

Supervised Autonomous Locomotion and Manipulation for Disaster Response with a Centaur-like Robot

Tobias Klamt, Diego Rodriguez, Max Schwarz, Christian Lenz, Dmytro Pavlichenko, David Droschel, and Sven Behnke

Abstract—Mobile manipulation tasks are one of the key challenges in the field of search and rescue (SAR) robotics requiring robots with flexible locomotion and manipulation abilities. Since the tasks are mostly unknown in advance, the robot has to adapt to a wide variety of terrains and workspaces during a mission. The centaur-like robot Centauro has a hybrid legged-wheeled base and an anthropomorphic upper body to carry out complex tasks in environments too dangerous for humans. Due to its high number of degrees of freedom, controlling the robot with direct teleoperation approaches is challenging and exhausting. Supervised autonomy approaches are promising to increase quality and speed of control while keeping the flexibility to solve unknown tasks. We developed a set of operator assistance functionalities with different levels of autonomy to control the robot for challenging locomotion and manipulation tasks. The integrated system was evaluated in disaster response scenarios and showed promising performance.

I. INTRODUCTION

In many SAR scenarios, humans cannot work due to risks such as radiation or collapsing structures. Mobile manipulation robots are promising to help solving tasks in these cases. Respective environments, e.g., the damaged nuclear plant in Fukushima, are mostly man-made but cluttered with debris and unpredictable. Hence, a suitable platform needs to provide a wide range of capabilities to solve occurring tasks and address unforeseen difficulties.

The Centauro robot has been developed in the European H2020 project CENTAURO¹ for such scenarios (Fig. 1). Its lower body consists of four articulated legs ending in steerable wheels which allows for omnidirectional driving as well as for stepping locomotion. The anthropomorphic upper body possesses two 7 DoF arms ending in two hands with different capabilities. One of them is an anthropomorphic Schunk hand which allows for precise manipulation in man-made workspaces. Additionally, several sensors such as a 3D laser scanner and cameras perceive the environment and enable the operators to obtain situation awareness.

Teleoperation of such highly flexible robots is challenging, though. Common approaches, like the control in joint space or Cartesian end-effector space, are only suitable for simple tasks. For more complex tasks, the high number of DoF and typical constraints of multi-legged robots, such as stability and collision avoidance, put a high cognitive load on the operator which may result in slow and dangerous

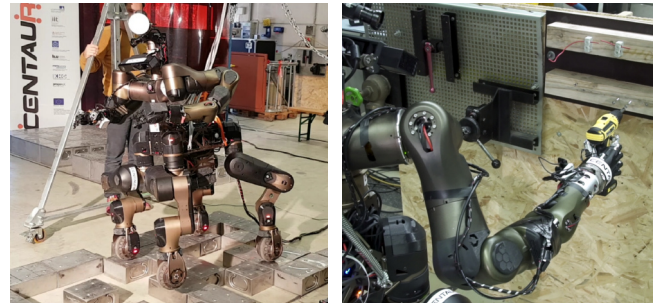


Fig. 1. Centauro robot controlled by our proposed teleoperation system: overcoming a step field (l.) and operating an off-the-shelf power drill (r.).

operations. One way to address this challenge is by means of immersive exoskeletons, such as the recently introduced Master Maneuvering System for the Toyota humanoid robot T-HR3². These direct control interfaces are as complex as the controlled robot and require a low-latency, high-bandwidth data connection. Other teleoperation approaches utilize pre-defined motion primitives. They reduce the operator's cognitive load but the generation of these primitives requires knowledge about specific tasks in advance. Obviously, this restricts the platform flexibility and applicability to unknown tasks. Supervised autonomy is promising to provide fast and reliable control while keeping a high flexibility.

We developed a set of teleoperation interfaces with different levels of autonomy for solving a wide variety of locomotion and manipulation tasks with Centauro. For example, we perform autonomous grasping of unknown tools or semi-autonomous stepping over irregular terrain. Other interfaces with less autonomy include wrist control via a 6D input device. All interfaces provide a high degree of intuition which leads to a limited cognitive load for the operator. This results in less operator failures and extended operation times before the operator needs to recover or must be exchanged.

The integrated functionalities were evaluated in experiments which are typical for disaster-response missions. Locomotion capabilities were evaluated in tasks like driving up a ramp, overcoming a gap, and moving through an irregular step field. Manipulation interfaces were evaluated in experiments like grasping and using different power tools, physically connecting and disconnecting objects such as electrical plugs, or scanning surfaces, e.g., for radiation. A combination of locomotion and manipulation capabilities was required for opening and passing a door. Most of the tasks were solved quickly and without previous training.

All authors are with the Autonomous Intelligent Systems group of University of Bonn, Germany klamt@ais.uni-bonn.de. This work was supported by the European Union's Horizon 2020 Programme under Grant Agreement 644839 (CENTAURO).

¹ www.centauro-project.eu

² <https://newsroom.toyota.co.jp/en/download/20110424>

II. RELATED WORK

Several mobile manipulation robots have been developed for SAR missions. Those robots vary in their locomotion strategy as well as their manipulation setup. Pure wheeled, tracked or legged robots can either overcome long, sufficiently even distances quickly or can navigate in challenging terrain with isolated footholds, but a combination of both capabilities is only available for hybrid driving-stepping locomotion platforms. Manipulation capabilities depend on the number and design of arms and especially the end-effectors. However, independent from robot design details, key to the applicability in a wide range of scenarios are teleoperation interfaces which ideally enable teleoperators to use all robot capabilities while keeping the cognitive load low and the applicability to unknown tasks high. In 2015, the DARPA Robotics Challenge (DRC) pushed research teams to develop robots that are capable of performing several mobile manipulation tasks which indirectly put the focus on the development of suitable teleoperation interfaces.

