Paper:

Teleoperating Assistive Robots: A Novel User Interface Relying on Semi-Autonomy and 3D Environment Mapping

Zdeněk Materna*, Michal Španěl*, Marcus Mast, Vítězslav Beran*, Florian Weisshardt***, Michael Burmester**, and Pavel Smrž***

*Centre of Excellence IT4Innovations, Faculty of Information Technology, Brno University of Technology
Bozetechova 1/2, 612 66 Brno, Czech Republic
E-mail: {imaterna, spanel, beranv}@fit.vutbr.cz
**Information Experience and Design Research Group, Stuttgart Media University
Nobelstr. 10, 70569 Stuttgart, Germany
E-mail: ux@mhmast.com, burmester@hdm-stuttgart.de
***Fraunhofer Institute for Manufacturing Engineering and Automation IPA
Postfach 800469, 70504 Stuttgart, Germany
E-mail: florian.weisshardt@ipa.fraunhofer.de
[Received December 17, 2015; accepted December 8, 2016]

Despite remarkable progress of service robotics in recent years, it seems that a fully autonomous robot which would be able to solve everyday household tasks in a safe and reliable manner is still unachievable. Under certain circumstances, a robot's abilities might be supported by a remote operator. In order to allow such support, we present a user interface for a semi-autonomous assistive robot allowing a non-expert user to quickly asses the situation on a remote site and carry out subtasks which cannot be finished automatically. The user interface is based on a mixed reality 3D environment and fused sensor data, which provides a high level of situational and spatial awareness for teleoperation as well as for telemanipulation. Robot control is based on low-cost commodity hardware, optionally including a 3D mouse and stereoscopic display. The user interface was developed in a human-centered design process and continuously improved based on the results of five evaluations with a total of 81 novice users.

Keywords: human-robot interaction, user interface, teleoperation, remote manipulation

1. Introduction

Autonomous systems cannot yet be programmed to handle all possible situations. A remote human operator may help the robot to solve many difficult situations. The collaboration between humans and robots, often referred to as either shared autonomy or human in the loop, might be highly useful in cases where robots often fail, e.g., in object recognition and environment manipulation. On the other hand, an operator should not be bothered by repetitive low-level tasks which can be solved by the robot itself. Then, the operator is not overloaded with solving trivial issues and may concentrate on the important ones and, for example, control more robots due to the time freed. The challenging issue is to equip a potential human operator with easy-to-use but powerful interaction and control tools to act appropriately and effectively in various situations.

This paper describes a novel 3D interactive user interface and its components. The interface allows a user to assess the situation on a remote site, safely navigate in environments with obstacles and with narrow passages where autonomous navigation is likely to fail and to grasp previously untrained objects in cluttered scenes, in various poses and on non-flat surfaces. It is based on common low-cost hardware and can be optionally used with a 3D mouse for intuitive robot navigation and arm control. Additionally, stereoscopic display may be used for improved depth perception. It also includes a module for building a memory-efficient 3D map of the environment, which is used for both visualization purposes and for the planning of collision-free arm trajectories.

The interface has been developed as part of a larger system within the SRS project.[1] The goal of the SRS project [1, 2] was to develop a personal robot able to support elderly people in independent living at their residence. Based on the results of a survey of user-demanded features and on considering what is realistic to implement on current hardware [3], when designing the remote user interface, our primary objectives were navigation and manipulation capabilities.

The SRS project adopts a semi-autonomous paradigm, where under normal circumstances the robot is controlled by its autonomous system, which follows instructions

1. Multi-Role Shadow Robotic System for Independent Living, http://srs-project.eu [Accessed December 7, 2015], technical documentation available at http://wiki.ros.org/srs_public [Accessed December 7, 2015].

given by the elderly person. Local control is based on a mobile device, which allows the user to initiate autonomous actions such as "bring an object." So most of the time, the robot is controlled by its autonomous system without any remote intervention. In case a problem occurs with task execution, there is a second, more advanced interface, which is typically used by a family member who lives separately. The family member can, through a tablet-based interface, control the robot to help the elderly person physically with their daily living tasks. If there is a problem unsolvable by the previous two interfaces, a professional operator is called who can remotely control the robot through the most advanced interface (the one described in this work) and use semi-autonomous functionality to guide the robot, e.g., to bring an object unknown to its autonomous system. The autonomous system and its connection to various interfaces are further described in [2].

The Care-O-bot 3[2] service robot [4] was used as a project demonstration platform. It is based on an omnidirectional platform with positionable torso and a sensor head, a Kuka LBR dexterous manipulator (7 DOF) equipped with a Schunk SDH three-finger hand (7 DOF) and tactile sensors. The robot uses three 2D laser scanners for obstacle avoidance and a Microsoft Kinect RGB-D camera for 3D perception.

To create the interface, we have combined various existing components with newly designed and developed ones in a novel way, enabling semi-autonomous operation of the robot. The results of two experiments with novice users [5, 6] have suggested high effectiveness and suitability of the approaches incorporated in our user interface. Even a short simulation-based training of 60 minutes (including introduction to the robot) was sufficient for achieving high success rates in navigational, search, and manipulation tasks in a home-like environment. In previous publications we have described the overall usage concept underlying the present user interface [3, 7], the framework enabling its semi-autonomy [2], and results of experiments on user interface components [5, 6]. The present paper describes the latest iteration of the user interface, iteratively improved based on the results of several evaluations.

This paper describes a user interface for a semi-autonomous robot. Section 2 presents related work. Section 3 gives an overview of the goals that motivated development. Section 4 describes the development and evaluation procedure. The interface architecture and its basic functionality are detailed in Section 5. Sections 6 and 7 describe two main use cases for our interface: remote navigation and manipulation. Section 8 draws conclusions.

2. Related Work

In this section, we will give a brief overview of the previous work related to remotely operated robots from different perspectives.

2.1. Robot Control Architecture

Various approaches exist for assistive robot control architecture. For instance, the robot presented by Michaud et al. is fully teleoperated and focuses mainly on establishing communication between teleoperator and elderly person [8]. When a teleoperator is not available, the robot is not able to perform any task. To overcome the lack of true autonomy, some approaches introduce nearly full autonomy with the possibility of human intervention when necessary. These approaches are referred to as semi-autonomy, shared autonomy, adjustable autonomy, or human in the loop. Such systems may provide to the operator tools with various levels of autonomy. For instance, the system proposed by Muszynski et al. based on egoperspective visualization offers three levels of autonomy [9]. A similar approach was designed by Bruemmer et. al. where the robot also offers different levels of autonomy [10]. Their user study has shown that users performed better when using tools with more autonomy. Similar results suggesting that more autonomy leads to an improved teleoperator performance were obtained in [11, 12]. The recent efforts utilize human semantic knowledge to help robots perform better [13], which might lead to less operator intervention and thus to decreased workload. Using a robot's motion planner instead of low-level joint control can be also considered a semi-autonomous approach and according to [14] it is also more effective. Using high-level arm control including a Cartesian planner and collision avoidance according to [12] allows users to focus fully on the cognitive part of the task, which is usually the most challenging for the robot.

2.2. Visualization and User Interaction

Traditional video-based interfaces transmitting images from a camera mounted on a robot provide low situational and spatial awareness and increase the risk of collisions [15]. The lack of human-robot awareness, e.g., knowledge of the robot's state and the state of the environment are the primary causes of incidents during teleoperation [16]. The main problem of video-based teleoperation lies in the limited field of view and the absence of depth data [17]. Traditionally, additional information is shown to the user in a separate window or overlaid over the video on the sides. Individual information on the state of the robot and the environment must be mentally correlated, which increases cognitive load. The ecological interface paradigm [18], on the other hand, fuses as much information as possible into a one coherent virtual scene and acts as a form of a mixed-reality. Interfaces based on this paradigm appear to provide better situation awareness and require less mental load [15]. A virtual scene presented to the operator can be based on a manually created 3D model [19], an extruded 2D map [10], or a continuously updated 3D model based on sensor measurements. Results of a study by Mast et al. [6] have suggested the

2. http://www.care-o-bot.de/en/care-o-bot-3.html [Accessed December 7, 2015]

usefulness of an automatically built and updated 3D environment model for navigating a robot remotely.

