Dexterous Underwater Manipulation from Distant Onshore Locations

A. Birk, T. Doernbach, C.A. Müller, T. Luczynski, A. Gomez Chavez, D. Köhntopp, A. Kupcsik, S. Calinon, A.K. Tanwani, G. Antonelli, P. di Lillo, E. Simetti, G. Casalino, G. Indiveri, L. Ostuni, A. Turetta, A. Caffaz, P. Weiss, T. Gobert, B. Chemisky, J. Gancet, T. Siedel, S. Govindaraj, X. Martinez, P. Letier

Underwater manipulation is a challenging problem. The state of the art is dominated by Remotely Operated Vehicles (ROV). ROV operations typically require an offshore crew consisting of at least an intendant, an operator, and a navigator. This crew often has to be duplicated or even tripled due to work shifts. In addition, customer representatives often wish to be physically present offshore. Furthermore, underwater intervention missions are still dominated by a significant amount of low-level, manual control of the manipulator(s) and of the vehicle itself. While there is a significant amount of research on Autonomous Underwater Vehicles (AUV) in general and there are even already fieldable solutions for inspection and exploration missions, there is still quite some room for adding intelligent autonomous functions for interventions.

We present here work to reduce the amount of robot operators required offshore - hence reducing cost and inconveniences - by facilitating operations from an onshore control center and reducing the gap between low-level teleoperation and full autonomy (Fig. 1). The basic idea is that the user interacts with a real time simulation environment, and a cognitive engine analyzes the user's control requests and turns them into movement primitives that the ROV needs to autonomously execute in the real environment - independently of communication latencies.

This article focuses on the results of intensive field trials from 26th of June until 7th of July 2017 in the Mediteranean Sea offshore of Marseille. Seven extended experimental dives were performed with the ROV while being connected via satellite to the command center in Brussels. Four different sites were used with different water depths (8m, 30m, 48m, 100m).

I. System Components

A. Overview

The presented work is targeted at a high technology readiness level (TRL) of 6, i.e., it is developed and validated beyond just lab experiments. The research vessel JANUS II of COMEX with a 2500 msw SubAtlantic Apache ROV is used for this purpose (Fig. 2). For our research, the ship is equipped with satellite communications (Sec. I-D) to allow the control of the ROV by pilots who are located in a command center in Brussels, Belgium (Sec. I-E). Furthermore, a skid is added to the ROV to carry additional components used for our research, namely an electric manipulator, respectively two manipulators in a bi-manual setup (Sec. I-B) and a vision system (Sec. I-C).

B. The Underwater Manipulator(s)

Our manipulator has been designed starting from the Underwater Modular Arm (UMA). Two kinds of electrically-driven joints are available, with one, respectively two motion axes, which are complemented by a set of links for connecting the joints. Different kinematic configurations can be obtained by varying the number of basic modules, i.e., joints and links, and/or by varying the way in which they are interconnected. The arm is characterized by 6 degrees of freedom (DOF), which are obtained by connecting three modules with each one with 2 DOF forming a pitch-roll configuration. The overall length when totally stretched is slightly more than 1 meter. However, the arm is also fully foldable, in order to minimize its size when parked in the ROV skid during the navigation phases. Both a single arm and a dual arm set-up can be used. During the 2017 trials, a mock-up of grippers that are under development were used.

C. The Vision System

An intelligent underwater vision system with computing power on-board the ROV is used to minimize the traffic over the umbilical cable from the ROV to the vessel. It is based on high-resolution firewire (IEEE 1934b) cameras in pressure housings connected to an embedded computer, which can be used for vision processing and adaptive video compression on-board of the ROV.

The firewire bus supports among others the synchronization of the cameras. They can hence be used for stereo, respectively multi-camera set-ups to generate depth information from different views with a known relative geometry. Due to payload constraints of the Apache ROV, a stereo set-up with two cameras is used in the 2017 trials. The compute bottle of the vision system on-board of the robot also services the core navigation sensors in form of a LinkQuest NavQuest 600P Micro Doppler Velocity Log (DVL) and a Xsens MTi-300 Inertial Measurement Unit (IMU).

D. Satellite Communication

Satellite communications services for mobile offshore maritime operations are associated with bandwidth limitations (uplink and downlink), inherent delays, and disruptions and

