one DDOF, e.g. alternating the absolut value of the velocity while moving on a path with fixed curvature. However, its steering wheels can be adjusted in a way allowing movements in any direction. Thus Care-O-bot® 3 is able to navigate through narrow gaps and cope with cluttered environments. Due to the fact that the robot's number of wheels ($n = 4$) is higher than its degree of steerability ($\delta_{\mathrm{S}} = 2$) the synchronization of the steering motion plays an important role.

B. Control Architecture of Mobile Platform

The control-software of the mobile platform has a multilayer structure, ordered hierarchically from abstract/deliberative to explicit/reactive. The first layer is composed by a probabilistic roadmap method (PRM) based path planner. After setting a target position, the planner generates a path based on the known map and current readings from the two laserscanners. Based on this first plan and regularly updated sensor readings a reactive planner based on an elastic band method adapts the target path of the robot every 100 ms. While both, the global as well as the local planner, take into account the robot's geometry and footprint, the specific kinematics are ignored at this level.

Subsequently an interpolator generates motion commands based on the robot's position on the planned trajectory. These commands constitute a feedforward term in the actual trajectory controller, which also runs at 10 Hz. In contrast to many other approaches [6], [7], [8], [9] the output of the trajectory controller are Cartesian velocity commands $(v_x, v_y, \omega)^T$. While, this is an antagonism to the robot's kinematical properties - namely a degree of mobility of one - it allows to unify the command interface to the robot base. Thus it becomes possible to easily switch between different execution modes. This enables the higher level control to change from a planner and trajectory controller which exploits all DOFs of the system to one that mimics differential drive behaviour (e.g. to approach persons) or to directly control the robot via a manual, possibly holonomic user interface, such as a joypad.

Beneath this unified interface follows a controller instance that synchronizes the motion of the four wheels. This is followed by a last cascade, separate for each wheel module, which synchronizes steer and drive motor.

C. State-Space & Control on Undercarriage-level

The state-space representation employed for Care-O-bot® 3 is similar to the representation proposed by Thuillot, D'Andréa-Novel and Micaelli in [9] and is based on reformulating the ICM. In [10] we showed that our reformulation leads to a representation of the ICM that is literally equivalent to the representation of the robot's twist vector $\vec{t_c}$ in spherical coordinates $\vec{t_s}$:

$$\vec{t_c} = \begin{pmatrix} v_x \\ v_y \\ \omega \end{pmatrix}, \vec{t_s} = \begin{pmatrix} \rho \\ \phi \\ \theta \end{pmatrix}. \tag{6}$$

Given that the steering angles take the commanded values (the lower level controllers have to care for that) the state-space representation of the robot can be written in the following form

$$\begin{pmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta}_r \\ \dot{\rho} \\ \dot{\phi} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \rho \cdot \cos(\theta) \cdot \cos(\phi) \\ \rho \cdot \cos(\theta) \cdot \sin(\phi) \\ \frac{\rho}{d_{\max}} \cdot \sin(\theta) \\ u_1 \\ u_2 \\ u_3 \end{pmatrix}, \tag{7}$$

where $d_{\max}$ is a parameter introduced to render ω the same dimension as v_x and v_y, and $\vec{u}$ is the vector of the input variables. It has to be noted that this workspace encompasses singular spots. To drive the system towards its target configuration, while avoiding the critical regions a potential field based control approach [11] was implemented:

$$\vec{u} = -M^{-1} \left(\sum \nabla U_{Si} \big|_{\vec{t_s}} + \nabla U_G \big|_{\vec{t_s}} + k_\nu \dot{\vec{t}}_s \right), \tag{8}$$

where U_{Si} is the repulsive potential originated by the i-th singularity, U_G is the potential that drives the system towards the target configuration and k_ν is a stabilizing damping term and M resembles the inertia matrix of the system. However, note that M is not the inertia matrix itself. It is used in this context to tune the dynamical behaviour of the controlled system.