In case of video-based egocentric interfaces aimed at robot navigation, a joystick was used to be a frequent choice. New ways of control were introduced for virtual reality-based interfaces, which are using exocentric display perspective such as "point and click" [11], when a goal position for the robot is specified by clicking a place in the virtual environment. Most recent interfaces tend to use virtual widgets, also called interactive markers [11, 12]. The advantage of these markers is that they are an integral part of the virtual scene and no special device is required as opposed to control using, e.g., the Phantom device [20], motion capture [21], data gloves [22], or brain-computer interfaces [23]. A crucial issue associated with the difference between the input devices and the visualization is the potential problem of display-control misalignments introduced by using different coordinate systems. Thus, the remote operator has to keep switching mentally between the coordinate systems. This issue has been addressed by either using artificial cues [24] or by choosing an appropriate coordinate system.

2.3. Imaging Equipment

A conventional 2D display can only convey depth perception based on monocular depth cues, consisting of perspective, occlusion, lighting and shadows, relative object size, surface textures, etc. Stereoscopic displays on the other hand enable users to naturally judge relations between objects, based on provided binocular cues [25]. Potential advantages of stereoscopy have been investigated in several studies. For instance [26] suggested that there was no significant difference in completion times between stereo and mono display in a navigation task. On the other hand, there was a substantial difference in the number of collisions against the environment, which were lower for the stereo condition. Utility of stereo display for dexterous manipulation has been investigated in [27]. In their comparison of an interface based on multiple 2D views of the scene versus stereoscopic display, the stereoscopic mode resulted in a 60% decrease of task completion time. Influence of mono and stereo visualization of 3D scan data on users' ability to understand the environment has been investigated in [28]. This work points out that the stereoscopic visualization reduces the risk of misunderstanding the environment. Various technologies for stereoscopic display have been compared in [29] and it was found that shutter glasses provide depth impression comparable to much more expensive polarized walls or cave.

2.4. Conclusion

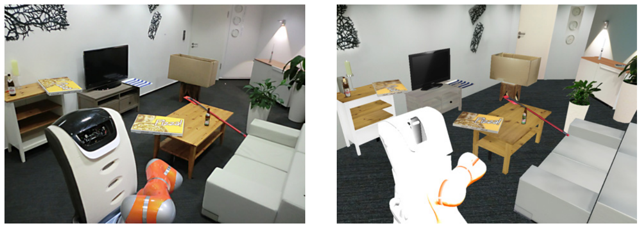
Until fully autonomous assistive robots will be available, some form of teleoperation will likely be necessary. Using a semi-autonomous approach a robot remote operator's workload can be lowered and at the same time performance increased. The degree of the underlying autonomy plays a crucial role in operators' performance. Another important factor is the user interface, its design, capabilities and ability to convey rich information. There are approaches focused on particular aspects however there is currently none utilizing a synergy of these aspects, moreover using affordable hardware for user interaction.

3. User Interface Design Goals

The vision underlying our user interface is a robot that acts autonomously as much as possible. Only when it fails to accomplish a task by itself, a human operator takes over remotely and intervenes with navigation or manipulation. During the intervention it is up to the operator to select appropriate tool with given level of autonomy leading to the lowest workload and safe operation. To be able to solve a wide range of problems, users were to have a high degree of control over the robot. The user interface further had to be easy to use as it was primarily aimed at tele-assistants, i.e., non-roboticists who were only to receive basic training [3]. Our goals were thus to maintain a high degree of robot autonomy while allowing a high degree of controllability, in a system that would still be easy to use. We identified a number of interesting approaches for achieving these goals:

- Techniques for assisted, semi-autonomous remote manipulation and navigation, aiming to take away load from the operator and allow safe operation over unstable network connections, e.g., [8, 10, 12].
- The ecological interface paradigm that enables an operator to directly infer possible actions from the visualized environment and thereby aims to reduce cognitive load and improve situation awareness and user interface usability, e.g., [15, 18].
- 3D visualization of the large-scale environment outside the robot's current field of view for better spatial orientation, e.g., [10, 19].
- Utilization of contemporary 3D sensors able to generate live colored 3D point clouds for a high degree of realism and detail, e.g., [12, 15].
- 3D environment mapping based on 3D sensor data for realistic large-scale representation of the environment, aiming to improve spatial orientation and situational awareness, e.g., [30, 31].

While each of these approaches is promising on its own, they had so far been used in a rather isolated way. For example, ecological interfaces were restricted to either navigation [18] or manipulation [15] or did not employ semi-autonomy. Some previous interfaces relying on 3D environment visualization were based on manually created 3D models [10, 19] rather than on sensor-based environment models that can be generated and kept up to date automatically. Applications of 3D environment mapping using 3D sensors were not used for visualization in the user interface [31]. We thus aimed to create a holistic solution for both semi-autonomous remote manipulation and navigation, using modern technology and integrating the above-mentioned approaches into a consistent user experience. We relied on commonly available low-cost hardware and, where possible, on software components already available. We developed own components or extensions to existing ones where necessary.



(a) Living room in reality (b) Modeled living room

Fig. 1. Realistic apartment model designed for evaluating the user interface; includes living room, bedroom, kitchen, corridors, and 80 household and furniture items.

4. Iterative Development and Evaluation

The user interface was developed following a human-centered design process [32] in several iterations of development and testing, evolving from a conceptual prototype into a fully functional user interface. A total of 430 prospective users were involved in studies directly and indirectly related to this user interface, carried out in the SRS project [2]. Early studies focused on eliciting user requirements [3, 33] and on the development of an overall usage concept also including two reduced-functionality mobile user interfaces not described here [3, 7]. The present user interface was tested five times at different stages of development with a total of 81 users. All evaluations were carried out with non-expert users. As the focus of the present paper is the description of the user interface, we just give a brief overview of the evaluations here and, where available, refer to the publications describing them for more detail.

The first evaluation was a usability test carried out in Germany at Stuttgart Media University's User Experience Research Lab employing a horizontal prototype of the user interface (static screens simulating interaction) [3]. Seven teleassistants from home telesupport centers were recruited for this study. We determined 18 usability problems that lead to 10 design changes in the horizontal prototype.

In the second evaluation an early implementation of the user interface was tested. This evaluation was carried out by project partner Don Gnocchi Foundation in Milan. Five users remotely navigated the robot through a realistic model apartment purpose-built for evaluations. This study gave insight into the strengths and weaknesses of various control modes for remote robot navigation. Also, numerous technical and usability issues were uncovered and addressed in subsequent development.

The third evaluation was again carried out in the lab in Stuttgart and employed the Gazebo robot simulator [34]. We created a detailed apartment model for carrying out evaluations in simulation under realistic conditions (**Fig. 1**). It consists of three rooms, connected by corridors, and contains 80 household and furniture items with realistic physical properties such as weights and friction resistances. The apartment was precisely modeled after the site used in our later experiments. We have made this model freely available so it can be used by other researchers.[3] 14 users participated in this evaluation. The

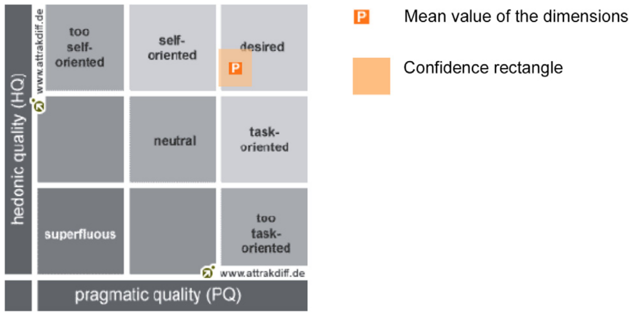


Fig. 2. Results of the most recent user experience assessment, based on the user interface's stereo mode: mean user ratings for pragmatic quality (usability) and hedonic quality.

evaluation focused on strengths and weaknesses of various approaches for visualizing the remote environment in the user interface. It also served as a comprehensive pilot study for the experiments carried out subsequently in reality.

When the user interface had reached a fully functional and stable state, we carried out two experiments with more narrowly specified questions and larger numbers of participants in a purpose-built model apartment on Fraunhofer IPA's premises in Stuttgart. The first experiment, i.e., the fourth evaluation, with 27 participants investigated the utility of two different types of global 3D environment maps (voxel-based and geometric) visualized in the user interface for remotely resolving navigational problems the robot cannot handle autonomously. Results are briefly summarized in Section 5.3 and described in detail in [6].