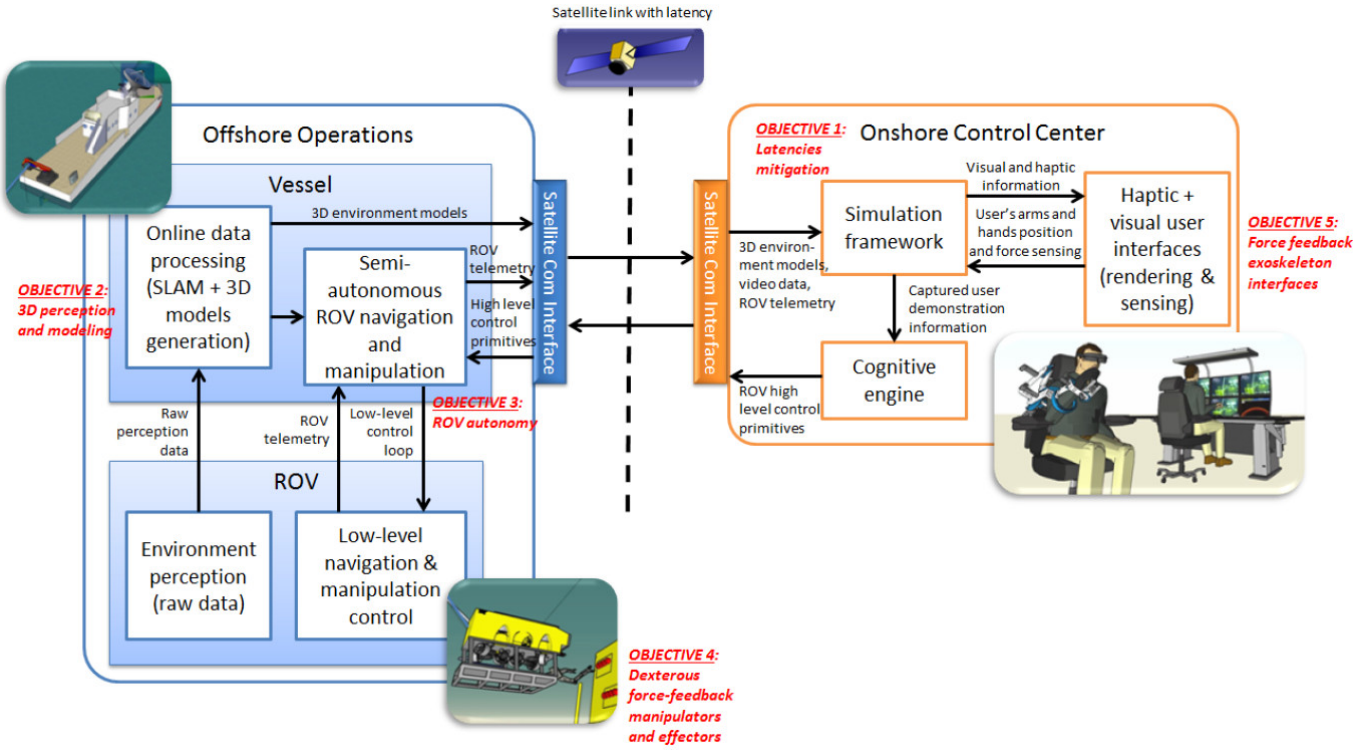

Fig. 1. Underwater manipulation from an onshore control center lessens operation costs and eases the involvement of mission specific experts, but it requires increased autonomous intelligent functions to compensate for latency and bandwidth constraints. An overview of our system components and their interplay is shown here.

Fig. 2. The system tested in field trials consists of an Apache ROV extended with newly developed components including a dual arm set-up and an intelligent vision system (left), which is deployed from the COMEX Janus II vessel (center) with a satellite connection to a control center in Brussels. A mockup panel structure is used to test different application scenarios (right).

they require a complex stabilized satellite tracking antenna. In the context of this research, a maritime VSAT solution is employed from a service provider (Omniaccess) that includes a Ku band Cobham Sailor 800 tracking antenna, its controller and the related modems. The nominal data bandwidth for the uplink from the vessel is 768 kb/s and the downlink to the vessel is 256 kb/s with an inherent nominal round trip delay of 620 ms.

E. The Control Center and the Exoskeleton

The onshore control center in Brussels consists of a monitoring and control room that features a double 7 DOF arm and 6 DOF hand force feedback exoskeleton. It is based in part on a design for the European Space Agency (ESA) [1] that was further improved in the EU-FP7 project ICARUS [2]. It is designed as a modular solution, allowing each arm and hand exoskeleton subsystem to be easily and conveniently connected and removed from the rest of the setup. It features furthermore a passive gravity compensation system that connects to the arm exoskeletons, and that can be calibrated to compensate for the full mass of the exoskeletons physical setups as well as the mass of the user arms. The user is hence given the impression to operate in neutral buoyancy, i.e., typically as a diver would. This reduces users fatigue during the ROV operation.

F. A Test Panel for System Validation

A test panel was developed for validation, which also served as target for the trials to emulate different scenarios, e.g., offshore oil&gas facilities or the handling of archeological artifacts (Figs 2 and 3). The panel consists of three sides. Each one is equipped with mockup elements. One side is used to test functionalities in offshore oil&gas interfaces based on the ISO 13628 standard including, e.g., valves or wet-mate connectors. Furthermore, a biologic panel including mockup corals and an archeological box including mockup ceramics are included.

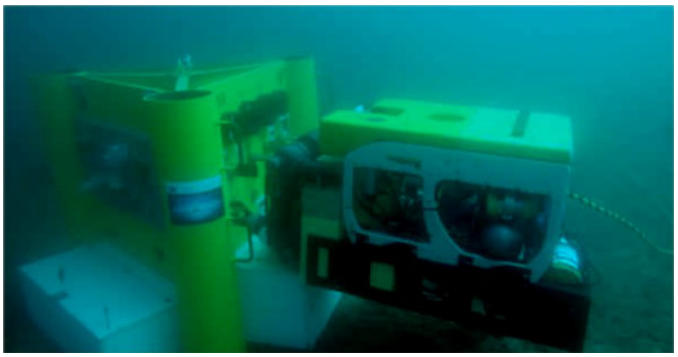

Fig. 3. The ROV and the test panel during the field trials.