RoboSimian [1] is a quadrupedal robot with four generalized limbs, developed for the DRC. Each limb ends in an under-actuated hand which allows for solving both stepping locomotion and manipulation tasks. Furthermore, RoboSimian has two active wheels at its trunk and two caster wheels at its limbs which allow for driving on even terrain. The operator interface is a standard laptop from which the operator can design, parametrize, and sequence predefined behaviors. The DRC winner robot DRC-HUBO [2] and the third placed platform CHIMP [3] both have roughly anthropomorphic bodies. Both are capable of walking and driving via additional wheels/tracks on their body. Both robots have two arms which end in hands with three fingers. Operation of DRC-HUBO is apportioned among three operators with different tasks which control the robot by selecting and adapting predefined poses while CHIMP is operated through task-specific motions which are configured through wizards by the operator before their execution.

Our centaur-shaped robot Momaro [4] came in 4th in the DRC using multiple teleoperation interfaces and showed autonomy solving known tasks at the DLR SpaceBot Cup [5]. Similar to Centauro, it has four legs ending in steerable wheels and an anthropomorphic upper body. In contrast to Centauro, it lacks hip yaw joints for the legs which restricts stepping capabilities. Moreover, its two arms end in 4-finger grippers which cannot provide the grasping capabilities of a human hand. Driving locomotion can be controlled by a joystick; leg motions are predefined or can be controlled during mission in joint space or Cartesian end-effector space. Furthermore, a semi-autonomous stepping controller was presented which relies on perceived terrain heights [6]. For telemanipulation, the operator used two hand-held controllers with magnetic trackers whose movements were projected to the robot arms. Although this approach appears to be intuitive, the operator experienced a high cognitive load due to imprecision in the motion mapping and the lack of feedback. Grasping was controlled by predefined gripper

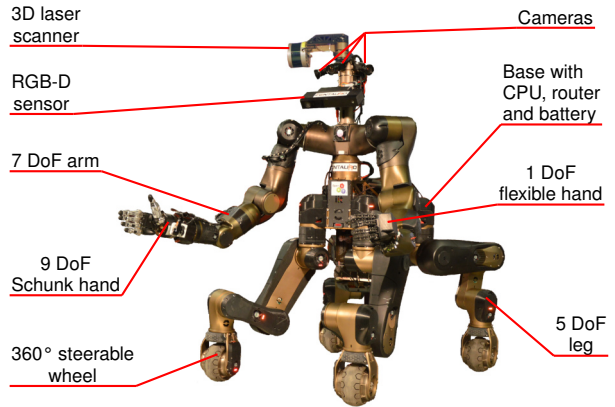


Fig. 2. The Centauro robot.

configurations.

Momaro can be seen as the predecessor of Centauro. Key design features, such as the general lower and upper body kinematics and the sensor setup, were transferred. Weak points, such as chosen actuators, missing hip yaw joints or restricted end-effectors, were improved.

Regarding the operator interface, we enrich Momaro's teleoperation interfaces by adding more intuitive control devices. Additionally, we focus on solving unknown tasks by incorporating flexible autonomous capabilities. An overview over robot control approaches with different levels of autonomy is given by Kiu et al. [7]. As shown recently by Marturi et al. [8] and earlier by Leeper et al. [9], the task efficiency and accuracy are improved by incorporating further interfaced and autonomous functionalities. Muelling et al. [10], for example, developed an integrated system of computer vision with manipulation capabilities, in which known objects with simple geometries are recognized, localized and grasped using depth images. Peer et al. [11] and Salvietti et al. [12] present telemanipulation approaches by mapping operator hand configurations to the robot hand and provide force feedback. Although such interfaces seem intuitive, they generally require a large amount of operator training to provide a satisfying grasp quality. Havoutis et al. [13] learn manipulation tasks online for semi-autonomous teleoperation applications where large communication latency make direct teleoperation unfeasible. In our current system, we use both perception and learning approaches to enrich our teleoperation capabilities.

III. HARDWARE

Centauro (Fig. 2) was designed by the Istituto Italiano di Tecnologia (IIT), bringing together Momaro's kinematic concept and Walk-Man's compliant actuation [14]. Centauro's kinematic designs aims to provide a wide range of locomotion and manipulation capabilities to solve any occurring disaster response task while the robot size is suitable for man-made environments and workspaces [15].

Centauro's lower body features four articulated 5-DoF legs which end in 360° steerable, directly driven wheels. This design allows for both omnidirectional driving and stepping locomotion. In addition, Centauro can perform locomotion

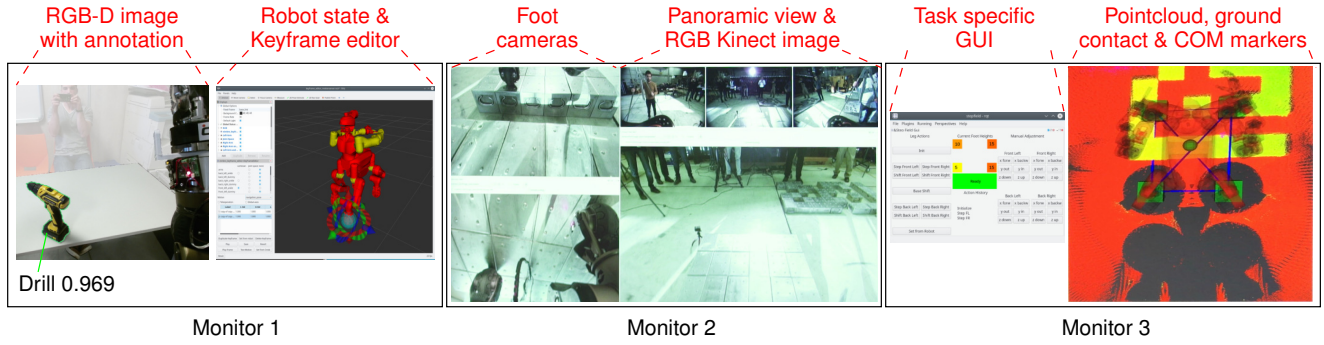


Fig. 3. Environment and robot state visualization for the operators.

actions which are not possible for driving or legged platforms, such as shifting individual feet while keeping ground contact.