The second experiment and fifth evaluation [5] was carried out with 28 participants at the Fraunhofer site. Its first purpose was to investigate potential advantages of stereoscopic presentation of the user interface for remotely resolving problematic situations with object manipulation and robot navigation. These results are briefly summarized in Section 5.5 and described in detail in [5]. The second purpose was to obtain an assessment of the quality of users' experience of interacting with the interface. This included ratings of usability and hedonic quality, measured with the AttrakDiff instrument [35]. The main user experience results are visualized in **Fig. 2** (based on stereo mode, which scored higher). The user interface overall

3. http://wiki.ros.org/srs_user_tests [Accessed December 7, 2015]

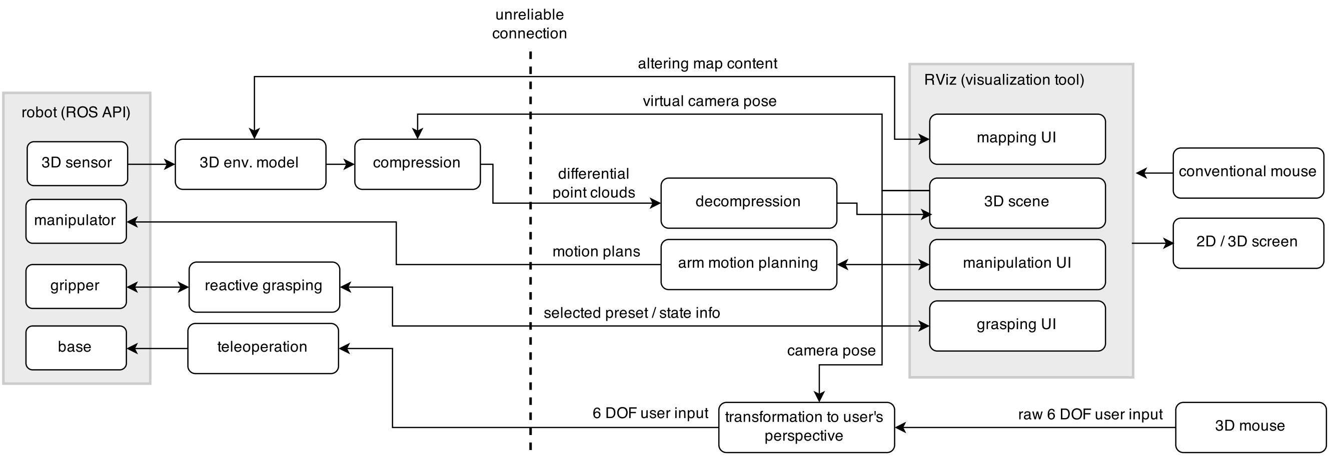


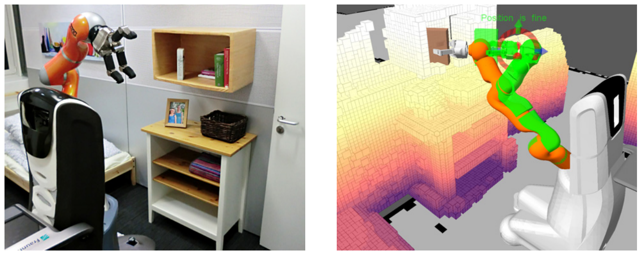
Fig. 3. Simplified diagram showing interactions between main components of the user interface, their connection to the robot and input and output devices.

falls just into the range of "desired," which is a highly encouraging result but there is also still some room for improvement. More details on these results can be found in [5].

5. Visualization and Interaction Approach

The interface consists of many components, the main ones being depicted in **Fig. 3**. It runs on two computers – one on the robot in a Wi-Fi network, and a remote user station. The front-end user interface is based on a visualization tool combining the interactive 3D scene showing most of the information and the side-panels with conventional elements like buttons etc. The user is provided with a 2D mouse, a 3D mouse and a conventional or a stereoscopic screen. The user station also hosts an arm motion planning component providing, among others features, inverse kinematics which is used for visualization. The robot's computer hosts, apart from low-level drivers etc., components for mapping, grasping and teleoperation. All components communicate using the ROS middleware and thus can be easily reused.

The interface specific feature is an API which can be used by the autonomous system to ask the user for help if a problem arises. Normally, the interface is disabled. When the robot's autonomous system cannot complete some task (see **Fig. 4(a)**), it sends a request to the interface. The interface then leads the user through the task giving text instructions for completing respective sub-tasks and automatically enabling necessary components such as an interactive virtual arm (see **Fig. 4(b)**). When dealing with a task, the user may at some point (sub-task) decide that the main problem is solved and hand back control to the autonomous system. Alternatively, he or she may decide that the task would be too difficult to complete for the robot and finish it manually. With this approach, the operator's time is conserved as much as possible.



(a) Robot having a problem in autonomous mode (cannot recognize object)

(b) View from the interface with the object already segmented and grasped

Fig. 4. The interface allows the user to manipulate an untrained object which cannot be handled autonomously.

5.1. 3D Mixed Reality Environment

The user interface is based on RViz,[4] a modular 3D visualization tool, for which we developed several custom plugins and an extension for stereoscopy. The largest portion of the user interface is dedicated to a rendered view of a 3D environment. The mixed-reality environment consists of a 2D map relevant for localization and navigation, a continuously updated 3D map, a robot model in proper scale and configuration according to the robot's proprioception. Moreover, there is in-scene visualization of data from three 2D laser scanners and the RGB-D camera. The 3D scene also contains interactive markers for robot control, object representation, etc. Elements of the user interface are automatically switched on and off based on the current context.

5.2. User Interaction

The user interface can be controlled exclusively by a common 2D pointing device. Optionally, a 3D mouse may be used for some tasks. During our pre-tests, 3D mouse-based control proved to be comfortable, easy to learn, and sufficiently precise even for manipulation in complex scenes. The 2D mouse is used to set the scene

4. http://wiki.ros.org/rviz [Accessed December 7, 2015]

view to any angle and distance, to interact with the in-scene 3D widgets, and to control the conventional part of the interface. The 3D mouse may be used to teleoperate the robot's base and to control the end effector goal pose.

The 3D mouse we used, SpaceNavigator,[5] is a low-cost device with six degrees of freedom. When using the 3D mouse, all cursor movements are encoded as a vector $(t_x, t_y, t_z, r_y, r_p, r_r)$ where (t_x, t_y, t_z) represents the translational part and (r_y, r_p, r_r) the rotational part in the form of yaw, pitch, and roll angles. We consider the pose of the camera observing the mixed-reality scene and transform control inputs from the 3D mouse coordinate system to the camera perspective. This leads to controlling robot movement in the user's rather than in the robot's coordinate system. The transformation is rather simple – the translation vector (t_x, t_y, t_z) introduced by the 3D mouse is rotated along the z-axis, i.e., the one perpendicular to the floor plane in the scene, according to the current camera pose so that the translation along the z-axis t_z remains unchanged:

$$(t'_x, t'_y) = (t_x, t_y) \cdot \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix}. \quad . \quad . \quad . \quad (1)$$

Here α is the current yaw angle of the camera pose in the scene coordinate system. As this transformation of the control commands to the user perspective requires much less mental rotations it should help to lower cognitive load on a user.

To enable the user to control the robot's base during teleoperation or end effector during telemanipulation very precisely at low velocities and at the same time to move fast across longer distances we have introduced a non-linearity into the SpaceNavigator outputs. The following equation is applied to each component of the 6DoF vector $(v_1, v_2, v_3, v_4, v_5, v_6)$ resulting from the 3D mouse:

$$n_i = \left(\frac{v_i}{v_i^{\max}}\right)^2 \cdot v_i^{\max}, \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (2)$$

where v_i is the original value, n_i is the transformed value and $v_i^{\max}$ is the maximal allowed value of the i-th component.

5.3. 3D Voxel-Based Environment Model

The robot's Kinect camera provides standard RGB images as well as colored point clouds at 30 Hz. The sensor has a limited field of view (57° horizontally and 43° vertically), a considerable level of noise and depth resolution decreasing quadratically with increasing distance from the sensor [36]. Mainly due to the limited field of view, using only live point clouds from the sensor for situation assessment or finding obstacles or objects to fetch would be complicated for a remote operator.