II. Data Flows between the Vessel/ROV and the Control Center

To sustain effective remote ROV operations, multiple data flows are required such as ROV commands, video streams, pose updates, 2.5/3D environmental maps, status updates, etc. between the onshore and offshore nodes, which need to be transmitted via the satellite link. It is hence critical to optimize the bandwidth usage by prioritizing data flows with specific Quality of Service (QoS) information, shaping the traffic [3] to avoid network overload and to ensure data reliability with minimal overheads. To address the described challenges, DDS OpenSplice (Prismtech) middleware is used to exchange data between the onshore and offshore nodes over the bandwidth-constrained satellite network.

The onshore control center and the ROV control and perception framework embed command and data interfaces in the Robot Operating System (ROS) through asynchronous publish-subscribe mechanisms over named and type-specific topics. A ROS2DDS bridge has been developed, which can be configured to interface with existing ROS topics in a system. For each ROS topic, the ROS2DDS bridge automatically creates a corresponding DDS data reader, data writer and named topics across the distributed nodes with associated QoS policies according to the Object Management Group's DDS QoS specification.

The architecture of the bridge is scalable to deploy multiple nodes with dynamic discovery of distributed ROS2DDS entities. The maximum burst sizes indicate the amount in bytes to be sent at maximum every "resolution" milliseconds. With reliable QoS, i.e., guaranteed data delivery, the maximum burst values are typically set just below the maximum bandwidth available for the uplink from the offshore side. For example, in the presence of a 768 kb/s satellite uplink, the maximum burst size of 650 kb/s, i.e., around 85% of the bandwidth, was found to be efficient. The remaining bandwidth is made available for retransmissions of data packets. In the case of best effort QoS, i.e., it is not necessary to re-send or acknowledge any received packets, the maximum burst size can be set to the available bandwidth. During the marine trials, the satellite link is shared between multiple data flows that are assigned different priorities and the maximum burst sizes are adapted proportionately.

III. Manipulation Tasks with the Cognitive Engine

The main purpose of the Cognitive Engine (CE) is to overcome teleoperator control delays in the face of satellite communication latency. Prior to the mission, the teleoperator first demonstrates a set of tasks (e.g., turning a valve, grasping a handle, etc.) using the exoskeleton in the onshore control center, which the CE encodes as statistical models. These models are transferred to the offshore vessel, where control of the ROV takes place. During mission execution, the offshore model assists the teleoperator-guided manipulation such that the tasks are replicated by adapting them to the current environmental situations. This reduces the cognitive load on the teleoperator, who can concentrate on selecting the tasks in the virtual environment.

The CE can assist the teleoperator in two different modes [4]:

Shared control: the teleoperator input is directly combined with the motion predicted by the task model. The adaptation is weighted based on the variability of the demonstrations in the parts of the task that are currently executed. For parts of the task requiring accuracy, the model assists the teleoperator by correcting deviations from the original demonstrations. For parts of the task allowing more variations, the teleoperator is free to move within the regions corresponding to the demonstrations.

Semi-autonomous control: the task is executed by generating the most likely trajectory starting from the current pose of the robot. This mode is particularly useful when delays or interruptions from the satellite communication is expected. In this control mode, the teleoperator visualizes and triggers the execution of the movement, which is executed until a new signal from the teleoperator is given [5].

In the following we give an overview on task learning and how motions are reproduced in varying situations.

A. Learning adaptive tasks from demonstrations

Our application requires the CE to learn skills from only a handful of demonstrations (typically up to 10). Additionally, we require skills that can be reproduced in novel environmental situations, for which no demonstration is available. To achieve this goal, the CE relies on a Task-Parametrized Gaussian Mixture Model (TP-GMM) [6] to encode demonstrations executed in different situations (Fig. 4). The task parameters are frames of references (coordinated systems with position and orientation information) associated to virtual landmarks or objects/tools in the environment. For example, in a valve

