The EU Framework Programme for Research and Innovation H2020
Research and Innovation Action

Deliverable D6.1 CENTAURO Manipulation Concept

Dissemination Level: Public

Project acronym:	CENTAURO
Project full title:	Robust Mobility and Dexterous Manipulation in Disaster Response by Fullbody Telepresence in a Centaur-like Robot
Grant agreement no.:	644839
Lead beneficiary:	UBO – Rheinische Friedrich-Wilhelms-Universität Bonn
Authors:	Tobias Rodehutskors, Max Schwarz, Giulia Meneghetti, and Sven Behnke
Work package:	WP6 – Manipulation
Date of preparation:	2015-09-30
Type:	Report
Version number:	1.0

Document History

Version	**Date**	**Author**	**Description**
0.1	2015-09-28	Tobias Rodehutskors, Max Schwarz, and Sven Behnke	Initial version
0.2	2015-09-29	Tobias Rodehutskors	Integrated internal review comments
0.3	2015-09-29	Giulia Meneghetti	Improved concept for Task 6.1
1.0	2015-09-30	Tobias Rodehutskors	Submission version

Executive Summary

This deliverable describes the manipulation concept for the CENTAURO system. As the CENTAURO project is an active research project, the content in this deliverable will most likely be subject to change. Nevertheless, this report is necessary to lay out the further path of development and research which will be conducted in Work package 6 of the CENTAURO project. We propose methods for object and workspace perception which will enable the CENTAURO robot to detect and recognize objects in its vicinity that are relevant to the envisioned manipulation tasks and to estimate their pose. The 3D structure of the manipulation work space will be modeled from the robot sensors. Using this representation, we will build a system for collision aware motion generation based on the idea of dynamic potential fields. We will then use this module to assist the operator during telemanipulation by generating meaningful force feedback in the presence of obstacles to guide him around obstacles. We will develop autonomous grasp and motion planning for the CENTAURO robot to reduce the cognitive workload for the operator by executing simple manipulation tasks autonomously under the supervision of the operators. For motion planning, we will extract motion primitives from recorded demonstrations of manipulation tasks performed by the human operator. These motion primitives will be adapted to the current situation and autonomously executed. Furthermore, we will chain motion primitives to sequences to solve simple single-handed and two-handed manipulation tasks, which are similar, but not necessarily identical to the observed ones. This data-driven approach of motion generation based on previous observations will result in human-like behavior of the robot which will reduce the operator workload during supervision of the execution of motions and respect task constraints. Similarly, we are going to record grasps on objects demonstrated by the operator and store them in a database of feasible grasps. These grasps can then be utilized if the same objects are perceived again.

Contents

1 Introduction

This deliverable reports on the conceptual design for the manipulation in the CENTAURO project as specified in Grant Agreement [60]. This deliverable and the corresponding deliverables in the other work packages constitute the specification for how to design the manipulation system and its components. Proper integration requires that the interfaces between the work packages and modules are identified and agreed upon amongst the project partners. Assumptions made about the performance of certain parts of the system made in other parts of the system much be matched.

The objective of this deliverable is to summarize the concept for single- and dual-arm object manipulation with the CENTAURO robot. It belongs to Work Package 6 Manipulation (WP6). The relationship of WP6 to the other work packages within the CENTAURO Project is displayed in Figure 1. Contributions to this deliverable and future work, come primarily from the CENTAURO partners UBO, LIU, IIT, and SSSA. This manipulation concept is a the starting point for the research and development within WP6.

This work package will focus on the development of autonomous single- and dual-arm manipulation capabilities of the CENTAURO platform to support the human operator during manipulation. It complements the teleoperated interaction of the robot with its environment during complex and challenging tasks, which will be realized using the telepresence station developed in Work package 3.

To assist the operator, WP6 will develop means to avoid collisions during teleoperation. To this end, the Momaro robot will monitor its environment with 3D sensors and provide the exoskeleton with information about the distance to obstacles in the vicinity of the robot. These

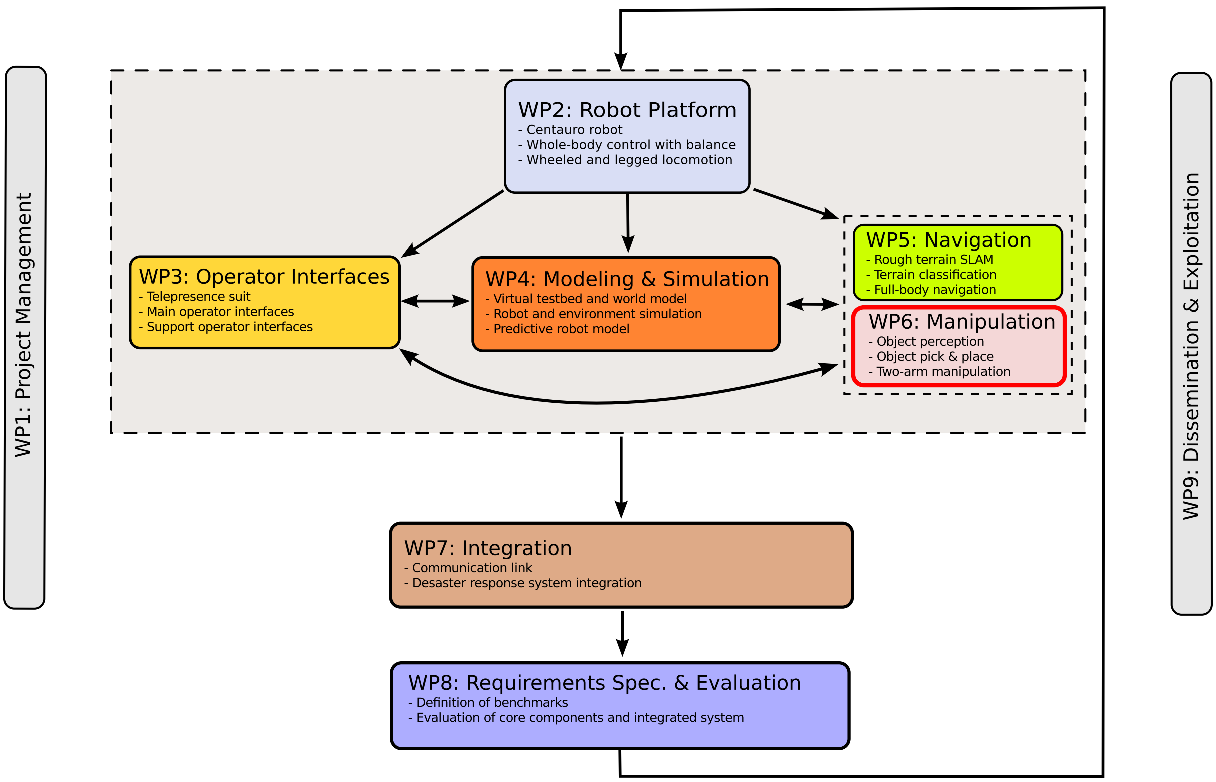

Figure 1: CENTAURO PERT Chart: Dependencies and work flow between Work package 6 and the other work packages within the CENTAURO project. From [60].

information can then be used to render forces onto the exoskeleton to give the operator haptic hints about possible collisions. These hints can also be visually displayed to the operators and sonified.

The autonomous manipulation functionality developed in WP6 aims to relieve the operator from repetitive and tedious tasks like picking and placing objects. To identify objects which can be grasped and to spot obstacles in the vicinity of the CENTAURO robot a detailed object and workspace perception is necessary. The operator should be able to select objects to pick up and the desired location for their placement. The components developed in this work package will plan the necessary collision-free motions and execute them under the supervision of the operator. Planning of autonomous motions should be fast to avoid waiting time for the operators. To achieve this, we will create a knowledge base of solutions for manipulation tasks by employing a data driven-approach that learns from demonstrations recorded during teleoperation of the robot. A big advantage of the CENTAURO system is its capability to generate the necessary training samples for motions during operation with the telepresence system. The learned motion primitives will be adapted and extended to comparable situations. We will also integrate state-of-the art methods for grasp sampling and motion planning based on traditional sampling and trajectory optimization techniques, as a fall back if no correspondence to a motion primitive can be established.

Work package 6 is tightly coupled to Work package 5 – Navigation – as can be seen in Figure 1. This is necessary to extend the reachable manipulation workspace of the robot. If objects, which are outside the current manipulator workspace, should be manipulated, the robot needs to reposition itself with respect to the object. This can be done either teleoperated or autonomously using the functionality developed in WP5. Repositioning might also be considered if no feasible grasps can be found for an object. Furthermore, it might be necessary to adjust the height of the robot by extending or retracting the legs to manipulate objects positioned at different heights. In addition, if heavy objects need to be lifted, a stable stance is required to ensure stability during manipulation.

The objectives of this work package are listed below and the conceptual approach to solve them is presented in Section 4:

- Develop robot perception for bottom-up scene segmentation into objects, for learning 3D models of specific objects, for detecting them and estimating their pose.
- Establish workspace perception for collision-aware motion control.
- Integrate dynamic whole-body motion control into collision-avoiding robot control.
- Implement grasp and motion planning for autonomous single-arm and bimanual object pick and place.
- Develop grasp learning from human demonstrations through telemanipulation.
- Finally, implement autonomous execution of either grasps or motion plans on operator request.

1.1 Inputs

As can be seen in Figure 6, the tasks of Work package 6 are tightly coupled to many tasks within other work packages. Their dependency is described in more detail in Section 4 and Section 5. The inputs needed in this work package are listed in the Table 1. In addition, we require the specification for the interfaces to other modules. A close interaction with WP7 is necessary.