To overcome this limitation, we have introduced an environment model which combines point clouds into a consistent global map as the robot travels around the environment (see **Fig. 5**). Our solution is based on the Octomap library [30], which models the environment as a grid of cubic volumes of varying size. This grid is hierarchically organized in an octree structure where each node represents a space contained in the cubic volume, and this volume is recursively subdivided into eight subvolumes until a preset minimum voxel size is reached. The Octomap library uses probabilistic occupancy mapping to fuse input sensor data suffering from errors and uncertainty into robust estimation of the true state of the environment. The continuously updated global map is displayed to the user and used for collision-free arm trajectory planning. The approach allows the user to see and consider the whole environment around the robot. See **Fig. 6** for an example of a visualization of a room from a home-like environment using a voxel resolution of 0.025 m. This resolution seems to be sufficient for the model to serve as a clue for spatial awareness and for obstacle avoidance. For high-precision tasks, users can rely on more detailed live sensor data (see Section 5.4).

To cope with limited network bandwidth, especially over unreliable wireless networks, we have developed modules for compressed transfer of differential frames representing the modified parts of the whole global map. They consider the position of the robot's 3D camera in the environment and its field of view and then compute and

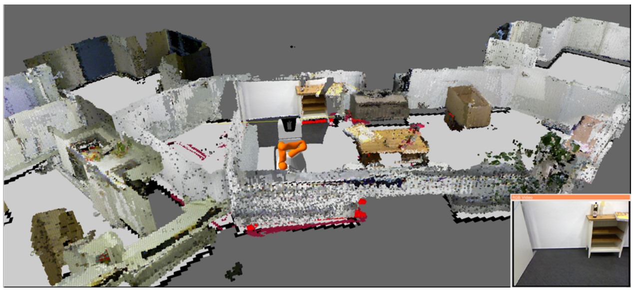


Fig. 5. Automatically generated and updated 3D model of home-like environment covering an area of 100 m^2.

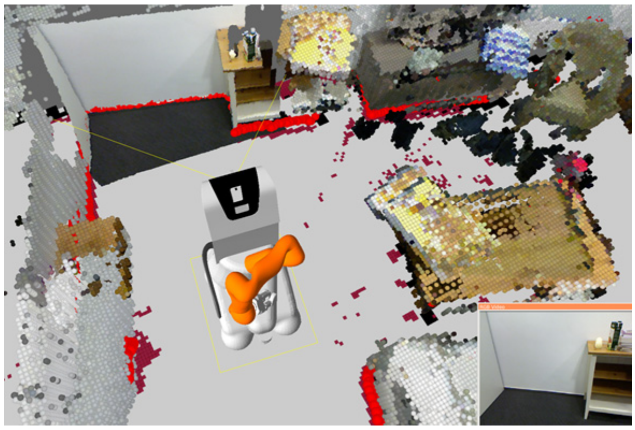


Fig. 6. The 3D mixed reality environment consisting of a robot model, 2D laser data, a 2D map, a combination of the live RGB-D data in current field of view of the robot (visualized using thin lines) and the 3D voxel-based map outside it and a video stream.

5. http://www.3dconnexion.com/products/spacenavigator.html [Accessed December 7, 2015]

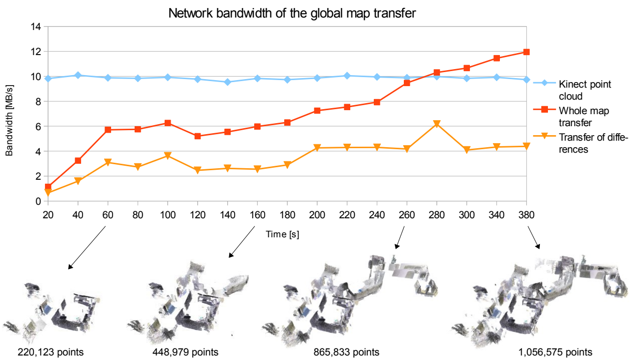


Fig. 7. Network bandwidth for whole global map transfer is compared to sending of map differences. Input RGB-D data and environment mapping were throttled to process 1 frame per second. The whole map was sent after each 5 differential frames.

send to the user's PC the corresponding point cloud in a compressed form. At the user's PC, the point cloud is decompressed and the respective part of the global map updated. Once per 5 to 10 differential frames, the whole map is sent to be able to recover from failures. **Fig. 7** shows the network bandwidth we measured during a test run around the evaluation apartment. Results show that the differential approach can save 65% of the network bandwidth for the resulting global map of 1056575 points. Memory requirements of the internal Octomap representation were growing up to 1.015 GB in this case. To further save network capacity, RGB camera images are transfered using the Theora codec. There are many other possibilities to cope with network issues but these remain to future work.

We have further extended the functionality of the standard Octomap library by:

- Allowing the user to manually modify a part of the map – either by clearing out a region of the map hindering arm trajectory planning, or by adding an artificial object to prevent the robot from going there.
- Filtering incoming point clouds for ground parts and speckles so that they do not obstruct the view and the 2D map.
- Removing noise and outdated parts of the 3D map using a ray-cast technique that clears out outdated parts of the environment when they are newly observed by the robot.

We investigated the usefulness of visualizing global 3D environment maps in the user interface in an experiment [6]. We compared the voxel-based mapping approach described above with an alternative geometric mapping approach, optimized for low network bandwidth consumption [37], and further with a condition without any global 3D mapping. Participants accomplished various object search and obstacle navigation tasks with the robot in a home-like environment. Global 3D environment mapping showed to have substantial temporal advantages when users were searching for objects in the apartment and it lead to fewer collisions when navigating the robot around elevated obstacles. During one navigation task where all obstacles were located on the floor, 3D mapping did not show temporal advantages – presumably because all relevant environment information was already contained in the 2D laser range data. User performance with the voxel-based technique tended to be better than with the simplified geometric visualization, presumably due to higher visual detail and realism [6].

5.4. Combining 3D Environment Visualizations

An important question is how to combine the "historical" data stored in the 3D map of the environment with the live RGB-D data. It is obviously important to show the remote operator the latest data and to not obstruct the view with any artifacts stored in the 3D environment map – e.g., the previous yet outdated recordings, noise, and speckles. Moreover, the resolution of the 3D map is lower than the resolution of the live data especially for close objects.

Our approach uses the information about the current position and orientation of the robot's torso to cut out the part of the 3D map inside the current field of view and show the live RGB-D data there. We limit the maximum distance from the camera at which the points are filtered because the effective range of the sensor is limited too. To communicate the difference between live and historical data to the user, the current field of view of the sensor is visualized using two thin lines, which do not obstruct the view (see **Fig. 6**).

5.5. Stereoscopic Display

Stereoscopic display can improve user performance [38] and user experience [39]. It has the potential to simplify tasks that depend on the operator's depth judgments, for example reaching and grasping of objects, robot navigation in the room including obstacle avoidance, judging the robot's arm position, or the relative positions and distances of objects in the scene. Without stereo visualization the operator may be less accurate and may need to adjust the viewpoint more often to see the scene from different perspectives.

There are several commercial solutions for stereo display in computer graphics. To achieve the stereoscopic effect, we use the Nvidia 3D Vision 2 stereoscopic kit.[6] This kit consists of LC shutter glasses and driver software. The glasses use a wireless IR protocol to communicate with the emitter providing a timing signal. The stereo driver software performs the stereoscopic conversion by using 3D models transmitted by the application and rendering two separate views from two slightly different points. A fast stereo LCD monitor (120 Hz) shows these two images alternately and the shutter glasses controlled by the emitter present the image intended for the left eye while blocking the right eye's view and vice versa. The scene in RViz is generated using the Ogre library,[7]

6. http://www.nvidia.com/object/3d-vision-main.html [Accessed December 7, 2015]
7. http://www.ogre3d.org [Accessed December 7, 2015]

which, however, is not ready for the stereoscopic display on Linux in the version included in ROS Electric (1.7.3). Thus it was necessary to modify the Ogre library as well as RViz itself.

To assess the usefulness of stereoscopic display for this user interface, we carried out an experiment [5]. 28 participants accomplished remote manipulation and robot navigation tasks – half of the participants under stereoscopic and the other half under monoscopic display. For the task of specifying the gripper's target position for grasping an object in the remote environment (see Section 7.3 and **Figs. 10(c)** and **(d)**), there was a clear temporal advantage of using stereoscopic display. Participants also reached the goals faster under stereo display for the two other types of task, i.e., defining the shape of an object to be grasped (see Section 7.2 and **Figs. 10(a)** and **(b)**) and navigating the robot around obstacles (see Section 6.4). However, the differences were not as pronounced here and not statistically significant after multiplicity correction. We thus concluded that stereoscopic display seems to be a useful additional display mode for this kind of user interface but that its utility may vary depending on the task [5].