The anthropomorphic upper body consists of a torso yaw joint and two 7-DoF arms ending in end-effectors with different capabilities. The right end-effector is a 9-DoF anthropomorphic Schunk hand which allows for dexterous, human-like manipulation [16]. The left end-effector is a flexible 1-DoF SoftHand which can be used for robust manipulation [17]. The overall upper body design results in a workspace equal to an adult-sized human.

Centauro's head comprises a *Microsoft Kinect V2* RGB-D sensor [18], an array of three *PointGrey BlackFly BFLY-U3-23S6C* wide-angle color cameras, and a rotating *Velodyne Puck VLP-16* 3D laser scanner with a spherical field-of-view. A *VectorNav VN-100* IMU is mounted in the torso. Two further RGB cameras were mounted under the robot base to obtain a view on the feet. Furthermore, the robot base incorporates three computing units as well as the communication routers and the robot battery.

IV. TELEOPERATION ARCHITECTURE

Although the considered disaster environments are too dangerous for a human to work in, the human capabilities of situation assessment, mission planning, and his experience are key to a successful SAR mission. The teleoperation interface enables the operators to transfer these capabilities into the scene by providing them an awareness of the situation and enabling them to control the robot. Both require a communication infrastructure, since a direct line of sight is not available.

A. Communication

For data transmission between the operator station and the robot, we use Ethernet connection or a standard IEEE 802.11ac 5 GHz WiFi link. All communication takes place using ROS, which is either directly accessed using ROS network transparency, or encoded with FEC for robustness using the *nimbros_network* developed for Momaro [4]. For extending the reach, a WiFi repeater can be carried by the Centauro robot and dropped at an appropriate location.

B. Situation Awareness

We developed several visualizations of the environment and the robot state to provide good situation awareness for

the operators. RGB camera images from the three cameras in the robot head are arranged to show a panoramic view from the robot head perspective which is helpful for a general scene understanding. In addition, images from the two RGB cameras under the robot base are arranged to give a detailed assessment for the terrain under the robot base which was key to a safe stepping locomotion operation. We rotated the image of the camera showing the two rear feet by 180° for intuitive visualization (Fig. 3). Moreover, laser scanner measurements are processed to registered point clouds which are visualized in RVIZ (Sec. V-B). This visualization is helpful for both locomotion and manipulation tasks. Finally, colored RGB-D point clouds are displayed to support manipulation. Those are enriched by semantics from the object detection (Sec. V-C).

The robot state is visualized by applying measured joint angles and IMU data to a 3D robot model in RVIZ. Further information, such as foot ground contact detection and the robot center of mass (CoM) are also displayed. We developed multiple robot control GUIs for different task classes. All visualization elements were arranged on three monitors as shown in Fig. 3.

C. Control Interfaces

We propose multiple locomotion and manipulation control interfaces which are suitable for different task classes. The whole set of control interfaces aims at enabling the operator to solve as many—previously known and unknown—typical disaster response tasks as possible. Hence, a key requirement is to address the whole range of kinematic capabilities of the robot while keeping the control itself intuitive. Different levels of autonomy are utilized to fulfill these requirements. The individual control interfaces are presented in Sec. VI and Sec. VII. Some of them require processed sensor input which is described in the following.

V. ADVANCED ENVIRONMENT PERCEPTION

The chosen sensor setup produces data of several types. While some sensor measurements, such as foot camera images, can be directly shown to the operators, other data is processed. The results serve as more intuitive visualizations or as input for some of the autonomous control functions.

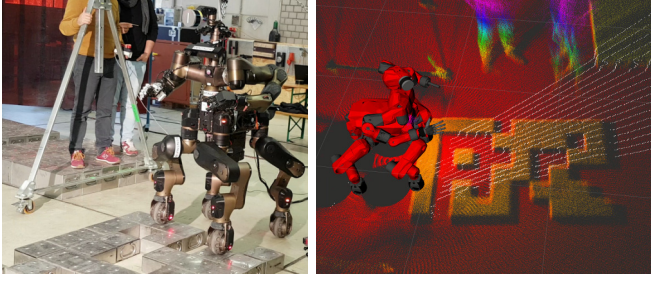


Fig. 4. Centauro robot traversing a step field. Left: photo of the scene, right: laser-based 3D map (colored points) and current scan (white points).

A. Ground Contact Detection

To understand the robot positioning in challenging terrain and to enable semi-autonomous stepping, it is helpful to detect, if a foot has ground contact. By measuring the joint torques of the respective leg and by applying a forward dynamics approach, we compute the 6D force vector which is applied to the foot. The vertical force component is extracted and compensated for gravity. If the resulting force exceeds a given threshold, ground contact is detected.