VI. Manipulation

Care-O-bot® 3 consists of three independent kinematic chains (manipulator, sensor carrier, tray) with different degrees of freedom and different geometries. Therefore, a flexible architecture is needed that allows to control different setups, distinguished by control, kinematics, simulation and (re)configuration. In particular, a sustainable safety concept is inevitable to avoid damage both to the robot and to the environment through the three independently moving kinematic chains.

In the following, the main building blocks of the manipulation framework are presented. Afterwards a short overview of the safety concept is given.

A. Main building blocks of Manipulation Component

The manipulation component is constituted by a layered architecture (see Fig. 6). The different layers, low level control, kinematics and collision avoidance, are connected by generic interfaces, which allow for a highly modular development. All classes in the low level control module are hardware dependent and a driver or a simulation class has to be written that implements the *LowLevelControl* interface. All other modules are hardware-independent.

In the kinematics module class libraries are provided that implement forward kinematic calculations and an analytical solution for the inverse kinematics. The redundancy is resolved by the closest solution to the current configuration.

Collision free movements are ensured by the *Collision Avoidance* module, which offers class libraries for offline path planning or online collision checking. The models necessary for the collision checking algorithms are built and maintained from static configuration data, from the current joint angles of all manipulators and from sensor data.

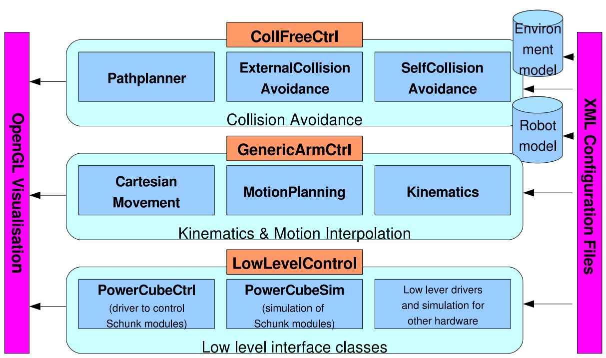


Fig. 6. Building blocks of the manipulation architecture.

B. Safety Aspects

As Care-O-bot® 3 is intended to perform fetch-and-carry tasks and to handle the infrastructure of a household, e.g. cupboard, dishwasher, fridge, etc., safe manipulation is needed. Safe hereby means both safe for the robot and safe for the environment including persons. These two aspects are referred to in the following by intrinsic and extrinsic safety.

1) Intrinsic Safety: Intrinsic safety for Care-O-bot is guaranteed by continuously performing collision checks based on an articulated model of the robot.

The key idea of the collision checking concept is to divide all model parts in a first step into potential colliding sets and collision free sets by considering the current velocity vectors attached to the individual model parts [12]. In the second step, all model parts contained in a potential collision set are checked on collisions by means of intersection tests. The procedure has been split into two parts, because the effort for determining potential colliding sets is usually much less than for calculating the actual distance of the model parts. Note that the collision checking concept is independent from the used models and the actual algorithm to determine if any parts of the model are bound to collide.

2) Extrinsic Safety: The Care-O-bot® 3 concept of distinguishing working side and user interaction side (see section II) reduces the problem of extrinsic safety to a certain extent, as robot's and human's workspace are fairly separated. All static parts of the environment (e.g., cupboards, tables, shelves, etc.) can be modeled in the same way as the robot parts in the obstacle model. The dynamic parts of the environment need to be captured and tracked by 3D sensors. In the hardware-setup of Care-O-bot® 3 (see section III), the laser scanners and the time-of-flight camera are available to update the world model. Currently, we are working on reducing the 3D sensor data (which are available as 3D point clouds) appropriately before adding them to the obstacle model in order to keep the calculation effort on a tolerable level.

The collision detection algorithm was also incorporated into a path planning algorithm. It is based on a fast single-query probabilistic road map path planner with lazy collision checking (SBL) which is included in the Motion Planning Toolkit [13], [14].

Currently, a simple model consisting of oriented bounding boxes (OBBs) [15] is implemented for Care-O-bot® 3. The OBB model of the robot is depicted along with static parts of the environment in Fig. 7.