6. Assisted Navigation

Safe and reasonably fast movement of an assistive robot can be considered an essential functionality. Contemporary robot navigation systems are quite mature and able to assure 2D navigation even in complex and dynamic environments. However, because of safety concerns, these systems are usually tuned to be conservative, to use wide safety margins, etc. This leads to improved safety but it limits the robot's abilities on the other hand. In our semi-autonomous solution, a remote operator can be contacted if there is a problem with navigation, for instance if the robot cannot move to a desired location.

To solve navigation issues, the operator may use tools with different levels of autonomy depending on the current situation and personal preferences:

- Autonomous waypoint navigation.
- In-scene teleoperation.
- 3D mouse teleoperation (with the option to switch off collision avoidance).

Ecological approaches for teleoperation have typically used a non-interactive 3D scene with rather simplistic visualization of an environment and a joystick to control robot movement [8, 18]. Our approach is similar to previous ones in terms of visualization using a common reference frame and the ability to freely adjust the viewpoint. Beyond this, it provides rich visual information and enables the user to choose an appropriate tool for teleoperating the robot suitable for the particular situation. The 3D scene in our approach is interactive so two of the available navigation tools are integrated into it.

6.1. Scenarios

Under normal circumstances, the robot navigates autonomously using path planning based on the ROS Navigation Stack.[8] While the autonomous navigation is capable of coping with most situations it fails in some cases. A typical example is a very narrow passage where the robot physically fits but, because of safety settings, is not able to pass autonomously. Autonomous navigation also cannot reach its goal if there is an obstacle blocking the path. In semi-autonomous mode, the obstacle can be removed using the manipulator or pushed away with the robot's base.

6.2. Autonomous Waypoint Navigation

The teleoperation tool with most autonomy enables the operator to send intermediate waypoints to the robot's navigation system. This can be useful for moving the robot over a longer distance or when an optimal trajectory, which would normally be chosen by the navigation system, is for some reason not feasible, e.g., when there is a risk of collision. The operator sets waypoints by clicking at a desired position and also specifies the robot's target orientation by rotating the arrow before releasing the left mouse button. After that the trajectory is planned and the plan is visualized to the operator so he or she can easily predict the robot's movement.

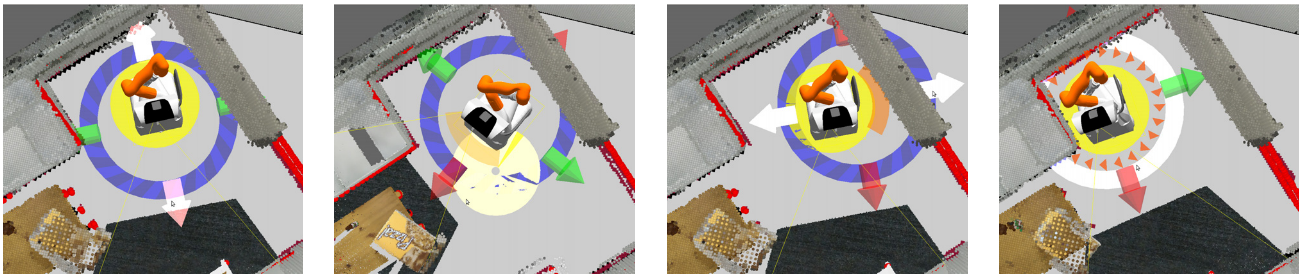
6.3. In-Scene Teleoperation

In order to provide an intuitive way to drive the robot directly within the 3D scene, we have designed a special in-scene teleoperation control that is based on ROS Interactive Markers.[9] The robot can be teleoperated for translational movement in two axes using arrows, and for rotation on the spot using the circle (**Fig. 8**). This type of control is suitable for small and precise movements in a tight space. A more comfortable and faster way of teleoperation is realized by a disk in the middle – when grabbed, the robot follows it. This type of control is more suitable for traversing larger distances in free space. However, while it allows control of more degrees of freedom at the same time, it does not provide precise control for navigation in tight environments. The in-scene control, especially the disc-following concept, was designed as an easy tool to manually drive the robot. When using the disc-following concept, the robot motion is derived from the current disc position (p_x, p_y) relative to the robot base:

$$L_M(x) = sign(x) * \min(M, |x|), \quad (3)$$
$$v_{fwd} = L_M(C_x * p_x), \quad (4)$$
$$v_{rot} = sign(p_x) * L_M(C_y * p_y). \quad (5)$$

Function $L_M(x)$ limits the maximum robot speed, C_x and C_y are constant scaling factors, v_{fwd} is the forward motion velocity and v_{rot} is the robot rotation velocity. Until the user grabs and moves the disc the position (p_x, p_y) is zero. These equations result in a smooth motion of the

8. http://wiki.ros.org/navigation [Accessed December 7, 2015]
9. http://wiki.ros.org/interactive_markers [Accessed December 7, 2015]



(a) Translation in free space (b) Following the disc (c) Translation towards the obstacle (d) Rotation next to the obstacle

Fig. 8. Driving the robot using the inscene teleop. Translation is achieved using arrows (a). Rotation is performed using the circle. The robot can be driven to a specified position by moving the disc (b). Velocity limited marker shown when the robot cannot move in a particular direction (c) and rotate in place (d).

robot when the robot simultaneously turns to face the disc and moves towards the disc.

In many real-world situations, the robot's collision avoidance system based on two 2D laser scanners prevents moving or rotating the platform in some directions because the platform or the arm is very close to either moving or static obstacles. When the robot is close to an obstacle, it automatically reduces its velocity until zero in this particular direction to avoid a collision. In these situations it may be frustrating if the remote operator cannot easily decide in which directions movement is allowed and in which direction the robot cannot be moved. Therefore, we designed a velocity limited indicator to help the remote operator decide in which directions he or she can manually drive the robot. Indicators are shown around the robot in the 3D scene to illustrate in which directions the velocity of the robot is limited (**Fig. 8(c)**) or if the rotational velocity is limited (**Fig. 8(d)**). This helps the remote operator to quickly decide what is the problematic obstacle and how to drive the robot around it.

6.4. 3D Mouse Teleoperation

As an alternative to the in-scene robot control that uses a conventional 2D mouse we have developed a 3D mouse control. It is up to the user's preferences and the problem at hand which way of control will be used. When using a 3D mouse, the indicators for velocity limitations due to imminent collision are available too. Compared to in-scene control using arrows and the blue ring, the 3D mouse allows the user to perform translational and rotational movements simultaneously.

7. Assisted Manipulation

When problems occur, fully autonomous manipulation can be substituted by a semi-autonomous solution, which has been developed as a part of the user interface. Assisted manipulation can be used in cases where automated planning of the arm trajectory fails or is not applicable. It offers a complete pipeline for manipulation tasks consisting of object detection, arm trajectory planning, and grasping.

The approach uses a collision-aware trajectory planner and offline execution. It allows the user to set a desired target position and orientation of the end effector by adjusting its virtual representation in the 3D scene. The scene includes visualization of the whole arm with proper joint positions computed by inverse kinematics. The user may visualize the trajectory animation and eventually let the robot execute it. In case of an emergency, the user can stop its execution. Due to the absence of low-level telemanipulation, latency-related problems are eliminated and thus our approach is also highly usable through unreliable wireless networks and through the Internet.

Previous approaches for remote manipulation were restricted to stationary manipulators [15], used only a video stream for user interaction [24] or used one or more joysticks for robot control [14, 15, 24]. More advanced semi-autonomous approaches often use humans' cognitive skills for selecting objects in cluttered scenes [40] or choosing appropriate grasp points on already detected objects [13] but they do not give users full manual control for cases when a particular automated procedure fails. Our approach allows the user to carry out all steps for object manipulation manually, if necessary. Decoupled motion planning and execution make the interface highly suitable for remote operation when compared to direct telemanipulation [14, 24]. Moreover, usage of a global 3D map updated in realtime provides the user better spatial and situational awareness when compared to interfaces using single 3D snapshots [11, 12, 15].