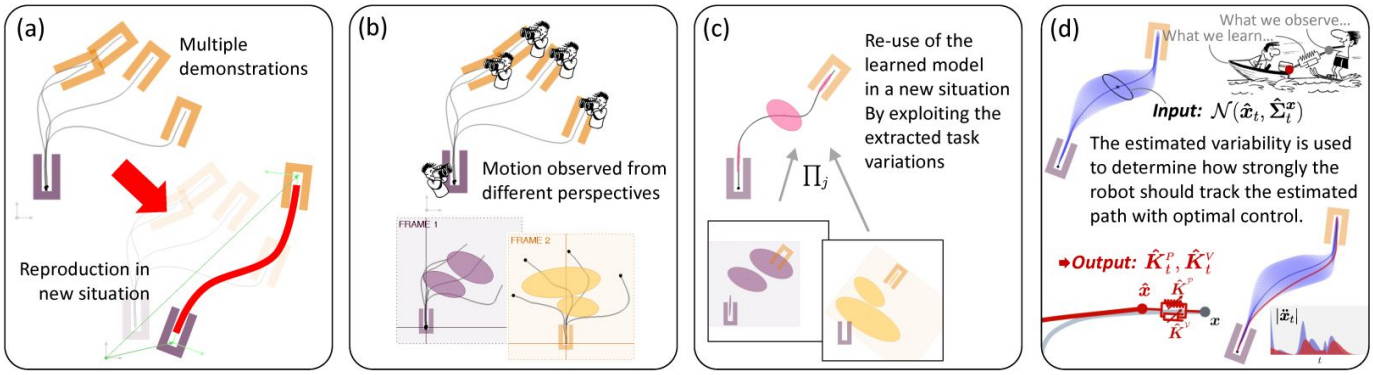

Fig. 4. Overview of the learning approach used in the Cognitive Engine. **a)** Demonstrations are collected with different task parameters (frames of reference). **b)** The demonstrations are transformed in each particular frame and a GMM is learnt in each frame. **c)** In a new situation, a new GMM is computed with a product of linearly transformed Gaussians. **d)** The computed trajectory distribution provides a variance estimate for each set-point, which determines how accurately the robot should pass through these set-points.

turning task, such frames may refer to the robot base frame, the current valve pose and the targeted valve pose. Fig. 4 depicts the learning and retrieval process. First, demonstrations are collected in varying situations (each time with different task parameters). In order to capture the variance of the demonstrations, Gaussian Mixture Models (GMMs) are learnt in each task-relevant frame. Learning the models in each individual frame allows the system to generalize the observed tasks to new situations.

B. Task reproduction with adaptation to new situations

In a novel environmental situation, the Gaussian mixture components are transformed using the newly observed task parameters (Fig. 4.c). The retrieved GMM is exploited differently according to the selected control mode, both aiming at reducing the cognitive load on the teleoperator when executing a set of tasks.

In the *shared* control mode (Fig. 5), Gaussian Mixture Regression (GMR) is used on both teleoperation and robot sides to generate probabilistic trajectory distributions, represented as a tube in the figure. On the teleoperator side, this tube is adapted locally to match the situation of the virtual environment in which the user is immersed - here, the model can for example be used for haptic corrections. On the robot side, the same model adapts to the situation that is locally detected. This situation can potentially differ with the one currently experienced by the user, as depicted in the figure: the two tubes have different shapes but share the same GMM parameters.

In this way, the robot is provided with a fast adaptation technique that can directly exploit the locally sensed information, i.e., the tube is adapted online without passing through the slow satellite communication. This type of assistance is relevant to handle small transmission delays, i.e., to cope with the discrepancy of situations due to the slow refreshing rate. For longer delays, a semi-autonomous mode, as described next, is usually preferred.

In the *semi-autonomous* control mode (Fig. 6), a Linear Quadratic Tracking (LQT) controller in operational space is

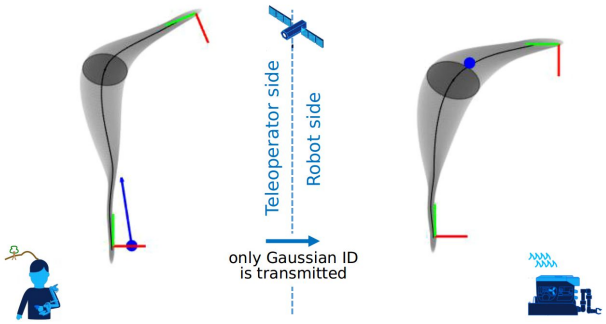

Fig. 5. Shared control mode with two frames of reference on each side. The current poses of the teleoperator and robot are displayed with blue dots.

used to generate a reference trajectory starting from the current robot pose. These acceleration commands in operational space are used by the robot controller until the teleoperator decides to abort the task or switch to another task. On the teleoperator side, the retrieved trajectory is used for visualization purpose.

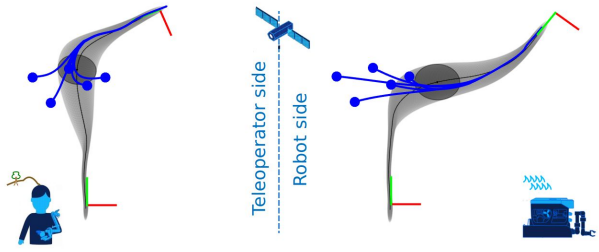

Fig. 6. Semi-autonomous control mode, with acceleration commands and an associated trajectory computed from any robot/teleoperator poses (displayed as blue points) using the model.

IV. ROV Control for Intervention Missions with Communication Latencies

A. Task Priority Inverse Kinematics

From a control point of view, the DexROV system is much more similar to an AUV than an ordinary ROV, in the sense

that it needs to take care of many control objectives on its own, with high-level inputs coming from the user through the cognitive engine.