B. Laser-based 3D Mapping and Localization

Laser range measurements from the 3D rotating laser scanner are aggregated to a dense 3D map of the environment using our local multiresolution surfel grid approach [19]. The laser provides $\sim 300,000$ range measurements per second with a maximum range of 100m and is rotated at 0.1 rotations per second, resulting in a dense omnidirectional 3D scan per half rotation. We acquire one full 3D scan every five seconds and compensate for sensor motion during acquisition by incorporating IMU measurements.

Consecutive scans are registered to a dense egocentric map. The resulting egocentric maps from different view poses form nodes in a pose graph to allow for allocentric mapping of the environment. They are connected by edges representing spatial constraints, which result from aligning these maps with each other. The global registration error is minimized using graph optimization. The resulting 3D map allows for localizing the robot in an allocentric frame. A resulting 3D map is shown in Fig. 4.

C. Object Segmentation

We apply our object segmentation approach to RGB images from the *Kinect V2* [20]. This approach is able to produce pixel- (or point-)wise segmentation directly. It uses the RefineNet [21] architecture, which addresses the problem of low spatial resolution in later stages of the CNNs by subsequently upsampling and merging higher-level feature maps with lower-level features of higher spatial resolution—creating a representation of the input image with both highly semantic information and high spatial resolution, which is well suited for semantic segmentation.

We address the lack of large amounts of training data by generating new training scenes using data captured from a turntable setup. Automatically extracted object segments are inserted into precaptured scenes (Fig. 5).



Fig. 5. Scene synthesis. Synthetic training scene generated by inserting new objects into the scene. The right image shows the resulting color image, the left one shows synthetic ground truth for training the segmentation model.

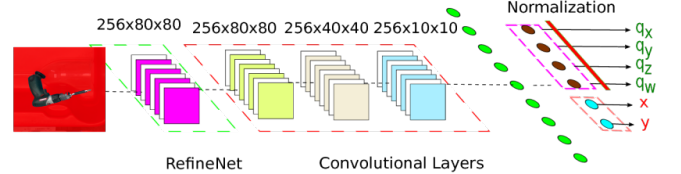


Fig. 6. Pose estimation network architecture.

D. Pose Estimation

For predicting poses efficiently, we augment the semantic segmentation pipeline with an additional CNN to estimate the object 5D pose (rotational, and X and Y of the translational components) from RGB-D crops of the objects from the scene. Those crops are extracted from the bounding boxes of detected contours. To encode the segmentation results, pixels classified as non-object are pushed towards red (Fig. 6). This representation allows the network to focus on the specific object for which the pose should be estimated. The pretrained RefineNet network from the semantic segmentation is used to extract features. To generate the ground truth poses for training the network, the data acquisition pipeline described in [20] was extended to record turntable poses automatically and fuse captures with different object poses or different objects with minimal user intervention.

VI. LOCOMOTION CONTROL

Centauro's lower body design allows for omnidirectional driving as well as stepping locomotion and, hence, provides a wide range of locomotion capabilities which have to be addressed by the respective control interface. Driving locomotion allows for fast, energy efficient and stable navigation on sufficiently even terrain while stepping locomotion increases the platform's capabilities to terrains where only isolated footholds are available. Besides the listed control interfaces, we developed a hybrid driving-stepping locomotion planner [22], [23] which lifts the level of autonomy even higher but has not been evaluated on the real platform, yet.

A. 4D Joystick

Omnidirectional driving can be controlled by a *Logitech Extreme 3D Pro* joystick with four axis (Fig. 7). Robot base velocity components v_x , v_y and v_θ are mapped to the three corresponding joystick axis. Foot-specific velocities and orientations are derived from this robot base velocity and the individual foot positions. The joystick throttle controller jointly scales all three velocity components.

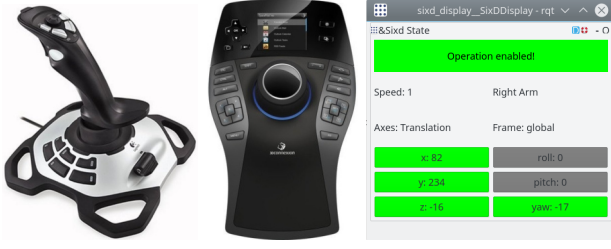


Fig. 7. Operator input devices. Left: *Logitech Extreme 3D Pro* for omnidirectional driving control, center and right: *3DConnexion SpacePilot Pro* and respective operator GUI for dexterous manipulation.

B. Keyframe Editor

A keyframe editor generates robot motions by interpolating between given keyframes [5]. Keyframes for joint groups (e.g., the front left leg) can either be specified in joint space or in Cartesian end-effector space. Longer motion sequences can be designed by queuing keyframes. The operators can either predefine keyframes, modify them during the mission and send them to the robot, or modify the robot configuration live. The RVIZ-based GUI (Fig. 8) allows for keyframe definition by either graphically moving joint group markers with the mouse or by entering numerical values for desired joint angles or end-effector positions.

C. Semi-autonomous Stepping Locomotion

Stepping locomotion can be controlled by a semi-autonomous controller. It provides a set of motions which can be triggered by the operator. Available motions are: step with a chosen foot, drive a chosen foot forward, and shift the robot base forward. For stepping motions, the controller balances the robot by shifting the robot base longitudinally and laterally, and by rolling it around its longitudinal axis. If a stable pose is established, the stepping foot is lifted, extended by a given length and lowered. The lowering motion stops when ground contact is detected. Hence, the robot adapts to the terrain automatically. The proposed controller triggers queues of keyframes, as described in Sec. VI-B.

We developed an intuitive GUI which provides buttons to trigger the described motions (Fig. 8). It also contains buttons to manually move individual feet in Cartesian space. Moreover, the GUI displays detected terrain heights under the four feet and a history of the recently triggered motions which is helpful to execute repetitive motion sequences.