7.1. Scenarios

The SRS autonomous system [2] offers object recognition and grasping, however its functionality is not available under certain circumstances. First, the object to be grasped must be learnt in advance. This is unproblematic for most of the objects of daily use, however there might be a need to handle an unknown object. Further, detection of a known object may fail because of occlusion in a cluttered scene, low illumination levels, or due to inappropriate robot position. Finally, even in case of a known and detected object, it might be impossible for the au-

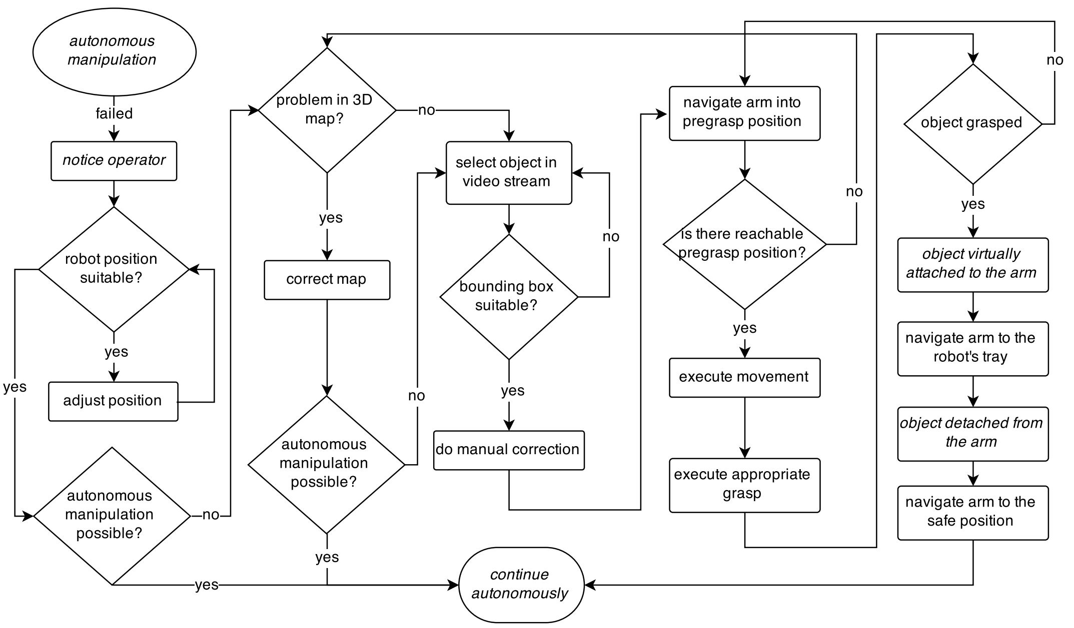


Fig. 9. Manipulation workflow diagram. Each motion planning/execution step can be repeated or divided into more subsequent steps.

tonomous system to reach any of the precomputed grasping positions for various reasons. In all of these cases, a remote operator is called for providing assistance.

When there is a request for remote intervention, for instance when an unknown object shall be fetched, appropriate tools in the user interface are enabled and an operator is instructed with text messages to perform the following steps:

1. Drive the robot to a proper position (the robot is then prepared automatically for the task – the torso is tilted forward, the camera is flipped to the right direction, the arm prepared in the appropriate position, and the tray lifted up).

2. Correct 3D map (i.e., remove noise) if necessary.

3. Manually segment the object from the 3D scene.

4. Navigate the arm to the proper grasp position.

5. Select an appropriate grasp strategy (see Section 7.4) and execute it.

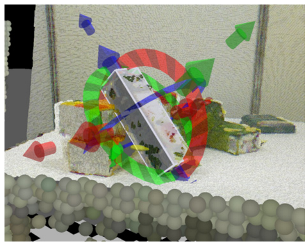
6. Navigate the arm to place the object above the tray (the gripper opens automatically).

7. Check if the object is on the tray and navigate the arm to a safe position.

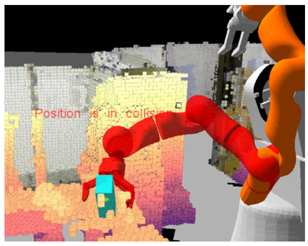
8. Hand back control to the autonomous system.

From this sequence, some steps can be repeated and at some points it is also possible to give the autonomous system the next try after the operator fixed the problem as shown in **Fig. 9**.

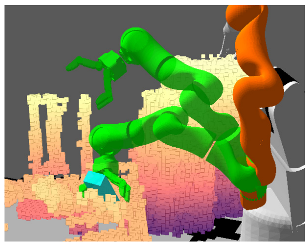
(a) Selecting an object in the video stream



(b) Adjusting bounding box in the 3D scene



(c) Goal position not reachable due to collision



(d) Visualization of planned trajectory

Fig. 10. Assisted arm navigation used to perform a pick-and-place task.

7.2. Object Segmentation

In order to use semi-autonomous manipulation for unknown or unrecognized objects, the dimensions of the object to be grasped need to be defined first. We implemented a tool which accelerates this process. When there is a need for specifying an object shape, an operator is asked to draw a box over the object in the video stream (**Fig. 10(a)**). Based on this region of interest, we fit a bounding box to the corresponding 3D points. The estimated bounding box is shown in the 3D scene (**Fig. 10(b)**), and the user can then adjust its pose and size according to either live 3D data or the 3D voxel-based model as there might be a lot of noise or occlusion in the original sensor data. This bounding box is then considered when planning the collision-free arm trajectory.

7.3. Interactive Arm Navigation

The visualization for arm navigation in the user interface consists of a 3D scene containing a robot representation with manipulator, a 3D collision map, the bounding box of a detected or user-specified object, and of an RViz plugin providing several functions via buttons, which are hidden by default. When a remote operation session is initiated, the operator is notified by a pop-up window and the appropriate controls become active.

For arm navigation (**Fig. 10**), the operator is required to set a goal position for the end effector in the 3D scene. It can be done using interactive 3D widgets or more intuitively by a 3D mouse. While adjusting the virtual end effector position, the real manipulator does not move. Through color coding of the arm as well as a text overlay in the 3D scene, the interface indicates if the desired position is reachable by the arm and whether there are collisions with the environment model or objects. A collision-free trajectory from the start position to the goal position is planned on the user's request. If the planner cannot find a trajectory, the user may try planning with a different goal position or even with a revised robot position. Before executing the planned trajectory, the operator can run its visualization (**Fig. 10(d)**) several times and decide if it is safe. The operator may decide to plan several trajectories for one task. When finished, the operator marks the task as completed and hands back control to the robot.

The solution for trajectory planning is based on functionality provided by the arm_navigation stack. It contains components for generating a robot-specific configuration, maintaining representation of the environment and recognized objects for collision checking, trajectory planning and filtering, inverse kinematics computation, visualization tools, etc. Our main contributions lie in making the user interface adequate for non-expert users, in providing the ability to use a 3D mouse as an input device, and in an API for integration with the autonomous system.

7.4. User-Assisted Reactive Grasping

Our approach for grasping was designed to work for objects unknown to the robot, meaning that there is no known model of an object. This precludes grasping approaches based on prior shape knowledge [41].

We have developed software for the SDH[10] gripper equipped with tactile sensors, which allow easy to use, safe, and robust remote grasping. There is a predefined list of empirically determined target joint configurations with associated maximum forces for each tactile pad. The user selects an appropriate preset according to the object (e.g., "full beverage carton"). Then velocities for the joints are calculated so all joints will reach the target configuration at the same time including acceleration and deceleration ramps of configurable lengths. Any joint is stopped during the process of grasping if the maximum force from its tactile array exceeds a value defined in the chosen preset.

During informal experiments using this approach, we have been able to grasp various objects of daily use and of different shapes. However, results to a certain extent depend on previous steps and experience of the operator.

10. Schunk Dexterous Hand.

8. Conclusions and Future Work

The interface presented in this article enables intervention of a remote operator who may navigate the robot and perform manipulation of objects which cannot be handled autonomously. The interface's central features are a 3D scene display, global 3D mapping with interactive features, tools for teleoperation and telemanipulation, stereoscopic display, and control relying on a 3D mouse. The solution is built on already available and widely used components from ROS and newly designed and developed ones, such as an intuitive user interface for manipulation and a component for the efficient transport of 3D maps. Usage of the 3D interface with fused visualization of all relevant data requires only short training, shown by the fact that novice users in our experiments were all able to complete all tasks we asked them to solve. We believe that the concept of a semi-autonomous robot is promising as even remote manipulation tasks can be accomplished within reasonable time and with reasonable effort.

In order to improve user interaction, we are experimenting with head tracking to introduce motion parallax, which might be useful especially for manipulation. Another option we investigate is to allow a user to change the viewpoint with a 3D mouse. Regarding global 3D mapping, we envision a solution that avoids the influence of imprecise robot localization on a created map. For limited-bandwidth connections, user experience could be improved by using techniques like adaptive frame rates for images and point clouds.