What	When	From Whom
Existing robot model similar to CENTAURO	M6	UBO
Model of the CENTAURO robot	M10	IIT
Sample data from an existing telemanipulation input device	M6	UBO
Definition of the tasks the system should be able to fulfill	M6	LIU
Data of the upper body exoskeleton	M17	SSSA
Access to the simulation environment	M10	RWTH
Detailed local 3D model from local data	M12	KTH
Detailed local 3D model from Centeral World Model	M18	KTH

Table 1: The inputs to the manipulation work package.

1.2 Consumers

The users of the results are listed in Table 2 below. Here we focus on consumers external to this work package.

What	When	To Whom
Collision-avoidance information	M17	SSSA
Grasp hints and motion proposals for integration into the control interface	M17, M27	SSSA
Motion and Grasp planning module	M16, M34	RWTH
Software components for integration	continuously	PGX

Table 2: The external consumers of results from the manipulation work package.

2 Related Work

Telemanipulation systems have been developed for applications that are unaccessible or hazardous for humans such as minimally invasive surgery [3, 23], explosive ordnance disposal [40], and undersea [87] or space applications [56]. Only few research focuses on intuitive dexterous telemanipulation with human-like hands, since dexterous hands and haptic feedback at the hands are delicate to implement. The Japanese telexistence robot Telesar V [15] has an anthropomorphic upper body with dexterous five-finger hands. The setup provides auditory, visual, haptic, and thermal feedback to the operator. The teleoperated robot demonstrates several skills such as pouring the content of one cup into another, stacking cubes, and Japanese calligraphy. The German Aerospace Centre (DLR) developed telepresence in the dual-arm mobile manipulation robot Rollin' Justin [35]. The robot was controlled to handle a cable connector via a bimanual haptic device that is also built from light-weight arms and specialized control devices with haptic feedback for the hands. The modular prosthetic limb (MPL) system [19] is equipped with two human-like arms and five-finger hands. The fingers contain rich sensory equipment to measure force, vibration, fine point contact, and temperature. Hand and body motion of the operator is recorded using data gloves and orientation sensors at wrist and elbow. The operator is provided with feedback in the hands through force-feedback data gloves and vibrotacticle feedback devices for the fingers. In a simplified setup without kinesthetic feedback in the fingers, the robot could be controlled to manipulate bind rails and to handle clothes pegs. In previous work, we combined 3D visualization and tracking of operator head and hand motions to an intuitive interface for bimanual teleoperation [61]. 3D point clouds acquired from the robot were visualized together with a 3D robot model and camera images using a tracked 3D head-mounted display. 6D magnetic trackers capture the operator hand motions which were mapped to the grippers of our two-armed robot Momaro. The system checked for self-collisions and displayed the links which were nearly in collision color-coded to the operator. Furthermore, the system stopped motions which would lead to self-collisions. With this setup, we were able to successfully solve all manipulation tasks encountered during the DARPA Robotics Challenge Finals [36]. Additionally, we demonstrated in lab experiments the ability of the system to successfully connect two flexible hoses. In contrast, the CENTAURO system will be able to provide haptic feedback to the operator and therefore allow more fine grained control for manipulation.

Many groups are working on enabling robots to learn motions from human demonstrations [50, 64, 44, 41]. Often, motions are represented using the well known framework of dynamic movement primitives proposed by Ijspeert et al. [37]. This formalism allows to represent arbitrary demonstrated movements with a set of differential equations. The shape of the represented trajectory can be modified by adding a perturbing force or changing the desired goal, while the trajectory is still stable and its convergence to the goal is guaranteed. Park et al. [54] extended the existing framework of dynamic movement primitives to incorporate obstacle avoidance in movement reproduction of task space trajectories for a 7 DoF robot arm. They used dynamic potential fields around obstacles which exert a repelling force onto the end effector and guide it around obstacles. The resulting trajectories were still stable and reached the desired goal positions. The used dynamic potential field varies its magnitude based on the angle between obstacle and end effector and the velocity of the end effector, which results in smooth trajectory. Reinhart et al. [59] presented a modular architecture for bi-manual skill acquisition from kinesthetic teaching for the humanoid robot iCub. They were able to learn skill specific constraints which arise using two hands on one object due to the closed kinematic chain and demonstrated the generation and execution of different learned skills like paddling and weight lifting. A simple state machine was used to sequence the individually extracted motion primitives to a meaningful motion sequence. Gräve and Behnke presented an integrated approach to identify and recognize human actions and reproduce them in previously unseen situations [20]. For this purpose, they chose manipulation tasks in a table top setting and recorded human motions using a camera based motion tracking system. They proposed a set of task space features to construct probabilistic models of action classes. Their combined segmentation and classification algorithm was able to reliably locate transitions between actions but required a training set of pre-segmented actions. Later work of Gräve and Behnke combined learning from human demonstrations with reinforcement learning for sequential tasks [21] and task hierarchies [22].

The premise of telemanipulation systems is to utilize the cognitive capabilities of a human operator to solve dexterous manipulation tasks such as object pick-and-place, opening a door, or tool-use. However, this approach also leaves significant load to the operator. Implementing the required cognitive capabilities in autonomous systems in general is an open research topic. Many approaches to grasp and motion planning in unstructured environments assume that the geometry of objects is known and identify a set of stable grasps on objects in an offline phase [47, 8, 52]. Stability is often measured by criteria of form closure [42] or in the grasp wrench space [39]. To grasp the objects in the actual scene, the grasp set is pruned by identifying those grasps that are reachable under kinematic and free-space constraints. In [52], UBO developed mobile bin picking with an anthropomorphic service robot. Objects are perceived as compounds of geometric shape primitives, and grasps are sampled on the shape primitives and pruned in a multi-stage process. Hsiao *et al.* [34] proposed a reactive approach to grasping that plans grasps at raw partial 3D measurements of objects, evaluating the resulting grasps for reachability by time-costly motion planning. In own work at UBO [78], this approach has been extended to provide grasps and reaching motions very fast in situations, where parametric arm motion primitives can be used instead of costly planning of reaching motions. Goldfeder *et al.* [18] propose to store grasps on specific 3D object geometries in a database and to retrieve them for partial scans of the objects. Some approaches have been proposed to bimanual grasp and motion planning, mainly in the field of humanoid robots [85, 49]. Vahrenkamp *et al.* [85] find grasp candidates for each arm using the medial axis transform on the object geometry. They score bimanual grasps by the manipulability of the arms, i.e., they quantify how movable the arms are at the grasp. Mohan *et al.* [49] formulate motion generation as attractor dynamics subject to external forces that implement the task.

Collision avoidance in unstructured environments requires methods for perceiving free, unknown, and occupied space in the local vicinity of the robot. Vahrenkamp *et al.* [84] represent the environment by geometric shape primitives and apply lazy collision checking and enlarged robot models to speed up collision checks. Hornung *et al.* [32] integrate measurements of a laser scanner and textured-light stereo cameras into sparse 3D occupancy maps that are represented in octrees. Sensor motion is determined from joint position encoders and wheel odometry.

The lead beneficiary for this work package (UBO) has extensive experience in autonomous manipulation. Fig. 2 shows some of the manipulation robots developed at University of Bonn. Manipulation scenes can be segmented visually into object candidates through bottom-up cues without explicit knowledge about specific objects. For instance, Holz *et al.* segmented depth images of a time-of-flight camera into geometric shape primitives using a fast multi-resolution RANSAC approach [30]. This approach was extended by Berner et al. [9] to detect compounds of 3D and 2D shape primitives. Holz et al. [29] proposed fast methods for the segmentation of planes in RGB-D images. Rusu *et al.* [62] proposed to describe and classify the local context of points by Fast Point Feature Histograms. They regularize the classification in a conditional random field on the point cloud to obtain coherent object segments. Holz and Behnke [28] represent depth images in a triangular mesh and apply fast curvature-based segmentation.

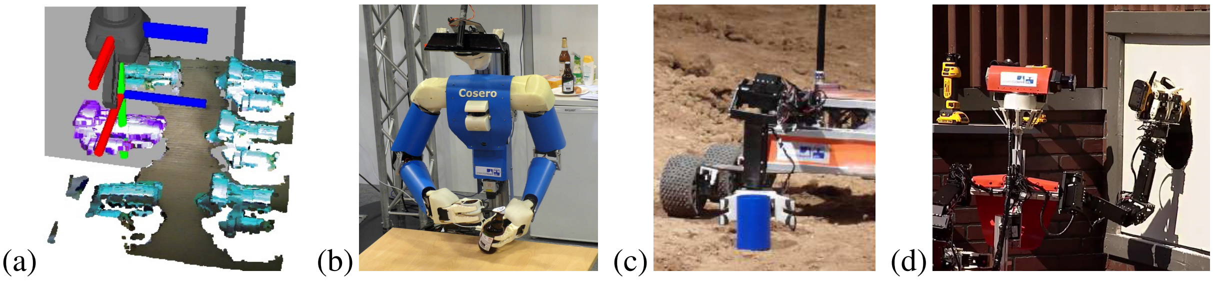

Figure 2: Example manipulation robots at University of Bonn: (a) Depalettizing with Universal Robots UR10 manipulator and Robotiq three-finger hand [31]; (b) Bottle opening with cognitive service robot Cosero [74]; (c) Grasping a water-filled mug with space robot Explorer [77]; (d) Cutting drywall with mobile manipulation robot Momaro [61].

A strong segmentation cue is common motion. Stückler and Behnke [75] proposed an efficient method for segmenting RGB-D video into moving rigid bodies. They used observed common motion to infer object hierarchies [73].