D. Motion Execution

The Centauro robot uses a keyframe interpolation method developed for Momaro to generate joint space trajectories [4]. Keyframes consist of joint space or 6D Euclidean space poses for each of the robot's limbs. The interpolation system produces jerk-free joint-space trajectories obeying velocity and acceleration constraints set per keyframe.

VII. MANIPULATION CONTROL

Regarding the robot's manipulation capabilities, the Centauro system possesses several levels of autonomy: starting at low-level direct joint control; over inverse kinematics control with end-effector poses coming from either an 6D input

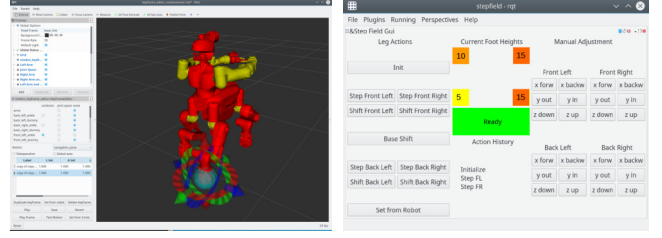


Fig. 8. Left: Keyframe editor, right: semi-autonomous stepping GUI.

device, or 6D markers on the screen; keyframe motions with collision avoidance; and finally, autonomous pick-and-place actions triggered by the operator. For manipulation, we also use the same interface as described in Sec. VI-B. Thus, we will only describe in this section the novel 6D input interface and the autonomous grasping capabilities.

A. Dexterous Wrist Manipulation

We developed a user interface for dexterous manipulation using a *3DConnexion SpacePilot Pro* which is a 6D input device with additional buttons (Fig. 7). The interface establishes the connection between the device and a motion player, which interpolates between the desired and current poses and executes the motion.

The following control parameters can be easily adjusted by the GUI (Fig. 7) or by the device buttons: the *controlled end-effector* (e.g., a wrist for arm control or an ankle for leg control), the *reference frame* (e.g., robot base frame, end-effector frame, or a custom frame), the translational and rotational *axes* in which the end-effector is allowed to move, and the maximum end-effector *speed*.

This teleoperation interface is well suited for manipulation tasks where very precise arm movement along certain directions is required (e.g., moving the arm along a plane surface or turning an object around a specified axis).

B. Autonomous Grasping

To achieve autonomous manipulation, several components need to be developed and integrated. We propose a pipeline composed of: *semantic segmentation* (Sec. V-C), *pose estimation* (Sec. V-D), and *grasp planning* that generates a feasible motion (set of keyframes), which later is combined with a *trajectory optimization* that produces the final joint trajectory given a collision map generated by the *laser SLAM* (Sec. V-B) perception module (Fig. 9).

1) *Grasp Planning*: Our grasp planning method is based on the observation that objects within a category exhibit several similarities in their extrinsic geometry. We transfer

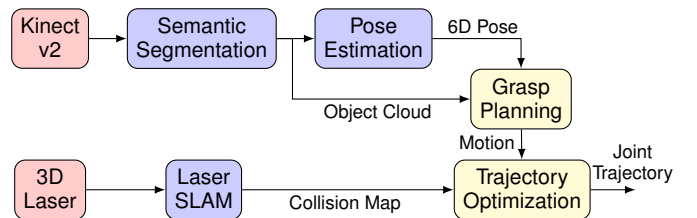


Fig. 9. Overview of autonomous manipulation: integrated sensors (red), perception modules (purple), and manipulation planning (yellow).

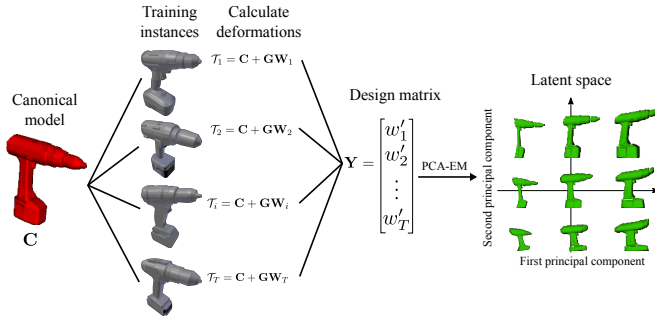


Fig. 10. The latent shape space is built by applying PCA-EM on a matrix containing the deformation fields w_i of each training instance toward the canonical model.

grasp poses from a known instance—called the canonical model—to novel instances of the same category.

This transfer happens as the result of a non-rigid registration method based on a learned latent shape space. For building this latent shape space, we define a single canonical model for the category, and calculate the deformation fields relating the canonical model to all other instances by using Coherent Point Drift (CPD) [24]. This provides a single matrix whose number of elements equals the number of points in the canonical model for each instance. A design matrix containing all deformation fields is consequently assembled as column vectors. Finally, we apply Principle Component Analysis Expectation Maximization (PCA-EM) on the design matrix to find a lower-dimensional manifold of deformation fields, i.e., the latent shape space (Fig. 10).

We add a global rigid transformation for each instance to reduce the impact of minor misalignments in the pose between the canonical shape and the observed shape. For registering a new instance, we use gradient descent to simultaneously optimize for pose and shape. In general, we aim for an aligned dense deformation field that matches best the canonical model toward the observed instance. Associated grasping control poses of the canonical model are also transformed to the observed instance and used for the final grasping motion. We orthonormalize the transformed poses since the warping process can violate the orthogonality of the orientation.

Fig. 11 illustrates how the canonical model and associated grasping control poses of a *Drill* category are warped to fit to the observed point cloud. A complete analysis and discussion of this method is available in [25] and [26].