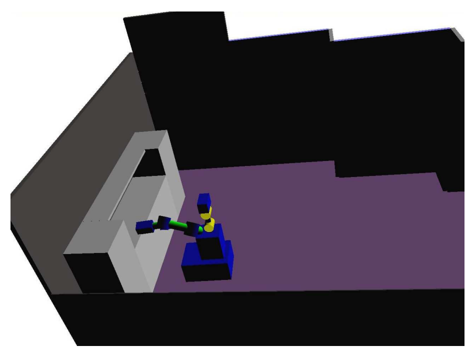


Fig. 7. Obstacle model containing the robot model (blue) along with static environmental model parts.

VII. Results & Experiments

As the system as a whole is hard to benchmark in terms of a quantitative analysis this chapter is structured as follows. First experimental results for the single components object recognition, mobility and manipulation are presented. Then a representative scenario is used to give a metric for the robots performance and robustness. The scenario includes the tasks navigation, object detection, manipulation and grasping and finally mobile manipulation (door opening and passing).

A. Object Recognition

For the scenario (see section VII-D) three different bottles were learned by the method described in section IV. The object detection algorithm described in section IV-B was executed on a 2.0 GHz PC with 1 GB RAM in 300 - 500 ms. Due to the stable feature points [3], the objects were always detected correctly as long as they were not occluded by more than 50 percent and the object view was learned in the training phase. The accuracy of the detected position is directly correlated to the accuracy of the used time-of-flight sensor. In the scenario a position error of about 2-5 cm was obtained when detecting objects in a distance of 2 m.

B. Mobility

The main difference between Care-O-bot® 3's mobility concept and common approaches is constituted in its pseudo-omnidirectional base in combination with the interface allowing to set holonomic commands to the non-holonomic robot platform. The following charts show the resulting system trajectory within the spherical ICM space $(\rho, \phi, \theta)^T$ and the according errors in the steering angles for an exemplary input.

During the experiments the robot is controlled remotely via a joypad, generating the velocity commands. Fig. 8 depicts the results for a 40 s run in which the system was repeatedly driven close to singular configurations. In Fig. 8(a) one can see how the system follows the target configuration and how it is repelled if this configuration passes close to a singularity. As shown in Fig. 8(b) the errors in the wheel's steering angles $\delta\vec{\varphi}_S$ stay reasonably small during these maneuvers.

C. Manipulation

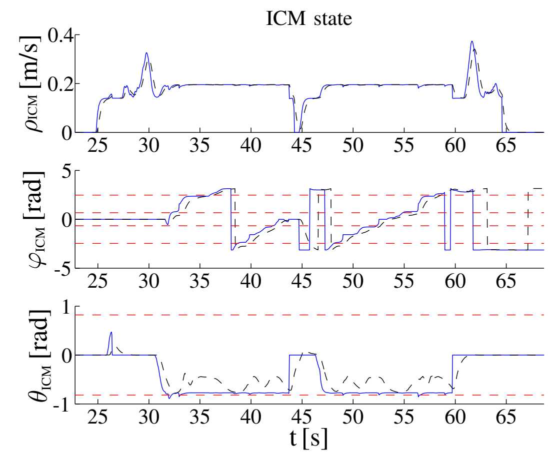
The continuous self collision checking introduced in section VI was implemented with OBB models using the separating axis theorem [15]. Table II shows results of performance tests on a 2.0 GHz PC with 1 GB RAM for different robot movements. Collision checking is executed concurrently while either the arm, the torso or both manipulators are moving. As one collision check lasts less than 1 ms, the manipulators can be moved at a satisfyable speed. Obviously the calculation time for the collision checks correlates to the number of model parts.

TABLE II
DURATION OF ONE COLLISION CHECKING CYCLE FOR DIFFERENT MODEL SIZES.