Keypoint-based approaches have been a leap forward in the recognition and pose estimation of specific objects (e.g. [43]). However, keypoints require textured objects. Recent approaches overcome this drawback by exploiting intensity edges [26, 11] or geometric shape, if dense depth is available. Hinterstoisser *et al.* [26] propose a real-time object template detection approach based on dominant color gradients and surface normals. Different view poses onto the object need to be trained as separate templates to implement pose estimation. Choi *et al.* [11] extract 2D edge templates in several view poses from a 3D mesh model of an object. The edge templates are detected in gray-scale images and tracked using a particle filter in real-time. To estimate the pose of objects in depth images, several variants of the Fast Point Feature Histogram have been proposed [62]. Signatures of histograms of orientations (SHOT) describe local shape and color context at 3D key points [82]. The method determines a local 3D orientation at the keypoints to define a reference frame for context description and pose voting. Each keypoint votes for a relative pose of the object in a Hough voting scheme. Drost *et al.* [14] proposed a method to match meshes that borrows concepts from the Generalized Hough Transform. They build surface element (surfel) pairs and associate them efficiently with hash maps. Since each surfel-pair defines a unique reference frame, pose can be estimated from surfel pair correspondences through Hough voting. The method has been extended by contours [12] and color [10]. Stueckler and Behnke [72] also propose multi-resolution surfel maps for 3D modeling and real-time tracking of objects. Multi-resolution surfel maps represent 3D point clouds and RGB-D measurements by shape and color distributions in voxels within an octree. The maps can be efficiently aggregated from RGB-D images and support real-time registration. Full-view models of objects are obtained through simultaneous localization and mapping (SLAM) to which then RGB-D images are registered in real-time to estimate object pose. This approach has been applied for SLAM in indoor scenes [71]. It was extended by McElhone et al. [46] to the joint detection and tracking of objects. Other recent methods for object proposal exploit assumptions made on the map structure, on appearance in RBG images, on motion fields or on a combination of the information coming from these modalities [24, 86, 38, 88].

In recent years, deep learning approaches [5, 65] are the leading methods for many pattern recognition tasks. In previous work, we applied them successfully to image denoising [4], image categorization [6, 63, 83, 69], object detection [7, 67], and object-class segmentation [66, 27, 51, 55]. Another popular method for object-class segmentations are random forests, which we implemented on GPU [68]. By aggregating semantic segmentations in 3D, semantic maps can be created [80].

3 Progress in CENTAURO beyond the State-of-the-Art

In contrast to previous work on telemanipulation systems, the CENTAURO system will provide full kinesthetic feedback for the upper body of the main operator. The CENTAURO robot will be able to perform dexterous dual-arm manipulation skills under direct control of the main operator as well as elementary tasks semi-autonomously under the supervision of the operators. While elementary autonomous skills such as picking and placing objects with one and two arms will be implemented using grasp and motion planning, grasps for complex capabilities like door-opening or tool-use will be learned from human demonstrations that are given through the telemanipulation interface. CENTAURO will create the foundations for teaching robots dexterous dual-arm manipulation through direct human demonstration. The planned or learned grasp and motion strategies will support the operators in decision-making by displaying suggested grasps and motions using augmented reality techniques, based on object and workspace perception from robot sensors. While the operators could perform the task themselves, they may also let the robot execute suggestions autonomously. For manipulating objects, stability of the CENTAURO robot will be established through dynamic whole-body control of CENTAURO's compliant robot mechanism. We pursue two means for planning grasps on the object. If no example grasps are known for the object, for instance, if the robot shall move debris, grasps will be planned on the measured 3D shape of the object. We will also build a grasp database for specific objects such as door handles or tools. These grasps are learned by the robot from human demonstrations through telemanipulation. In contrast to many previous approaches to learning from human demonstrations, our approach does not require a mapping from the human body to the robot kinematics, since the robot directly experiences the grasps in its own body.

To examine the reachability of grasps and to execute them, we will generate a direct obstacle-avoiding reach towards the grasps through simulation of the whole-body dynamics. Learned motions from previous demonstrations will be adapted to the current situation perceived by the robot. Furthermore, new motion sequences will be generated automatically by meaningful combination of known motions. Telemanipulation will handle more complex manipulation scenarios that would require more sophisticated task and motion planning.

We pursue novel means to perceive the robot workspace that consolidate research in workspace perception with state-of-the-art mapping and motion estimation, investigating the possibility to exploit a combination of 2D , 2.5D and 3D sensors. We will augment the maps with information about free, unknown, and occupied space through signed distance to occupied voxels. The map representation supports fast volume queries for obstacle-avoiding motion generation. Live images and scans will be efficiently aligned within a single multi-resolution occupancy map of the local surrounding of the robot thus improving over time the detail level of the map as well as the local egomotion estimates using state-of-the-art simultaneous localization and mapping (SLAM) approaches and bundle adjustment techniques.

For perceiving objects, we will extend existing methods for object proposals and classification, aiming at exploitation of the multimodal information (RGB, depth, occupancy and motion) available through our workspace map. Furthermore, we will investigate the advantages and disadvantages of alternative methods for object pose estimation such as fine-grained classification by means of an extensive model dataset or fitting of a generic deformable model describing the whole object class.

4 Manipulation Concept

4.1 Object and Workspace Perception (Task 6.1)

A high-quality, intuitive visualization of the CENTAURO robot surroundings supports the operator to better understand the disaster area. The variety of disaster scenarios and manipulation tasks that the system will face implies the need to follow a dynamic approach in the process of creating spatial maps. As described in the Deliverable 5.1 Sec. 4.2, a representation on three layers (or levels) of details will be investigated and proposed for the CENTAURO system to address the needs of different tasks.

The workspace for manipulation tasks will be mapped as a highly detailed 3D model (or *occupancy map*) catering to the needs of such a high precision task. The intended idea is to have a dense structure estimation pipeline at very close range. The maximum resolution of the maps gradually decreases with the distance from the sensor. In this way, measurement principles and noise properties of typical sensors such as cameras, RGB-D cameras and/or 3D laser scanner are considered, and processing is focused on measurable detail. The maps maintain information in voxels about free, occupied, and unknown space through signed distance towards the sampled surface (positive: free, negative: unknown).

This form of occupancy map provides detailed information that can be exploited for collision avoidance when moving in the workspace or when manipulating objects. However, when the task does not require high precision, collision detection can be performed efficiently by enclosing occupied space in simple shapes such as spheres or parallelepipedes (*bounding spheres/boxes*). These can be derived from the appropriate level of the map depending on the accuracy required by the task.

As described in D5.1, this highly detailed occupancy map is created on demand, i.e., when the robot needs to perform a manipulation task, it makes use of active sensing to acquire the views needed to build a detailed model. The actual resolution/accuracy of the model will vary from millimeters to centimeters depending on the needs and the sensor limits.

Once an accurate map of the workspace is built, a step of object identification and tracking can follow with the purpose of aiding the operator in understanding the scene (teleoperation mode) or allowing the robot to perform selected manipulation tasks autonomously. In the case of autonomous mode, this step becomes of fundamental importance.

Object identification and tracking will be handled in two ways, depending on whether an object belongs to a known class (for example a tool) or is to be treated as generic object in the context of the manipulation task (for example some rubble on the workspace). Also a transition between the two cases is possible (e.g. generic model of a door handle).

The handling of objects within the workspace begins with the identification of the presence of an object in the constructed map and in the relative RGB(-D) images. Object hypotheses will be derived from local regularity assumptions made on the map structure (objects lying on a plane) [45, 24, 2], on appearance (color and/or depth discontinuities) [86, 38, 1] or motion (when interacted with, objects do not follow the static scene assumption, i.e. they can be identified as entities moving in the workspace when touched and interacted with) [25, 16, 88].

When an object is detected in the scene, its appearance and shape can be tested within a classification framework to identify whether the object belongs to a known class (for example a set of tools the robot knows how to handle) or not. In the latter case, the object is classified as an unknown object, in the former the object is identified as either a tool or a known element of the environment (for example a door handle).

To facilitate grasping a tool or other objects with task constraints, its pose must be identified.

For this purpose, the perceived appearance and 3D shape of the object is registered with a previously learned model so that its pose can be identified with respect to a standard frame. This can be achieved by classifying the object as a specific instance of the class chosen from a large dataset of detailed models or by fitting a generic, deformable model to the perceived instance [57]. The actual method to implement is an open problem and it will be chosen taking into account complexity, flexibility and and accessibility of the required data.

With the aim to facilitate motion planning and grasping, the maps shall support fast collision checking with bounding boxes and spheres derived from the detected objects as well as obstacles. The specifics of how collision detection and avoidance will be handled are described in the following sections.

4.2 Collision-aware Motion Generation (Task 6.2)

Task 6.2 develops collision-aware motion control for use during telemanipulation as well as autonomous manipulation. It will consider self collision avoidance as well as environment collision avoidance. This task depends on Task 6.1 which implements the perception of objects in the workspace and generates a workspace map which contains information about occupied and free regions that will be used in this task to calculate distances and potential collisions between the robot and its environment. The results of this task will be used in Task 6.3, 6.4, and 6.5 for autonomous planning and execution of single-arm as well as bimanual motions. Furthermore, the results of this task will be used for control of the robot during telemanipulation by displaying potential collisions to the main operator through the kinesthetic feedback of the exoskeleton.

The Flexible Collision Library (FCL) [53] will be used for collision checking. It supports collision detection, distance computation, and supplies contact information if objects are in collision with one another. It supports checks between different shape primitives (sphere, box, cone, cylinder) as well as meshes and octrees, which are represented using the OctoMap library [33]. The FCL is also integrated into ROS through the MoveIt![1] package. Even though the FCL is capable of computing distances between different objects, this functionality is not fully available through the interface of the MoveIt! package yet. Thus, this interface needs to be adapted to fit our needs.