2) *Trajectory Optimization*: We use arm trajectory optimization to generate collision free and fast arm trajectories with low actuator load. Our approach [27] is based on Stochastic Trajectory Optimization for Motion Planning (STOMP) [28]. The method receives a point cloud describing the environment and an initial trajectory as input. It outputs a trajectory, that is optimized with respect to a cost function. The initial trajectory may be very naïve, for example a straight interpolation between the start and the goal configuration. The trajectories are represented as sets of keyframes in joint space. The optimization is performed in an iterative manner in order to gradually minimize the costs. In contrast to the original STOMP, our cost function is defined as a sum

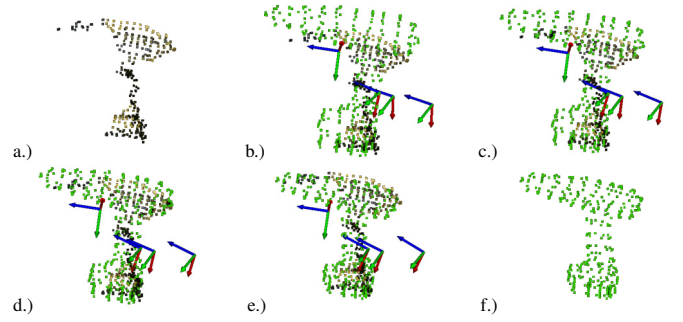


Fig. 11. Transferring grasping knowledge to the presented novel drill. a) novel view; b)-e) grasping control poses of the canonical model are transformed; f) inferred shape.

of costs of *transitions* between the consequent keyframes instead of the keyframes themselves. The cost function includes trajectory duration, collision avoidance, and required joint torques. Since cost components are normalized, they can be weighted to introduce a prioritized optimization.

For collision avoidance, we assume the robot base and the environment to be static and describe both with signed Euclidean Distance Transforms (EDT) which allow for fast collision checking against the moving robot parts, represented as spheres. An example is shown in Fig. 12.

VIII. EVALUATION

We evaluated the Centauro system with task-level tests at facilities of the Kerntechnische Hilfsdienst GmbH in Karlsruhe, Germany, which is a provider of systems and knowledge for disaster response in nuclear power plants. All tasks were performed without direct visual contact such that the operators had to rely on information provided by our interfaces. There were no training runs for any of the tasks. A video with footage from the experiments is available online³. The results are summarized in Table I.

A. Locomotion Tasks

The tested locomotion tasks mainly focused on proving that the robot can effectively navigate different complex terrain types. In the simplest task, the robot was required to drive up a ramp with 20° incline, which was accomplished using joystick teleoperation. In the door experiment (Fig. 13), the robot had to open a door and drive through it. The

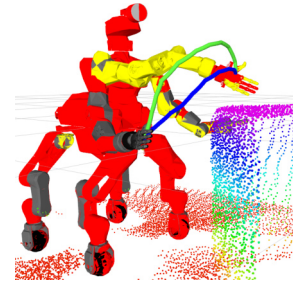


Fig. 12. Two qualitatively different trajectories generated by our trajectory optimization: priority on obstacle avoidance (green) and priority on trajectory duration (blue). The point cloud visualizes the environment.

³https://www.ais.uni-bonn.de/videos/IROS_2018_Centauro

TABLE I
EVALUATED TASKS.

| Locomotion | | Manipulation | |
|------------|---------------|---------------------|---------------|
| Task | Success/Tries | Task | Success/Tries |
| Door | 3/3 | Surface detection | 2/2 |
| Ramp | 3/3 | Plug | 2/3 |
| Gap | 3/4 | Screw driver | 3/3 |
| Step field | 2/2 | Autonomous grasping | 7/14 |
| Stairs | 0/1 | | |

manipulation part was accomplished using the 6D mouse control without any problem.

More complex locomotion capabilities were tested in the gap and step field tests. The gap test required the robot to overcome a 30 cm gap, which was accomplished using predesigned stepping motion primitives, which were interleaved with joystick driving commands (see Fig. 14).

A more challenging test was performed by climbing a set of stairs (see Fig. 15). For this purpose, motion primitives were designed offline before the test, and executed under supervision of the operators, who could take corrective actions using the joystick input. Due to hardware problems, it was only possible to make one serious attempt at climbing the stairs, which had to be stopped after an actuator shutdown halfway up—with the robot at least completely on the stairs.

Another task was to traverse a step field consisting of $20 \times 20 \times 10$ cm blocks which were placed on the ground (see Fig. 16). The operators issued stepping commands via the semi-automatic stepping GUI described in Sec. VI-C. The task was solved two out of two attempts.

Overall, the locomotion capabilities were demonstrated successfully during the Karlsruhe evaluation. The more complex tasks would have been impossible to finish in acceptable time without autonomy functions.

B. Telemanipulation Tasks

The first task required the robot to sweep a planar surface with a (dummy) radiation sensor without touching the surface. This task was successfully performed using the 6D mouse for wrist control and locomotion via joystick.

An electrical plug had to be inserted by the robot (Fig. 17), which was performed using the 6D mouse. After two successful attempts, a plastic part in the robot wrist broke due to excessive force during the third attempt—the operators had misjudged the situation slightly.