For this reason, following an approach similar to the one adopted for the TRIDENT [7] and MARIS [8] projects, the developed control is a Task Priority Inverse Kinematics (TPIK) algorithm that allows to set a priority order among several tasks and to find the system velocity vector $\dot{\boldsymbol{y}}$ that accomplishes them simultaneously at best, following the priority order. Given a hierarchy composed by k tasks $\dot{\boldsymbol{\sigma}}_1 \dots \dot{\boldsymbol{\sigma}}_k$, the target velocity $\dot{\boldsymbol{y}}$ can be computed as:

$$\dot{\boldsymbol{y}} = \dot{\boldsymbol{y}}_1 + \boldsymbol{N}_1 \dot{\boldsymbol{y}}_2 + \cdots + \boldsymbol{N}_{1,k-1} \dot{\boldsymbol{y}}_k \tag{1}$$

where each $\dot{\boldsymbol{y}}_i$ is the velocity contribution of the task i and $\boldsymbol{N}_{1,i}$ is the null space of the augmented Jabobian matrices from $\boldsymbol{\sigma}_1$ to $\boldsymbol{\sigma}_i$. If there are conflicting tasks, the projection of the velocity contribution of the lower priority tasks into the null space of the higher priority ones guarantees that the priority order is always respected. This control framework has been extended to handle also set-based tasks [9], [10] like, for example, arm joint mechanical limits or obstacle avoidance, in which the control objective is to keep the task value above a lower threshold or below an upper threshold. In order to effectively and safely operate the system, it is useful to divide all the tasks in three groups and to exploit this classification to assign priority levels: 1) Safety tasks such as ROV autoaltitude, mechanical joint limits, obstacle avoidance, that assure the integrity of the system and of the environment in which it operates. 2) Operational tasks that contain all the tasks commanded by the user, such as ROV guidance from point A to B, end-effector position or configuration. 3) Optimization tasks, that contain all those tasks that are not strictly necessary for the actual accomplishment of the operation, but they help to do it in a more efficient way, e.g., the arm manipulability. The hierarchy can be changed in terms of the number and the order of priority of the tasks as a function of the action that needs to be performed.

B. Experimental Validation

The developed control framework has been validated and tested during the 2017 field trials. In the following, the results of an experiment where only the manipulator is controlled with the proposed TPIK algorithm are shown. Then, a simulation showing the coordinated control of the vehicle and the arm is described.

Regarding the field experiment, the chosen task hierarchy is composed by two tasks: arm joint limits avoidance and the end-effector position. The end-effector is commanded to follow a simple circular trajectory, while the joint 3 upper and joint 5 lower thresholds have been chosen in order to get active during the motion of the end-effector, to test the priority mechanism. Figure 7 shows the position error and the joint values with the corresponding thresholds respectively. The effectiveness of the control algorithm is clear, as the end-effector follows the desired trajectory and the joint values never exceed the desired upper and lower thresholds.

A simulation experiment further illustrates results of the coordinated control. The task hierarchy is as follows:

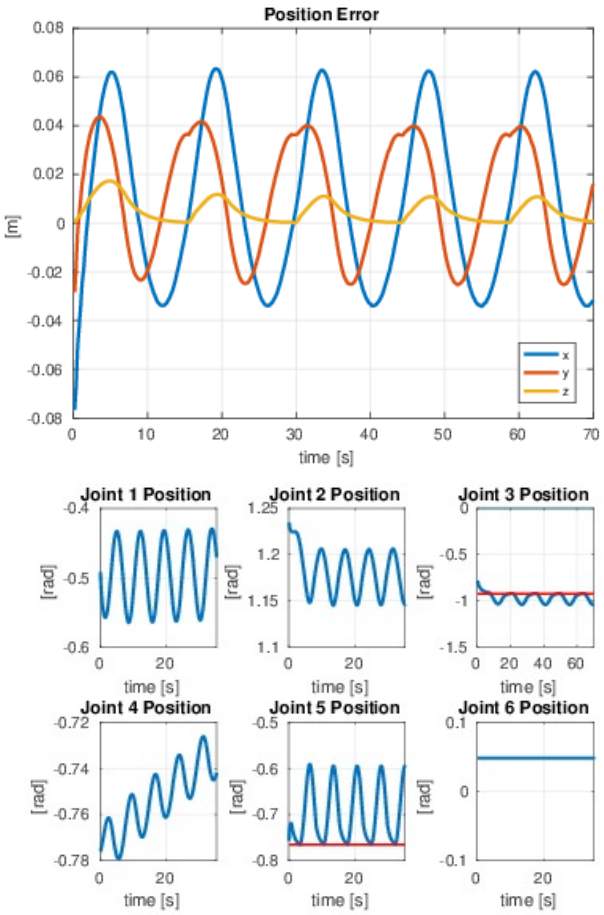

Fig. 7. The position error over time during the experiments shown on top. The error is relatively high because the control gains were maintained low for safety reasons during these tests. Below, the joint positions and the minimum/maximum thresholds are shown (in red). The plots show how the TPIK approach enforces the validity of the joint limits.

- **Arm manipulability:** a minimum value of 0.029 is set for the measure of manipulability of the arm, in order to avoid singular configurations
- **Virtual box:** a set of 6 virtual walls surrounding the arm base frame that assures that the arm never tries to move the end-effector outside its workspace
- **End-effector configuration:** a constant set-point for the position and the orientation of the arm has been set.

Figure 8 shows the results. The system's initial position is set far away from the desired waypoint, and it moves both the vehicle and the arm in order to reach it with a null error, while the arm manipulability and the end-effector position expressed in the arm base frame never exceed the desired thresholds.