Acknowledgements

This research was supported by the European Commission, FP7 project SRS, Grant Agreement No. 247772; and by The Ministry of Education, Youth and Sports of the Czech Republic from the National Programme of Sustainability (NPU II); project IT4Innovations excellence in science – LQ1602.

References:

[1] L. Pigini, D. Facal, A. Garcia, M. Burmester, and R. Andrich, "The proof of concept of a shadow robotic system for independent living at home," Computers Helping People with Special Needs, pp. 634-641, Springer, 2012.

[2] R. Qiu, Z. Ji, A. Noyvirt, A. Soroka, R. Setchi, D. T. Pham, S. Xu, N. Shivarov, L. Pigini, G. Arbeiter et al., "Towards robust personal assistant robots: Experience gained in the srs project," 2012 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 1651-1657, 2012.

[3] M. Mast, M. Burmester, K. Krüger, S. Fatikow, G. Arbeiter, B. Graf, G. Kronreif, L. Pigini, D. Facal, and R. Qiu, "User-centered design of a dynamic-autonomy remote interaction concept for

manipulation-capable robots to assist elderly people in the home," J. of HRI, Vol.1, No.1, 2012.

[4] U. Reiser, C. Pascal Connette, J. Fischer, J. Kubacki, A. Bubeck, F. Weisshardt, T. Jacobs, C. Parlitz, M. Hägele, and A. Verl, "Care-o-bot® 3-creating a product vision for service robot applications by integrating design and technology," IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), Vol.9, pp. 1992-1998, 2009.

[5] M. Mast, Z. Materna, M. Španěl, F. Weisshardt, G. Arbeiter, M. Burmester, P. Smrž, and B. Graf, "Semi-autonomous domestic service robots: Evaluation of a user interface for remote manipulation and navigation with focus on effects of stereoscopic display," Int. J. of Social Robotics, Vol.7, No.2, pp. 183-202, 2015.

[6] M. Mast, M. Španěl, G. Arbeiter, V. Štancl, Z. Materna, F. Weisshardt, M. Burmester, P. Smrž, and B. Graf, "Teleoperation of domestic service robots: Effects of global 3d environment maps in the user interface on operators' cognitive and performance metrics," Social Robotics, pp. 392-401, Springer, 2013.

[7] M. Mast, M. Burmester, B. Graf, F. Weisshardt, G. Arbeiter, M. Španěl, Z. Materna, P. Smrž, and G. Kronreif, "Design of the human-robot interaction for a semi-autonomous service robot to assist elderly people," Ambient Assisted Living, pp. 15-29, Springer, 2015.

[8] F. Michaud, P. Boissy, D. Labonté, S. Briere, K. Perreault et al., "Exploratory design and evaluation of a homecare teleassistive mobile robotic system," Mechatronics, Vol.20, No.7, pp. 751-766, 2010.

[9] S. Muszynski, J. Stuckler, and S. Behnke, "Adjustable autonomy for mobile teleoperation of personal service robots," 2012 IEEE RO-MAN, pp. 933-940, 2012.

[10] D. J. Bruemmer, D. A. Few, R. L. Boring, J. L. Marble, M. C. Walton, and C. W. Nielsen, "Shared understanding for collaborative control," IEEE Trans. on Systems, Man and Cybernetics, Part A: Systems and Humans, Vol.35, No.4, pp. 494-504, 2005.

[11] T. L. Chen, M. Ciocarlie, S. Cousins, P. Grice, K. Hawkins, K. Hsiao, C. C. Kemp, C.-H. King, D. A. Lazewatsky, A. Leeper et al., "Robots for humanity: A case study in assistive mobile manipulation," IEEE Robotics & Automation Magazine, 2013.

[12] A. E. Leeper, K. Hsiao, M. Ciocarlie, L. Takayama, and D. Gossow, "Strategies for human-in-the-loop robotic grasping," Proc. of the seventh annual ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI), pp. 1-8, 2012.

[13] T. Witzig, J. M. Zollner, D. Pangercic, S. Osentoski, R. Jakel, and R. Dillmann, "Context aware shared autonomy for robotic manipulation tasks," 2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 5686-5693, 2013.

[14] J. D. Will, K. L. Moore, and I. K. Lynn, "Optimizing human-robot teleoperation interfaces for mobile manipulators," Industrial Robot: An Int. J., Vol.40, No.2, pp. 173-184, 2013.

[15] J. A. Atherton and M. A. Goodrich, "Supporting remote manipulation with an ecological augmented virtuality interface," Proc. of the AISB Symposium on New Frontiers in Human-Robot Interaction, Edinburgh, Scotland, 2009.

[16] J. L. Drury, J. Scholtz, and H. A. Yanco, "Awareness in human-robot interactions," IEEE Int. Conf. on SMC 2003, Vol.1, pp. 912-918, 2003.

[17] J. Zhu, X. Wang, and M. Rosenman, "Fusing multiple sensors information into mixed reality-based user interface for robot teleoperation," IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC) 2009, pp. 876-881, 2009.

[18] C. W. Nielsen, M. A. Goodrich, and R. W. Ricks, "Ecological interfaces for improving mobile robot teleoperation," IEEE Trans. on Robotics, Vol.23, No.5, pp 927-941, 2007.

[19] D. Labonte, P. Boissy, and F. Michaud, "Comparative analysis of 3-d robot teleoperation interfaces with novice users," IEEE Trans. on Systems, Man, and Cybernetics (SMC), Part B: Cybernetics, Vol.40, No.5, pp. 1331-1342, 2010.

[20] J. Franz, A. Maciel, and L. Nedel, "Assessment of a user centered interface for teleoperation and 3d environments," Proc. of the 28th Annual ACM Symposium on Applied Computing, pp. 953-960, 2013.

[21] J. Kofman, X. Wu, T. J. Luu, and S. Verma, "Teleoperation of a robot manipulator using a vision-based human-robot interface," IEEE Trans. on Industrial Electronics, Vol.52, No.5, pp. 1206-1219, 2005.

[22] H. Hu, J. Li, Z. Xie, B. Wang, H. Liu, and G. Hirzinger, "A robot arm/hand teleoperation system with telepresence and shared control," Proc., 2005 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics (AIM), pp. 1312-1317, 2005.

[23] C. J. Bell, P. Shenoy, R. Chalodhorn, and R. P. N. Rao, "Control of a humanoid robot by a noninvasive brain-computer interface in humans," J. of neural eng., Vol.5, No.2, p. 214, 2008.

[24] K. Chintamani, A. Cao, R. D. Ellis, and A. K. Pandya, "Improved telemanipulator navigation during display-control misalignments using augmented reality cues," IEEE Trans. on SMC, Part A: Systems and Humans, Vol.40, No.1, pp. 29-39, 2010.

[25] M. Ferre, R. Aracil, and M. Sanchez-Uran, "Stereoscopic human interfaces," IEEE Robotics & Automation Magazine, Vol.15, No.4, pp. 50-57, 2008.

[26] D. R. Scribner and J. W. Gombash, "The effect of stereoscopic and wide field of view conditions on teleoperator performance," Technical report, DTIC Document, 1998.

[27] W.-k. Fung, W.-t. Lo, Y.-H. Liu, and N. Xi, "A case study of 3d stereoscopic vs. 2d monoscopic tele-reality in real-time dexterous teleoperation," 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 181-186, 2005.

[28] T. Fujiwara, T. Kamegawa, and A. Gofuku, "Stereoscopic presentation of 3d scan data obtained by mobile robot," 2011 IEEE Int. Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 178-183, 2011.

[29] S. Livatino and G. Muscato, "Robot 3d vision in teleoperation," World Automation Congress (WAC) 2012, pp. 1-6, 2012.

[30] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," Autonomous Robots, Vol.34, pp. 189-206, 2013. Software available at http://octomap.github.com [Accessed December 7, 2015].

[31] M. Yguel and O. Aycard, "3d mapping of outdoor environment using clustering techniques," 2011 23rd IEEE Int. Conf. on Tools with Artificial Intelligence (ICTAI), pp. 403-408, 2011.

[32] International Organization for Standardization, "Ergonomics of Human-system Interaction: Part 210: Human-centred Design for Interactive Systems," ISO, 2010.

[33] M. Mast, M. Burmester, E. Berner, D. Facal, L. Pigini, and L. Blasi, "Semi-autonomous teleoperated learning in-home service robots for elderly care: A qualitative study on needs and perceptions of elderly people, family caregivers, and professional caregivers," Proc. of the twentieth ICROM, pp. 06-09, 2010.