The most complex telemanipulation task required the robot to drive a screw into a wooden block (Fig. 18). The

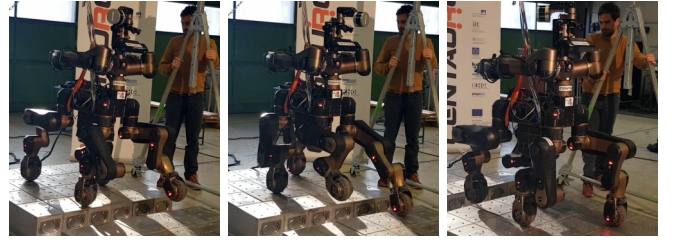


Fig. 14. Overcoming a gap with the Centauro robot.

robot used a cordless screw driver for this task, starting with the tool in hand. The wooden block was approached using joystick locomotion, mainly guided by camera images and the 3D laser scanner point cloud. Next, the tip of the screw driver was aligned with the screw using 6D mouse control, guided by camera images. For gaining an additional perspective, a small webcam was mounted on the other hand, providing a controllable-viewpoint perspective to the operators. After alignment was visually confirmed, the cordless screwdriver was activated using the index finger of the robot. During the screwing process, the operators had to ensure that the tool tip was in constant contact with the screw head, which was facilitated using the single-axis mode of the 6D mouse interface. Overall, three out of three attempts were successful.

C. Autonomous Manipulation

The objective of this test was to detect, segment, and estimate the pose of a previously unknown cordless driller in front of the robot (Fig. 19). After pose estimation, a grasping pose was to be transferred from a known model to the new instance and the driller was to be grasped.

We performed this experiment many times, since it had a higher failure rate due to the complexity and the number of involved components. While the system performed well on the operator side, failures cases on the system side include imprecise segmentation or misregistration, both resulting in missed grasps, and hardware failures. Overall, the success rate improved during testing.

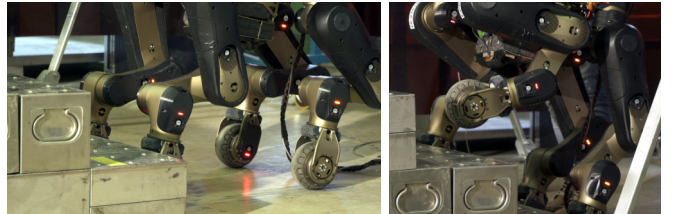


Fig. 15. Climbing stairs.

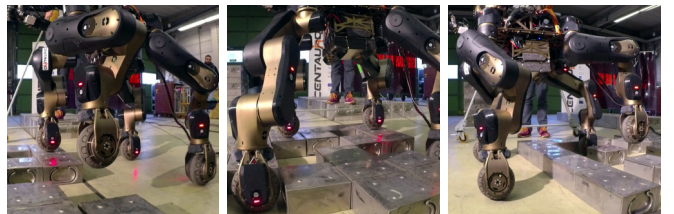


Fig. 16. Traversing a step field with the Centauro robot.

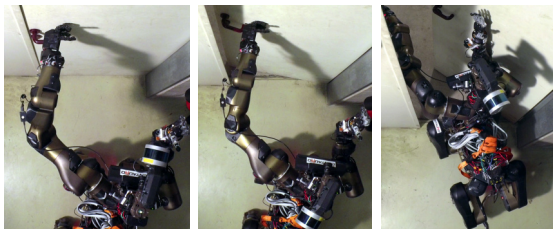


Fig. 13. Opening the door and driving through it.



Fig. 17. Inserting an electrical plug.



Fig. 18. Driving a screw into plywood. Left: Robot arm in front of the screw. Center/right: Detail on fine alignment and screwing.

IX. CONCLUSION

On the example of the Centauro robot, we successfully demonstrated several useful autonomous functions that assist the operators on different levels of autonomy. Their efficiency was especially demonstrated considering that all experiments were performed without any previous training. Operation time was often shortened or task fulfillment was enabled. We are convinced that such strong autonomy functions are needed for disaster response robots to make rapid deployment in unknown scenarios possible.

REFERENCES

- [1] P. Hebert, M. Bajracharya, J. Ma, N. Hudson, A. Aydemir, J. Reid, C. Bergh, J. Borders, M. Frost, *et al.*, "Mobile manipulation and mobility as manipulation-design and algorithms of RoboSimian," *Journal of Field Robotics*, vol. 32, no. 2, pp. 255–274, 2015.
- [2] M. Zucker, S. Joo, M. X. Grey, C. Rasmussen, *et al.*, "A general-purpose system for teleoperation of the DRC-HUBO humanoid robot," *Journal of Field Robotics*, vol. 32, no. 3, pp. 336–351, 2015.
- [3] A. Stentz, H. Herman, A. Kelly, E. Meyhofer, G. C. Haynes, D. Stager, *et al.*, "CHIMP, the CMU highly intelligent mobile platform," *Journal of Field Robotics*, vol. 32, no. 2, pp. 209–228, 2015.
- [4] M. Schwarz, T. Rodehutsors, D. Droschel, M. Beul, M. Schreiber, N. Araslanov, I. Ivanov, C. Lenz, *et al.*, "NimbRo Rescue: Solving disaster-response tasks with the mobile manipulation robot Momaro," *Journal of Field Robotics*, vol. 34, no. 2, pp. 400–425, 2017.
- [5] M. Schwarz, M. Beul, D. Droschel, S. Schüller, A. S. Periyasamy, C. Lenz, M. Schreiber, and S. Behnke, "Supervised autonomy for exploration and mobile manipulation in rough terrain with a centaur-like robot," *Frontiers in Robotics and AI*, vol. 3, p. 57, 2016.
- [6] M. Schwarz, T. Rodehutsors, M. Schreiber, and S. Behnke, "Hybrid driving-stepping locomotion with the wheeled-legged robot Momaro," in *Int. Conf. on Robotics and Automation (ICRA)*, 2016.
- [7] Y. Liu and G. Nejat, "Robotic urban search and rescue: A survey from the control perspective," *Journal of Intelligent & Robotic Systems*, vol. 72, pp. 147–165, 2013.
- [8] N. Marturi, A. Rastegarpanah, C. Takahashi, M. Adjigble, *et al.*, "Towards advanced robotic manipulation for nuclear decommissioning: A pilot study on tele-operation and autonomy," in *Int. Conf. on Robotics and Automation for Humanitarian Applications*, 2016.
- [9] A. Leeper, K. Hsiao, M. Ciocarlie, L. Takayama, and D. Gossow, "Strategies for human-in-the-loop robotic grasping," in *Int. Conf. on Human-Robot Interaction (HRI)*, 2012.
- [10] K. Muelling, A. Venkatraman, J.-S. Valois, J. E. Downey, J. Weiss, S. Javdani, M. Hebert, *et al.*, "Autonomy infused teleoperation with application to brain computer interface controlled manipulation," *Autonomous Robots*, vol. 41, no. 6, pp. 1401–1422, 2017.