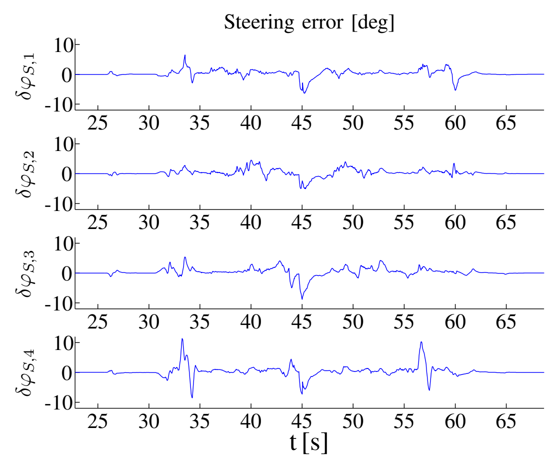
Robot parts moving	Calculation time [ms]
Manipulator 4 DOF	0.14
Manipulator 7 DOF	0.16
Both Manipulators	0.19
Manipulator 7 DOF and Pointcloud (25k points)	86.14

D. Use case “Serving Drinks”

To assess the overall performance of the system and as proof of the ‘two sides’ design concept, Care-O-bot® 3 was tested in context of a task that encompasses all the above discussed components. The use case "‘Serving drinks"’ starts with offering a selection of drinks to the user. After choosing a drink on the touch-screen the robot drives to a bar and tries to detect the chosen drink. If the bottle is succesfully detected, it is grasped and placed on the tray. Thereafter the robot proceeds towards a door which it has to open and pass. Behind the door the robot graps a cup to put it on top of the bottle. Then it drives back to the user and serves bottle and cup via the tray. Fig. 9 shows a short excerpt from the scenario.



(a) Set point values (blue solid) and resulting controlled values (black dashed) for (ρ, φ, θ)



(b) Error in steering angle for all wheels

Fig. 8. Experimental results obtained with Care-O-bot® 3 during a 40 s testdrive where the robot was controlled remotely. The red dashed lines indicate the critical values for φ_{ICM} and θ_{ICM}, on which the singular spots are situated.

Within this scenario the person is familiar with the robot and knows how to operate the touch-screen menu. The drinks are offered and served at a pre-defined position - person-detection is not used during this scenario. Also the tactile sensors were not activated. The environment and especially the position and size of the door which needs to be opened is known a priori.

The scenario was played through 15 times. It was checked how many times the robot succeeded to accomplish the whole task and which component caused the abortion if the robot failed. Table III shows an overview of the errors and error sources.

While every single component succeeded in over 70% of the cases - some reached 100% success-rate - the system succeeded to accomplish the whole process only 6 times. The most critical point were the grasping of the bottle and

Fig. 9. The ‘two sides’ concept: The robot is commanded via the touchscreen to fetch a drink and serve it to the user. During the interaction, the working side of the robot with its arm points away from the user.

TABLE III

SUCCESS RATES OF COMPONENTS AND ROBOT; N_s GIVES THE ABSOLUTE NUMBER OF SUCCESSES; R_s GIVES THE SUCCESS RATE OF THE SINGLE COMPONENTS AND THE FULL SCENARIO IN PERCENT

	Hardware	drive	detect object	grasp bottle	open door	place cup	task done
N_s	13	15	15	11	14	13	6
R_s	86.7%	100%	100%	73.3%	93.9%	86.7%	40%

the placing of the cup on the bottle later on. The analysis shows clearly that the most critical component was the noisy position-estimation based on the swiss-ranger measurements. This caused the roboter to fail three times during the grasp-procedure. Also the two failures while placing the cup on top of the bottle are due to insufficient position-estimation.

VIII. CONCLUSION AND OUTLOOK

The motivation for the development of the service robot Care-O-bot® 3 was presented. The development resulted in an outstanding high mechatronical integration and an iconic, user oriented design with many interaction possibilities. Several innovations from the areas of mechatronics, software, material science and multimedia were included. By having sophisticated software for navigation, manipulation and vision, a solid basis was created for the development of challenging manipulation tasks in everyday environments. A simple object delivery use case was implemented for a first evaluation of the performance of the integrated components and the whole system.