In order to physically interact with the environment through the whole robot body, telemanipulation needs to support a mode which allows physical interaction between the robot and its environment. Then again, it is difficult for the operator to continuously monitor collisions for all body parts. To assist the operator in cases that do only require physical interaction of robot hands and feet with the world, a collision-avoiding mode for the remaining body parts will be developed. This will be implemented by the specification of a symmetrical matrix for all body parts (e.g. left arm, front right leg) and the environment, which defines if

(A) both objects can be in collision,

(B) both objects are not allowed to be in collision, or

(C) the collision between both objects is not checked at all to save computing time.

An example for such a matrix might look like this:

[1] http://moveit.ros.org/

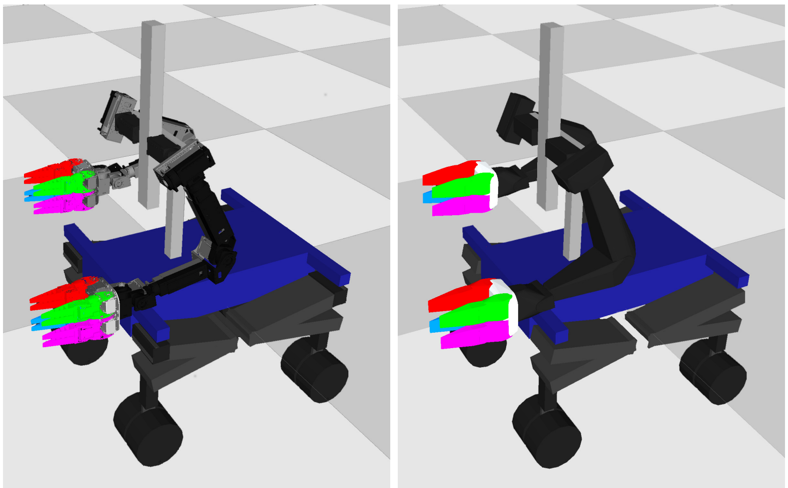

Figure 3: Left: Detailed rendering of the Momaro robot. Right: Fine collision model of the same robot to ease collision checking based on the convex hull of its limbs.

	Left Hand	Right Hand	Left Arm	...	Environment
Left Hand		A	B		A
Right Hand	A		B		A
Left Arm	B	B			B
Back Left Leg	C	C	C		A
...					
Environment	A	A	B		

This matrix would allow interaction between the hands of the robot and its environment, but checks for collisions of the robot arms with the hands and the environment. Furthermore, collisions between the back left leg and and the hands and arms are not checked at all, as these robot parts cannot collide during typical tasks. These matrices can be swapped during operation of the CENTAURO robot w.r.t. the current task.

As the collision checking module is not only used during execution of motions on the robot but also for motion planning, different abstractions of the robot model can be used for collision checking depending on the current requirements regarding available computing time. For planning of motions and simulation of planned motions in simulation prior to the execution on the real robot, a coarse and conservative collision shape of the robot can be employed as the time between the beginning of the planning and the actual execution of the motion should be minimized. More costly collision checking using a finer collision model can be used during the actual execution of the motion as more time is available. This might prove useful, as the results of this checking are more accurate. Hence, we plan to provide two collision models based on the actual robot model:

- A **fine collision model** which is based on the convex hull of the exact robot model. This model can be used for collision avoidance computation during (teleoperated) motion execution on the robot (see Fig. 3).

- A **coarse collision model** which simplifies the coarse model by representing robot body parts by locally circumscribing shape primitives such as spheres and cubes to ease col-

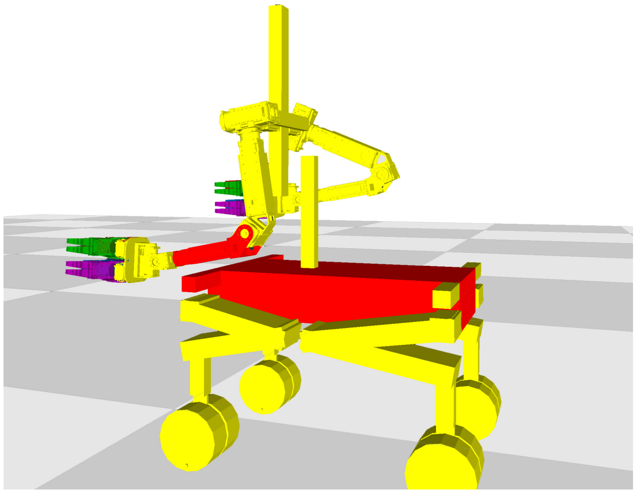

Figure 4: Rendering of the robot model for the support operator indicating close proximity between the lower left arm and the base.

lision checking. This approach has been used by Dietrich et al. [13] to realize collision avoidance with a sampling time of 1 ms using a sphere approximation of their robot Rollin' Justin.

Obstacle avoidance will then be implemented through virtual forces that act from the found obstacles onto the robot limbs. These forces are used for collision-avoiding control of the robot as well as for displaying potential collisions to the main operator through the kinesthetic feedback of the exoskeleton. For each limb of the upper body of the robot, the distance to all other limbs and obstacles within the local workspace will be calculated based on the previously defined collision matrix. If the distance between a given pair is lower than a given threshold, the direction and distance of this pair will be used to create a potential field which generates a repulsive force between these potentially colliding parts. This information will also be sent to the operator interface to render forces through the kinesthetic feedback of the exoskeleton to the operator. Furthermore, it can also be visually displayed to the main and support operators, e.g. by coloring the limbs of the rendered robot model according to the nearness of the given robot limb to an obstacles (see Fig. 4) or sonified, e.g. by beeping with increased frequency when the distance to obstacles decreases.

4.3 Grasp and Motion Planning for Single-arm Object Pick-and-Place (Task 6.3)

Task 6.3 develops an assistance module, which helps the operators in performing basic pick-and-place tasks using a single arm by semi-autonomously finding suitable grasps and executing motion primitives fitted for the current situation. This will reduce the operator workload for simple and repetitive tasks. The required motion primitives will be either designed by the developer of manipulation planning or will be extracted from previous demonstrations of similar motions, which have been executed by the human operator using the exoskeleton. A big advantage of the CENTAURO system is that these demonstrations can be recorded during normal operation of the system and therefore allow the system to improve its capabilities over time. Finding appropriate grasps on objects is necessary for the majority of manipulation problems. To this end, the module developed in this task will contain two grasp planners. The first one will be a state-of-the-art grasp planner, which can offer grasps on any kind of object measurable by the robot sensors. The second generator will learn grasps from telemanipulation demonstrations. This task depends heavily on the recognition of objects developed in Task 6.1 and the recording of the motions of the operator wearing the exoskeleton, which will be developed in Work package 3. Its results will be used in Task 6.5 to allow the autonomous execution of motions and also in the operator interface developed in Task 3.5 to display possible grasps to the operator.

4.3.1 Grasp Planning

This module will contain two grasp generators. The first one will be a state-of-the-art grasp planner, which can offer grasps on any kind of objects measurable by the robot sensors using heuristics. The second generator will query a grasp database of known grasps for the object which should be grasped. This database will be filled by previously observed grasps during telemanipulation. Grasps from the learning generator are expected to be more useful for specific grasps occurring during tool use or other interaction motions, while the grasp planner will be useful for previously unknown objects or when the object merely needs to be carried, not used. The grasps candidates generated by both generators will be displayed to the operator by the user interface developed in Task 3.5. for evaluation and selection. If a grasp is selected, the robot will execute it autonomously under the supervision of the operators using the motion planner described in 4.3.2. The next two paragraphs will explain the two grasp planners in more detail.

Grasp Sampling This grasp planner will sample grasps for perceived objects following a state-of-the-art approach such as medial axis transform (MAT) based grasp sampling [58] or primitive-based sampling [79] which we have used in previous work. These methods will generate multiple grasp candidates for each object, which need to be ranked by their applicability. A grasp is defined by the pose of the end effector relative to the perceived object at which the fingers of the hand are closed. Additionally, a pre-grasp pose is automatically derived from the sampled grasp pose, to allow a safe approach of the object by the gripper.

Several checks are performed on each grasp candidate to discard unfeasible grasp and to evaluate grasp quality:

- The reachability of the grasp pose and pre-grasp pose will be checked using a precomputed reachability map, which is a look-up table for the existence of valid IK solutions. If either the grasp pose or the pre-grasp pose is not reachable from the robot's current position, the grasp candidate is discarded.
- If valid IK solutions have been found for the grasp and pre-grasp pose, these configurations are checked for collisions with the environment. If either produces a collision, this grasp is discarded.
- The path between the pre-grasp and grasp pose of the grasp candidate is checked for collisions between the robot hand and the workspace map. If the path is not collision-free, the grasp is discarded.
- Next, a full whole-body simulation of the reaching motion with collision avoidance will be performed using the motion planning described below. If the simulated execution of the motion is not successful, e.g. due to insufficient balance of the robot, the grasp is discarded.