As concerns the vehicle related tasks, the proposed solution [11] builds on standard techniques leading to a proportional-integral (PI) controller including an anti wind-up mechanism.

V. Underwater Perception for Manipulation

A. Camera Calibration

For manipulation - with both teleoperation as well as with autonomy - it is essential that the sensor system is well calibrated to correctly capture the environment. We use a novel

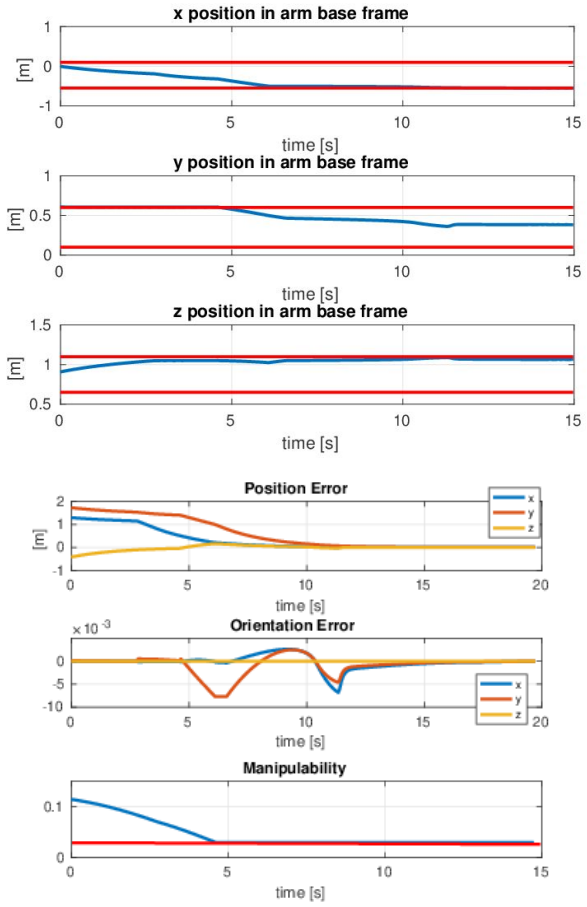

Fig. 8. Top: end-effector position expressed in the arm base frame with the limits imposed by the virtual walls (red). Bottom: end-effector position and orientation error and measure of manipulability with the corresponding minimum threshold. All the set-based tasks stay within their limits while the position and orientation errors reach a null value.

calibration and refraction correction process for underwater cameras with flat-pane interfaces that is very easy and convenient to use while providing very accurate rectification results [12].

The correction is derived from an analysis of the axial camera model for underwater cameras, which is physically correct but which is among others computationally hard to tackle. It can be shown how realistic constraints on the distance of the camera to the window can be exploited, which leads to an approach dubbed *pinax* model as it combines aspects of a virtual pinhole model with the projection function from the axial camera model. The pinax model is not only convenient as it allows in-air calibration, it also outperforms standard methods in accuracy [12].

B. 3D Mapping and Object Recognition

The data from the camera system with its stereo set-up is online processed to generate dense 2.5D point-clouds, which get integrated in 3D maps (Fig.9). The well known octree data-structure is used for this purpose. More precisely, our implementation builds upon the popular OctoMap library [13], which is extended for differential update operations to support an efficient, low-latency transmission of the 3D representation over the satellite link to the onshore command center [14]. Furthermore, an efficient strategy for underwater color updates is added.

Underwater images suffer from challenging light conditions, especially wave-length dependent attenuation as well as forward and back scattering. When coloring the octomap, a very simple but efficient strategy is used: as attenuation is wavelength and distance dependent, the brightest measurement is used, which corresponds to the closest and hence most accurate sample [14]. For substantial image enhancement - at much higher computational cost - a new variant of the Dark Channel Prior for underwater vision is used [15]. Even though there are known adaptations of this method to underwater applications, further improvements are made by reformulating the problem. Bright regions in the dark channel appear in outdoor images in air on non-sky regions due to the backscattering, which is used in the original Dark Channel Prior. In contrast to other methods adapting it to underwater applications, the estimation of the ”atmospheric” light is adjusted, which leads to clear improvements [15].

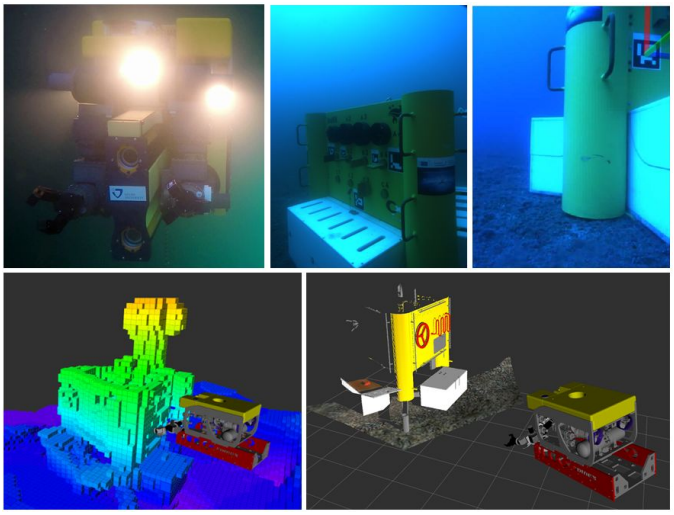

Fig. 9. When the vehicle (top left) approaches the mockup panel structure (top center) Augmented Reality marker (top right) can be used to aid the navigation, respectively to validate different navigation methods. Among others, a 3D octomap (bottom left) is generated in real-time, which is transmitted to the offshore control center. As the mockup panel structure is - like in oil&gas operations - a priori known, perceived parts can be used to determine its pose and to project the known model in the scene to aid the execution of tasks (bottom right).