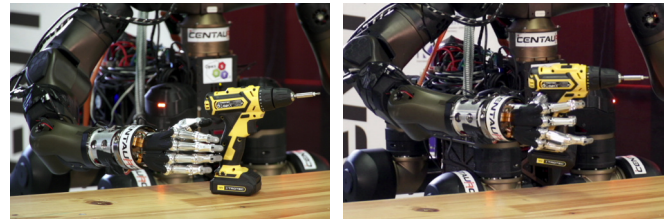


Fig. 19. Autonomous Grasping: approaching (l.) and lifting the drill (r.).

- [11] A. Peer, S. Eidenkel, and M. Buss, "Multi-fingered telemanipulation-mapping of a human hand to a three finger gripper," in *Int. Symposium on Robot and Human Interactive Communication*, 2008.
- [12] G. Salvietti, L. Meli, G. Gioioso, M. Malvezzi, and D. Prattichizzo, "Multi-contact bilateral telemanipulation with kinematic asymmetries," *Tr. on Mechatronics*, vol. 22, no. 1, pp. 445–456, 2017.
- [13] I. Havoutis, A. K. Tanwani, and S. Calinon, "Online incremental learning of manipulation tasks for semi-autonomous teleoperation," in *Int. Conf. on Intelligent Robots and Systems, Workshop on Closed Loop Grasping and Manipulation: Challenges and Progress*, 2016.
- [14] N. G. Tsagarakis, D. G. Caldwell, F. Negrello, *et al.*, "WALK-MAN: A high performance humanoid platform for realistic environments," *Journal of Field Robotics*, vol. 34, no. 7, pp. 1225–1259, 2017.
- [15] L. Baccelliere, N. Kashiri, L. Muratore, A. Laurenzi, M. Kamedula, A. Margan, S. Cordasco, J. Malzahn, and N. G. Tsagarakis, "Development of a human size and strength compliant bi-manual platform for realistic heavy manipulation tasks," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2017.
- [16] S. W. Ruhl, C. Parltz, G. Heppner, A. Hermann, A. Roennau, and R. Dillmann, "Experimental evaluation of the Schunk 5-finger gripping hand for grasping tasks," in *Int. Conf. on Robotics and Biomimetics (ROBIO)*, 2014.
- [17] M. G. Catalano, G. Grioli, E. Farnioli, A. Serio, C. Piazza, and A. Bicchi, "Adaptive synergies for the design and control of the Pisa/IIT SoftHand," *The Int. Journal of Robotics Research*, vol. 33, no. 5, pp. 768–782, 2014.
- [18] P. Fankhauser, M. Bloesch, D. Rodriguez, R. Kaestner, M. Hutter, and R. Siegwart, "Kinect v2 for mobile robot navigation: Evaluation and modeling," in *Int. Conf. on Advanced Robotics (ICAR)*, 2015.
- [19] D. Droschel, M. Schwarz, and S. Behnke, "Continuous mapping and localization for autonomous navigation in rough terrain using a 3D laser scanner," *Robotics and Autonomous Systems*, vol. 88, pp. 104–115, 2017.
- [20] M. Schwarz, C. Lenz, G. M. Garca, S. Koo, A. S. Periyasamy, M. Schreiber, and S. Behnke, "Fast object learning and dual-arm coordination for cluttered stowing, picking, and packing," in *Int. Conf. on Robotics and Automation (ICRA)*, 2018.
- [21] G. Lin, A. Milan, C. Shen, and I. Reid, "Refinenet: Multi-path refinement networks with identity mappings for high-resolution semantic segmentation," in *Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [22] T. Klamt and S. Behnke, "Anytime hybrid driving-stepping locomotion planning," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2017.
- [23] T. Klamt and S. Behnke, "Planning hybrid driving-stepping locomotion on multiple levels of abstraction," *Int. Conf. on Robotics and Automation (ICRA)*, 2018.
- [24] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 12, pp. 2262–2275, 2010.
- [25] D. Rodriguez, C. Cogswell, S. Koo, and B. Sven, "Transferring grasping skills to novel instances by latent space non-rigid registration," in *Int. Conf. on Robotics and Automation (ICRA)*, 2018.
- [26] D. Rodriguez and S. Behnke, "Transferring category-based functional grasping skills by latent space non-rigid registration," in *IEEE Robotics and Automation Letters (RA-L)*, 2018, pp. 2662–2669.
- [27] D. Pavlichenko and S. Behnke, "Efficient stochastic multicriteria arm trajectory optimization," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2017.
- [28] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor, and S. Schaal, "STOMP: Stochastic trajectory optimization for motion planning," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011.