The next steps will consist of implementing more complex household scenarios including e.g., safe manipulation of kitchen infrastructure like fridges or drawers. Simultaneously, existing software packages for manipulation, navigation and vision will be enhanced and further improved with respect to reliability and robustness, offering an even more advanced development platform for service robotic applications. The platform will be used as demonstrator for the software developments of joint national and international research projects in the service robotic domain.

REFERENCES

[1] B. Graf and O. Barth, “Entertainment robotics: Examples, key technologies and perspectives,” in *IROS-Workshop ”Robots in Exhibitions”*, 2002.

[2] M. Hans and B. Graf, “Robotic home assistant Care-O-bot II,” in *Advances in Human-Robot Interaction*, E. P. et al., Ed. Heidelberg, Germany: Springer Berlin / Heidelberg, 2004, pp. 371–384.

[3] M. Grabner, H. Grabner, and H. Bischof, “Fast approximated sift,” in *ACCV 2006: Asian Conference on Computer Vision*. Springer-Verlag, 2006, pp. 918–927. [Online]. Available: http://dx.doi.org/10.1007/11612032_92

[4] B. Graf, M. Hans, and R. Schraft, “Care-O-bot II - development of a next generation robotic home assistant,” *Autonomous Robots*, vol. 16, no. 2, pp. 193–205, March 2004.

[5] G. Campion, G. Bastin, and B. D’Andréa-Novel, “Structural properties and classification of kinematic and dynamic models of wheeled mobile robots,” *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 47–62, Feb 1996.

[6] K. Moore and N. Flann, “A six-wheeled omnidirectional autonomous mobile robot,” *IEEE Control Systems Magazine*, vol. 20, no. 6, pp. 53–66, Dec 2000.

[7] E. N. Y. Mori and T. Takahashi, “Mechanism, control and design methodology of the nonholonomic quasi-omnidirectional vehicle ”odv9”,” *International Journal of robotics Research*, vol. 21, no. 5-6, pp. 511–525, May-June 2002.

[8] M. Lauria, I. Nadeau, P. Lepage, Y. Morin, P. Giguére, F. Gagnon, D. Létourneau, and F. Michaud, “Design and control of a four steered wheeled mobile robot,” *IEEE 32nd Annual Conference on Industrial Electronics*, pp. 4020–4025, Nov 2006.

[9] B. Thuilot, B. D’Andréa-Novel, and A. Micaelli, “Modeling and feedback control of mobile robots equipped with several steering wheels,” *IEEE Transactions on Robotics and Automation*, vol. 12, no. 3, pp. 375–390, June 1996.

[10] C. Connette, A. Pott, M. Hägele, and A. Verl, “Control of an pseudo-omnidirectional, non-holonomic, mobile robot based on an icm representation in spherical coordinates,” *IEEE Conference on Decision and Control*, Dec 2008.

[11] C. Connette, C. Parlitz, M. Hägele, and A. Verl, “Singularity avoidance for over-actuated, pseudo-omnidirectional, wheeled mobile robots,” *IEEE International Conference on Robotics and Automation*, May 2009.

[12] U. Reiser, R. Volz, and F. Geibel, “Manipa: A flexible manipulation framework for collision avoidance and robot control.” *ISR 2008 : 39th International Symposium on Robotics*, pp. 407–411, October 2008.

[13] F. Schwarzer, M. Saha, and J.-C. Latombe, “Adaptive dynamic collision checking for single and multiple articulated robots in complex environments,” *IEEE Transactions on Robotics and Automation*, vol. 21, no. 3, pp. 338– 353, June 2005.

[14] G. Sanchez and J.-C. Latombe, “A single-query bi-directional probabilistic roadmap planner with lazy collision checking,” *International Symposium on Robotics Research*, November 2001.

[15] S. Gottschalk, M. C. Lin, and D. Manocha, “OBBTree: A hierarchical structure for rapid interference detection,” *Computer Graphics*, vol. 30, no. Annual Conference Series, pp. 171–180, 1996.