The core navigation of the vehicle is based on the data of the according sensors connected to the vision compute bottle, i.e., the NavQuest DVL and the Xsens IMU. This data is fused in an Extended Kalman Filter (EKF). It can be aided by the registration of the stereo scans and the tracking of objects up to the level of full Simultaneous Localization and Mapping (SLAM). The mockup panel structure is equipped with Augmented Reality (AR) markers (Fig.9), which can further aid the navigation, respectively which can be used for the validation of the other navigation methods. The lower-level image processing, i.e., the rectification and the image enhancement, are important for the robust recognition of the

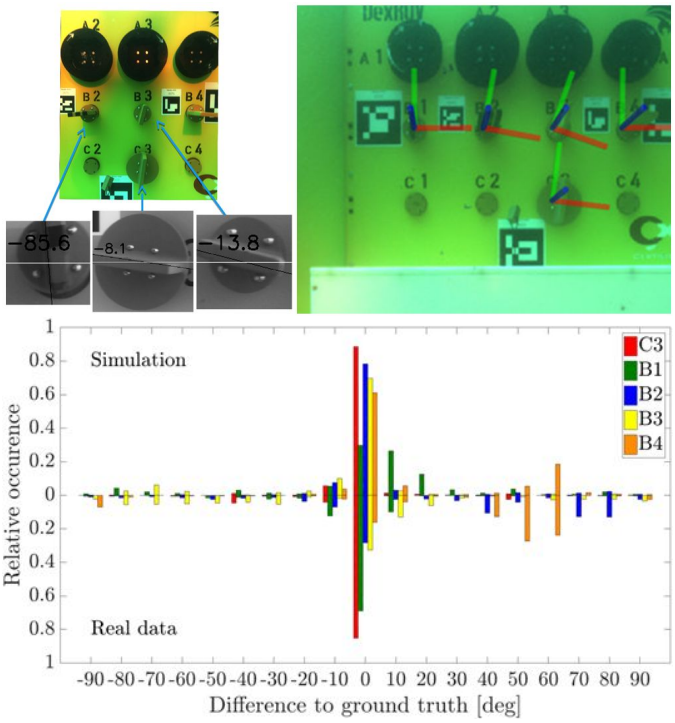

Fig. 10. Top: As basis for (semi-)autonomous manipulation, the orientations of the valves on the mockup structure are determined by an active contours method with super-ellipse fitting in combination with a Hough transform. Bottom: The errors in a single frame state detection are within a few degrees and they behave very similar in the sea trials (real data) as in the high-fidelty simulation of the system (simulation).

markers.

The 3D octomap can be run with a 2 cm grid cell resolution. This is well suited for autonomous operations like collision avoidance and path-planning, and it is also sufficient to give human operators in the onshore command center an overview of the environment including unexpected obstacles and the terrain. To aid manipulation, the fact can be exploited that in many application cases, e.g., for oil&gas operations, the structures that are to be dealt with are known. The detailed simulation framework that is used for component and system validation [16] can in this context be used to provide a virtual visualization of the real underwater operations (Fig.9), i.e., when the vision system of the ROV detects, respectively tracks (parts of) the structure of interest, the structure's pose is transmitted to the onshore command center and the known model can be projected into the scene. Furthermore, there are dedicated vision processes to detect and localize crucial objects as basis for (semi-)autonomous manipulation. For example, the valves on the mockup structure are detected and localized with an active contours method with super-ellipse fitting. Their orientations are determined with a Hough transform to estimate the predominate edges within the fitted ellipses (Fig.10).

VI. Conclusions

An approach to underwater manipulation was presented, which facilitates the use of a distant onshore control center with an exoskeleton based on: 1) efficient transmission of multiple data-streams over a satellite link, 2) a cognitive engine to mitigate communication latencies by encoding statistical models of manipulation tasks, 3) the vehicle control, which is more oriented towards AUV than ROV operations, 4) an intelligent vision system, which provides perception capabilities. The approach was tested in July 2017 in a first field campaign over two weeks in the Mediterranean Sea near Marseille. Seven extended dives with about eleven hours of experimental data were performed where the ROV interacted with a mockup panel structure to validate the different system components and their interplay.

The main lessons learned from the field trials are: 1) ROV operation from an onshore control center via a satellite link is in principle feasible despite latencies and low bandwidth, 2) but it is important that the system is capable of detecting the operators intentions and uses this to (semi-)autonomously carry out tasks, 3) there is no black or white with respect to autonomy but there are different levels that can be useful or even necessary depending on the communication conditions, and 4) good situational awareness through constant update of the onshore environment model is important.