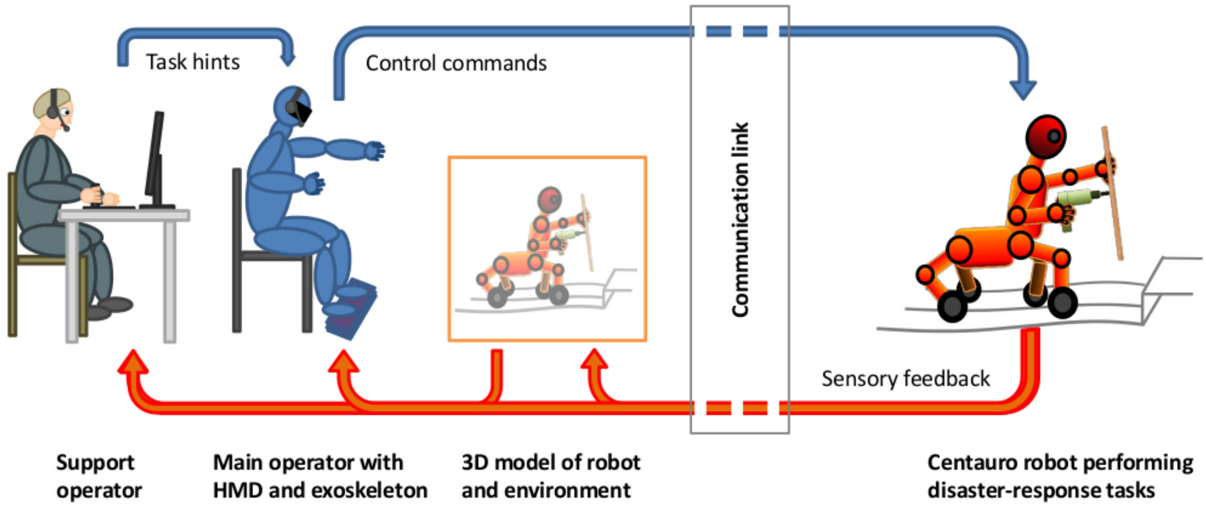

Figure 5: Control commands from the main operator as well as the resulting sensory information (e.g. measured interaction forces) of the robot will be recorded to solve tasks autonomously by learning from demonstrations.

The resulting grasps, which passes these tests, will be ranked according to their properties e.g. by checking how much the object is caged to estimate the stability of the grasp and the estimated energy consumption to reach this specific grasp. If no grasp passes these tests, the system will indicate a problem to the operator and rely on the operator to resolve the situation.

Grasp Database While telemanipulation via the operators will be required for more complex, previously unknown tasks, the CENTAURO system should be able to learn continuously from the demonstrations recorded during telemanipulation and offer an autonomous alternative the next time the task is encountered. To this end, we will build a grasp database for specific objects such as door hinges or tools that are tracked using the perception modules developed in Task 6.1. Grasps demonstrated during telemanipulation will be represented relative to the local reference frame of the object and made persistent in the grasp database. Control commands and sensory feedback of the robot (see Fig. 5), e.g. the interaction forces between the fingers of the robot hand and the object, during the teleoperated grasp will be recorded and stored in the database along with the grasp pose. If the object is perceived again, grasps of the objects will be retrieved from the database and transformed according to the actual pose of the object. The object is non-rigidly registered to the learned template. While methods for non-rigid instance-instance registration of objects exist (e.g. [76]), we would like to employ non-rigid instance-category registration. To this end, the template would be represented as the distribution over all observed object instances in a given category. The non-rigid registration allows skill transfer to the currently observed object. The resulting grasp candidates will be checked for feasibility using the same checks described above. Resulting feasible grasps will be provided to the control station to be suggested to the support operator in Task 3.5. If a grasp is selected and executed, the system will try to mimic the previously measured interaction forces of the fingers with the object while grasping. Hence, lightweight or fragile objects as well as heavy objects can be grasped safely without the risk of crushing or dropping a grasped object.

If no feasible grasp can be retrieved from the grasp database, the system will try to sample an appropriate grasp using the method mentioned above.

4.3.2 Motion Planning

Human pick and place motions are similar for recurring tasks, suggesting that humans do not plan these motions from scratch each time a task of this kind needs to be solved. We want to adapt this idea for motion planning and avoid planning from scratch for tasks which have already been solved. Instead, we want to rely on previous solutions and adapt them to the current situation. To this end, we will record the control commands issued by the operator wearing the exoskeleton during pick and place tasks. We will extract motion primitives based on these recordings, which we have already done in previous work [20]. In contrast to our previous work, we will rely on a more general probabilistic approach which is described in Section 4.5. If similar pick and place tasks need to be solved, the extracted motion primitives will be adapted to the new situation and executed. This approach will result in more human-like movements than planning from scratch, which will make it easier for the human operator to observe the executed motions and to assess their utility, thus reducing the cognitive workload for supervision of the system. The demonstrated motions will implicitly also encode task constrains, such as holding open fluid containers upright.

Motions will be represented as 6D (position and orientation) trajectories of the end effector in task space, encoded relative to the starting pose of the motion and the ending pose, e.g. the object which should be manipulated. They will also contain information on how the redundancy of the manipulator is resolved. Additionally, obstacles in the vicinity of the object will also be incorporated to help explaining deviations from direct paths. Furthermore, the interaction forces measured during the motion will be recorded as well. During an adapted execution of the motion primitive in a similar situation, the system will try to mimic the force it encountered during the recording. This could be useful, e.g., to pick up a small object from a table by pressing the finger onto the table with similar force to previously recorded demonstrations and then closing the fingers. Additionally, this would allow for learning the interaction forces which are necessary for certain manipulation tasks such as turning valves. To interact with the environment, the motion generator will use the compliant control of the robot developed in Task 2.3.

To avoid collisions during the execution of adapted motion primitives, motions will be executed under the influence of a repelling force based on Task 6.2. The end effector will be repelled from obstacles in its vicinity and thereby deviate from the intended motion of the motion primitive. To do this, a dynamic potential field [54] is used based on the distance calculations performed in Task 6.2. The magnitude of the potential field increases with the speed of the end effector and decreases with the angle between the current velocity direction of the end effector and the direction towards the obstacle, thus allowing for smoother trajectories. Reaching and retracting motions will be simulated using a full whole-body simulation to access their feasibility prior to their actual execution on the CENTAURO robot. For the retracting section during grasp motions, the grasped object is virtually attached to the end effector for collision checking. If the simulated execution fails or no suitable motion primitive is available for the current situation, the system can ask the operator the perform a demonstration for the task which will be used the next time a similar situation occurs.

As a fallback system, we will also integrate motion planning using state-of-the-art sampling based methods provided by the Open Motion Planning Library [81] which is available in the ROS ecosystem trough the previously mentioned MoveIt! plug-in. Furthermore, state of the art optimization based motion planning algorithms like STOMP which we already used in previous work [70], can be integrated as a fallback in case the motions generated from the demonstrations are not sufficient. This might be the case if not enough demonstrations were provided to the system for the situation at hand.

As the generated motions are represented as position and orientation of the end effector in the workspace, the redundancy resolution of the robot arm has to be resolved independently. For this purpose, we are going to use a gradient projection method which projects the gradient of a cost function into the nullspace of the robot arm similar to the methods embodied by Gienger et al. [17]. The cost function will be based on the distances which are defined in Task 6.2 to guide the limbs of the robot arm away from obstacles. If collision avoidance is not possible using only nullspace optimization, the system will also be able to change the desired pose of the end effector and deviate from the planned end effector trajectory.

4.4 Grasp and Motion Planning for Bimanual Object Pick-and-Place (Task 6.4)

This task extends the work of Task 6.3 to grasp and motion planning for both arms simultaneously. Bimanual grasping is typically required for objects, which are too large or too heavy to grasp with one arm. It might also be required for more complex manipulation tasks (e.g. opening a container). Furthermore, the strength of two arms might be required for manipulating objects that require larger interaction forces. e.g. a stuck valve.

Grasp candidates will be generated as explained in section 4.3.1. The difference to single-armed manipulation is now that two reachable, non-colliding grasps on the object are required. Concurrent grasps for both arms are additionally stored in the grasp database.

The system will—similar to Task 6.3—be able to learn bimanual motions from demonstrations. Both arms will be treated in the same way. In addition to the individual arm motions, constraints between the end-effectors, such as their distance or interaction forces will be recorded. The motions will be represented in task space relative to the relevant object as in Task 6.3. As the motion of this closed kinematic chain is overconstrained, the compliance of the robot will be used to resolve potential conflicts. The same collision avoidance technique as in Task 6.3 will be used, but now the potential fields will affect the motion of both end effectors at the same time, with the effects mediated by the simultaneously grasped object.

4.5 Autonomous Execution of Manipulation Commands (Task 6.5)

The CENTAURO system will offer grasp candidates generated in Task 6.3 and Task 6.4 for objects in the immediate vicinity of the robot to the operator in real-time. A visual display (e.g. a transparent model of the end effector) will show the grasp poses, so that the operator can decide whether a generated grasp is feasible and useful. Details of the actual interface will be defined in Task 3.5. As soon as the operator triggers a grasp, the planned grasp motion will be executed using the collision-aware dynamic whole body control. If the robot is near collision during the motion, execution will be stopped and the operator will be notified about the problem and which parts of the robot are in danger of collision (see Fig. 4).

While grasp planning and motion planning for pick and place tasks is important for a large part of manipulation, we will also investigate possibilities for reducing the operator load by offering assistance during other manipulation tasks (e.g. turning valves, pushing objects, etc.). Because the space of possible manipulation actions is very large, we plan to use a probabilistic approach. Recent works in the field of computer animation, for example Motion Graphs++ [48], can generate new motions from a motion database, subject to kinematic and semantic constrains. Motions recorded during teleoperation will be segmented and inserted into the motion graph. Therefore, we will develop a method for segmenting demonstrated motions into short clips between keyframes. These keyframes will be used for parametrization of the motions prior to their execution. Keyframes will be placed automatically at key points during the motion, e.g. at contact changes or stationary points. Another possibility to detect segment boarders is based on the recognition of velocity zero-crossings [59] or zero-crossings of acceleration. New motions can be generated using a graph search in the motion graph and chaining the motions along the resulting path together. This will enable the system to generate new motions for similar situations by combining the available motion primitives in a meaningful way. Key prerequisite for this is the establishment of correspondences between the known models and the current situation via perception of the manipulated objects in Task 6.1. The motions can then be warped to the situation at hand in a non-rigid way.