[34] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," Proc., 2004 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), Vol.3, pp. 2149-2154, 2004.

[35] M. Hassenzahl, M. Burmester, and F. Koller, "Attrakdiff: A questionnaire to measure perceived hedonic and pragmatic quality," Mensch & Computer, pp. 187-196, 2003.

[36] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," Sensors, Vol.12, No.2, pp. 1437-1454, 2012.

[37] G. Arbeiter, R. Bormann, J. Fischer, M. Hägele, and A. Verl, "Towards geometric mapping for semi-autonomous mobile robots," Spatial Cognition VIII, pp. 114-127, Springer, 2012.

[38] R. Edmondson, K. Light, A. Bodenhamer, P. Bosscher, and L. Wilkinson, "Enhanced operator perception through 3d vision and haptic feedback," SPIE Defense, Security, and Sensing, pp. 838711-838711, 2012.

[39] N. Broy, E. André, and A. Schmidt, "Is stereoscopic 3d a better choice for information representation in the car?," Proc. of the 4th AutomotiveUI, pp. 93-100, ACM, 2012.

[40] B. Pitzer, M. Styer, C. Bersch, C. DuHadway, and J. Becker, "Towards perceptual shared autonomy for robotic mobile manipulation," 2011 IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 6245-6251, 2011.

[41] N. Chen, C.-M. Chew, K. P. Tee, and B. S. Han, "Human-aided robotic grasping," 2012 IEEE RO-MAN, pp. 75-80, 2012.

Name:
Zdeněk Materna

Affiliation:
Ph.D. Student, Centre of Excellence IT4Innovations, Faculty of Information Technology, Brno University of Technology

Address:
Bozetechova 2, 61266 Brno, Czech Republic

Brief Biographical History:
2010- Brno University of Technology

Main Works:
• Z. Materna, M. Kapinus, M. Španěl, V. Beran, and P. Smrž, "Simplified Industrial Robot Programming: Effects of Errors on Multimodal Interaction in WoZ experiment," Robot and Human Interactive Communication (RO-MAN), pp. 1-6, New York City: Institute of Electrical and Electronics Engineers, 2016.
• M. Mast, Z. Materna, M. Španěl, F. Weisshardt, G. Arbeiter, M. Burmester, and P. Smrž, "Semi-Autonomous Domestic Service Robots: Evaluation of a User Interface for Remote Manipulation and Navigation With Focus on Effects of Stereoscopic Display," Int. J. of Social Robotics. 2015, pp. 183-202, 2015.
• M. Mast, M. Španěl, G. Arbeiter, V. Štancl, Z. Materna, F. Weisshardt, M. Burmester, and P. Smrž, "Teleoperation of Domestic Service Robots: Effects of Global 3D Environment Maps in the User Interface on Operators' Cognitive and Performance Metrics," Proc. of Int. Conf. on Social Robotics, pp. 392-401, Bristol: Springer Verlag, 2013.

Name:
Michal Španěl

Affiliation:
Assistant Professor, Centre of Excellence IT4Innovations, Faculty of Information Technology, Brno University of Technology

Address:
Bozetechova 1, 612 66 Brno, Czech Republic

Brief Biographical History:
2001-2005 HW Programmer, HEXAGON Systems, s.r.o., Virtual Reality Media, s.r.o., VR Simulators, s.r.o.
2003-2011 Ph.D. Student, FIT, Brno University of Technology (BUT)
2008- Co-founder & Head of Software Development, 3Dim Laboratory s.r.o.
2011- Assistant Professor, Researcher, Faculty of Information Technology, Brno University of Technology

Main Works:
• "Collar Line Segments for Fast Odometry Estimation from Velodyne Point Clouds," IEEE Int. Conf. on Robotics and Automation, pp. 4486-4491, 2016.
• "Semi-Autonomous Domestic Service Robots: Evaluation of a User Interface for Remote Manipulation and Navigation With Focus on Effects of Stereoscopic Display," Int. J. of Social Robotics, Vol.2015, No.7, pp. 183-202, 2015.
• "Continuous Plane Detection in Point-cloud Data Based on 3D Hough Transform," J. of Visual Communication and Image Representation, Vol.25, No.1, pp. 86-97, 2013.

Name:
Marcus Mast

Affiliation:
Research Scientist, Information Experience and Design Research Group, Stuttgart Media University

Address:
Nobelstr. 10, 70569 Stuttgart, Germany

Brief Biographical History:
2008- Joined Stuttgart Media University
2014 Received Ph.D. in Cognitive Science
2015- MH Mast User Experience Consulting

Main Works:
• "User-Centered Design of a Dynamic-Autonomy Remote Interaction Concept for Manipulation-Capable Robots to Assist Elderly People in the Home," J. of Human-Robot Interaction, Vol.1, pp. 96-118, 2012.
• "Semi-Autonomous Domestic Service Robots: Evaluation of a User Interface for Remote Manipulation and Navigation with Focus on Effects of Stereoscopic Display," Int. J. of Social Robotics, Vol.7, pp. 183-202, 2015.

Name:
Vítězslav Beran

Affiliation:
Assistant Professor, Centre of Excellence IT4Innovations, Faculty of Information Technology, Brno University of Technology

Address:
Bozetechova 1/2, 612 66 Brno, Czech Republic

Brief Biographical History:
2002-2003 Center for Vision and Speech, Signal and Signal Processing, University of Surrey
2007-2011 Assistant, Faculty of Information Technology, Brno University of Technology
2011- Assistant Professor, Faculty of Information Technology, Brno University of Technology

Main Works:
• "On-line Video Synchronization Based on Visual Vocabularies," Int. J. of Signal and Image Processing, Vol.2010, No.2, p. 7, 2010.
• "Fast and Accurate Plane Segmentation in Depth Maps for Indoor Scenes," IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 1-6, 2012.
• "Simplified Industrial Robot Programming: Effects of Errors on Multimodal Interaction in WoZ experiment," IEEE Int. Symposium on Robot and Human Interactive Communication (RO-MAN), New York, pp. 1-6, 2016.

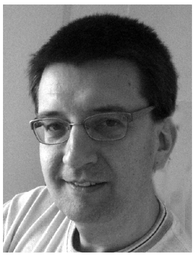
Name:
Florian Weisshardt

Affiliation:
Fraunhofer Institute for Manufacturing Engineering and Automation IPA

Address:
Nobelstr. 12, 70569 Stuttgart, Germany

Brief Biographical History:
2009- Research Associate and Project Manager, Service Robotics Group, Fraunhofer IPA

Main Works:
• F. Weisshardt et al., "Enhancing software portability with a testing and evaluation platform," ISR/Robotik 2014, 2014.
• M. Mast et al., "Semi-Autonomous Domestic Service Robots: Evaluation of a User Interface for Remote Manipulation and Navigation With Focus on Effects of Stereoscopic Display," Int. J. of Social Robotics, Vol.7, No.2, pp. 183-202, 2015.



Name:
Pavel Smrž

Affiliation:
Associate Professor, Centre of Excellence IT4Innovations, Faculty of Information Technology, Brno University of Technology

Address:
Bozetechova 2, 61266 Brno, Czech Republic

Brief Biographical History:
1998- Masaryk University
2006- Brno University of Technology

Main Works:
• L. Polok and P. Smrz, "Increasing Double Precision Throughput on NVIDIA Maxwell GPUs," Proc. of the 24th High Performance Computing Symposium, Pasadena / Los Angeles, pp. 1-8, 2016.

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineers (IEEE)
• American Association for Artificial Intelligence (AAAI)

Name:
Michael Burmester

Affiliation:
Vice Dean Research, Professor for Ergonomics and Usability, Director of the User Experience Research Lab, Speaker of the Information Experience and Design Research Group, Stuttgart Media University

Address:
Nobelstr. 10, D-70569 Stuttgart, Germany

Brief Biographical History:
1992- Fraunhofer Institute for Industrial Engineering IAO
1997- Siemens AG Corporate Technology
2000- User Interface Design GmbH
2002- Professor, Stuttgart Media University

Main Works:
• research on positive user experiences and positive design for work contexts
• usability and user experience method development
• human-robot interaction

Membership in Academic Societies:
• European Association for Cognitive Ergonomics (EACE)
• Professional Member German Gesellschaft für Informatik (GI)
• Association for Computing Machinery (ACM)
• Usability Professionals Association (UPA and GC-UPA)