VII. Acknowledgments

The presented research is carried out in the project "Effective Dexterous ROV Operations in Presence of Communications Latencies (DexROV)", which is funded by the European Commissions Horizon 2020 Framework Programme for Research and Innovation, under the topic "Blue Growth: Unlocking the Potential of Seas and Oceans", BG6-2014 "Delivering the sub-sea technologies for new services at sea".

References

[1] P. Letier, E. Motard, and J. P. Verschueren, "Exostation : Haptic exoskeleton based control station," in *2010 IEEE International Conference on Robotics and Automation*, 2010, pp. 1840–1845.

[2] G. D. Cubber, D. Doroftei, Y. Baudoin, D. Serrano, K. Chintamani, R. Sabino, and S. Ourevitch, "ICARUS: An EU-FP7 project providing unmanned search and rescue tools," in *IROS Workshop on Robots and Sensors integration in future rescue INformation system (ROSIN'12)*, 2012.

[3] M. Karaliopoulos, R. Tafazolli, and B. G. Evans, "Providing differentiated service to tcp flows over bandwidth on demand geostationary satellite networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 2, pp. 333–347, Feb 2004.

[4] A. K. Tanwani and S. Calinon, "A generative model for intention recognition and manipulation assistance in teleoperation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2017, pp. 43–50.

[5] A. Tanwani and S. Calinon, "Learning robot manipulation tasks with task-parameterized semitied hidden semi-markov model," *Robotics and Automation Letters, IEEE*, vol. 1, no. 1, pp. 235–242, 2016.

[6] S. Calinon, "A tutorial on task-parameterized movement learning and retrieval," *Intelligent Service Robotics*, vol. 9, no. 1, pp. 1–29, January 2016.

[7] E. Simetti, G. Casalino, S. Torelli, A. Sperinde, and A. Turetta, "Floating underwater manipulation: Developed control methodology and experimental validation within the TRIDENT project," *Journal of Field Robotics*, vol. 31, no. 3, pp. 364–385, 2014.

[8] E. Simetti, F. Wanderlingh, S. Torelli, M. Bibuli, A. Odetti, G. Bruzzone, D. Rizzini, J. Aleotti, G. Palli, L. Moriello, and U. Scarcia, "Autonomous underwater intervention: Experimental results of the MARIS project," *IEEE Journal of Oceanic Engineering*, pp. 1–20, 2017.

[9] S. Moe, G. Antonelli, A. R. Teel, K. Y. Pettersen, and J. Schrimpf, "Set-based tasks within the singularity-robust multiple task-priority inverse kinematics framework: General formulation, stability analysis and experimental results," *Frontiers in Robotics and AI*, vol. 3, p. 16, 2016.

[10] F. Arrichiello, P. D. Lillo, D. D. Vito, G. Antonelli, and S. Chiaverini, "Assistive robot operated via P300-based brain computer interface," in *IEEE International Conference on Robotics and Automation*. IEEE, 2017, pp. 6032–6037.
[11] D. D. Palma and G. Indiveri, "Underwater vehicle guidance control design within the DexROV project: preliminary results," *IFAC-PapersOnLine*, vol. 49, no. 23, pp. 265–272, 2016.
[12] T. Luczynski, M. Pfingsthorn, and A. Birk, "The pinax-model for accurate and efficient refraction correction of underwater cameras in flat-pane housings," *Ocean Engineering, Vol. 133, pp. 9-22, March 2017*, vol. 133, pp. 9–22, 2017.
[13] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, 2013.
[14] T. Luczynski, T. Fromm, S. Govindaraj, C. A. Mueller, and A. Birk, "3d grid map transmission for underwater mapping and visualization under bandwidth constraints," in *IEEE Oceans*. IEEE press, 2017.
[15] T. Luczynski and A. Birk, "Underwater image haze removal with an underwater-ready dark channel prior," in *IEEE Oceans*. IEEE press, 2017.
[16] T. Fromm, C. A. Mueller, M. Pfingsthorn, A. Birk, and P. D. Lillo, "Efficient continuous system integration and validation for deep-sea robotics applications," in *IEEE Oceans*. IEEE press, 2017.

Andreas Birk, Tobias Fromm, Christian Atanas Müller, Tomasz Luczynski, Arturo Gomez Chavez, Daniel Köhntopp : c/o a.birk@jacobs-university.de, Jacobs University Bremen, Germany

Andras Kupcsik, Sylvain Calinon, Ajay Tanwani : c/o sylvain.calinon@idiap.ch, Idiap Research Institute, Switzerland

Gianluca Antonelli, Paolo di Lillo, Enrico Simetti, Giuseppe Casalino, Giovanni Indiveri, Luigi Ostuni : c/o antonelli@unicas.it, Interuniversity Center of Integrated Systems for the Marine Environment (ISME), Italy

Alessio Turetta, Andrea Caffaz : c/o alessio.turetta@graaltech.it, Graal Tech, Italy

Peter Weiss, Thibaud Gobert, Bertrand Chemisky : c/o p.weiss@comex.fr, Compagnie Maritime d'Expertises (COMEX), France

Jeremi Gancet, Torsten Siedel, Shashank Govindaraj, Xavier Martinez, Pierre Letier : c/o jeremi.gancet@spaceapplications.com, Space Applications Services (SpaceApps), Belgium