5 Implementation Plan

This section provides a plan for the implementation of the manipulation concept. The implementation of the different tasks will not be independent from one another. Instead, the tasks of this work package depend on each other and there are also dependencies to tasks of other work packages, as can be seen in Figure 6. As the implementation is an iterative process, interfaces defined between different tasks may be subject to change. We will constantly integrate the individual components into working subsystems to identify weak points in the implementation by regularly evaluating the performance of subcomponents w.r.t. to the tasks which should be tackled by the CENTAURO system. This will allow us to identify weak points of the system and to take counter measurements to mitigate possible impacts on the whole project. To ensure this, evaluation of the components developed in this work package will take place in close coordination with the partners leading the other work packages.

UBO has the most person months (30 PM out of 54 PM) in Work package 6 and will lead the implementation of Tasks 6.2–6.5. It will be supported by SSSA (6 PM) and IIT (6 PM) in Tasks 6.2, 6.3 and 6.4. LIU has the second highest number of person months (12 PM) in Work package 6 and will lead the implementation of Task 6.1. LIU will be supported by UBO during this task.

The goal of this work package is to integrate components to a first working system as fast as possible and extend it successively. To make this possible, we will rely on place holder components and already available software until the respective components are available. As in the whole CENTAURO project, ROS will be used as a middleware. It allows to independently

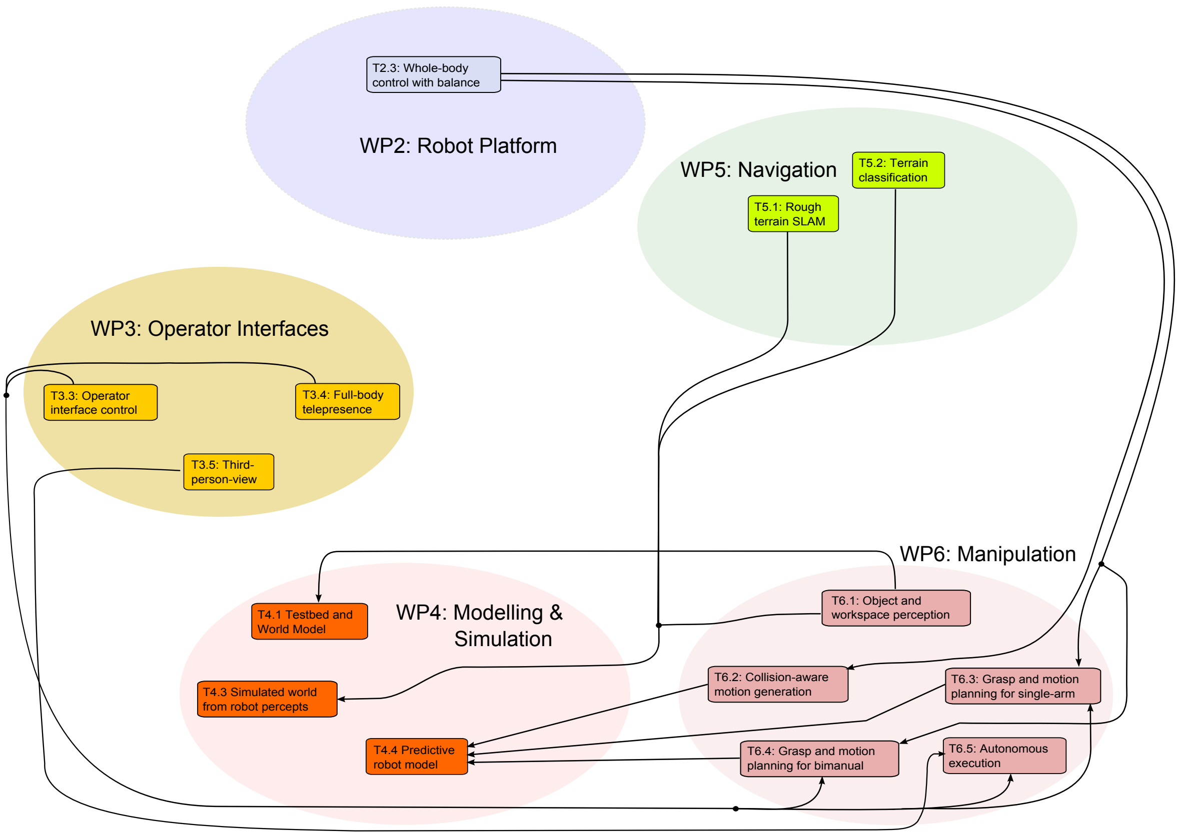

Figure 6: Interaction of tasks of Work package 6 with tasks of the other work packages.

implement components, which are loosely coupled to one another. Furthermore, ROS allows easy definition of interfaces between different components and therefore enables us to easily exchange components during development. To speed up the development, we will use the ROS plug-in MoveIt! for the implementation of the motion planning system. It already provides sampling based planners through the Open Motion Planning Library, which will be integrated in this work package as a fallback system. Furthermore, we will use the Flexible Collision Library for collision detection and distance computations and the OctoMap Library for the representation of the workspace.

As the CENTAURO robot is not available at the beginning of the development, we will make use of replacement components during the implementation of this work package. At the beginning of the development phase, the simulation environment envisioned in the CENTAURO project will neither be available, therefore development will start based on the software components already available through ROS, e.g. Gazebo as a simulator. As soon as the simulation environment with the model of the CENTAURO robot is provided by RWTH, we will adapt the developed techniques to this simulation. While simulation is heavily used during development to facilitate the implementation and integration process, it will be necessary to use real hardware as soon as possible to make sure that the software components are capable of handling realistic scenarios. To this end, we will use our mobile manipulation robot Momaro as a surrogate robot platform, which has a similar configuration as the CENTAURO robot. The individual software components developed in this work package will be designed to be easily portable from the Momaro robot to the CENTAURO robot.

Figure 7 shows a rough estimate for the schedule of the implementation of Work package 6. LIU will start the development process of Task 6.1 early in the project, but its results may not be directly usable for the implementation of collision-aware motion generation, since this task will be started in parallel by UBO. Thus, the implementation of Task 6.2 will initially be based on artificial input. As soon as first results of Task 6.1 are available, both tasks will be evaluated in conjunction. The results of Task 6.2 will be used by SSSA to give the operator feedback over obstacle in the vicinity of the robot.

As the generation of motions based on previous demonstrations is an open research topic and its applicability and reliability cannot be guaranteed, we will start by implementing and integrating the fallback motion planning methods based on already available state-of-the-art sampling algorithms. This ensures that motion planning for pick and place tasks will be available regardless of the outcome of the research and can be integrated as a core component in MS2 in M17. Similarly, we will start by integrating grasp planning used in our previous work [79] and evaluate its performance on objects relevant for the manipulation tasks envisioned for the CENTAURO system.

As soon as the fallback method is integrated, we will start the development of motion generation based on previously recorded human demonstrations. As the exoskeleton is not available

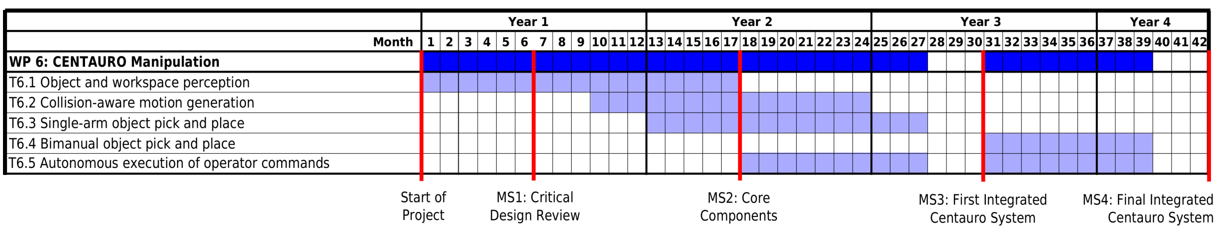

Figure 7: Gantt chart of the tasks of Work package 6.

at the start of the development process, we will use 6D magnetic trackers to track the hand motions of the operator during demonstrations, as we already did in previous work [61]. In contrast to the CENTAURO exoskeleton, which will measure the joint angles of the operator and map those to the joints of the CENTAURO robot, the currently available magnetic trackers only measure the position and orientation of the operator hands in the workspace. Nevertheless, as the motions extracted in Task 6.3 and 6.4 will be represented in task space, this will not pose a problem during development and porting the solution to the exoskeleton should not require major changes. Since the Momaro platform is—in contrast to the CENTAURO robot—not equipped with 6D force sensors in its wrists, we will adapt the developed motion generation framework to be able to handle forces and reply them during execution at a later time.

After the development of single-arm motions, which will be available for integration in M27, we will extend the developed methods to bimanual capabilities for the Final Integrated CENTAURO System.

In parallel to the development of Task 6.3 and Task 6.4, we will start to integrate the autonomous execution of operator commands. This tasks will be done in close collaboration with SSSA, which is responsible for the operator interface.

6 Conclusions

In this deliverable we have presented the manipulation concept for the CENTAURO robot.

We are going to implement workspace and object perception to recognize obstacles in the vicinity of the robot and to identify objects which can be manipulated by the robot. Based on this perception, we will develop collision-aware motion generation for the CENTAURO robot. This will be based on potential fields which will produce repelling forces onto the robot limbs. These forces can also be rendered as kinesthetic feedback to the exoskeleton to make the operator aware of obstacles. Furthermore, we are going to develop grasp and motion planning to enable the robot to autonomously execute simple manipulation tasks on operator request. Our approach to grasp and motion planning is data driven and we will build a database of grasps for known objects as well as a primitives of arm motions. This grasp and motion primitive database will be filled by example grasps and example motions generated by the operator controlling the CENTAURO robot by means of the exoskeleton. The grasps and motion primitives will be instantiated in similar situations by adapting them according to the work space and object perception and chaining them to motion graphs.

UBO will lead the development in this work package and will be supported by LIU, SSSA and IIT. To ensure the successful implementation and integration of this work package into the overall CENTAURO system, we will iteratively develop the individual software components and evaluate their performance in close collaboration with all concerned partners of the other work packages. Furthermore, we will integrate state-of-the-art components which are already available as a fallback for situations in which we have not collected sufficient a sufficient number of demonstrations.

References

[1] A. Aydemir and P. Jensfelt. Exploiting and modeling local 3d structure for predicting object locations. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2012.

[2] Alper Aydemir, Kristoffer Sjöö, John Folkesson, Andrzej Pronobis, and Patric Jensfelt. Search in the real world: Active visual object search based on spatial relations. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2818–2824. IEEE, 2011.

[3] Garth H Ballantyne and Fred Moll. The da Vinci telerobotic surgical system: The virtual operative field and telepresence surgery. *Surg Clin North Am*, 83(6):1293–304, vii, 2003.

[4] Sven Behnke. Learning iterative image reconstruction in the neural abstraction pyramid. *International Journal of Computational Intelligence and Applications*, 1(4):427–438, 2001.

[5] Sven Behnke. *Hierarchical Neural Networks for Image Interpretation*, volume 2766 of *Lecture Notes in Computer Science*. Springer, 2003.

[6] Sven Behnke. A two-stage system for meter value recognition. In *Image Processing (ICIP), IEEE International Conference on*, pages 549–552, 2003.

[7] Sven Behnke. Face localization and tracking in the neural abstraction pyramid. *Neural Computing and Applications*, 14(2):97–103, 2005.

[8] Dmitry Berenson and Siddhartha Srinivasa. Grasp synthesis in cluttered environments for dexterous hands. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2008.

[9] Alexander Berner, Jun Li, Dirk Holz, Jörg Stückler, Sven Behnke, and Reinhard Klein. Combining contour and shape primitives for object detection and pose estimation of prefabricated parts. In *Image Processing (ICIP), IEEE International Conference on*, pages 3326–3330, 2013.

[10] Changhyun Choi and Henrik I. Christensen. 3d pose estimation of daily objects using an rgb-d camera. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2012.

[11] Changhyun Choi and Henrik I. Christensen. 3D textureless object detection and tracking: An edge-based approach. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 3877 –3884, 2012.

[12] Changhyun Choi, Y. Taguchi, O. Tuzel, Ming-Yu Liu, and S. Ramalingam. Voting-based pose estimation for robotic assembly using a 3D sensor. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2012.

[13] Alexander Dietrich, Thomas Wimböck, Alin Albu-Schäffer, and Gerd Hirzinger. Reactive whole-body control: Dynamic mobile manipulation using a large number of actuated degrees of freedom. *Robotics & Automation Magazine, IEEE*, 19(2):20–33, 2012.

[14] B. Drost, M. Ulrich, N. Navab, and S. Ilic. Model globally, match locally: Efficient and robust 3D object recognition. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[15] Charith Lasantha Fernando, Masahiro Furukawa, Tadatoshi Kurogi, Sho Kamuro, Katsunari sato, Kouta Minamizawa, and Susumu Tachi. Design of telesar v for transferring bodily consciousness in telexistence. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 5112 –5118, 2012.

[16] Ross Finman, Thomas Whelan, Michael Kaess, and John J Leonard. Toward lifelong object segmentation from change detection in dense rgb-d maps. In *Mobile Robots (ECMR), 2013 European Conference on*, pages 178–185. IEEE, 2013.

[17] Michael Gienger, Marc Toussaint, and Christian Goerick. Whole-body motion planning–building blocks for intelligent systems. In *Motion Planning for Humanoid Robots*, pages 67–98. Springer, 2010.

[18] Corey Goldfeder, Matei Ciocarlie, Hao Dang, and Peter K. Allen. The columbia grasp database. In *Proceedings of the IEEE Intl. Conf. on Robotics and Automation*, 2009.

[19] J.L. Graham, S.G. Manuel, M.S. Johannes, and R.S. Armiger. Development of a multi-modal haptic feedback system for dexterous robotic telemanipulation. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3548–3553, 2011.

[20] Kathrin Gräve and Sven Behnke. Incremental action recognition and generalizing motion generation based on goal-directed features. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, pages 751–757, 2012.

[21] Kathrin Gräve and Sven Behnke. Learning sequential tasks interactively from demonstrations and own experience. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, pages 3237–3243, 2013.

[22] Kathrin Gräve and Sven Behnke. Bayesian exploration and interactive demonstration in continuous state MAXQ-learning. In *Robotics and Automation (ICRA), IEEE International Conference on*, 2014.

[23] Ulrich Hagn, Rainer Konietschke, Andreas Tobergte, Mathias Nickl, Stefan Jörg, Bernhard Kübler, Georg Passig, Martin Gröger, Florian Fröhlich, Ulrich Seibold, Luc Le Tien, Alin Albu-Schäffer, Alexander Nothhelfer, Franz Hacker, Markus Grebenstein, and Gerd Hirzinger. DLR MiroSurge: a versatile system for research in endoscopic telesurgery. *Int. J. Computer Assisted Radiology and Surgery*, 5(2):183–193, 2010.

[24] Johan Hedborg, Per-Erik Forssén, and Michael Felsberg. Fast and Accurate Structure and Motion Estimation. In *International Symposium on Visual Computing*, volume Volume 5875 of *Lecture Notes in Computer Science*, pages 211–222, Berlin Heidelberg, 2009. Springer-Verlag.

[25] Evan Herbst, Peter Henry, Xiaofeng Ren, and Dieter Fox. Toward object discovery and modeling via 3-d scene comparison. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2623–2629. IEEE, 2011.

[26] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, , and N. Navab. Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. 2012.

[27] Nico Höft, Hannes Schulz, and Sven Behnke. Fast semantic segmentation of RGB-D scenes with gpu-accelerated deep neural networks. In *37th Annual German Conference on AI (KI)*, pages 80–85, 2014.

[28] Dirk Holz and Sven Behnke. Fast range image segmentation and smoothing using approximate surface reconstruction and region growing. In *Proceedings of the 12th International Conference on Intelligent Autonomous Systems (IAS-12)*, 2012.

[29] Dirk Holz, Stefan Holzer, Radu Bogdan Rusu, and Sven Behnke. Real-time plane segmentation using RGB-D cameras. In *RoboCup 2011: Robot Soccer World Cup XV*, volume 7416 of *Lecture Notes in Computer Science*, pages 306–317. Springer, 2012.

[30] Dirk Holz, Ruwen Schnabel, David Droeschel, Jörg Stückler, and Sven Behnke. Towards semantic scene analysis with time-of-flight cameras. In *RoboCup 2010: Robot Soccer World Cup XIV*, volume 6556 of *Lecture Notes in Computer Science*, pages 121–132. Springer, 2011.

[31] Dirk Holz, Angeliki Topalidou-Kyniazopoulou, Joerg Stueckler, and Sven Behnke. Real-time object detection, localization and verification for fast robotic depalletizing. In *Intelligent Robots and Systems (IROS), IEEE International Conference on*, 2015.

[32] Armin Hornung, Mike Phillips, Edward Gil Jones, Maren Bennewitz, Maxim Likhachev, and Sachin Chitta. Navigation in three-dimensional cluttered environments for mobile manipulation. In *Proceedings of the IEEE Int. Conference on Robotics and Automation (ICRA)*, 2012.

[33] Armin Hornung, Kai M Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 34(3):189–206, 2013.

[34] K. Hsiao, S. Chitta, M. Ciocarlie, and E. G. Jones. Contact-reactive grasping of objects with partial shape information. In *Proceedings of the International Conference on Robots and Systems (IROS)*, 2010.

[35] T. Hulin, K. Hertkorn, P. Kremer, S. Schatzle, J. Artigas, M. Sagardia, F. Zacharias, and C. Preusche. The dlr bimanual haptic device with optimized workspace. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3441–3442, 2011.

[36] K. Iagnemma and J. Overholt. Special issue: DARPA robotics challenge (DRC). *Journal of Field Robotics*, 32(2), 2015.

[37] Auke Jan Ijspeert, Jun Nakanishi, and Stefan Schaal. Movement imitation with nonlinear dynamical systems in humanoid robots. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 2, pages 1398–1403. IEEE, 2002.

[38] Fahad Shahbaz Khan, Rao Muhammad Anwer, Joost van de Weijer, Andrew D. Bagdanov, Maria Vanrell, and Antonio M. Lopez. Color Attributes for Object Detection. In

Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2012, pages 3306–3313. IEEE, 2012.

[39] David Kirkpatrick, Bhubaneswar Mishra, and Chee-Keng Yap. Quantitative steinitz's theorems with applications to multifingered grasping. *Discrete & Computational Geometry*, 7:295–318, 1992.

[40] A. Kron, G. Schmidt, B. Petzold, M.I. Zah, P. Hinterseer, and E. Steinbach. Disposal of explosive ordnances by use of a bimanual haptic telepresence system. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2004.

[41] Dana Kulić, Christian Ott, Dongheui Lee, Junichi Ishikawa, and Yoshihiko Nakamura. Incremental learning of full body motion primitives and their sequencing through human motion observation. *The International Journal of Robotics Research*, page 0278364911426178, 2011.

[42] K. Lakshminarayana. Mechanics of form closure. *ASME Paper*, 78-DET-32, 1978.

[43] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[44] Guilherme Maeda, Marco Ewerton, Rudolf Lioutikov, Heni Ben Amor, Jan Peters, and Gerhard Neumann. Learning interaction for collaborative tasks with probabilistic movement primitives. In *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*, pages 527–534. IEEE, 2014.

[45] Zoltan-Csaba Marton, Radu Bogdan Rusu, Dominik Jain, Ulrich Klank, and Michael Beetz. Probabilistic categorization of kitchen objects in table settings with a composite sensor. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 4777–4784. IEEE, 2009.

[46] Manus McElhone, Jörg Stückler, and Sven Behnke. Joint detection and pose tracking of multi-resolution surfel models in RGB-D. In *2013 European Conference on Mobile Robots (ECMR)*, pages 131–137, 2013.

[47] Andrew T. Miller, Steffen Knoopt, Henrik I. Christensen, and Peter K. Allent. Automatic grasp planning using shape primitives. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2003.

[48] Jianyuan Min and Jinxiang Chai. Motion graphs++: A compact generative model for semantic motion analysis and synthesis. *ACM Trans. Graph.*, 31(6):153:1–153:12, November 2012.

[49] V. Mohan, P. Morasso, G. Metta, and G. Sandini. A biomimetic, force-field based computational model for motion planning and bimanual coordination in humanoid robots. *Auton. Robots*, 27(3):291–307, 2009.

[50] Manuel Mühlig, Michael Gienger, and Jochen J Steil. Interactive imitation learning of object movement skills. *Autonomous Robots*, 32(2):97–114, 2012.

[51] Andreas C. Müller and Sven Behnke. Learning depth-sensitive conditional random fields for semantic segmentation of RGB-D images. In *Robotics and Automation (ICRA), IEEE International Conference on*, pages 6232–6237, 2014.

[52] Matthias Nieuwenhuisen, David Droeschel, Dirk Holz, Jörg Stückler, Alexander Berner, Jun Li, Reinhard Klein, and Sven Behnke. Mobile bin picking with an anthropomorphic service robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, to appear 2013.

[53] Jia Pan, Sachin Chitta, and Dinesh Manocha. Fcl: A general purpose library for collision and proximity queries. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3859–3866. IEEE, 2012.

[54] Dae-Hyung Park, Peter Pastor, Stefan Schaal, et al. Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields. In *Humanoid Robots, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on*, pages 91–98. IEEE, 2008.

[55] Mircea Serban Pavel, Hannes Schulz, and Sven Behnke. Recurrent convolutional neural networks for object-class segmentation of RGB-D video. In *Neural Networks (IJCNN), International Joint Conference on*, 2015.

[56] L. Pedersen, D. Kortenkamp, D. Wettergreen, and I. Nourbakhsh. A survey of space robotics. In *Proceedings of the 7th Internation Symposium on Artificial Intelligence, Robotics and Automation in Space*, 2003.

[57] Florian T. Pokorny, Yasemin Bekiroglu, and Danica Kragic. Grasp moduli spaces and spherical harmonics. In *2014 IEEE International Conference on Robotics and Automation (ICRA) :*, pages 389–396, 2014.

[58] Markus Przybylski, Tamim Asfour, and Rüdiger Dillmann. Planning grasps for robotic hands using a novel object representation based on the medial axis transform. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 1781–1788. IEEE, 2011.

[59] René Felix Reinhart, Andre Lemme, and Jochen Jakob Steil. Representation and generalization of bi-manual skills from kinesthetic teaching. In *Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on*, pages 560–567. IEEE, 2012.

[60] European Commission Directorate General for Communications Networks, Content and Technology Components and Systems: Robotics. Grant Agreement 644839: CENTAURO—Robust Mobility and Dexterous Manipulation in Disaster Response by Fullbody Telepresence in a Centaur-like Robot, 2014.

[61] Tobias Rodehutskors, Max Schwarz, and Sven Behnke. Intuitive bimanual telemanipulation under communication restrictions by immersive 3D visualization and motion tracking. In *IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2015.

[62] Radu Bogdan Rusu, Andreas Holzbach, Nico Blodow, and Michael Beetz. Fast geometric point labeling using conditional random fields. In *22nd IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2009.

[63] Dominik Scherer, Andreas C. Müller, and Sven Behnke. Evaluation of pooling operations in convolutional architectures for object recognition. In *Artificial Neural Networks (ICANN), 20th International Conference on*, pages 92–101, 2010.

[64] Alexander M Schmidts, Dongheui Lee, and Angelika Peer. Imitation learning of human grasping skills from motion and force data. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 1002–1007. IEEE, 2011.

[65] Hannes Schulz and Sven Behnke. Learning object-class segmentation with convolutional neural networks. In *20th European Symposium on Artificial Neural Networks, ESANN 2012, Bruges, Belgium, April 25-27, 2012*, 2012.

[66] Hannes Schulz and Sven Behnke. Learning object-class segmentation with convolutional neural networks. In *20th European Symposium on Artificial Neural Networks (ESANN)*, 2012.

[67] Hannes Schulz and Sven Behnke. Structured prediction for object detection in deep neural networks. In *Artificial Neural Networks (ICANN) International Conference on*, pages 395–402, 2014.

[68] Hannes Schulz, Benedikt Waldvogel, Rasha Sheikh, and Sven Behnke. CURFIL: random forests for image labeling on GPU. In *10th International Conference on Computer Vision Theory and Applications (VISAPP)*, pages 156–164, 2015.

[69] Max Schwarz, Hannes Schulz, and Sven Behnke. RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features. In *Robotics and Automation (ICRA), IEEE International Conference on*, pages 1329–1335, 2015.

[70] Ricarda Steffens, Matthias Nieuwenhuisen, and Sven Behnke. Continuous motion planning for service robots with multiresolution in time. In *Intelligent Autonomous Systems (IAS), 13th International Conference on*. IEEE, 2014.

[71] Jörg Stückler and Sven Behnke. Integrating depth and color cues for dense multi-resolution scene mapping using RGB-D cameras. In *Proc. of the IEEE Int. Conf. on Multisensor Fusion and Information Integration (MFI)*, 2012.

[72] Jörg Stückler and Sven Behnke. Model learning and real-time tracking using multi-resolution surfel maps. In *Proc. of the AAAI Conference on Artificial Intelligence (AAAI-12)*, 2012.

[73] Jörg Stückler and Sven Behnke. Hierarchical object discovery and dense modelling from motion cues in RGB-D video. In *Artificial Intelligence (IJCAI), 23rd International Joint Conference on*, 2013.

[74] Jörg Stückler and Sven Behnke. Adaptive tool-use strategies for anthropomorphic service robots. In *14th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 755–760, 2014.

[75] Jörg Stückler and Sven Behnke. Efficient dense rigid-body motion segmentation and estimation in RGB-D video. *International Journal of Computer Vision*, 113(3):233–245, 2015.

[76] Jörg Stückler and Sven Behnke. Perception of deformable objects and compliant manipulation for service robots. In *Soft Robotics*, pages 69–80. Springer, 2015.

[77] Jörg Stückler, Max Schwarz, Mark Schadler, Angeliki Topalidou-Kyniazopoulou, and Sven Behnke. Nimbro explorer: Semi-autonomous exploration and mobile manipulation in rough terrain. *Journal of Field Robotics*, 2013.

[78] Jörg Stückler, Ricarda Steffens, Dirk Holz, and Sven Behnke. Efficient 3D object perception and grasp planning for mobile manipulation in domestic environments. *Robotics and Autonomous Systems*, 61(10):1106–1115, 2013.

[79] Jörg Stückler, Ricarda Steffens, Dirk Holz, and Sven Behnke. Efficient 3d object perception and grasp planning for mobile manipulation in domestic environments. *Robotics and Autonomous Systems*, 61(10):1106–1115, 2013.

[80] Jörg Stückler, Benedikt Waldvogel, Hannes Schulz, and Sven Behnke. Dense real-time mapping of object-class semantics from RGB-D video. *Journal of Real-Time Image Processing*, 2014.

[81] Ioan Sucan, Maciej Moll, Lydia E Kavraki, et al. The open motion planning library. *Robotics & Automation Magazine*, 19(4):72–82, 2012.

[82] Federico Tombari and Luigi DiStefano. Hough voting for 3D object recognition under occlusion and clutter. *IPSJ Transactions on Computer Vision and Applications*, 4:20–29, 2012.

[83] Rafael Uetz and Sven Behnke. Large-scale object recognition with cuda-accelerated hierarchical neural networks. In *Intelligent Computing and Intelligent Systems (ICIS), IEEE International Conference on*, 2009.

[84] Nikolaus Vahrenkamp, Tamim Asfour, and Rüdiger Dillmann. Efficient motion planning for humanoid robots using lazy collision checking and enlarged robot models. In *IEEE International Conference on Intelligent Robots and Systems, IROS*, 2007.

[85] Nikolaus Vahrenkamp, Markus Przybylski, Tamim Asfour, and Rüdiger Dillmann. Bimanual grasp planning. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2011.

[86] Marcus Wallenberg, Michael Felsberg, Per-Erik Forssén, and Babette Dellen. Channel Coding for Joint Colour and Depth Segmentation. In *Proceedings of Pattern Recognition 33rd DAGM Symposium, Frankfurt/Main, Germany, August 31 - September 2*, volume 6835 of *Lecture Notes in Computer Science*, pages 306–315. SpringerLink, 2011.

[87] L.L. Whitcomb. Underwater robotics: out of the research laboratory and into the field. In *In Proceedings of the IEEE International Conference on Robotics and Automation*, 2000.

[88] Vasileios Zografos, Reiner Lenz, Erik Ringaby, Michael Felsberg, and Klas Nordberg. Fast segmentation of sparse 3D point trajectories using group theoretical invariants. In *Computer Vision – ACCV 2014*, volume 9003–9007 of *Lecture Notes in Computer Science*. Springer, 2014.